Investigation and Solution of Problems for Applying Identification Methods to Real Systems

Jae Hyeuk Suk

A dissertation submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

University of Washington

2009

Program Authorized to Offer Degree: Mechanical Engineering

University of Washington
Graduate School

This is to certify that I have examined this copy of a doctoral dissertation by

Jae Hyeuk Suk

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Chair of the Supervisory Committee:

_____

Brian C. Fabien

Reading Committee:

_____

Martin C. Berg

_____

Santosh Devasia

_____

Brian C. Fabien

Date: _____

In presenting this dissertation in partial fulfillment of the requirements for the doctoral degree at the University of Washington, I agree that the Library shall make its copies freely available for inspection. I further agree that extensive copying of this dissertation is allowable only for scholarly purposes, consistent with "fair use" as prescribed in the U.S. Copyright Law. Requests for copying or reproduction of this dissertation may be referred to Proquest Information and Learning, 300 North Zeeb Road, Ann Arbor, MI 48106-1346, 1-800-521-0600, to whom the author has granted "the right to reproduce and sell (a) copies of the manuscript in microform and/or (b) printed copies of the manuscript made from microform."

Signature_____

Date_____

University of Washington

Abstract

Investigation and Solution of Problems for Applying Identification Methods to Real Systems

Jae Hyeuk Suk

Chair of the Supervisory Committee:
Professor Brian C. Fabien
Mechanical Engineering

This dissertation will investigate and develop techniques for identifying linear dynamic systems. Here, new proposed techniques are a new identification technique using a complex variable transformation and a new model order selection method. In this dissertation, we will compare our techniques to other methods using simulated and experimental data of real applications, i.e., Flywheel Energy Storage System and Two-Linked-Arm Robot. To achieve those goals, several identification methods, including new techniques, will be introduced, compared, and evaluated specifically, Eigensystems Realization Algorithm, Fast Eigensystems Realization Algorithm, and Auto-Regressive Models.

Development of new techniques starts from the complex variables transformation which is for a compact and efficient representation of a gyroscopic systems. The transformation can convert a non-square plant to a square plant and eliminate the skew-symmetric term in gyroscopic systems. These advantages of the transformation make it possible for us to develop the new identification technique. Before the identification methods will be derived and applied to a simple example, proofs of the controllability matrix, the observability matrix, and responses of initial/forced condition using complex variable transformation will be done. Finally, new identification technique will be derived and applied to the simulated and real data. In addition, a practical model order selection method, which is in accord with new identification technique, is proposed and evaluated with real applications.

TABLE OF CONTENTS

Page

List of Figures iv

List of Tables vi

Chapter 1: Introduction 1
- 1.1 Introduction 1
- 1.2 Approach 9

Chapter 2: ERA - Eigensystem Realization Algorithm 11
- 2.1 Introduction of ERA 11
- 2.2 Basic Concepts 12
- 2.3 Identification 15
- 2.4 Extracting M, D_{amp}, and K Matrices 20

Chapter 3: Fast ERA 25
- 3.1 Introduction 25
- 3.2 Basic Concepts 25
- 3.3 Identification 28
- 3.4 Extracting $\mathbf{M}$, $\mathbf{D}_{amp}$ and $\mathbf{K}$ Matrices 34
- 3.5 Comparison with ERA 35

Chapter 4: AR Model - Auto-Regressive 37
- 4.1 Introduction 37
- 4.2 Concepts of a Random Process 37
- 4.3 Predictor 49
- 4.4 Parameter Selection 52
- 4.5 Extraction of State-Space Model 54
- 4.6 Comparison 54

Chapter 5: Algorithm Evaluation 55
5.1 Introduction 55
5.2 Three-DOF System without noises 55
5.3 3-DOF System with noise 68
5.4 Gyroscopic System 80

Chapter 6: Complex Variable Coordinate System with ERA 83
6.1 Introduction 83
6.2 Applying Modern Control Theory 86
6.3 Applying the ERA Identification Method 100

Chapter 7: Model Order Determination 106
7.1 Introduction 106
7.2 Model Order Selection Techniques 106
7.3 Model Reduction 108
7.4 Example 114

Chapter 8: Flywheel Energy Storage System 116
8.1 Introduction 116
8.2 Simulated Data 117
8.3 Experimental Data 127

Chapter 9: Two-Linked-Arm Robot System 140
9.1 Introduction 140
9.2 Background 141
9.3 Result of Applying Complex Variables ERA 146
9.4 Applying Model Reduction 146

Chapter 10: Conclusion 150
10.1 Conclusion 150
10.2 Future Research 151

Bibliography 152

Appendix A: Proofs 160
A.1 Observability and Controllability Matrices for Continuous State-space Model 160
A.2 Gerschgorin Disks Theorem 166

Appendix B: M-files 167
B.1 Main Program 167
B.2 Subroutine Functions 185

LIST OF FIGURES

Figure Number Page

1.1 Macroscopic diagram of system identification methods 5

4.1 An explanation of a random process with waveforms, a sample space, and probability measure. 40

4.2 ARX model structure 45

4.3 ARMAX Model Structure 46

4.4 Output-Error structure 48

4.5 Box-Jenkins Structure 48

5.1 Spring, Damping and Mass System 56

5.2 Simulation result of three springs-dampers-masses system without noises . . . 58

5.3 Identified system with ERA method 61

5.4 Identified System with Fast ERA 65

5.5 Identified System with ARX 67

5.6 Block diagram for adding white noise to state-space model system. 69

5.7 Identified System with ERA when size of Hankel matrix is 90 until 1.5 seconds. 70

5.8 Identified System with ERA when size of Hankel matrix is 90 after 1.5 seconds. 72

5.9 Singular Values of *Hankel matrix* 73

5.10 Identified System with ERA when size of state-space model (A) is 6 after reduction 74

5.11 Identified System with Fast ERA when size of Hankel matrix is 90 until 1.5 seconds. 77

5.12 Singular Values of D_q 78

5.13 Identified System with Fast ERA when size of state-space model is 6 after reduction 79

5.14 Comparison between ARX identified and original system with white noise . . 81

5.15 Two degree of freedom model for a shaft with non-constant angular velocity . 82

6.1 Initial responses of the gyroscopic example with real coordinates 88

6.2 Initial response of gyroscopic example with complex coordinates 88

6.3 Excitation responses of the gyroscopic example with real coordinates 90

6.4 Excitation responses of the gyroscopic example with complex coordinates . . 91
6.5 The time responses of the identified system. 105

7.1 The Magintude of the identified state-space $\bar{\mathbb{A}}_{id}$. 109
7.2 Gerschgorin's disks and eigenvalues of $\bar{A}$. 111
7.3 Flowchart of selecting system order with the Gerschgorin radii. 113
7.4 Parts of the magnitude of the identified state-space $\bar{\mathbb{A}}_{id}$, matrix. 114

8.1 The flywheel energy storage system parameters 118
8.2 Impulse responses of the FESS with $\omega = 1500$RPM 121
8.3 Complex variables Identified responses of the figure 8.2. 122
8.4 Errors between the figure 8.2 and the figure 8.3. 122
8.5 Magnitude of the identified state-space $\hat{\bar{\mathbb{A}}}_{id}$ matrix. 124
8.6 Identified responses of the simulated FESS with ARX model. 126
8.7 Non-spinning FESS experimental data (x_{top}, x_{bottom}, y_{top}, and y_{bottom}) with 0.005 sec sampling time. 128
8.8 Converted non-spinning FESS experimental data (x_{center}, y_{center}, θ_x, and θ_y) with 0.005 sec sampling time. 129
8.9 $H_{300\times300}$ and 4 order identified system responses 131
8.10 $H_{300\times300}$ and 2 order identified system responses 131
8.11 2500Hz Filter response 132
8.12 Time response of X- Y axis impulse after filtering 132
8.13 Average error percentages by size of Hankel matrix 134
8.14 Average error percentages by system order 135
8.15 The Magintude of the identified state-space $\bar{\mathbf{A}}$. 136
8.16 Identified responses of the experimental FESS data with ARX model. 138
8.17 Identified responses of the experimental FESS data with higher order ARX model. 139

9.1 The flexible electromechanical system - Two-Link-Arm Robot. 141
9.2 Schematic model of the test bed for mathematical model development. 142
9.3 The impulse responses of Two-Link-Arm robot with x-y coordinates 145
9.4 The impulse responses of the identified system with 1300×1300 Hankel matrix. 146
9.5 Magnitude of the identified state-space A matrix. 148

LIST OF TABLES

Table Number Page

1.1 Introduction of identification methods 5

3.1 Table for Comparison ERA and Fast ERA 36

4.1 Basic Functions and Notations in Probability and Statistics 39
4.2 Table for the equation (4.38) 49

5.1 Table for comparing $\mathbf{D}_{amp}$ and $\mathbf{K}$ matrix between derived and original values 63
5.2 Table for comparing $\mathbf{D}_{amp}$ and $\mathbf{K}$ matrix between derived and original values 66
5.3 The extracted eigenvalues of damping and stiffness matrix with ERA identified and reduced state-space model. 75
5.4 The extracted eigenvalues of damping and stiffness matrix with the Fast ERA identified and reduced state-space model. 80

7.1 Ratio of the Gerschgorin disk radius of the gyroscopic example 115
7.2 The system order depending on the model reduction methods. 115

8.1 Parameters of the flywheel energy storage system. 120
8.2 Error percentages table 123
8.3 Ratio of the Gerschgorin disk radius for the simulated data of FESS 125
8.4 The system order depending on the model reduction methods. 125
8.5 Error percentages with 4-order system and various Hankel matrix. 130
8.6 Error percentages - 700 RPM, 2 system order, and various size of Hankel matrix 133
8.7 Error percentages - 700 RPM, various system order 135
8.8 Error percentages - 700 RPM 137

9.1 Numerical values of Two-Link-Arm robot parameters 143
9.2 Characteristics of the shoulder and elbow actuator 144
9.3 The error percentage table for the identified TLA robot system. 147
9.4 Ratio of the Gerschgorin disc radius for Two-Link-Arm robot 148
9.5 The system order depending on the model reduction methods. 149

ACKNOWLEDGMENTS

I would like to appreciate to those who have helped me make my dissertation. Without their continuous help, I could not have finished my dissertation.

Especially, To my advisor, professor Brian C. Fabien, who helped me in terms of academical knowledge from my master degree to my doctoral degree, whenever I asked something. To professor Martin C. Berg, who supported my dissertation and gave me a good application. To professor Santoshi Devasia, who gave me a great feedback after my general exam. To professor Mehran Mesbahi, who encouraged me to finish my degree.

To my father, Soo-Ung, who has passed away after my master degree and gave me the greatest spirit. He is always with me. To my mother, Chan-Ha, who has made me comfortable and supported me during whole my life. To my sister, Eun-Joo, who has gave me many advices and inspirations. To my brother-in-law, Hyun-Suk, who has give me encouragement.
Also to my wife, Se-yeun Lee, who has been my greatest counselor although she had to study her own research. To my daughter, Christina Jae-Eun Suk, who has given me a lot of enjoyments.

To my colleagues, Jeff McCauley, Janet Ash, Richard Bankhead, and many college instructors, who gave me chances to work with them. My thanks to the Department of Mechanical Engineering in University of Washington for supporting my studies.

NOMENCLATURE

All notations will be presented again when they are used.

◦ Glossary

- ERA Eigensystem Realization Algorithm
- AR Auto-Regressive
- ARX Auto-Regressive eXternal inputs
- ARMAX Auto-Regressive Moving Average eXternal inputs
- FESS Flywheel Energy Storage System
- TLA Two-Link-Arm

◦ Basic Items

- $\mathfrak{R}$ = Set of all real numbers
- $\mathfrak{C}_{omplex}$ = Set of all complex numbers
- $\mathbf{I}_{N\times N}$ = N by N identity matrix
- $\mathbf{0}_{N\times M}$ = N by M zero matrix
- $\mathbf{M}$ = Mass matrix
- $\mathbf{D}_{amp}$ = Damping matrix

- $\mathbf{K}$ = Stiffness matrix
- $\mathbf{G}(\omega)$ = Gyroscopic matrix
- $\mathbf{H}_{sd}$ = Hankel matrix with s rows and d columns
- $\mathbf{M}_{arkov}(k)$ = Markov parameters
- $\mathcal{O}$ = Observability matrix
- $\mathcal{C}$ = Controllability matrix
- $W_{\mathcal{O}}$ = Observability Gramian
- $W_{\mathcal{C}}$ = Controllability Gramian
- $\mathcal{N}_{ull}(\cdot)$ = Null space of $\cdot$
- $\mathcal{R}_{ange}(\cdot)$ = Subspace of $\cdot$
- $\mathbf{rank}(\cdot)$ = Rank of $\cdot$
- $\mathrm{diag}[\cdots]$ = Diagonal matrix with elements
- exp = Exponential function
- $\mathbf{U}$ and $\mathbf{V}$ = Unitary matrix for singular value decomposition
- $\mathbf{\Sigma}$ = Singular value matrix for singular value decomposition
- $\mathbf{\Upsilon}$ = Eigenvector matrix for eigenvalue decomposition
- $\mathbf{\Lambda}$ = Eigenvalue matrix for eigenvalue decomposition

- i = Imaginary number

◦ Subscript

- N_{out} = Number of Outputs
- N_{in} = Number of Inputs
- s = Chosen Variable
- d = Chosen Variable[1]
- o = Theoretical original system
- c = Continuous time system
- ds = Discrete time system
- id = Identified system
- ie = Identified system with ERA

◦ Superscript

- $\dagger$= Pseudo-inverse
- T = Transpose matrix
- H = Hermitian matrix
- $\hat{\mathbb{A}} = e^{A_c T_s}$ [2]

---

[1] $N_{out} \times d = s \times N_{in}$

[2] T_s is a sampling time.

- $\hat{\mathbb{B}} = \int_0^{\Delta t} e^{A_c(\Delta t - \tau')} \mathrm{d}\tau' \; B_c$

- $(\,^{-}\,)$ = Complex variables(or matrices)

Chapter 1
INTRODUCTION

1.1 Introduction

As the modern technologies have been developed, many researchers require the identification of their very complicated dynamic systems to verify their theoretical modeling approaches and to design suitable control systems. During the last few decades, several identification methods have been developed in many fields. Many papers, i.e., [38], [47], [48], [80], [81] and books [32], [65], [67], [94] have been written to describe these identification techniques.

In this dissertation, several identification methods will be discussed in detail, however, we will focus on the problem of identifying a gyroscopic system using a complex variable transformation. All necessary conditions and requirements of the complex variable transformation, i.e., proofs, derivations, model order reduction, stability problems, and control theories, will be provided. In addition, a comparison of the different methods will be given with a simulation example and experimental examples for helping to understand how each identification method works. In this chapter, we will review the system identification research.

1.1.1 Fundamental System Model

For explaining concepts of a system model, let us review several terminologies[1] which will be frequently used in this dissertation.

A *system* is an object in which variables of different kinds interact and produce observable signals - called *outputs*. The system behavior can be influenced by signals called *inputs*. Some inputs which can be directly manipulated are called *control inputs*. The *system identification* techniques attempt to extract the system properties using recorded *control inputs* and *outputs*. Other inputs to system that can not be manipulated are called *disturbances*.

[1] All notation is based upon the Nomenclature.

It is assumed that the *disturbances* can not be measured directly.

In this dissertation, we consider the identification of $N_{out} \times N_{in}$ dynamic systems where N_{in} is the number of inputs and N_{out} is the number of outputs. The dynamics systems are generalized like the form

$$\begin{aligned} \mathbf{M}\ddot{q} + \mathbf{D}_{amp}\dot{q} + \mathbf{K}q &= \mathbf{E}f, \\ y &= \mathbf{c}q + \mathbf{d}f + v, \end{aligned} \tag{1.1}$$

as well as $N_{out} \times N_{in}$ gyroscopic systems

$$\begin{aligned} \mathbf{M}\ddot{q} + \mathbf{D}_{amp}\dot{q} + \mathbf{G}(\omega)\dot{q} + \mathbf{K}q &= \mathbf{E}f, \\ y &= \mathbf{c}q + \mathbf{d}f + v, \end{aligned} \tag{1.2}$$

where

$$\begin{gathered} q \in \Re^{N\times 1} : \text{ the states of system, } f \in \Re^{N_{in}\times 1} : \text{ controllable inputs,} \\ v \in \Re^{N_{out}\times 1} : \text{ disturbances, } \mathbf{M} \in \Re^{N\times N} : \text{ mass matrix,} \\ \mathbf{D}_{amp} \in \Re^{N\times N} : \text{ damping matrix, } \mathbf{K} \in \Re^{N\times N} : \text{ stiffness matrix,} \\ \mathbf{E} \in \Re^{N\times N_{in}} : \text{ input matrix,} \\ \mathbf{c} \in \Re^{N_{out}\times N} \text{ is a output matrix associated with system states,} \\ \mathbf{d} \in \Re^{N_{out}\times N_{in}} \text{ is a relationship matrix between inputs and outputs,} \\ \mathbf{G}(\omega) \in \Re^{N\times N} : \text{ gyroscopic matrix - skew-symmetric matrix,} \\ \omega : \text{ spinning speed.} \end{gathered}$$

Typically, control system designers describe these systems by first order differential equations, i.e., in the state-space form. To do so, consider the transformation

$$x = \begin{bmatrix} q \\ \dot{q} \end{bmatrix} \text{ and } f = u.$$

Then we obtain the following state-space model,

$$\begin{aligned} \dot{x} &= \mathbf{A}x + \mathbf{B}u, \\ y &= \mathbf{C}x + \mathbf{D}u + v. \end{aligned} \tag{1.3}$$

The system matrices are for equation (1.1),

$$\begin{aligned}
\mathbf{A} &= \begin{bmatrix} \mathbf{0}_{N\times N} & \mathbf{I}_{N\times N} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{D}_{amp} \end{bmatrix}, \\
\mathbf{B} &= \begin{bmatrix} \mathbf{0}_{N\times N_{in}} \\ \mathbf{M}^{-1}\mathbf{E} \end{bmatrix}, \\
\mathbf{C} &= \begin{bmatrix} \mathbf{c} & \mathbf{0} \end{bmatrix}, \\
\mathbf{D} &= \mathbf{d},
\end{aligned}$$

and for equation (1.2),

$$\begin{aligned}
\mathbf{A} &= \begin{bmatrix} \mathbf{0}_{N\times N} & \mathbf{I}_{N\times N} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}(\mathbf{D}_{amp} + \mathbf{G}(\omega)) \end{bmatrix}, \\
\mathbf{B} &= \begin{bmatrix} \mathbf{0}_{N\times N_{in}} \\ \mathbf{M}^{-1}\mathbf{E} \end{bmatrix}.
\end{aligned}$$

To identify the dynamics models for systems we perform the following tasks; (i) Construct a suitable analytical model for comparing with the identified model. (ii) Select an identification technique. (iii) Collect the data to be used by the identification method.

For the system identification methods considered in this dissertation, we need to store the output of the system at various times in the matrices

$$\mathbf{M}_{arkov}(k) = \left[y_1(kT_s) \quad y_2(kT_s) \quad \cdots \quad y_{N_{in}}(kT_s) \right], \tag{1.4}$$

where

$$k = 0,\ 1,\ 2,\ \cdots,\ \text{and } T_s = \text{sampling time},$$

and

$$\mathbf{H}_{qd}(k) = \begin{bmatrix} \mathbf{M}_{arkov}(k+1) & \mathbf{M}_{arkov}(k+2) & \cdots & \mathbf{M}_{arkov}(k+d) \\ \mathbf{M}_{arkov}(k+2) & \mathbf{M}_{arkov}(k+3) & \cdots & \mathbf{M}_{arkov}(k+d+1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}_{arkov}(k+q) & \mathbf{M}_{arkov}(k+q+1) & \cdots & \mathbf{M}_{arkov}(k+q+d-1) \end{bmatrix}, \tag{1.5}$$

where

$$\mathbf{M}_{arkov} : \text{Markov parameter, } \mathbf{H} : \text{Hankel Matrix,}$$
$$N_{out}\text{: a number of outputs, } N_{in}\text{: a number of inputs,}$$
$$s \text{ and } d : \text{chosen number with } N_{out} \times d = s \times N_{in}.$$

The matrix $\mathbf{M}_{arkov}$ is called the *Markov parameter*. The matrix $\mathbf{H}$ is called the *Hankel matrix*. The various identification techniques considered in this study use $\mathbf{H}$ to estimate $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$, $\mathbf{M}$, $\mathbf{K}$, $\mathbf{D}_{amp}$, $\mathbf{G}(\omega)$ and so on.

1.1.2 History of Identification

There are essentially two techniques for identifying a dynamic system of the types considered in this dissertation. The techniques are (i) Eigensystem Realization (ER) methods, and (ii) Auto-Regressive (AR) models. The ER methods are based on the concepts presented by Ho & Kalman [38] and was developed before the AR model. Many papers which are associated with Eigensystem Realization methods have been published since 1960s.

The ER methods use some linear algebra techniques i.e., the singular value decomposition [47], or the eigenvalue decomposition [80] to extract the system model. Because the ER methods extract system properties from the Hankel matrix, the identified responses exactly track the responses which are used to make the Hankel matrix. As a result, some identified system may be unstable (see [78]) because the ER methods use the Hankel matrix which can cause instability.

The AR models attempt to identify the dynamic system so that time series predictions, can be made. Since the 1970's this method has been rapidly developed by economists who are primarily interested time series predictions. This type of model has been utilized in other fields, e.g., biology [12, 43], geoscience [41], electromagnetic [70], speech signal [54], and environmental systems [84].

Figure 1.1 shows an overview of the ER and AR identification techniques. ER methods can be divided by basic decomposition methods and AR models are categorized by the structure of each model. With these categories, Table 1.1 shows the basic concepts and characteristics of each method which will be used in this dissertation.

The fundamental of the Eigensystem Realization Algorithm (ERA) are developed in the

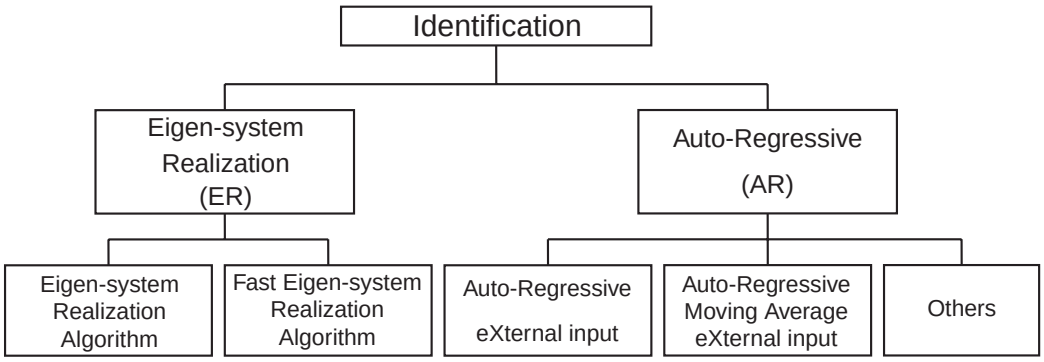

Figure 1.1: Macroscopic diagram of system identification methods

Table 1.1: Introduction of identification methods

Methods	Basic Concepts and Usages	Notes
ERA	Singular value decomposition	Linear Algebra
FastERA	Eigenvalue decomposition	Linear Algebra
ARMAX	Time series analysis	package in MATLAB, Probability Theory
ARX	MIMO system	Probability Theory

papers [38], [45], [47], and [48]. These papers show how to derive the algorithm in detail and apply it to various systems. The ERA identification technique basically uses Singular Value Decomposition (SVD) of the Hankel matrix to obtain the state-space model of system and extract properties of the system.

In Jer-Nan Juang's paper [46], he showed how to identify system with the Fast ERA method. Lee D. Peterson developed an eigensystem realization algorithm with eigenvalue decomposition instead of singular value decomposition in his paper [80]. He showed the advantages of the Fast ERA in terms of computer memory usages.

Ciann-Dong Yang *et al.* [101] and Alvin [3] explained a procedure of extracting the mass, damping, and stiffness matrices of the system. These references use a simple 3-DOF spring-mass-damping example to verify their theoretical approaches. Ciann-Dong Yang *et al.* [101] derived explicit expressions for $\mathbf{M}$, $\mathbf{D}_{amp}$, and $\mathbf{K}$ with a SVD technique. Alvin [3] developed procedures for identifying the system matrices using either (i) displacement measurement or (ii) velocity measurements or (iii) acceleration measurements.

Comparison studies of time domain identification methods are presented by K. A. Petsounis *et al* [81]. They compared four stochastic (Prediction Error Method, 2 Stage Least Squares, Linear Multi Stage, Instrumental Variable), and three deterministic (Least Squares, Prony, Eigensystem Realization Algorithm) time domain identification methods with simple simulation examples. Most of the stochastic methods are based on AR model and most of deterministic methods use fundamental linear algebra. In addition, they used simulation data instead of real data for comparison.

The papers [32], [65], [67] give the fundamentals of the AR models. AR models identify the system properties using the data in the matrix (1.4). AR models can be categorized into several models such as Auto-Regressive eXternal input (ARX), Auto-Regressive Moving Average eXternal input (ARMAX), Output-Error model, and Box-Jenkins model. The estimation of ARX is based on the least squares methods [7], [8]. Also, ARMAX (Auto-Regressive Moving Average eXternal input) is derived from the prediction error method [65]. ARMAX model is very useful for time series analysis. These categories also can be taken in terms of the structure of AR models.

For AR models most comparison studies are based on frequency response results such

as [19], [20], [78], [79]. These papers focused on frequency responses and AR models.

Finally, the books [27] and [30] had introduced the complex variable coordinate for studying the rotor-dynamics such as power generation, transmission, and fluid transport. Note that these texts focus on the vibration analysis of structure, not on system identification.

1.1.3 Problems

Even though many papers and books present various identification methods for the extraction of matrices, $\mathbf{M}$ (mass), $\mathbf{D}_{amp}$ (damping), and $\mathbf{K}$ (stiffness), none of these identification methods have been applied to gyroscopic systems. Furthermore, no techniques are available for constructing a $\mathbf{G}(\omega)$ (gyroscopic matrix) from experimental data. Also the complex variables coordinate has never used to identify the gyroscopic system.

Because the identification methods need a lot of data, the order of the system must be found with some algorithms. Many papers note that the ERA method uses the large Hankel matrix which can result in the identified system being unstable. Finally, some critical comparisons among identification methods with real experimental data are needed. The following items show problems which will be addressed in this dissertation.

1 Complex variable coordinated system:

This dissertation shows that many of the results from linear systems theory, based on the real state variables, can be applied to the systems described by complex state variables. Specifically (i) a initial condition response, (ii) a external input response, (iii) controllability, and (iv) observability. These results will be used in our proposed identification technique.

2 Identification methods with complex variable coordinates:

We will develop a system identification technique where the states are written using complex variables. The system identification method will be derived in detail and applied to examples - gyroscopic system and non-gyroscopic system.

3 System characteristics:

$\mathbf{M}$ (mass), $\mathbf{D}_{amp}$ (damping), and $\mathbf{K}$ (stiffness) matrices are extracted in many papers

[3], [20], [78], [101]. Here we show that the method developed in this dissertation. $\mathbf{G}(\omega)$ (gyroscopic) matrix can also be extracted.

4 System Order:

A new technique must be developed to determine system order when using the complex variable ERA method.

5 Comparison of methods:

Validation of the identification method must be performed using real and experimental data.

1.1.4 Objectives - What's New

Many papers - [48], [65], [80] - have introduced identification methods, however, few have applied many identification methods to same system for comparing the methods. A more specific objective is the fact that we can modify the system structure to get better response if we identify $\mathbf{M}$, $\mathbf{D}_{amp}$, $\mathbf{K}$ and $\mathbf{G}(\omega)$ matrices.

A new identification technique which uses the complex variable coordinate transformation will be presented for solving the following problems of identifying a gyroscopic system, i.e., (i) non-square system: A number of sensors and positions of sensors cause a system to be a non-square plant. Sometimes, only reduction of the number of states can make the non-square system square. The complex variables coordinate transformation make it possible to identify non-square plant system easily by reducing the number of states. (ii) skew-symmetric term in state-space model: The other advantage of the new identification technique is identifying gyroscopic systems because a skew-symmetric term by the gyroscopic matrix is eliminated. In order to use the complex variable coordinate transformation, this dissertation shows that the linear systems theory with real state variables can be applied to systems which are described by complex state variables.

The problem of system order is also important because we need the accurate system order for getting an efficient control system. When the ERA identification method is used, the identified responses may be unstable even though the original system is stable. Therefore,

identification using the ERA method must take account of the system order to ensure stability of the model.

This dissertation will consider several identification methods which are based on different concepts. This work will illustrate advantages and disadvantages of each method with simple mechanical example system which is simulated by computer. The computer simulation results make it easy to identify the system even when some white noise is added to simulation results. To illustrate the practical utility of the methods developed, some real data for identifying system and comparing identification methods will be used. In this dissertation, the identification methods are applied to real systems: a Two-Link-Arm Robot system and a flywheel energy storage system.

1.2 Approach

This dissertation is based on several methods for the identification of dynamic systems. These methods are presented in Chapters 2, 3, and 4. In Chapter 5, each identification method will be compared and evaluated with a simple mechanical system, i.e., 3 degree of freedom (DOF) spring-mass-damping system. For the 3 DOF system, simulated data by MATLAB will be used to evaluate the identification techniques. It has been observed that these identification methods work well with simulation data but sometimes fail when applied to experimental data. To validate the practical utility of these methods, actual systems are used for identification. In particular, we apply the methods to a flywheel energy storage device in Chapter 8 and a two-link-arm robot in Chapter 9.

1.2.1 Basic Experimental Concepts

Flywheel Energy Storage System

Flywheel Energy Storage (FES) systems have been developed for backup power supplies [4], [52], electromagnetic guns [17], seam-welding [1], electrical load leveling [36] and electric vehicles [44].

Even though the FES device has many applications, the FES device has still some difficulties. J. David Stenmier [91] showed some problems of developing the FES device in his

dissertation. Especially, he mentioned that a mathematical model of FES is necessary for developing a controller of FES. In addition, the FES device is an example of a gyroscopic system which makes the state-space model skew-symmetric. Due to the mathematical modeling problems and skew-symmetric terms in the state-space model, the FES device will be a good application for identification study. With results of some identification methods for FES, we will be able to develop a controller for FES.

Two-Link-Arm Robot

Two-Link-Arm robot is used to hold the parts within the designed work space of the robots. The industrial manufacturing also requires repeatability and accuracy of a robot very much.

Therefore, former researchers have developed Two-Link-Arm robot in terms of mathematical modeling, experimental results, and a suitable controller in their thesis [62], [82], [99]. In Lertpiriyasuwat's thesis, he had developed the mathematical modeling with the assumed modes method [73] and discussed unstable behavior for a consistently linearized model. The flexible Two-Link-Arm robot will be a good example for applying system identification methods.

Chapter 2

ERA - EIGENSYSTEM REALIZATION ALGORITHM

2.1 Introduction of ERA

The Eigensystem Realization Algorithm is a very effective method for identifying system using the Hankel matrix. Most parts of algorithm are related to linear algebra manipulations with the singular value decomposition of the Hankel matrix. As a result, all manipulations and decomposition methods of matrices must be understood to use it.

The main goal of this chapter is to introduce the Eigensystem Realization Algorithm (ERA) for the identification of discrete time linear system. In particular, suppose a discrete time linear system is given by

$$\begin{aligned} x_{k+1} &= \mathbb{A}x_k + \mathbb{B}u_k, \\ y_k &= \mathbf{C}x_k + \mathbf{D}u_k. \end{aligned} \tag{2.1}$$

where

$\mathbb{A}$ ($\in \mathfrak{R}^{2N\times 2N}$), $\mathbb{B}$ ($\in \mathfrak{R}^{2N\times N_{in}}$), $\mathbf{C}$ ($\in \mathfrak{R}^{N_{out}\times 2N}$), and $\mathbf{D}$ ($\in \mathfrak{R}^{N_{out}\times N_{in}}$)
are discrete time linear state-space model matrices,
u_k ($\in \mathfrak{R}^{N_{in}\times 1}$) is the input, x_k ($\in \mathfrak{R}^{2N\times 1}$) is the state at time $t = kT_s$,
and T_s is the sampling time.

Then using the responses of the system due to an impulse input, the ERA determines an equivalent linear system.

$$\begin{aligned} x_{k+1} &= \mathbb{A}_{id}x_k + \mathbb{B}_{id}u_k \\ y_k &= \mathbf{C}_{id}x_k + \mathbf{D}_{id} \end{aligned} \tag{2.2}$$

where $\mathbb{A}_{id}$, $\mathbb{B}_{id}$, $\mathbf{C}_{id}$, and $\mathbf{D}_{id}$ are the identified system model.

In the section 2.2, some basic concepts - Markov parameter and Hankel Matrix - are explained. The section 2.3 introduces the basic matrices of identification of Eigensystem Realization Algorithm in detail.

In this dissertation, the commercial program - MATLAB, will be used. Because MATLAB has a lots of function for linear algebra, we don't need to make many subroutine functions. In this chapter, we only consider the algorithm of the ERA. Chapter 5 will consider the cases, i.e., (i) no noise in the system data and (ii) noise in the system data.

2.2 Basic Concepts

Before we identify systems, we need to store output data for handling data easily. The Hankel matrix is used to store outputs of system for the ERA method. The Hankel matrix consists of the Markov parameters [47], [48]. Those concepts are a fundamental rule for arranging output data. The important point for the Hankel matrix is the fact that outputs are the responses of impulse inputs from zero initial conditions.

In addition, some matrices i.e., controllability matrix, observability matrix [22], and singular value matrices [93] are used. In the derivation of the ERA method, these concepts which are related to the linear algebra are used.

Markov parameter and Hankel matrix

Assume that the dimension of system is $N_{out} \times N_{in}$, where N_{in} is the number of control inputs and N_{out} is the number of outputs. For calculating the ith response for unit impulse inputs, the continuous time state-space form (1.3) can be manipulated as follows.

The continuous time State-Space matrix form for general N_{in} inputs, N_{out} outputs, and N states system will be expressed as,

$$\begin{aligned} \dot{x} &= \mathbf{A}_c x(t) + \mathbf{B}_c u(t), \\ y(t) &= \mathbf{C}_c x(t) + \mathbf{D}_c u(t). \end{aligned} \tag{2.3}$$

where the subscript 'c' is used for continuous time state-space matrices,

$$\begin{aligned}
\mathbf{A}_c &= \begin{bmatrix} \mathbf{0}_{N\times N} & \mathbf{I}_{N\times N} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{D}_{amp} \end{bmatrix}, \\
\mathbf{B}_c &= \begin{bmatrix} \mathbf{0}_{N\times N_{in}} \\ \mathbf{M}^{-1}\mathbf{I}_{N\times N_{in}} \end{bmatrix}, \\
\mathbf{C}_c &= \begin{bmatrix} \mathbf{I}_{N_{out}\times N} & \mathbf{0}_{N_{out}\times N} \end{bmatrix}, \\
\mathbf{D}_c &= [\mathbf{0}_{N_{out}\times N_{in}}].
\end{aligned}$$

It is impossible to store all continuous outputs in terms of a usage of memory. Therefore, outputs $y(t)$ must be expressed by discrete time form with sampling time T_s for making outputs a Markov parameter. Then the equations (2.3) becomes.

$$\begin{aligned}
\dot{x} &= \mathbf{A}_c x(t) + \mathbf{B}_c u(t), \\
y(kT_s) &= \mathbf{C}_c x(kT_s),
\end{aligned} \tag{2.4}$$

where $k = 0,\ 1,\ 2,\ \cdots$. For calculating an arbitrary output [22], the kth response can be written with discrete time based outputs (2.4) and matrix manipulation

$$y(kT) = \mathbf{C}_c \hat{\mathbb{A}}^{(k-1)} \hat{\mathbb{B}}, \tag{2.5}$$

where

$$\begin{aligned}
\hat{\mathbb{A}} &\equiv e^{\mathbf{A}_c T_s}, \\
\hat{\mathbb{B}} &\equiv \int_0^{\Delta t} e^{\mathbf{A}_c(\Delta t - \tau')} d\tau' \mathbf{B}_c.
\end{aligned}$$

Using the equation (2.5), we can define the Markov parameter as

$$\mathbf{M}_{arkov}(k) = \begin{bmatrix} y_1(kT_s) & y_2(kT_s) & \cdots & y_{N_{in}}(kT_s) \end{bmatrix}, \tag{2.6}$$

where $k = 0,\ 1,\ 2,\ \cdots$, and $\mathbf{M}_{arkov}(k) \in \mathfrak{R}^{N_{out}\times N_{in}}$.

The Hankel matrix which consists of the Markov parameters is defined as

$$\mathbf{H}_{qd}(k) = \begin{bmatrix} \mathbf{M}_{arkov}(k+1) & \mathbf{M}_{arkov}(k+2) & \cdots & \mathbf{M}_{arkov}(k+d) \\ \mathbf{M}_{arkov}(k+2) & \mathbf{M}_{arkov}(k+3) & \cdots & \mathbf{M}_{arkov}(k+d+1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}_{arkov}(k+q) & \mathbf{M}_{arkov}(k+q+1) & \cdots & \mathbf{M}_{arkov}(k+q+d-1) \end{bmatrix}, \tag{2.7}$$

where

$$N_{out} \text{ is the number of outputs, } q \text{ and } d \text{ are chosen numbers like}$$
$$N_{out} \times q = d \times \text{number of inputs}(= N_{in}),$$
$$\mathbf{H}_{qd}(k) \in \Re^{(N_{out} \times q) \times (N_{in} \times d)}.$$

Singular Value Decomposition

The ERA method uses the singular value decomposition of the Hankel matrix so that we need to review the definition of the singular value decomposition (SVD). According to numerical linear algebra texts [57], [83], [93], [96], the singular value decomposition[1] can be used for 1) a spectral norm of a matrix, 2) image processing, 3) calculation of effective rank of matrix, 4) a polar decomposition, 5) solution for least square problem, 6) a pseudo-inverses, 7) condition number of matrix. The singular value decomposition of the matrix $\mathbf{H} \in \Re^{q \times d}$ is defined

$$\mathbf{H} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T, \tag{2.8}$$

where

$$\text{The superscript of } T \text{ denotes the transpose matrix,}$$
$$\mathbf{U} = \mathbf{U}(1:q, 1:N) \text{ is an unitary matrix,}$$
$$\mathbf{\Sigma} = \mathbf{\Sigma}(1:N, 1:N) \text{ is a singular value matrix - diagonal,}$$
$$\mathbf{V} = \mathbf{V}(1:d, 1:N) \text{ is an unitary matrix.}$$

The unitary matrix is defined, if its hermitian conjugate[2] equals its inverse. And the matrix $\mathbf{\Sigma}$ is a diagonalized matrix with singular values of the matrix $\mathbf{H}$.

---

[1] More detail Properties of the SVD will be presented in chapter 5 and 7 for reducing system order.

[2] Transpose matrix of complex-conjugate matrix is called hermitian conjugate.

Controllability and Observability Matrix

For the discrete time system like equations (2.1) and (2.4), the controllability matrix, $\mathcal{C}$, and the observability matrix $\mathcal{O}$ are defined as

$$\mathcal{C} = \left[\hat{\mathbb{B}}, \hat{\mathbb{A}}\hat{\mathbb{B}}, \cdots, \hat{\mathbb{A}}^{(N-1)}\hat{\mathbb{B}}\right], \tag{2.9}$$

$$\mathcal{O} = \begin{bmatrix} \mathbf{C}_c \\ \mathbf{C}_c\hat{\mathbb{A}} \\ \vdots \\ \mathbf{C}_c\hat{\mathbb{A}}^{(N-1)} \end{bmatrix}. \tag{2.10}$$

$\hat{\mathbb{A}} \in \mathfrak{R}^{N\times N}$, $\hat{\mathbb{B}} \in \mathfrak{R}^{N\times N_{in}}$, $\mathbf{C} \in \mathfrak{R}^{N_{out}\times N}$,

N is a number of the states,

N_{in} is a number of inputs,

and N_{out} is a number of outputs.

Note from (2.4) and(2.5) that $\hat{\mathbb{A}}$, $\hat{\mathbb{B}}$, and $\mathbf{C}_c$ are the discrete time equivalents of the continuous time system.

Those matrices are able to check the controllability and the observability of the given system. If the controllability matrix (or the observability matrix) has the full rank, the system is controllable (or observable). The detail derivations are in Appendix A.

2.3 Identification

In the previous section, basic concepts for the ERA method are introduced briefly e.g., the Markov Parameter, the Hankel Matrix, the singular value decomposition method, the controllability matrix, and the observability matrix. In this section, the procedure of identifying system will be derived from given data with those concepts.

The first step in the identification procedure is the storage of given output data with the Markov parameter. The Markov parameters can be obtained by equations (2.6) and (2.5)

like

$$\begin{aligned}
\mathbf{M}_{arkov}(1) &= \mathbf{C}_c\hat{\mathbb{A}}^0\hat{\mathbb{B}} = \mathbf{C}_c\hat{\mathbb{B}}, \\
\mathbf{M}_{arkov}(2) &= \mathbf{C}_c\hat{\mathbb{A}}^1\hat{\mathbb{B}} = \mathbf{C}_c\hat{\mathbb{A}}\hat{\mathbb{B}}, \\
\vdots \quad & \quad \vdots \\
\mathbf{M}_{arkov}(k) &= \mathbf{C}_c\hat{\mathbb{A}}^{k-1}\hat{\mathbb{B}} = \mathbf{C}_c\hat{\mathbb{A}}^{k}\hat{\mathbb{B}}.
\end{aligned} \tag{2.11}$$

Plug equations (2.11) into the Hankel matrix (2.7) as

$$\begin{aligned}
\mathbf{H}_{qd}(0) &= \begin{bmatrix} \mathbf{M}_{arkov}(1) & \mathbf{M}_{arkov}(2) & \cdots & \mathbf{M}_{arkov}(d) \\ \mathbf{M}_{arkov}(2) & \mathbf{M}_{arkov}(3) & \cdots & \mathbf{M}_{arkov}(d+1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}_{arkov}(q) & \mathbf{M}_{arkov}(q+1) & \cdots & \mathbf{M}_{arkov}(q+d-1) \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{C}_c\hat{\mathbb{B}} & \mathbf{C}_c\hat{\mathbb{A}}\hat{\mathbb{B}} & \cdots & \mathbf{C}_c\hat{\mathbb{A}}^{(d-1)}\hat{\mathbb{B}} \\ \mathbf{C}_c\hat{\mathbb{A}}\hat{\mathbb{B}} & \mathbf{C}_c\hat{\mathbb{A}}^2\hat{\mathbb{B}} & \cdots & \mathbf{C}_c\hat{\mathbb{A}}^{(d)}\hat{\mathbb{B}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_c\hat{\mathbb{A}}^{(q-1)}\hat{\mathbb{B}} & \mathbf{C}_c\hat{\mathbb{A}}^{(q)}\hat{\mathbb{B}} & \cdots & \mathbf{C}_c\hat{\mathbb{A}}^{(q+d-2)}\hat{\mathbb{B}} \end{bmatrix}.
\end{aligned} \tag{2.12}$$

Multiplying the observability matrix, $\mathcal{O}$ and the controllability matrix $\mathcal{C}$ gives us

$$\begin{aligned}
\mathcal{O}_q\mathcal{C}_d &= \begin{bmatrix} \mathbf{C}_c \\ \mathbf{C}_c\hat{\mathbb{A}} \\ \vdots \\ \mathbf{C}_c\hat{\mathbb{A}}^{(q-1)} \end{bmatrix} \left[\hat{\mathbb{B}}, \hat{\mathbb{A}}\hat{\mathbb{B}}, \cdots, \hat{\mathbb{A}}^{(d-1)}\hat{\mathbb{B}}\right] \\
&= \begin{bmatrix} \mathbf{C}_c\hat{\mathbb{B}} & \mathbf{C}_c\hat{\mathbb{A}}\hat{\mathbb{B}} & \cdots & \mathbf{C}_c\hat{\mathbb{A}}^{(d-1)}\hat{\mathbb{B}} \\ \mathbf{C}_c\hat{\mathbb{A}}\hat{\mathbb{B}} & \mathbf{C}_c\hat{\mathbb{A}}^2\hat{\mathbb{B}} & \cdots & \mathbf{C}_c\hat{\mathbb{A}}^{(d)}\hat{\mathbb{B}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_c\hat{\mathbb{A}}^{(q-1)}\hat{\mathbb{B}} & \mathbf{C}_c\hat{\mathbb{A}}^{(q)}\hat{\mathbb{B}} & \cdots & \mathbf{C}_c\hat{\mathbb{A}}^{(q+d-2)}\hat{\mathbb{B}} \end{bmatrix}.
\end{aligned} \tag{2.13}$$

The Hankel matrix, $\mathbf{H}_{qd}(0)$ - (2.12), is equal to the multiplication of the observability matrix and the controllability matrix - (2.13).

$$\mathbf{H}_{qd}(0) = \mathcal{O}_q\mathcal{C}_d. \tag{2.14}$$

With same logic of getting the equation (2.14), the Markov parameters in the Hankel matrix, $\mathbf{H}_{qd}(1)$, are substituted as

$$\begin{aligned}
\mathbf{H}_{qd}(1) &= \begin{bmatrix} \mathbf{M}_{arkov}(2) & \mathbf{M}_{arkov}(3) & \cdots & \mathbf{M}_{arkov}(d+1) \\ \mathbf{M}_{arkov}(3) & \mathbf{M}_{arkov}(4) & \cdots & \mathbf{M}_{arkov}(d+2) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}_{arkov}(q+1) & \mathbf{M}_{arkov}(q+2) & \cdots & \mathbf{M}_{arkov}(q+d+1) \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{C}_c\hat{\mathbb{A}}\hat{\mathbb{B}} & \mathbf{C}_c\hat{\mathbb{A}}^1\hat{\mathbb{B}} & \cdots & \mathbf{C}_c\hat{\mathbb{A}}^{(d)}\hat{\mathbb{B}} \\ \mathbf{C}_c\hat{\mathbb{A}}^2\hat{\mathbb{B}} & \mathbf{C}_c\hat{\mathbb{A}}^3\hat{\mathbb{B}} & \cdots & \mathbf{C}_c\hat{\mathbb{A}}^{(d+1)}\hat{\mathbb{B}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_c\hat{\mathbb{A}}^{(q)}\hat{\mathbb{B}} & \mathbf{C}_c\hat{\mathbb{A}}^{(q+1)}\hat{\mathbb{B}} & \cdots & \mathbf{C}_c\hat{\mathbb{A}}^{(q+d)}\hat{\mathbb{B}} \end{bmatrix}. \qquad (2.15)
\end{aligned}$$

Multiplying the observability matrix, $\mathcal{O}$, the state-space $\hat{\mathbb{A}}$, and the controllability matrix $\mathcal{C}$ gives us

$$\begin{aligned}
\mathcal{O}_q\hat{\mathbb{A}}\mathcal{C}_d &= \begin{bmatrix} \mathbf{C}_c \\ \mathbf{C}_c\hat{\mathbb{A}} \\ \vdots \\ \mathbf{C}_c\hat{\mathbb{A}}^{(q-1)} \end{bmatrix} \hat{\mathbb{A}} \left[\hat{\mathbb{B}}, \hat{\mathbb{A}}\hat{\mathbb{B}}, \cdots, \hat{\mathbb{A}}^{(d-1)}\hat{\mathbb{B}}\right] \\
&= \begin{bmatrix} \mathbf{C}_c\hat{\mathbb{A}}\hat{\mathbb{B}} & \mathbf{C}_c\hat{\mathbb{A}}^2\hat{\mathbb{B}} & \cdots & \mathbf{C}_c\hat{\mathbb{A}}^d\hat{\mathbb{B}} \\ \mathbf{C}_c\hat{\mathbb{A}}^2\hat{\mathbb{B}} & \mathbf{C}_c\hat{\mathbb{A}}^3\hat{\mathbb{B}} & \cdots & \mathbf{C}_c\hat{\mathbb{A}}^{(d+1)}\hat{\mathbb{B}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_c\hat{\mathbb{A}}^q\hat{\mathbb{B}} & \mathbf{C}_c\hat{\mathbb{A}}^{(q+1)}\hat{\mathbb{B}} & \cdots & \mathbf{C}_c\hat{\mathbb{A}}^{(q+d-1)}\hat{\mathbb{B}} \end{bmatrix}. \qquad (2.16)
\end{aligned}$$

The Hankel matrix, $\mathbf{H}_{qd}(1)$ - (2.15), is equal to the multiplication of the observability matrix, the state-space $\hat{\mathbb{A}}$, and the controllability matrix - (2.16).

$$\mathbf{H}_{qd}(1) = \mathcal{O}_q\hat{\mathbb{A}}\mathcal{C}_d. \qquad (2.17)$$

Equations (2.14) and (2.17) are starting equations for the ERA method. For getting some relationships from equation (2.14), apply the singular value decomposition to the Hankel matrix - $\mathbf{H}_{qd}(0)$

$$\begin{aligned}
\mathbf{H}_{qd}(0) &= \mathbf{U}_q\mathbf{\Sigma}_{qd}\mathbf{V}_d^T \qquad (2.18) \\
&= \mathcal{O}_q\mathcal{C}_d.
\end{aligned}$$

With equations (2.14) and (2.18), controllability, observability, pseudo-inverse of controllability, and pseudo-inverse of observability matrix can be obtained like

$$\mathcal{O}_q = \mathbf{U}_q \, \mathbf{\Sigma}_{qd}^{(1/2)} \rightarrow \mathcal{O}_q^{\dagger} = \mathbf{\Sigma}_{qd}^{(-1/2)} \, \mathbf{U}_q^T, \tag{2.19}$$

$$\mathcal{C}_d = \mathbf{\Sigma}_{qd}^{(1/2)} \, \mathbf{V}_d^T \rightarrow \mathcal{C}_d^{\dagger} = \mathbf{V}_d \, \mathbf{\Sigma}_{qd}^{(-1/2)}. \tag{2.20}$$

where the superscript of † denotes the pseudo-inverse.

The other starting equation (2.17) is a least-squares problem formation for $\hat{\mathbb{A}}$. With the pseudo-inverse of controllability and observability matrices, the least-squares problem can be solved for $\hat{\mathbb{A}}$.

$$\hat{\mathbb{A}} = \mathcal{O}_q^{\dagger} \mathbf{H}_{qd}(1) \mathcal{C}_d^{\dagger}. \tag{2.21}$$

From the equations (2.14) and (2.18), the pseudo-inverse of controllability and observability matrices have already been obtained. Put equations (2.20) and (2.19) into the equation (2.21), then the identified discrete state-space $\hat{\mathbb{A}}$ can be written

$$\hat{\mathbb{A}}_{id} = \mathcal{O}_q^{\dagger} \mathbf{H}_{qd}(1) \mathcal{C}_d^{\dagger} = \mathbf{\Sigma}_{qd}^{(-1/2)} \, \mathbf{U}_q^T \mathbf{H}_{qd}(1) \mathbf{V}_d \, \mathbf{\Sigma}_{qd}^{(-1/2)}. \tag{2.22}$$

For identifying the $\hat{\mathbb{B}}$ matrix of state-space model, use the controllability matrix $\mathcal{C}$ (2.9). The $\hat{\mathbb{B}}$ matrix must equal the first N_{in} columns of the controllability matrix. Therefore, the $\hat{\mathbb{B}}$ matrix can be obtained by multiplying calculated controllability matrix (2.20) to a matrix which eliminates the rest of the columns. Obviously, a number of system order, N_{order}, must be chosen by a proper algorithm[3].

$$\hat{\mathbb{B}}_{id} = \mathcal{C} \begin{bmatrix} \mathbf{I}_{N_{order} \times N_{in}} \\ \mathbf{0}_{(d-N_{order}) \times N_{in}} \end{bmatrix} = \mathbf{\Sigma}_{qd}^{(1/2)} \, \mathbf{V}_d^T \begin{bmatrix} \mathbf{I}_{N_{order} \times N_{in}} \\ \mathbf{0}_{(d-N_{order}) \times N_{in}} \end{bmatrix}. \tag{2.23}$$

With same logic for identifying $\hat{\mathbb{B}}$, the state-space identified $\mathbf{C}$ matrix can be obtained, i.e., in the observability matrix $\mathcal{O}$ (2.10), the state-space $\mathbf{C}$ matrix have to equal the first N_{out} rows of the observability matrix. Therefore, multiplying the calculated observability matrix (2.19) to a matrix which will get rid of rest rows makes the state-space $\mathbf{C}$ matrix,

$$\mathbf{C}_{id} = \begin{bmatrix} \mathbf{I}_{N_{out} \times N_{order}} & \mathbf{0}_{N_{out} \times (q-N_{order})} \end{bmatrix} \mathcal{O} = \begin{bmatrix} \mathbf{I}_{N_{out} \times N_{order}} & \mathbf{0}_{N_{out} \times (q-N_{order})} \end{bmatrix} \mathbf{U}_q \, \mathbf{\Sigma}_{qd}^{(1/2)}. \tag{2.24}$$

[3]Model reduction algorithm will be introduced in Chapter 7.

The last step for the ERA is getting the state-space $\mathbf{D}$ matrix. The state-space $\mathbf{D}$ matrix must equal $\mathbf{M}_{arkov}(0)$ because the $\mathbf{D}$ matrix gives us information about the direct relationship between inputs and outputs.

Up to now, the Hankel Matrix is $q \times d$ matrix for generalizing. Usually the Hankel Matrix will be made by square matrix for getting a state-space matrix easily. Then, the state-space matrix $\hat{\mathbb{A}}_{id}$, $\hat{\mathbb{B}}_{id}$, $\mathbf{C}_{id}$ and $\mathbf{D}_{id}$ are obtained like

$$\begin{aligned}
\hat{\mathbb{A}}_{id} &= \mathbf{\Sigma}^{(-1/2)}\mathbf{U}^T\mathbf{H}(1)\mathbf{V}\mathbf{\Sigma}^{(-1/2)}, \\
\hat{\mathbb{B}}_{id} &= \mathbf{\Sigma}^{(1/2)}\mathbf{V}^T \begin{bmatrix} \mathbf{I}_{N_{order}\times N_{in}} \\ \mathbf{0}_{(d-N_{order})\times N_{in}} \end{bmatrix}, \\
\mathbf{C}_{id} &= \begin{bmatrix} \mathbf{I}_{N_{out}\times N_{order}} & \mathbf{0}_{N_{out}\times(q-N_{order})} \end{bmatrix} \mathbf{U}\mathbf{\Sigma}^{(1/2)}, \\
\mathbf{D}_{id} &= \mathbf{M}_{arkov}(0).
\end{aligned}$$

For summarizing the ERA method, the following steps can be itemized.

(1) Set up the Hankel matrix using the Markov parameter:

$$\mathbf{H}_{qd}(0) = \begin{bmatrix} \mathbf{M}_{arkov}(1) & \mathbf{M}_{arkov}(2) & \cdots & \mathbf{M}_{arkov}(d) \\ \mathbf{M}_{arkov}(2) & \mathbf{M}_{arkov}(3) & \cdots & \mathbf{M}_{arkov}(d+1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}_{arkov}(q) & \mathbf{M}_{arkov}(q+1) & \cdots & \mathbf{M}_{arkov}(q+d-1) \end{bmatrix}.$$

(2) Decompose $\mathbf{H}_{qd}(0)$ into its singular value factors:

$$\mathbf{H}_{qd}(0) = \mathbf{U}_q\mathbf{\Sigma}_{qd}\mathbf{V}_d^T.$$

(3) From $\mathbf{H}_{qd}(1)$:

$$\mathbf{H}_{qd}(1) = \begin{bmatrix} \mathbf{M}_{arkov}(1) & \mathbf{M}_{arkov}(2) & \cdots & \mathbf{M}_{arkov}(d) \\ \mathbf{M}_{arkov}(2) & \mathbf{M}_{arkov}(3) & \cdots & \mathbf{M}_{arkov}(d+1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}_{arkov}(q) & \mathbf{M}_{arkov}(q+1) & \cdots & \mathbf{M}_{arkov}(q+d+1) \end{bmatrix}.$$

(4) Solve for $\hat{\mathbb{A}}$ using the controllability matrix $\mathcal{C}$ and the observability $\mathcal{O}$:

$$\hat{\mathbb{A}}_{id} = \mathbf{\Sigma}^{(-1/2)}\mathbf{U}^T\mathbf{H}_{qd}(1)\mathbf{V}\mathbf{\Sigma}^{(-1/2)}.$$

(5) Solve for $\hat{\mathbb{B}}$:

$$\begin{aligned}\hat{\mathbb{B}}_{id} &= \mathbf{\Sigma}_{qd}^{(1/2)}\,\mathbf{V}_d^{\mathbb{T}}\begin{bmatrix}\mathbf{I}_{N_{order}\times N_{in}}\\ \mathbf{0}_{(d-N_{order})\times N_{in}}\end{bmatrix}\\ &= \mathbf{\Sigma}_{qd}^{(1/2)}\,\mathbf{V}_d^{T}(1:N_{order},1:N_{in}).\end{aligned}$$

(6) Solve for $\mathbf{C}$:

$$\begin{aligned}\mathbf{C}_{id} &= \begin{bmatrix}\mathbf{I}_{N_{out}\times N_{order}} & \mathbf{0}_{N_{out}\times(q-N_{order})}\end{bmatrix}\mathbf{U}_q\,\mathbf{\Sigma}_{qd}^{(1/2)}\\ &= \mathbf{U}_q(1:N_{out},1:N_{order})\,\mathbf{\Sigma}_{qd}^{(1/2)}.\end{aligned}$$

(7) Solve for $\mathbf{D}$:

$$\mathbf{D}_{id} = \mathbf{M}_{arkov}(0).$$

2.4 Extracting M, D_{amp}, and K Matrices

Even though the state-space model is identified completely with the equations (2.25), $\mathbf{M}$ (mass), $\mathbf{D}_{amp}$ (damping), and $\mathbf{K}$ (stiffness) matrices are still unknown. The ERA identification method is able to extract $\mathbf{M}$, $\mathbf{D}_{amp}$ and $\mathbf{K}$ matrices with similarity transformation matrix in the paper [101]. Alvin [3] developed procedures for identifying the system matrices with each measurement condition e.g., displacement, velocity, or acceleration measurement, focusing on the eigenvalues of $\mathbf{D}_{amp}$ and $\mathbf{K}$.

For extracting $\mathbf{M}$, $\mathbf{D}_{amp}$ and $\mathbf{K}$ matrices, we assume that the order of state-space model is exactly correct with that of real system. The assumption means that the dimension of continuous time state-space A_c matrix must be $2N \times 2N$ if a system has N inputs, N outputs and N states. In addition, the assumption is associated with the correction of system order[4] because the extraction with incorrect model order will lead unsuitable information of system.

[4]The reduction of system order will be presented later in chapter 7.

This extraction method uses the similarity transformation matrix P between ideal real ($\mathbf{A}_c$, $\mathbf{B}_c$, $\mathbf{C}_c$) and identified ($\hat{\mathbb{A}}_{id}$, $\hat{\mathbb{B}}_{id}$, $\mathbf{C}_{id}$) state-space model. Then a similarity transformation matrix P satisfies

$$\mathbf{A}_c = P\hat{\mathbb{A}}_{id}P^{-1}, \tag{2.25}$$

$$\mathbf{B}_c = P\hat{\mathbb{B}}_{id}, \tag{2.26}$$

$$\mathbf{C}_c = \mathbf{C}_{id}P^{-1}. \tag{2.27}$$

The first step for extracting $\mathbf{M}$, $\mathbf{D}_{amp}$ and $\mathbf{K}$ matrices is from the equation (2.27). The matrix P can be expressed like

$$P = \begin{bmatrix} p_{up} \\ p_{low} \end{bmatrix}, \tag{2.28}$$

where

$$p_{up},\ p_{low} \in \mathfrak{R}^{N\times 2N}.$$

The matrix P in the equation (2.27) must be multiplied to the state-space matrix $\mathbf{C}_c$ like

$$\begin{gathered} \mathbf{C}_c = \mathbf{C}_{id}P^{-1} \rightarrow \mathbf{C}_c P = \mathbf{C}_{id}, \\ [\mathbf{I}_{N\times N} \quad \mathbf{0}_{N\times N}] \begin{bmatrix} p_{up} \\ p_{low} \end{bmatrix} = \mathbf{C}_{id}, \\ \text{From 1st } N \text{ rows} \Rightarrow p_{up} = \mathbf{C}_{id}. \end{gathered} \tag{2.29}$$

We can obtain the upper part of the similarity transformation matrix P from the equation (2.29). Then let us take a look the equation (2.25).

$$\mathbf{A}_c = P\hat{\mathbb{A}}_{id}P^{-1} \rightarrow \mathbf{A}_c P = P\hat{\mathbb{A}}_{id}. \tag{2.30}$$

The state-space matrix $\mathbf{A}_c$, (2.3), consists of the zero, identity, mass, damping, and stiffness matrices. Then do multiplication of the equation (2.30).

$$\begin{gathered} \begin{bmatrix} \mathbf{0}_{N\times N} & \mathbf{I}_{N\times N} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{D}_{amp} \end{bmatrix} \begin{bmatrix} \mathbf{C}_{id} \\ p_{low} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_{id} \\ p_{low} \end{bmatrix} \hat{\mathbb{A}}_{id}, \\ \begin{bmatrix} p_{low} \\ -\mathbf{M}^{-1}\mathbf{K}\mathbf{C}_{id} - \mathbf{M}^{-1}\mathbf{D}_{amp}p_{low} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_{id}\hat{\mathbb{A}}_{id} \\ p_{low}\hat{\mathbb{A}}_{id} \end{bmatrix}. \end{gathered} \tag{2.31}$$

The similarity transformation matrix P can be obtained completely from the first N rows of equation (2.31),

$$\begin{aligned} p_{low} &= \mathbf{C}_{id}\hat{\mathbb{A}}_{id} \\ P &= \begin{bmatrix} \mathbf{C}_{id} \\ \mathbf{C}_{id}\hat{\mathbb{A}}_{id} \end{bmatrix} \end{aligned} \tag{2.32}$$

The similarity transformation matrix P is calculated by manipulating the identified state-space model matrices. Then the matrix P and several linear algebra calculation make it possible to extract the properties of system. The known matrix P, (2.32), can multiply the equation (2.26) like

$$\mathbf{B}_c = P\hat{\mathbb{B}}_{id} = \begin{bmatrix} \mathbf{C}_{id} \\ \mathbf{C}_{id}\hat{\mathbb{A}}_{id} \end{bmatrix} \hat{\mathbf{B}}_{id} = \begin{bmatrix} \overbrace{\mathbf{C}_{id}\hat{\mathbb{B}}_{id}}^{=0} \\ \mathbf{C}_{id}\hat{\mathbb{A}}_{id}\hat{\mathbb{B}}_{id} \end{bmatrix}. \tag{2.33}$$

The last N rows of the equation (2.33) and the matrix $\mathbf{B}_c$ (2.3) can extract the mass matrix like

$$\mathbf{M}_{id} = (\mathbf{C}_{id}\hat{\mathbb{A}}_{id}\hat{\mathbb{B}}_{id})^{-1}. \tag{2.34}$$

The first N rows of the equation (2.31) are used for making the matrix P complete. The last N rows of the equation (2.31) will be used for the $\mathbf{D}_{amp}$ and $\mathbf{K}$ matrices.

$$-\mathbf{M}^{-1}\mathbf{K}\mathbf{C}_{id} - \mathbf{M}^{-1}\mathbf{D}_{amp}\mathbf{C}_{id}\hat{\mathbb{A}}_{id} = \mathbf{C}_{id}\hat{\mathbb{A}}_{id}^2.$$

Multiply $-\mathbf{M}$ on both sides,

$$\begin{aligned} \mathbf{K}\mathbf{C}_{id} + \mathbf{D}_{amp}\mathbf{C}_{id}\hat{\mathbb{A}}_{id} &= -\mathbf{M}\mathbf{C}_{id}\hat{\mathbb{A}}_{id}^2, \\ \begin{bmatrix} \mathbf{K} & \mathbf{D}_{amp} \end{bmatrix} \begin{bmatrix} \mathbf{C}_{id} \\ \mathbf{C}_{id}\hat{\mathbb{A}}_{id} \end{bmatrix} &= -\mathbf{M}\mathbf{C}_{id}\hat{\mathbb{A}}_{id}^2, \\ \begin{bmatrix} \mathbf{K}_{id} & \mathbf{D}_{amp_{id}} \end{bmatrix} &= -\mathbf{M}\mathbf{C}_{id}\hat{\mathbb{A}}_{id}^2 \begin{bmatrix} \mathbf{C}_{id} \\ \mathbf{C}_{id}\hat{\mathbb{A}}_{id} \end{bmatrix}^{-1}. \end{aligned} \tag{2.35}$$

The equations (2.34), (2.35) show that the identified mass $\mathbf{M}_{id}$, damping $\mathbf{D}_{amp_{id}}$, and stiffness $\mathbf{K}_{id}$ matrices can consist of the identified state-space model.

Alvin [3] had suggested another extraction procedure which focuses on the eigenvalue of the damping and stiffness matrices. This realization method requires damping ratio (ζ) and natural frequencies (ω_n) of system. To do so, he used a new transformation matrix P by which $\hat{\mathbb{A}}_{id}$, $2N \times 2N$ matrix, can be diagonalized with a nonsingular basis transformation. Let us review the another method.

First of all, the discrete identified state-space model is

$$\begin{aligned} x(k+1) &= \hat{\mathbb{A}}_{id}x(k) + \hat{\mathbb{B}}_{id}u(k), \\ y(k) &= \mathbf{C}_{id}x(k) + \mathbf{D}_{id}u(k). \end{aligned} \tag{2.36}$$

A new transformation matrix P can diagonalize the identified state-space $\hat{A}_{id}$ matrix, which means that the transformation matrix P is considered like

$$P^{-1}\hat{\mathbb{A}}_{id}P = \Lambda = diag(\bar{\lambda_1}, \bar{\lambda_2}, \cdots, \bar{\lambda_n}). \tag{2.37}$$

Any set $[P^{-1}\hat{\mathbb{A}}_{id}P,\ P^{-1}\hat{\mathbb{B}}_{id},\ \mathbf{C}_{id}P] = [\mathbb{A}_{id_{new}},\ \mathbb{B}_{id_{new}},\ \mathbf{C}_{id_{new}}]$ is a realization which may consist of complex variables. This transformation matrix makes new coordinates z such as

$$z(t) = P^{-1}x(t). \tag{2.38}$$

Using equation (2.38), the discrete identified model, (2.36), can be written as

$$\begin{aligned} \dot{z}(t) &= P^{-1}\hat{\mathbb{A}}_{id}Pz(t) + P^{-1}\hat{\mathbb{B}}_{id}u(t), \\ y(t) &= \mathbf{C}_{id}Pz(t) + \mathbf{D}_{id}u(t), \end{aligned} \tag{2.39}$$

where

$$\begin{aligned} P^{-1}\hat{\mathbb{A}}_{id}P &\equiv \Lambda = diag\left[\bar{\lambda_1}, \bar{\lambda_2}, \cdots, \bar{\lambda_n}\right], \\ \bar{\lambda_i} &= \begin{bmatrix} \sigma_i + j\omega_i & 0 \\ 0 & \sigma_i - j\omega_i \end{bmatrix}. \end{aligned} \tag{2.40}$$

Equations (2.39) and (2.40) are based on the discrete time identified state-space model which is originated from the continuous time state-space system. And this is the state-space modal

coordinates model.

$$\begin{bmatrix} \dot{\eta}_i \\ \ddot{\eta}_i \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\omega_{N_{in}}^2 & -2\zeta_i\omega_{N_{in}} \end{bmatrix} \begin{bmatrix} \eta_i \\ \dot{\eta}_i \end{bmatrix} + \begin{bmatrix} 0 \\ P^{-1}\bar{\mathbb{B}}_{id} \end{bmatrix} u,$$
$$y = [\mathbf{C}_{id}P \qquad 0] \begin{bmatrix} \eta_i \\ \dot{\eta}_i \end{bmatrix} + \mathbf{D}_{id}u. \tag{2.41}$$

With equations (2.39) and (2.41), eigenvalues of $\hat{\mathbb{A}}_{id}$ can be expressed in terms of damping ratios and natural frequencies like

$$\omega_{N_{in}}^2 = -\sigma_i^2 - \omega_i^2,$$
$$\zeta_i\omega_{N_{in}} = -2\sigma_i. \tag{2.42}$$

Chapter 3

FAST ERA

3.1 Introduction

The Fast ERA also uses very compact concepts which are almost similar with the ERA method [80]. Strictly the Fast ERA is just alternative method of the ERA because the Fast ERA method uses the eigenvalues decomposition of the inner product of the Hankel matrix instead of the singular value decomposition of the Hankel matrix. In terms of linear algebra, there is no differences between the singular value decomposition of the Hankel matrix and the eigenvalue decomposition of the inner product of the Hankel matrix [37], [96]. However, due to the decomposition method, those identification methods can be distinguished.

The section 3.2 just touches the basic concepts of the Fast ERA because the basic concepts of the Fast ERA are similar with those of the ERA. The section 3.3 provides a step-by-step procedure of the Fast ERA, and the section 3.5 introduces advantages, disadvantage, and differences between the ERA and the Fast ERA.

3.2 Basic Concepts

The Fast ERA method also uses the Markov parameter and the Hankel matrix to store given output data compactly. The given output data are basically the impulse responses of system. The main difference with the ERA is the basic usage of the Hankel matrix i.e., the Fast ERA method uses the inner product of the Hankel matrix instead of the Hankel matrix itself. In addition, the Fast ERA method uses the eigenvalue decomposition. The eigenvalue decomposition is less accurate than the SVD in terms of the linear algebra. Basically, the MATLAB programming uses the SVD for getting the eigenvalue decomposition. For using the eigenvalue decomposition of inner product of Hankel matrix, let us check several concepts and newly defined matrices for the Fast ERA method.

Markov parameter and Hankel matrix

As the Markov parameter and the Hankel matrix are introduced in section 2.2, those concepts are used for storing given impulse response data. In this section, detail concepts will be not presented because section 2.2 has already introduced. Recall the Markov parameters and the Hankel matrix are

$$\mathbf{M}_{arkov}(k) \quad = \quad [y_1(kT_s) \quad y_2(kT_s) \quad \cdots \quad y_{N_{in}}(kT_s)]\,, \tag{3.1}$$

where

$$\mathbf{M}_{arkov}(k) \in \Re^{N_{out}\times N_{in}},\ k = 0,\ 1,\ 2,\ \cdots,\ \text{and } T_s = \text{sampling time},$$

$$\mathbf{H}_{qd}(k) \quad = \quad \begin{bmatrix} \mathbf{M}_{arkov}(k+1) & \mathbf{M}_{arkov}(k+2) & \cdots & \mathbf{M}_{arkov}(k+d) \\ \mathbf{M}_{arkov}(k+2) & \mathbf{M}_{arkov}(k+3) & \cdots & \mathbf{M}_{arkov}(k+d+1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}_{arkov}(k+q) & \mathbf{M}_{arkov}(k+q+1) & \cdots & \mathbf{M}_{arkov}(k+q+d-1) \end{bmatrix}, \tag{3.2}$$

where

N_{out} is the number of outputs, q and d are chosen numbers like

$$N_{out} \times q = d \times \text{number of inputs}(= N_{in})$$

$$\text{and } \mathbf{H}_{qd}(k) \in \Re^{(N_{out}\times q)\times(N_{in}\times d)}.$$

Eigenvalue decomposition

With same logic of the ERA method, the Fast ERA also needs the decomposition technique for breaking down and manipulating matrices which is made by the given data. The Fast ERA method uses the eigenvalue decomposition method so that we need to review the definition of the eigenvalue decomposition. According to numerical linear algebra texts [37], [57], [83], [93], [96], the eigenvalue decomposition of the matrix $\mathbf{H}$ ($\in \Re^{n\times n}$) is defined

$$\mathbf{H} = \mathbf{\Upsilon}\mathbf{\Lambda}\mathbf{\Upsilon}^T, \tag{3.3}$$

where

The superscript of ^{T} denotes the transpose matrix,

$\boldsymbol{\Upsilon} = \boldsymbol{\Upsilon}(1:n, 1:n)$ is Eigenvector matrix - Hermitian

$\boldsymbol{\Lambda} = \boldsymbol{\Lambda}(1:n, 1:n)$ is Eigenvalue diagonalized matrix

Mathematically, the eigenvalue decomposition of inner products of the Hankel matrix is associated with the singular value decomposition of the Hankel matrix. The inner products of the Hankel matrix can be expressed by the singular value matrices like

$$\begin{aligned} \mathbf{H}\mathbf{H}^T &= (\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T)(\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T)^T \\ &= \mathbf{U}\boldsymbol{\Sigma}\underbrace{\mathbf{V}^T\mathbf{V}}_{=\mathbf{I}}\boldsymbol{\Sigma}^T\mathbf{U}^T \\ &= \mathbf{U}\boldsymbol{\Sigma}^2\mathbf{U}^T. \end{aligned} \tag{3.4}$$

Also, the eigenvalue decomposition of the inner product of the Hankel matrix is

$$eig(\mathbf{H}\mathbf{H}^T) = \boldsymbol{\Upsilon}\,\boldsymbol{\Lambda}\,\boldsymbol{\Upsilon}^T. \tag{3.5}$$

The equations (3.4) and (3.5) show that the relationship between the eigenvalue decomposition of the inner products of the Hankel matrix and the singular value decomposition of the Hankel matrix.

$$\boldsymbol{\Lambda} = \boldsymbol{\Sigma}^2. \tag{3.6}$$

Controllability and Observability Matrix

The controllability matrix $\mathcal{C}$ and the observability matrix $\mathcal{O}$ also are used for the Fast ERA method same as the ERA method in section 2.2.

$$\mathcal{C} = \left[\hat{\mathbb{B}}, \hat{\mathbb{A}}\hat{\mathbb{B}}, \cdots, \hat{\mathbb{A}}^{(N-1)}\hat{\mathbb{B}}\right], \tag{3.7}$$

$$\mathcal{O} = \begin{bmatrix} \mathbf{C}_c \\ \mathbf{C}_c\hat{\mathbb{A}} \\ \vdots \\ \mathbf{C}_c\hat{\mathbb{A}}^{(N-1)} \end{bmatrix}, \tag{3.8}$$

where

$$\hat{\mathbb{A}} \in \mathfrak{R}^{N\times N},\ \hat{\mathbb{B}} \in \mathfrak{R}^{N\times N_{in}},\ \mathbf{C} \in \mathfrak{R}^{N_{out}\times N}\ ,$$

N is a number of the states,

N_{in} is a number of inputs,

and N_{out} is a number of outputs.

Several Newly Defined Matrices

We need to define a new D_q matrix which is the fundamental matrix of the Fast ERA method. The role of the D_q is almost the same with the Hankel matrix in the ERA method. The matrix D_q, therefore, will be broken down and manipulated. Let us define the D_q matrix with the inner product of the Hankel matrix,

$$D_q \quad = \quad \mathbf{H}_{qd}(0)\mathbf{H}_{qd}(0)^T. \tag{3.9}$$

Sometimes, the matrix D_q can be changed by using $\mathbf{H}_{qd}(1)$ instead of $\mathbf{H}_{qd}(0)$. However, basic concepts of identifying a system do not change. The identified results will be changed by the matrix D_q. Another new matrix D_q^* is defined with a part of matrix D_q, i.e.,

$$D_q^* = D_q(N_{out}+1\ :\ qN_{out}\ ,\ 1\ :\ (q-1)N_{out}). \tag{3.10}$$

The matrices D_q (3.9) and D_q^* (3.10) have $[qN_{out} \times qN_{out}]$, $[(q-1)N_{out} \times (q-1)N_{out}]$ respectively.

Until now, the most important and basic concepts of the Fast ERA method are introduced in detail. In next section, the procedure of the Fast ERA method will be presented.

3.3 Identification

The starting point for the Fast ERA is the matrix D_q and the Hankel matrix (3.2). The multiplication of the controllability and the observability matrices is one expression of the Hankel matrix from equations (2.12) and (2.13) in section 2.3.

$$\begin{aligned} D_q &= \mathbf{H}_{qd}(0)\mathbf{H}_{qd}^T(0) \\ &= \mathcal{O}_q\mathcal{C}_d\mathcal{C}_d^T\mathcal{O}_q^T. \end{aligned} \tag{3.11}$$

Let us assume that the matrix $\hat{\mathbb{A}}$ in state-space model is an unitary matrix. This assumption makes the equation (3.11) simple i.e., the square of the controllability matrix $\mathcal{C}_d$ is identity without any loss of generality. Then the equation (3.11) can be changed like

$$D_q = \mathcal{O}_q \mathcal{O}_q^T. \tag{3.12}$$

The state-space $\hat{\mathbb{A}}$ matrix can be obtained from manipulating the observability matrix $\mathcal{O}$, (3.8), with following steps. First of all, let us choose the first $(q-1)$ row of the matrix $\mathcal{O}$, then define a new matrix $\mathcal{O}_q^{(1)}$ like

$$\mathcal{O}_q^{(1)} = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\hat{\mathbb{A}} \\ \vdots \\ \mathbf{C}\hat{\mathbb{A}}^{q-2} \end{bmatrix}. \tag{3.13}$$

In addition, another new matrix $\mathcal{O}_q^{(2)}$ which is a part of the equation (3.8) can be defined like

$$\mathcal{O}_q^{(2)} = \begin{bmatrix} \mathbf{C}\hat{\mathbb{A}} \\ \mathbf{C}\hat{\mathbb{A}}^2 \\ \vdots \\ \mathbf{C}\hat{\mathbb{A}}^{q-1} \end{bmatrix}. \tag{3.14}$$

When we compare the equation (3.13) to the equation (3.14), there is a relationship which is associated with the state-space $\hat{\mathbb{A}}$ matrix. The relationship equation between equations (3.13) and (3.14) can be expressed by the following equation which will be a basic equation for the state-space $\hat{\mathbb{A}}$ matrix.

$$\mathcal{O}_q^{(1)} \hat{\mathbb{A}} = \mathcal{O}_q^{(2)}. \tag{3.15}$$

We have manipulated the matrices D_q and observability $\mathcal{O}$ with the controllability $\mathcal{C}$, observability $\mathcal{O}$ and the state-space $\hat{\mathbb{A}}$ matrices. Now, with an arbitrary matrix P_q and the eigenvalue decomposition, we need to make the formation of the matrix D_q. As a result, the eigen analysis of the matrix D_q and the matrix P_q can express the matrix D_q.

$$D_q = P_q P_q^T = \boldsymbol{\Upsilon}_{D_q} \boldsymbol{\Lambda}_{D_q} \boldsymbol{\Upsilon}_{D_q}^T. \tag{3.16}$$

The equation (3.16) has same formation with the equation (3.12). Therefore, the matrix $\mathcal{O}_q$ is equal to the matrix P_q, which means that the partition of the matrix P_q is same with the partition of the matrix $\mathcal{O}_q$.

$$\begin{aligned} \mathcal{O}_q^{(1)} = P_q^{(1)} &= \begin{bmatrix} \tilde{P}_0 \\ \tilde{P}_1 \\ \vdots \\ \tilde{P}_{q-2} \end{bmatrix}, \\ \mathcal{O}_q^{(2)} = P_q^{(2)} &= \begin{bmatrix} \tilde{P}_1 \\ \tilde{P}_2 \\ \vdots \\ \tilde{P}_{q-1} \end{bmatrix}. \end{aligned} \tag{3.17}$$

Recall the equation (3.15) and solve for the matrix $\hat{\mathbb{A}}$ which will be a identified state-space $\hat{\mathbb{A}}_{id}$ matrix. In order to solve for $\hat{\mathbb{A}}_{id}$, the inverse matrix of the observability matrix is required. In (3.15), however, the observability matrix is not a square matrix. As a result, the pseudo-inverse of observability matrix is used. Then, substitute the observability matrix with equations (3.17). After that, following equations can be obtained.

$$\begin{aligned} \hat{\mathbb{A}}_{id} &= \left[\mathcal{O}_q^{(1)}\right]^{\dagger} \mathcal{O}_q^{(2)} \\ &= \left[P_q^{(1)}\right]^{\dagger} P_q^{(2)}. \end{aligned} \tag{3.18}$$

If we can calculate the unknown matrices - $P_q^{(1)}$ and $P_q^{(2)}$, the identified state-space $\hat{\mathbb{A}}_{id}$ matrix is obtained by the equation (3.18). The matrices D_q and D_q^* will be used for getting the $P_q^{(1)}$ and $P_q^{(2)}$.

When the state-space $\hat{\mathbb{A}}$ matrix is calculated, the matrix $\left[P_q^{(1)}\right]^{\dagger}$ is the most wanted matrix. In the part of the equation (3.16), the matrix P_q can be separated by the first $(q-1)$ rows of P_q and the other.

$$\begin{aligned} D_q &= P_q P_q^T = \begin{bmatrix} P_q^{(1)} \\ \tilde{P}_q \end{bmatrix} \left[\left[P_q^{(1)}\right]^T \tilde{P}_q^T\right] \\ &= \begin{bmatrix} P_q^{(1)} \left[P_q^{(1)}\right]^T & P^{(1)} \tilde{P}_q^T \\ \tilde{P}_q \left[P_q^{(1)}\right]^T & \tilde{P}_q \tilde{P}_q^T \end{bmatrix}. \end{aligned} \tag{3.19}$$

In equation (3.19), $P_q^{(1)}$ is the square root factorization of the upper left $(q-1) \times (q-1)$ of D_q, which is D_{q-1} exactly. In addition, from the inner product of the Hankel matrix, $P_q^{(1)}$ $[P_q^{(1)}]^T$ can be written like

$$P_q^{(1)} \left[P_q^{(1)}\right]^T = D_{q-1} = \mathbf{H}_{q-1d}(0)\mathbf{H}_{q-1d}(0)^T. \tag{3.20}$$

With the eigenvalue analysis (3.16), we can compose the matrix D_{q-1} to the eigen values and eigen vectors. Then we can get the eigenvalues and eigenvector matrix for D_{q-1} such as

$$D_{q-1} \quad = \quad \mathbf{\Upsilon}_{D_{q-1}} \mathbf{\Lambda}_{D_{q-1}} \mathbf{\Upsilon}_{D_{q-1}}^T. \tag{3.21}$$

The equations (3.20) and (3.21) show two ways to express the matrix D_{q-1}. Then, by dividing $\mathbf{\Upsilon}_{D_{q-1}}$ to $[\mathbf{\Upsilon}_{D_{q-1}}]^{1/2}$, $P_q^{(1)}$ can be calculated

$$P_q^{(1)} \quad = \quad P_{q-1} = \mathbf{\Upsilon}_{D_{q-1}} [\mathbf{\Lambda}_{D_{q-1}}]^{\frac{1}{2}}. \tag{3.22}$$

By the definition of inverse matrix and the fact that the eigenvector matrix is a hermitian matrix, the pseudo-inverse of $P_q^{(1)}$ can be calculated easily by

$$\left[P_q^{(1)}\right]^\dagger = [\mathbf{\Lambda}_{D_{q-1}}]^{-\frac{1}{2}} \mathbf{\Upsilon}_{D_{q-1}}^T. \tag{3.23}$$

For getting $P_q^{(2)}$, which is the other part of the equation (3.18), the other partition of D_q - D_q^* - will be used. The matrix D_q can be partitioned by the last $(q-1)$ rows like

$$\begin{aligned} D_q \quad &= \quad P_q P_q^T = \begin{bmatrix} \tilde{P}_o \\ P_q^{(2)} \end{bmatrix} \begin{bmatrix} \left[P_q^{(1)}\right]^T & \tilde{P}_q^T \end{bmatrix} \\ &= \quad \begin{bmatrix} \tilde{P}_o \left[P_q^{(1)}\right]^T & \tilde{P}_o \tilde{P}_q^T \\ P_q^{(2)} \left[P_q^{(1)}\right]^T & P_q^{(2)} \tilde{P}_q^T \end{bmatrix}. \end{aligned} \tag{3.24}$$

In equation (3.24), partitioned matrix $\tilde{P}_o$ and $\tilde{P}_q^T$ are not important. The matrix $P_q^{(2)}$ are focused on for calculating the state-space matrix $\hat{\mathbb{A}}_{id}$. For calculating the $P_q^{(2)}$ with linear algebra manipulations, the lower left partition D_q^*, which is defined like the equation (3.10), will be used like

$$D_q^* = P_q^{(2)} \left[P_q^{(1)}\right]^T. \tag{3.25}$$

With the pseudo-inverse of the matrix $P_q^{(1)}$, (3.23), the equation (3.25) can be solved for the matrix $P_q^{(2)}$.

$$P_q^{(2)} = D_q^* \left[P_q^{(1)}\right]^{T\dagger} = D_q^* \mathbf{\Upsilon}_{D_{q-1}} [\mathbf{\Lambda}_{D_{q-1}}]^{-\frac{1}{2}}. \tag{3.26}$$

Finally, we can get the state-space $\hat{\mathbb{A}}_{id}$ matrix which has been assumed as an unitary matrix in early this section. With equations (3.18), (3.23) and (3.26), the assumption can be confirmed with following equation.

$$\hat{\mathbb{A}}_{id} = \left[P_q^{(1)}\right]^{\dagger} P_q^{(2)} = \mathbf{\Lambda}_{D_{q-1}}^{-\frac{1}{2}} \mathbf{\Upsilon}_{D_{q-1}}^T D_q^* \mathbf{\Upsilon}_{D_{q-1}} \mathbf{\Lambda}_{D_{q-1}}^{-\frac{1}{2}}. \tag{3.27}$$

For getting the state-space $\hat{\mathbb{B}}_{id}$ matrix, use the concept of the Markov parameter. With the Markov parameter basic concept in section 2.2, new matrix which consists of Markov parameters can be defined

$$\mathcal{M} = \begin{bmatrix} \mathbf{M}_{arkov}(1) \\ \mathbf{M}_{arkov}(2) \\ \vdots \\ \mathbf{M}_{arkov}(q-1) \end{bmatrix} = \begin{bmatrix} \mathbf{C}\hat{\mathbb{B}} \\ \mathbf{C}\hat{\mathbb{A}}\hat{\mathbb{B}} \\ \vdots \\ \mathbf{C}\hat{\mathbb{A}}^{q-2}\hat{\mathbb{B}} \end{bmatrix}. \tag{3.28}$$

The new matrix $\mathcal{M}$ is a known matrix because it consists of the Markov parameters which are made by the impulse response of system. In the equation (3.28), the state-space $\hat{\mathbb{B}}$ matrix can be extracted

$$\mathcal{M} = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\hat{\mathbb{A}} \\ \vdots \\ \mathbf{C}\hat{\mathbb{A}}^{q-2} \end{bmatrix} \hat{\mathbb{B}} = P_q^{(1)} \hat{\mathbb{B}}. \tag{3.29}$$

The $P_q^{(1)}$ matrix has been obtained by the equation (3.22). Then, we need the pseudo-inverse matrix of $P_q^{(1)}$ so that we can solve for the $\hat{\mathbb{B}}$ matrix. With the equation (3.23), therefore, the state-space $\hat{\mathbb{B}}_{id}$ matrix will be calculated like

$$\hat{\mathbb{B}}_{id} = \left[P_q^{(1)}\right]^{\dagger} \mathcal{M} = \left[P_q^{(1)}\right]^{\dagger} \begin{bmatrix} \mathbf{M}_{arkov}(1) \\ \mathbf{M}_{arkov}(2) \\ \vdots \\ \mathbf{M}_{arkov}(q-1) \end{bmatrix}. \tag{3.30}$$

The starting point for getting simply state-space $\mathbf{C}_{id}$ matrix is the observability matrix $\mathcal{O}$. It is obvious that the first N_{in} rows imply exactly the state-space $\mathbf{C}$ matrix in the equation (3.13). Therefore, the state-space $\mathbf{C}$ matrix is the first N_{in} rows of the $P_q^{(1)}$ which is the part of the matrix $\mathcal{O}$.

$$\mathbf{C}_{id} = P_q^{(1)}(1:N_{out}, 1:N_{in}). \tag{3.31}$$

The last state-space matrix $\mathbf{D}$ can be obtained with same logic for the ERA method. The state-space matrix must equal $\mathbf{M}_{arkov}(0)$ because $\mathbf{M}_{arkov}(0)$ gives us the information about relationship between inputs and outputs.

With given impulse response data, the identified state-space matrix $\hat{\mathbb{A}}_{id}$, $\hat{\mathbb{B}}_{id}$, $\mathbf{C}_{id}$, and $\mathbf{D}_{id}$ are provided by

$$\hat{\mathbb{A}}_{id} = \mathbf{\Lambda}_{D_{q-1}}^{-\frac{1}{2}} \mathbf{\Upsilon}_{D_{q-1}}^{T} D_q^{*} \mathbf{\Upsilon}_{D_{q-1}} \mathbf{\Lambda}_{D_{q-1}}^{-\frac{1}{2}}, \tag{3.32}$$

$$\hat{\mathbb{B}}_{id} = \left[P_q^{(1)}\right]^{\dagger} \begin{bmatrix} \mathbf{M}_{arkov}(1) \\ \mathbf{M}_{arkov}(2) \\ \vdots \\ \mathbf{M}_{arkov}(q-1) \end{bmatrix}, \tag{3.33}$$

$$\mathbf{C}_{id} = P_q^{(1)}(1:n_y, 1:n_x) = \left[P_q^{(1)}\right]^{\dagger T} \mathbf{\Lambda}_{D_{q-1}}, \tag{3.34}$$

$$\mathbf{D}_{id} = \mathbf{M}_{arkov}(0). \tag{3.35}$$

Let us summarize the procedure of the Fast ERA computation. For identifying the system using the Fast ERA, just keep though following steps.

(1) Calculating the matrix D_q - inner product of Hankel Matrix like:

$$D_q = \mathbf{H}_{qd}(0)\mathbf{H}_{qd}(0)^T.$$

(2) Determine eigenvalues and eigenvectors of D_{q-1}which is the partition of D_q:

$$D_q = \mathbf{\Upsilon}_{D_q} \mathbf{\Lambda}_{D_q} \mathbf{\Upsilon}_{D_q}^{T}.$$

(3) Get the pseudo-inverse of $P_q^{(1)}$:

$$\left[P_q^{(1)}\right]^{\dagger} = \mathbf{\Lambda}_{D_{q-1}}^{-\frac{1}{2}} \mathbf{\Upsilon}_{D_{q-1}}^{T}.$$

(4) Solve for $P_q^{(2)}$:

$$P_q^{(2)} = D_q^* \left[P_q^{(1)}\right]^{\dagger T}.$$

(5) Get state-space matrix $\hat{\mathbb{A}}$:

$$\mathbb{A}_{id} = \mathbf{\Lambda}_{D_{q-1}}^{-\frac{1}{2}} \mathbf{\Upsilon}_{D_{q-1}}^T D_q^* \mathbf{\Upsilon}_{D_{q-1}} \mathbf{\Lambda}_{D_{q-1}}^{-\frac{1}{2}}.$$

(6) Find matrix $\hat{\mathbb{B}}$:

$$\hat{\mathbb{B}}_{id} = \left[P_q^{(1)}\right]^{\dagger} \begin{bmatrix} \mathbf{M}_{arkov}(1) \\ \mathbf{M}_{arkov}(2) \\ \vdots \\ \mathbf{M}_{arkov}(q-1) \end{bmatrix}.$$

(7) Obtain matrix $\mathbf{C}$:

$$\mathbf{C}_{id} = P_q^{(1)}(1:n_y, 1:n_x) = \left[P_q^{(1)}\right]^{\dagger T} \mathbf{\Lambda}_{D_{q-1}}.$$

(8) Solve for matrix $\mathbf{D}$:

$$\mathbf{D}_{id} = \mathbf{M}_{arkov}(0).$$

3.4 Extracting $\mathbf{M}$, $\mathbf{D}_{amp}$ *and* $\mathbf{K}$ *Matrices*

The Fast ERA method also uses the discrete time matrices $\hat{\mathbb{A}}$ and $\hat{\mathbb{B}}$ like the ERA method because Markov Parameters consist of the matrices $\hat{\mathbb{A}}$ and $\hat{\mathbb{B}}$. Therefore, the fast ERA method needs some conversion methods from the identified discrete time system to a continuous time system. Many textbooks introduced conversion methods [22], [55] - zero-order holder, Tustin, and so on[1].

The identified state-space matrices (3.32) $\sim$ (3.35) give us the way to extract the characteristics $\mathbf{M}$, $\mathbf{D}_{amp}$, and $\mathbf{K}$ matrices with same methods in the previous subsection 2.4.

In terms of extracting damping and stiffness matrices, the Fast ERA is better methods than the ERA method because the ERA method needs special procedures to obtain continuous time system state-space matrix.

[1]Those conversion methods are included by MATLAB toolbox.

3.5 Comparison with ERA

There are several common factors and differences between the ERA and the Fast ERA methods. In this section, the comparisons are shown for helping users to understand how much they are different. The comparison table will be presented.

3.5.1 Common Factors

Basically, both methods use the Markov parameters for creating the fundamental matrix such as Hankel matrix (2.12) and D_q (3.9). The Markov parameters and the Hankel matrix are used for storing given impulse responses. The decomposition also are used in both methods - (2.18) with the singular value decomposition and (3.16) with the eigenvalue decomposition.

Both methods use the decomposition method to break down and manipulate matrices which are associated with the Markov parameters and theHankel matrix. The procedures of both methods look totally different because each decomposition method is different. Even though, however, formations are different, the basic concept of both decompositions is same because of equations (3.36), (3.37), and (3.38), namely,

$$eig(\mathbf{A}\mathbf{A}^T) = \boldsymbol{\Upsilon}_{\mathbf{A}}\, \boldsymbol{\Lambda}_{\mathbf{A}}\, \boldsymbol{\Upsilon}_{\mathbf{A}}^T, \tag{3.36}$$

$$svd(\mathbf{A}) = \mathbf{U}\, \boldsymbol{\Sigma}\, \mathbf{V}^T, \tag{3.37}$$

$$\text{with} \quad (3.36) \quad (3.37)$$

$$\boldsymbol{\Lambda}_{\mathbf{A}} = \boldsymbol{\Sigma}^2. \tag{3.38}$$

Mathematically, the eigenvalue decomposition of inner products of the Hankel matrix are associated with the singular value decomposition like equation (3.38). Therefore the basic concept of decomposition is exact same between the ERA and the Fast ERA in the end.

Other common factor is the use of the controllability $\mathcal{C}$ matrix and the observability $\mathcal{O}$ matrix. The definition and manipulation of the $\mathcal{C}$ and $\mathcal{O}$ make it possible to calculate the state-space $\mathbf{B}$ and $\mathbf{C}$ matrix in both identification methods.

The last very useful common factor is $\mathbf{M}$ (mass), $\mathbf{D}_{amp}$ (damping), and $\mathbf{K}$ (stiffness) matrices extraction from state-space model with several linear algebra manipulations like

subsection 2.4.

3.5.2 Differences

First of all, the decomposed matrices which are decomposed with singular value and eigenvalue are differences. Therefore, the state-space model form will be totally different. In the ERA method, the Hankel matrix is decomposed with singular values, but in the Fast ERA, the D_q - the inner product of Hankel matrix is decomposed with eigenvalues. In addition, the state-space $\mathbb{A}$ matrix in the Fast ERA is a hermitian matrix. As explained, the decomposition method are different.

In addition, the last few steps which is related to get the state-space matrices $\mathbb{B}$, $\mathbf{C}$, and $\mathbf{D}$ are different. This is because the decomposition methods differ with the other. The other reason why there are differences is the different starting point. It is possible that these differences are neglected because most steps are similar each other.

For helping to checking the differences and common factors, following table is provided.

Table 3.1: Table for Comparison ERA and Fast ERA

	ERA	Fast ERA
Basic concept	Decomposition	Decomposition
Basic matrix	Hankel matrix	Hankel matrix
Decomposition Method	Singular Value Decomposition	Eigen Decomposition
Decomposed Matrix	Hankel Matrix	Inner product of Hankel matrix
Input	Impulse Input	Impulse Input

where “Input” is used when the Hankel matrix is obtained.

Chapter 4

AR MODEL - AUTO-REGRESSIVE

4.1 Introduction

This chapter introduces the Auto-Regressive (AR) identification method. AR models use the concept of the probability because the AR model comes from the study of the random processes [6], [23], [58], [68], [74], [75], and [88]. AR models are used in economic prediction [34], statistical analysis [33], time series analysis [32] [65] and aerospace technology [42].

To help understand the basis of AR models, some concepts from a random process, e.g., process, probability and statistic will be discussed briefly. Then we will present a description of the AR model identification algorithm implemented in MATLAB.

4.2 Concepts of a Random Process

4.2.1 Basic

The AR models use concepts from the theory of probability and statistics [6], [24], and [59]. Next we present some important definitions and results from these theories.

1 Expectation Value or Mean

Expectation value or *Mean* is defined by [24]and [59]

$$E[X] = \int_{-\infty}^{\infty} t f_X(t) dt. \tag{4.1}$$

If we choose $f_X(x)$ as the distribution of mass on the real line, then $E[X]$ represents the center of mass. The expected value of random variable corresponds to our intuitive notion of the "average of X". Also, if X is a discrete random variable, the equation (4.1) has the other expression.

$$E[X] = \sum_{k=-\infty}^{\infty} x_k P_X(x_k), \tag{4.2}$$

where $P_X(x_k)$ is the probability distribution function.

2 Variance

The expected value gives us great information of data, but still very limited because we are interested not only in the mean but also in random variable's variation. The *Variance* is an expression of variation with the expectation value (or mean) [24] and [59].

$$\begin{aligned} VAR[X] &= E[(X - E[X])^2] && (4.3) \\ &= E[X^2 - 2E[X]X + E[X]^2] \\ &= E[X^2] - 2E[X]E[X] + E[X]^2 \\ &= E[X^2] - E[X]^2. && (4.4) \end{aligned}$$

The equations (4.3) and (4.4) are well known definition of the variance.

3 Standard Deviation

The *standard deviation* is also another expression of variation with the variance. The standard deviation is obtained by taking the square root of the variance with same units as X [24] and [59].

$$STD[X] = \sqrt{E[(X - E[X])^2]}. \quad (4.5)$$

Usually, the standard deviation is used as a measure of the "width" or "spread" of the data distribution.

4 Correlation

The *correlation* $R_X(t_1, t_2)$ of a random process X(t) is defined as the joint moment of $X(t_1)$ and $X(t_2)$ [24] and [59].

$$R_X(t_1, t_2) = E[X(t_1)X(t_2)]. \quad (4.6)$$

If the correlation of X and Y is zero, then X and Y are orthogonal.

5 Covariance

The *covariance* of a random process $X(t)$ is defined as the variance of the correlation

[24], i.e.,

$$\begin{aligned} C_X(t_1,t_2) &= E[(X(t_1)-E[X(t_1)])(X(t_2)-E[X(t_2)])] \\ &= E[X(t_1)X(t_2)-E[X(t_1)]X(t_2) \\ &\qquad -X(t_1)E[X(t_2)]+E[X(t_1)]E[X(t_2)]] \\ &= E[X(t_1)X(t_2)]-E[X(t_1)]E[X(t_2)] \\ &\qquad -E[X(t_1)]E[X(t_2)]+E[X(t_1)]E[X(t_2)] \\ &= R_X(t_1,t_2)-E[X(t_1)]E[X(t_2)]. \end{aligned} \tag{4.7}$$

6 Conditional Probability

The probability of event A given that event B has occurred is called by *conditional probability*. The conditional probability can be calculated by following equation [59].

$$Pr[A|B] = \frac{Pr[A \cap B]}{Pr[B]} \quad for \quad Pr[B] > 0.$$

The basic definitions given above are summarized in Table 4.1.

Table 4.1: Basic Functions and Notations in Probability and Statistics

Function or Notation	Meaning
$\mathrm{E}[\cdot]$	Expectation Value of $(\cdot)$
$m = \mathrm{E}[x]$	Mean of x
$var = \mathrm{E}[(x-m)^2]$	Variance of x
$std = \sqrt{E[(x-m)^2]}$	Standard Deviation of x
$cov(x) = \mathrm{E}[x(n)x(n)]$	Covariance
$R(x) = \mathrm{E}[x(l)x(n)]$	Correlation
$Pr[A]$	Probability of event A
$Pr[A\|B]$	Conditional probability of event A with given event B

4.2.2 Random (or Stochastic) Process

This subsection, we consider a *random* (or *stochastic*) *process* because the AR model comes from the study of a random (stochastic) process. A random process is a family of time functions together with a probability measure [68]. In addition, a random process can be illustrated as shown Figure 4.1 where ξ_i are outcomes, $P(\xi_i)$ are probabilities, $X(t, \xi_i)$ are time waveforms, and $i = 1, 2, \cdot, N$. The random process, $X(t, \xi)$ is a function of two

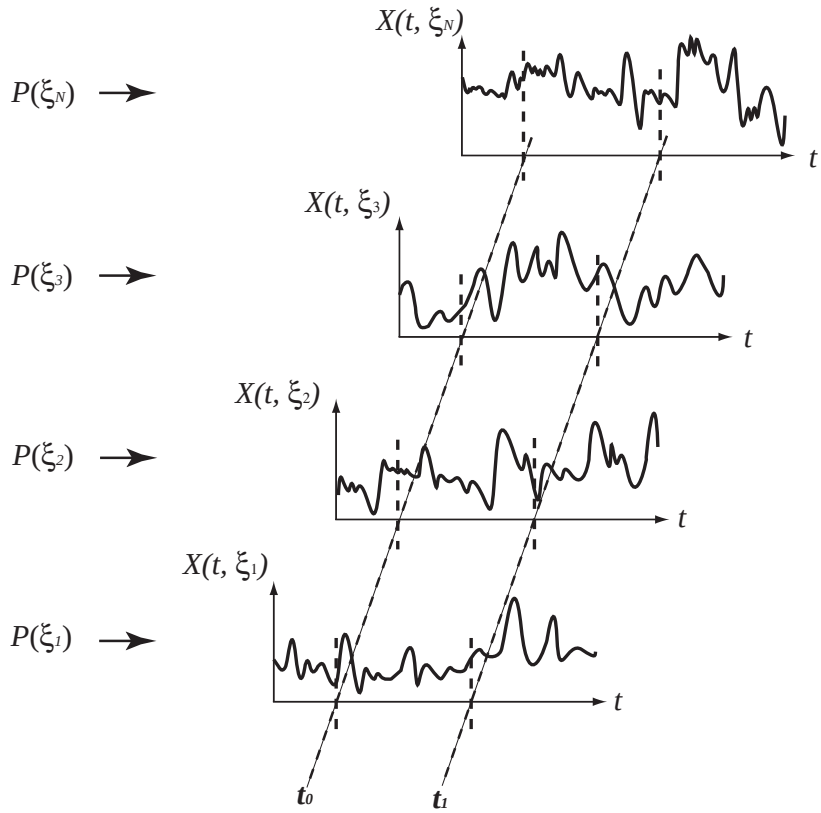

Figure 4.1: An explanation of a random process with waveforms, a sample space, and probability measure.

variables, t and ξ, where t is the time variable and ξ is the outcome variable. In particular, the specified time, $t = t_0$, makes the random process to be a random variable, $X(t_0, \xi)$. On the other hand, with the fixed outcome variable, $\xi = \xi_i$, the random process becomes a time function, i.e., $X(t, \xi_i)$ which is called a simple function or a realization of the process. If t

is time, the random process is called a random sequence or a discrete time random process.

ARMA Random Processes

Here, we review a special class of random processes, e.g., *Auto-Regressive Moving Average* (ARMA) random process, *Moving Average* (MA) random process, and *Auto-Regressive* (AR) random process [60] and [25]. First of all, let us consider time-invariant linear system, and review the terminologies of system analysis.

When the response to a certain input does not depend on time, the system is called time invariant system. Also, because the output of the system to any linear combination of inputs is the same linear combination of the outputs of the individual inputs, the system can be called a linear system. If $w(n)$ is the input to a linear time-invariant system and $x(n)$ is the output, the input-output relationship becomes

$$x(n) + a_1 x(n-1) + a_2 x(n-2) + \cdots + a_p y(n-p) = b_0 w(n) + b_1 w(n-1) + \cdots + b_q w(n-q), \quad (4.8)$$

where we assume that $p > q$, $n \geq 0$.

From equation (4.8), if $w(n)$ is a white sequence with $E[w(n)] = 0$ and $E[w(j)w(k)] = \sigma^2 \delta_{jk}$ for all j and k, the process $x(n)$ is called an autoregressive moving average (AMRA) random process of order p, q, which is usually written ARMA(p, q).

If all the $a_k = 0$ for all $k = 1, 2, \cdots, p$, the resulting random process $x(n)$ is called a moving average (MA) of order q, which is expressed by MA(q). Equation (4.8) becomes

$$x(n) = b_0 w(n) + b_1 w(n-1) + \cdots + b_q w(n-q). \quad (4.9)$$

If all the $b_k = 0$ for all $k = 1, 2, \cdots, q$, the random process $x(n)$ is an auto-regressive (AR) process of order q, written as AR(q) and equation (4.8) becomes

$$x(n) + a_1 x(n-1) + a_2 x(n-2) + \cdots + a_p y(n-p) = b_0 w(n). \quad (4.10)$$

The 'auto-regressive' means that it can be written as a linear function of its own previous values.

4.2.3 Storages - Markov parameter

All identification methods require the storage of the system data. For storing the given responses, AR models also use the Markov parameter, like the ERA (2.6) and the Fast ERA (3.1) methods, that is.

$$\mathbf{M}_{arkov}(k) \quad = \quad [y_1(kT_s) \quad y_2(kT_s) \quad \cdots \quad y_{N_{in}}(kT_s)]\,, \tag{4.11}$$

where

$$k = 0,\, 1,\, 2,\, \cdots,\, \text{and } T_s = \text{sampling time.}$$

The Hankel matrix, however, will not be used in the AR model. Even though AR models do not use the Hankel matrix, the storage of the given data is relatively larger than the ERA and the Fast ERA methods [20], [78]. The AR model needs the proper model reduction algorithm because of the larger data set. The MATLAB program provides simple commands for AR models which use only the state-space matrices **A**, **B**, **C** and **D**.

4.2.4 System Response

Consider a system with a scalar input $u(t)$ and a scalar output $y(t)$. When the system is a linear time invariant system, impulse response can be applied to the system by [26] and [61] as follows

$$y(t) = \int_{\tau=0}^{\infty} g(\tau)u(t-\tau)d\tau. \tag{4.12}$$

The stored data should be the discrete time response, because it is impossible to store all of the continuous response data. Therefore, the equation (4.12) needs to be transferred from continuous to discrete time. An arbitrary time t_k is defined by $t_k = kT_s$ $k = 1,\, 2,\, \cdots$, with a sampling time T_s. If the discrete time t_k is applied to the equation (4.12), then the discrete time output is

$$y(kT_s) = \int_{\tau=0}^{\infty} g(\tau)u(kT_s-\tau)d\tau, \tag{4.13}$$

where the sampling time T_s is the interval of getting data. With the discrete time concepts, the impulse output response (4.13) can be written as

$$y(t) = \sum_{k=1}^{\infty} g(k)u(t-k), \quad t = 0, 1, 2, \cdots . \tag{4.14}$$

The equation (4.14) shows that the exact solution can be obtained if the input signal is known. However, *disturbances* such as measurement noise, input noise and so on, occur in practical applications. Therefore, the equation (4.14) should be changed to account for these disturbances, i.e.,

$$y(t) = \sum_{k=1}^{\infty} g(k)u(t-k) + v(t). \tag{4.15}$$

where $v(t)$ is the disturbance. Here the disturbance is defined as

$$v(t) = \sum_{k=0}^{\infty} h(k)e(t-k), \tag{4.16}$$

where $e(t)$ are random variables with a known probability density function, and $h(t)$ is an arbitrary disturbance function. Usually $e(t)$ is taken as '*white noise*' which has a zero mean and λ covariance.

We need to characterize the disturbance. Specifically, we must establish the mean and the covariance. Apply the equation (4.16) to the definition of the mean (4.2) and covariance (4.7) to obtain

$$\begin{aligned}
E[v(t)] &= E\left[\sum_{k=0}^{\infty} h(k)e(t-k)\right] \\
&= \sum_{k=0}^{\infty} h(k)E[e(t-k)] \leftarrow E[e(t-k)] = 0; \ \because E[e(t)] = 0 \\
&= 0, \\
E[v(t)v(t-\tau)] &= E\left[\sum_{k=0}^{\infty} h(k)e(t-k) \sum_{l=0}^{\infty} h(l)e(t-\tau-l)\right] \\
&= \sum_{k=0}^{\infty}\sum_{l=0}^{\infty} h(k)h(l)E[e(t-k)e(t-\tau-l)] \\
&= \sum_{k=0}^{\infty}\sum_{l=0}^{\infty} h(k)h(l)\delta(k-\tau-l)\lambda \\
&= \lambda \sum_{k=0}^{\infty} h(k)h(k-\tau).
\end{aligned}$$

To obtain a transfer function from the equation (4.14), the definitions of forward shift operator q and backward shift operator q^{-1} are used as follows

$$qu(t) = u(t+1), \qquad q^{-1}u(t) = u(t-1).$$

With the definition of the forward and the backward shift operators, the equation (4.14) can be rewritten. Finally the transfer function of $G(q)$ which is associated with the input $u(t)$ and the output $y(t)$, can be defined as

$$\begin{aligned} y(t) &= \sum_{k=1}^{\infty} g(k)u(t-k) = \sum_{k=1}^{\infty} g(k)[q^{-k}u(t)] \\ &= \left[\sum_{k=1}^{\infty} g(k)q^{-k}\right] u(t) = G(q)u(t). \\ \rightarrow G(q) &= \sum_{k=1}^{\infty} g(k)q^{-k}. \end{aligned} \tag{4.17}$$

Using the same logic, the transfer function of $H(q)$, which is related to the disturbance $v(t)$ and the random variables $e(t)$, can be obtained by the equation (4.18). The transfer function of $H(q)$ gives us a new expression of $v(t)$ as follows

$$H(q) = \sum_{k=0}^{\infty} h(k)q^{-k}, \tag{4.18}$$

$$v(t) = H(q)e(t). \tag{4.19}$$

4.2.5 Model Structures

AR, MA, and ARMA random processes give us several ways to construct a sum process with inputs and outputs [68] and [88]. Let us consider structures for identifying a system.

ARX - Auto-Regressive eXternal input

The ARX - Auto-Regressive eXternal input with Multi Input Multi Output (MIMO) system in Figure 4.2 which is the first structures to be discussed. The difference equation of MIMO system based upon Figure 4.2 is as following.

$$y(t) + a_1 y(t-1) + \cdots + a_{n_a} y(t-n_a) = b_1 u(t-1) + \cdots + b_{n_b} u(t-n_b) + e(t). \tag{4.20}$$

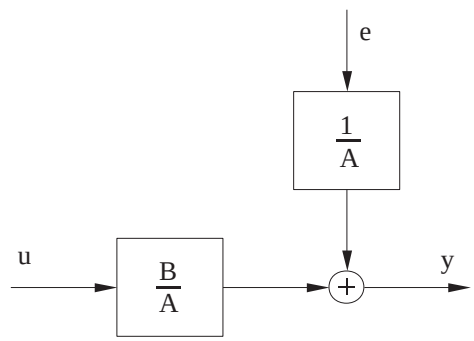

Figure 4.2: ARX model structure

The equation (4.20) is called an error equation model [65] because the $e(t)$ is included. The $e(t)$ is a representative of a white noise which has generally zero mean and λ covariance.

We can also write a time domain description of the system with transfer functions $G(q)$ (4.17) and $H(q)$ (4.18) as

$$y(t) = G(q)u(t) + H(q)e(t). \tag{4.21}$$

For realizing the $G(q)$ and $H(q)$ in the equation (4.21), let us convert the equation (4.20) to the transfer function formulation. Therefore, we can use the following polynomial in the delay operator equations (4.22) and (4.23) with adjustable parameters (4.24).

$$\begin{aligned} A(q) &= 1 + a_1 q^{-1} + \cdots + a_{n_a} q^{-n_a}, && (4.22) \\ B(q) &= b_1 q^{-1} + \cdots + b_{n_b} q^{-n_b}, && (4.23) \\ \Pi &= \begin{bmatrix} a_1 & a_2 & \cdots & a_{n_a} & b_1 & b_2 & \cdots & b_{n_b} \end{bmatrix}^T. && (4.24) \end{aligned}$$

If we specify transfer functions G and H with equations (4.22) $\sim$ (4.24), we can express those transfer functions as q^{-1} when n_k is the number of delays from input to output, i.e.,

$$\begin{aligned} G(q,\Pi) &= q^{-n_k}\frac{B(q)}{A(q)}, && (4.25) \\ H(q,\Pi) &= \frac{1}{A(q)}, && (4.26) \\ A(q)y(t) &= B(q)u(t-n_k) + e(t). && (4.27) \end{aligned}$$

With equations (4.22), (4.23), the equation (4.27) can be written like

$$\begin{aligned} y(t) + a_1 y(t-1) + \cdots + a_{n_a} y(t-n_a) &= \\ b_1 u(t-n_k) + b_2 u(t-n_k-1) + &\quad \cdots \quad + b_{n_b} u(t-n_k-n_b+1) + e(t), \end{aligned} \tag{4.28}$$

then we make following equation with equations (4.25), (4.26), (4.22), (4.23), and (4.24).

$$y(t) = G(q,\Pi)u(t) + H(q,\Pi)e(t). \tag{4.29}$$

Equations (4.28) and (4.29) are called the ARX model. Figure 4.2 shows the block diagram of the structure of the ARX model (4.28) and (4.29).

ARMAX - Auto Regressive Moving Average eXternal input

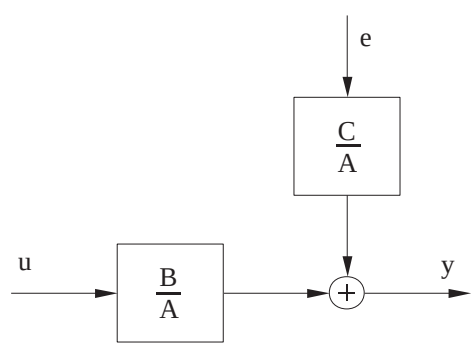

Figure 4.3: ARMAX Model Structure

The structure of ARX model is not enough to describe a disturbance term. The disturbance term of the ARX model is just $e(t)$ and $\frac{1}{A(q)}$. For better description of disturbance, therefore, we add the term $C(q)$ to equation (4.20) to get

$$\begin{aligned} & y(t) + a_1 y(t-1) + \cdots + a_{n_a} y(t-n_a) \\ & = b_1 u(t-1) + \cdots + b_{n_b} u(t-n_b) + e(t) \\ & \quad + c_1 e(t-1) + \cdots + c_{n_c} e(t-n_c). \end{aligned} \tag{4.30}$$

With same logic for equations (4.22), (4.23), and (4.24), the definition of $C(q)$ and new parameters are required as follows.

$$\begin{aligned} C(q) &= 1 + c_1 q^{-1} + \cdots + c_{n_c} q^{-n_c}, && (4.31) \\ \Pi &= \begin{bmatrix} a_1 & \cdots & a_{n_a} & b_1 & \cdots & b_{n_b} & c_1 & \cdots & c_{n_c} \end{bmatrix}^T. && (4.32) \end{aligned}$$

The description of ARMAX model with equations (4.30) $\sim$ (4.32) can be obtained as

$$\begin{aligned} & y(t) + a_1 y(t-1) + \cdots + a_{n_a} y(t-n_a) \\ & \quad = b_1 u(t-n_k) + \cdots + b_{n_b} u(t-n_b-n_k+1) \\ & \qquad + e(t) + c_1 e(t-1) + \cdots + c_{n_c} e(t-n_c). \end{aligned} \tag{4.33}$$

The ARMAX model in terms of the transfer function can be written using $A(q)$, $B(q)$, $C(q)$, $G(q,\Pi)$, and $H(q,\Pi)$, i.e.,

$$\begin{aligned} A(q)y(t) &= B(q)u(t-n_k) + C(q)e(t), && (4.34) \\ y(t) &= G(q,\Pi)u(t-n_k) + H(q,\Pi)e(t), && (4.35) \end{aligned}$$

where

$$G(q,\Pi) = q^{-n_k}\frac{B(q)}{A(q)}, \quad H(q,\Pi) = \frac{C(q)}{A(q)}.$$

Figure 4.3 shows the structure of ARMAX model with a block diagram. The ARMAX model is a standard tool in control and econometrics [65].

Other Methods

Other models - *Output-Error* model and *Box-Jenkins* model are introduced in [65], [66]. The Output-Error model is expressed like

$$\begin{aligned} y(t) &= \frac{B(q)}{F(q)}u(t-n_k) + e(t), && (4.36) \\ \text{with } F(q) &= 1 + f_1 q^{-1} + \cdots + f_{n_f} q^{-n_f}. \end{aligned}$$

According to the name of the Output-Error, the output error of system will directly affect

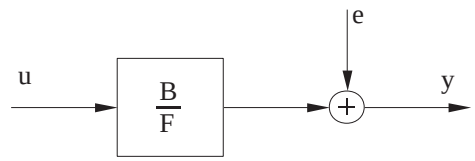

Figure 4.4: Output-Error structure

the output like measurement errors. Therefore, the Output-Error structure is Figure 4.4. In addition, the Box-Jenkins model is written as

$$\begin{aligned} y(t) &= \frac{B(q)}{F(q)}u(t-n_k) + \frac{C(q)}{D(q)}e(t), \\ \text{with } D(q) &= 1 + d_1 q^{-1} + \cdots + d_{n_d} q^{-n_d}. \end{aligned} \tag{4.37}$$

The Box-Jenkins model structure is a right block diagram of Figure 4.5.

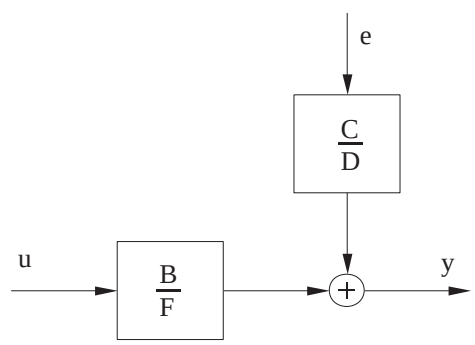

Figure 4.5: Box-Jenkins Structure

For summarizing and generalizing structures of AR models, equations (4.29), (4.35), (4.36), and (4.37) can be generalized as

$$\begin{aligned} A(q)y(t) &= q^{-n_k}\frac{\overline{B}(q)}{F(q)}u(t) + \frac{C(q)}{D(q)}e(t), \\ q^{-n_k}\bar{B}(q) &= B(q). \end{aligned} \tag{4.38}$$

where n_k is the number of delay from input to output. Typically, $n_k = 1$, therefore the equation (4.38) can be expressed by

$$\begin{aligned} A(q)y(t) &= q^{-1}\frac{\overline{B}(q)}{F(q)}u(t) + \frac{C(q)}{D(q)}e(t), \\ q^{-1}\bar{B}(q) &= B(q). \end{aligned} \tag{4.39}$$

The equations (4.38) and (4.39) look very complicated when compared to the ERA or Fast ERA method. However, each model has just 2 or 3 polynomials in transfer function equations. In other word, 2 or 3 polynomials in transfer function equations are neglected. When considering the equation (4.29) in the ARX model, the required polynomials are $A(q)$ and $B(q)$ polynomials. The polynomials $A(q)$, $B(q)$, and $C(q)$ are used for the ARMAX model (4.35). The Output-Error model uses the polynomials $B(q)$ and $F(q)$ like the equation (4.36). Finally, the Box-Jenkins model uses the $B(q)$, $C(q)$, $D(q)$, and $F(q)$ polynomials like the equation (4.37).

Therefore, Table 4.2 shows which polynomials in the equation (4.38) are used to make a model structure.

Table 4.2: Table for the equation (4.38)

Methods	Used polynomial	Methods	Used polynomial
ARX	A, B	ARMAX	A, B, C
Output-Error	B, F	Box-Jenkins	B, C, D, F

4.3 Predictor

In this section, the method of prediction will be presented for setting up the identification method. Therefore, the given system description (4.21) will be used how to predict outputs. The $v(t)$ can be predicted like

$$v(t) = H(q)e(t) = \sum_{k=0}^{\infty} h(k)e(t-k). \tag{4.40}$$

$H(q)$ must be stable because the equation (4.40) is meaningless when $v(t)$ will be divergent. Therefore, the following necessary condition must be satisfied.

$$\sum_{k=0}^{\infty} |h(k)| < \infty. \tag{4.41}$$

The equation (4.40) should have the invertible property for predicting some systems. Therefore, if $v(s)$, $s \leq t$, are known, $e(t)$ can be obtained like

$$\begin{aligned} e(t) = H^{-1}(q)v(t) &= \sum_{k=0}^{\infty} \widetilde{h}(k)v(t-k), \\ \text{where } \ H^{-1}(q) &= \frac{1}{H(q)}, \end{aligned} \tag{4.42}$$

also with

$$\sum_{k=0}^{\infty} |\widetilde{h}(k)| < \infty. \tag{4.43}$$

For predicting the output, let us assume that $v(s)$ for $s \leq t-1$ can be observed. Then the value of $v(t)$ is predicted with observed values $v(s)$. The equation (4.16) can be expressed like following equations.

$$v(t) = \sum_{k=0}^{\infty} h(k)e(t-k) = e(t) + \sum_{k=1}^{\infty} h(k)e(t-k). \tag{4.44}$$

If $v(s)$ is known, $e(t)$ must be known. Therefore the second term of (4.44) is known at time $t-1$. For simplifying the equation (4.44), let us denote

$$n(t-1) = \sum_{k=1}^{\infty} h(k)e(t-k). \tag{4.45}$$

The probability distribution function (PDF) must be needed for predicting - $e(t)$ is identically distributed.

$$Pr[x \leq e(t) \leq x + \Delta x] \approx f_e(x)\Delta x.$$

With the probability theory and conditional probability concepts, $f_v(t)$ which is the probability distribution function of $v(t)$ can be written such as

$$\begin{aligned} f_v(t)\Delta x &= Pr[x \leq v(t) \leq x + \Delta x | v_{-\infty}^{t-1}] = Pr[x \leq n(t-1) + e(t) \leq x + \Delta x] \\ &= Pr[x - n(t-1) \leq e(t) \leq x + \Delta x - n(t-1)] = f_e(x - n(t-1))\Delta x. \end{aligned}$$

When the equation (4.44) is considered again, a conditional expectation value $E[v(t)|v(t-1)]$ must be $n(t-1)$. Also, with z-transformation, (4.40), and (4.42), following equation can be derived.

$$\begin{aligned} E[v(t)|v(t-1)] &= n(t-1) = \sum_{k=1}^{\infty} h(k)e(t-k) \\ &= \left[\sum_{k=1}^{\infty} h(k)q^{-k}\right] e(t) = \left[\sum_{k=0}^{\infty} h(k)q^{-k} - 1\right] e(t) \\ &= [H(q)-1]\, e(t) = \frac{H(q)-1}{H(q)} v(t) \\ &= \left[1 - H^{-1}(q)\right] v(t) = \sum_{k=1}^{\infty} \left(-\widetilde{h}(k)v(t-k)\right). \end{aligned} \tag{4.46}$$

Equations (4.46) are expressions of one-step prediction of $v(t)$. With system description equation (4.21) and assumption - $y(s)$ and $u(s)$ are known for $s \leq t$-1, one-step prediction of $y(t)$ can be obtained by following procedures.

By the equation (4.21),

$$y(s) = G(q)u(t) + H(q)e(s) = G(q)u(t) + v(s) \tag{4.47}$$

$$\Rightarrow v(t) = y(t) - G(q)u(t). \tag{4.48}$$

The value of $v(t)$ have already been obtained like (4.46). Then following equations are able to derived by same logic with one-step prediction of $v(t)$ and (4.48).

$$\begin{aligned} E[y(t)|y(t-1)] &= G(q)u(t) + E[v(t)|v(t-1)] \\ &= G(q)u(t) + \left[1 - H^{-1}(q)\right] v(t) \\ &= G(q)u(t) + \left[1 - H^{-1}(q)\right] [y(t) - G(q)u(t)]. \end{aligned}$$

Using above equations, the conditional expectation value of $y(t)$ can be predicted by a following equation.

$$E[y(t)|y(t-1)] = H^{-1}(q)G(q)u(t) + \left[1 - H^{-1}(q)\right] y(t). \tag{4.49}$$

In real engineering, many systems are multi-variable systems. For a multi-variable system, the $m \times m$ matrix $H^{-1}(q)$ as

$$H^{-1}(q) = \sum_{k=0}^{\infty} \left(\widetilde{h}(k) q^{-k} \right). \tag{4.50}$$

The rest parts of prediction method are valid for multi-variable systems.

4.4 Parameter Selection

We just assume that we have a transfer function for AR models for categorizing AR models easily by structural understanding. Therefore, we need to calculate the coefficient of those polynomials as the parameters. For simplification, let us consider a linear, time-invariant and third-order system case, i.e.,

$$\frac{Y(q)}{U(q)} = G(q, \Pi) = \frac{B(q)}{A(q)} = \frac{b_1 q^2 + b_2 q + b_3}{q^3 + a_1 q^2 + a_2 q + a_3}, \tag{4.51}$$

where the parameter vector Π is

$$\Pi = [a_1 \;\; a_2 \;\; a_3 \;\; b_1 \;\; b_2 \;\; b_3]^T. \tag{4.52}$$

The given measurements $Y_m(1)$, $Y_m(2)$, and $Y_m(3)$ corresponding to the known inputs $U(0)$, $U(1)$, and $U(2)$ can determine the possible estimate of the magnitude of the estimates error for the parameter vector (4.52). From the transfer function of systems $G(q, \Pi)$, the true outputs and inputs of systems are associated with following equations

$$\begin{aligned} y(k+3) \quad &+ \quad a_1 y(k+2) + a_2 y(k+1) + a_3 y(k) \\ &= \quad b_1 u(k+2) + b_2 u(k+1) + b_3 u(k). \end{aligned} \tag{4.53}$$

Equation (4.53) can be rearranged and changed variables

$$\begin{aligned} y(k) \quad &= \quad -a_1 y(k-1) - a_2 y(k-2) - a_3 y(k-3) \\ &\quad\quad + b_1 u(k-1) + b_2 u(k-2) + b_3 u(k-3). \end{aligned} \tag{4.54}$$

For applying the *weighted least squares estimation*, the measured data $Y_m(k)$ can be expressed with error term.

$$\begin{aligned} Y_m(k) &= \phi(k)\Pi + v(k), \\ \textit{with} \quad & E[v(k)] = 0, \\ & E[v(k) \cdot v^T(k)] = \sigma^2(k) \end{aligned} \tag{4.55}$$

The matrix ϕ consists of the known data from measured vector $y(\cdot)$ and $v(k)$ is a unknown error vector which has zero mean and $\sigma^2(k)$ covariance.

$$\begin{bmatrix} y_m(1) \\ y_m(2) \\ y_m(3) \\ y_m(4) \\ \vdots \end{bmatrix} = \begin{bmatrix} -y(0) & 0 & 0 & u(0) & 0 & 0 \\ -y(1) & -y(0) & 0 & u(1) & u(0) & 0 \\ -y(2) & -y(1) & -y(0) & u(2) & u(1) & u(0) \\ -y(3) & -y(2) & -y(1) & u(3) & u(2) & u(1) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} + \begin{bmatrix} v(1) \\ v(2) \\ v(3) \\ v(4) \\ \vdots \end{bmatrix}$$

$$\rightarrow Y_m(k) = \phi(k)\Pi + v(k). \tag{4.56}$$

With equations (4.54) and (4.55), the weighted-least squares estimation of the parameter vector Π can be applied in order to get the estimation $\hat{\Pi}$. If we have enough measurement data $y_m(k)$ which is elements of ϕ matrix, the estimation result $\hat{\Pi}$ will be good.

$$\hat{\Pi} = (\phi^T R^{-1} \phi)^{-1} \phi^T R^{-1} Y_m(k), \tag{4.57}$$

with

$$R = \begin{bmatrix} \sigma^2(1) & 0 & \cdots & 0 \\ 0 & \sigma^2(2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma^2(n) \end{bmatrix}.$$

The weighted least square calculation is used in this section. Others estimation methods [1] also be used for examples Least Squares, Recursive Least Squares, Stochastic Least Squares, and Maximum Likelihood.

The parameters (4.24) and (4.32) can be applied to this parameter selection.

[1] Each method explanation is in chapter 12 of [22].

4.5 Extraction of State-Space Model

Usually, the state-space form of linear systems can be expressed like

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t) + v(t). \end{aligned} \tag{4.58}$$

With some steps of linear algebra manipulation, the transfer function $G(q)$ corresponds

$$G(q) = C(qI_{n_x} - A)^{-1}B + D. \tag{4.59}$$

If state-space form includes noises, the equation (4.58) can be written by following equations.

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t) + Ke(t), \\ y(t) &= Cx(t) + Du(t) + v(t). \end{aligned} \tag{4.60}$$

The matrix K can be computed from 'Kalman gain' if noises are stochastic processes with certain covariance properties. The transfer function $H(q)$ also corresponds

$$H(q) = C(qI_{n_x} - A)^{-1}K + I_{n_y}. \tag{4.61}$$

4.6 Comparison

AR models use probability theory for identifying some systems - usually time series prediction. According to concepts of AR models, many time series analysis can be applied. On the contrary, ERA methods are able to apply linear engineering systems easily. Many papers have tried to compare those methods [19], [20], [78], and [79]. They compared the size of computer memory storages when they are applied into personal computers. As the computer technologies are developed quickly, the problem of storages has became less significant.

Chapter 5
ALGORITHM EVALUATION

5.1 Introduction

In this chapter, some examples will be introduced to show how we apply the identification methods - ERA, Fast ERA, and ARX. A three degree of freedom mechanical system - three masses, four dampers, and four springs, is example. All identification methods are applied and evaluated. Then, adding some white noises to the responses of the same example makes the simulated data more realistic. Basically, MATLAB program simulates all examples to find the responses of the system due to an impulse input, identify the system using the given responses, and compare the identified system to the original system. Observation - The differences and the advantages among the various identification methods will also be explained in detail.

A simple description of a gyroscopic system is followed by the three degree of freedom example. The gyroscopic example will only be used for verifying a new identification technique which will be introduced in Chapter 6. After the verification of the new identification method and a model reduction algorithm using the complex variable transformation, the identification using actual experimental data of a Flywheel Energy Storage System is shown in Chapter 8.

5.2 Three-DOF System without noises

5.2.1 System

Figure 5.1 shows an example of multi springs, dampers and masses system. This example had been introduced in [3]. This example has the following equations of motion.

$$\begin{aligned} \mathbf{M}\ddot{q} + \mathbf{D}_{amp}\dot{q} + \mathbf{K}q &= U, \\ y &= \mathbf{c}q, \end{aligned} \tag{5.1}$$

where

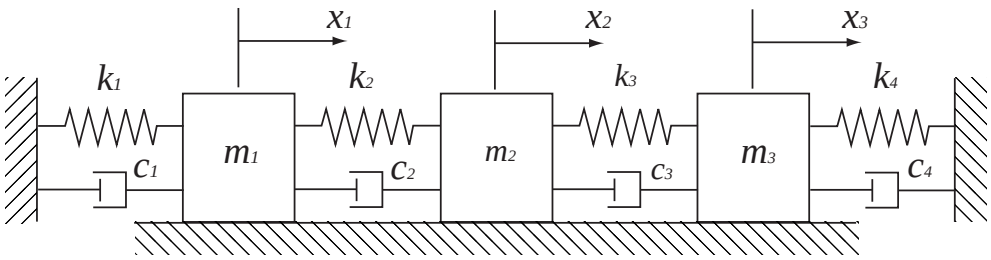

Figure 5.1: Spring, Damping and Mass System

$q \in \Re^{3\times 1}$: displacement vector for 3 DOF system, $y \in \Re^{3\times 1}$: Output vector for 3 DOF system,

$U \in \Re^{3\times 3}$: Input matrix due to the impulse input, $\mathbf{M} \in \Re^{3\times 3}$: Mass matrix - symmetry, $\mathbf{D}_{amp} \in \Re^{3\times 3}$: Damping matrix - symmetry, $\mathbf{K} \in \Re^{3\times 3}$: Stiffness matrix - symmetry, $\mathbf{c} \in \Re^{3\times 3}$: Output matrix - identity matrix.

The specific parameters used for this model are

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \\ \ddot{x}_3 \end{bmatrix} + \begin{bmatrix} 0.2 & -0.1 & 0 \\ -0.1 & 0.2 & -0.1 \\ 0 & -0.1 & 1.1 \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix}$$
$$+ \begin{bmatrix} 4 & -2 & 0 \\ -2 & 4 & -2 \\ 0 & -2 & 22 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} u_1 & 0 & 0 \\ 0 & u_2 & 0 \\ 0 & 0 & u_3 \end{bmatrix}, \tag{5.2}$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}. \tag{5.3}$$

Making the responses of impulse input is the first step of the ERA identification method which needs the Markov parameter and the Hankel matrix to store the given data. In this example, the sampling time of output response is 0.05 seconds. The required final time for getting good identified system must be larger than peak time of response.

In state-space form, the equations of motion can be written

$$\mathbf{A}_o = \begin{bmatrix} \mathbf{0}(3,3) & \mathbf{I}(3,3) \\ -\mathbf{K} & -\mathbf{D}_{amp} \end{bmatrix}, \tag{5.4}$$

$$\mathbf{B}_o = \begin{bmatrix} \mathbf{0}(3,3) \\ \mathbf{I}(3,3) \end{bmatrix}, \tag{5.5}$$

$$\mathbf{C}_o = \begin{bmatrix} \mathbf{I}(3,3) & \mathbf{0}(3,3) \end{bmatrix}, \tag{5.6}$$

$$\mathbf{D}_o = \begin{bmatrix} \mathbf{0}(3,3) \end{bmatrix}. \tag{5.7}$$

The subscript of '$_o$' denotes the original state-space matrix. With equations (5.4), (5.5), (5.6), (5.7), the '*impulse*' and '*ss*' commands in MATLAB are used to obtain the impulse responses as follows.

```
% Basic parameter
Ts = 0.05;                      % Sampling time
Tfinal=5;                       % Final time
T=0:Ts:Tfinal;                  % Time Duration
SYSo = ss(Ao,Bo,Co,Do);         % Continuous System

% Response for Impulse input
[yo,to,Xo] = impulse(SYSo,T);
```

Up to now, the simulated data - the responses due to impulse inputs, are obtained by the 'impulse' command in MATLAB in order to construct the Markov parameter and the Hankel matrix. In next sections, the simulated data will be used in the identification methods - ERA, Fast ERA, and AR models.

With above MATLAB codes, Figure 5.2 shows the simulated responses of the three degree of freedom example using the impulse inputs.

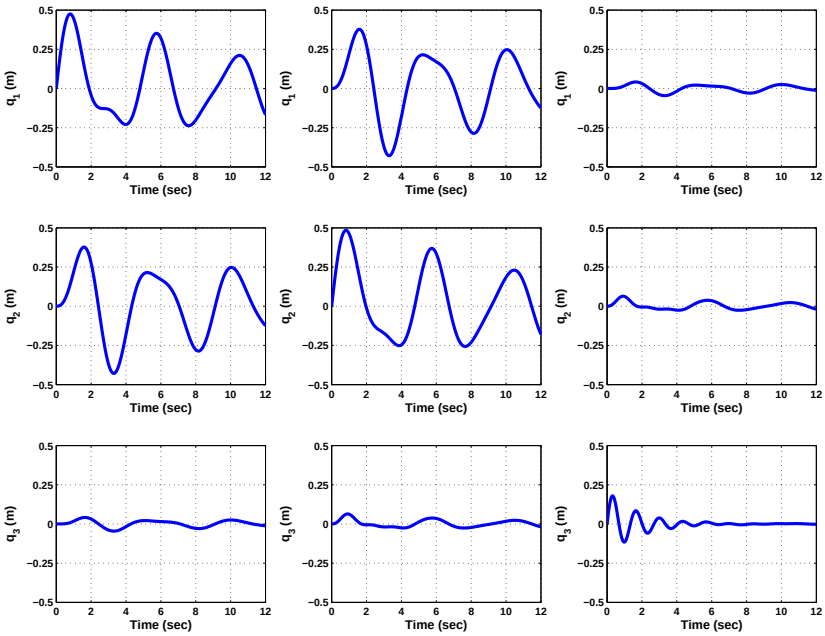

Figure 5.2: Simulation result of three springs-dampers-masses system without noises

5.2.2 The ERA method

Identification

The construction of the Hankel matrix with impulse responses is the first step for the ERA identification. The Hankel matrix consists of the Markov parameters - (2.12), i.e.,

$$\begin{aligned}
\mathbf{H}(0) &= \begin{bmatrix} yo(1,:,1)^T & yo(1,:,2)^T & yo(1,:,3)^T & yo(2,:,1)^T & yo(2,:,2)^T & yo(2,:,3)^T \\ yo(2,:,1)^T & yo(2,:,2)^T & yo(2,:,3)^T & yo(3,:,1)^T & yo(3,:,2)^T & yo(3,:,3)^T \end{bmatrix}, \\
\mathbf{H}(1) &= \begin{bmatrix} yo(2,:,1)^T & yo(2,:,2)^T & yo(2,:,3)^T & yo(3,:,1)^T & yo(3,:,2)^T & yo(3,:,3)^T \\ yo(3,:,1)^T & yo(3,:,2)^T & yo(3,:,3)^T & yo(4,:,1)^T & yo(4,:,2)^T & yo(4,:,3)^T \end{bmatrix}.
\end{aligned} \tag{5.8}$$

Instead of writing every response into the Hankel matrix $\mathbf{H}(0)$, an simple recursive subroutine function makes it easy to make the Hankel matrix. All subroutine functions are included in Appendix B.

It should be noted that the size of the Hankel matrix is an important factor to identify system with the ERA because the identified results will imitate the Hankel matrix. If the size of the Hankel matrix is not bit enough, the identified results cannot track the given data. A general rule is that the size of the Hankel matrix should be as large as twice as number of states of system (at least if the simulated data are used). If actual experimental data are used, the size of the Hankel matrix is usually determined by trial and error. However, in Chapter 7, we will describe a technique for finding the appropriate size of the Hankel matrix.

The Hankel matrix, $\mathbf{H}(0)$, is decomposed by the singular value decomposition, i.e., the '*svd*' command is used. After obtaining the Hankel matrix and the singular value matrices, let us get the identified matrix $\hat{\mathbb{A}}$, $\hat{\mathbb{B}}$, $\mathbf{C}$, and $\mathbf{D}$ using equations (2.22), (2.23), (2.24), and $\mathbf{D}$ which is $\mathbf{M}_{arkov}(0)$ as shown below.

```
[U,S,V]=svd(H0);                % Singular Value Decomposition
E=[eye(3,3), zeros(3,3)]';      % for Matrix C
E1=[eye(3,3)
    zeros(3,3)];                % for Matrix B
% Identified Matrix using ERA
```

```
Aie=S^(-0.5)*U'*H1*V*S^(-0.5);  % Identified A
Bie=S^(0.5)*V'*E1;              % Identified B
Cie=E'*U*S^(0.5);               % Identified C
Die=zeros(3,3);
```

In the MATLAB program, both of the transpose matrix and the complex conjugate transpose matrix can be formed by the '' command[1]. Those matrices `Aie`, `Bie`, `Cie`, and `Die` are based on discrete state-space model, where the subscript of '$_{ie}$' denotes the identification by the ERA. Therefore, those matrices must be transformed to continuous state-space model with '*d2c*' command.

```
SYSe = ss(Aie,Bie,Cie,Die,Ts);          % Continuous System
SYSe = d2c(SYSe);
% Response for Identified system (ERA)
[ye,te,Xe] = impulse(SYSe,T);
figure
impulse(SYSe,T)
```

The identified matrix `Bie` does not match with original matrix $\mathbf{B}_o$, because the realization of the system is not unique. With the above codes, following Figure 5.3 shows the results of identified system with the ERA method.

Observation

In general, we have a very good identified system, however making the Hankel matrix differently gives us slight different results. The equation (2.12) can be changed by choosing k=0 or k=1. Although this difference has not been explained in Chapter 2, several papers choose k=1 [45]. For getting better results, choose k=0 because data which are used in the ERA method include the first response of the original system.

When the result of the identified system is observed, the magnitudes of the identified responses are exactly same at the sampling time times as original response.

[1]The '*transpose*' command is for the transpose matrix and the '*ctranspose*' command is for the complex conjugate transpose (or Hermitian) matrix.

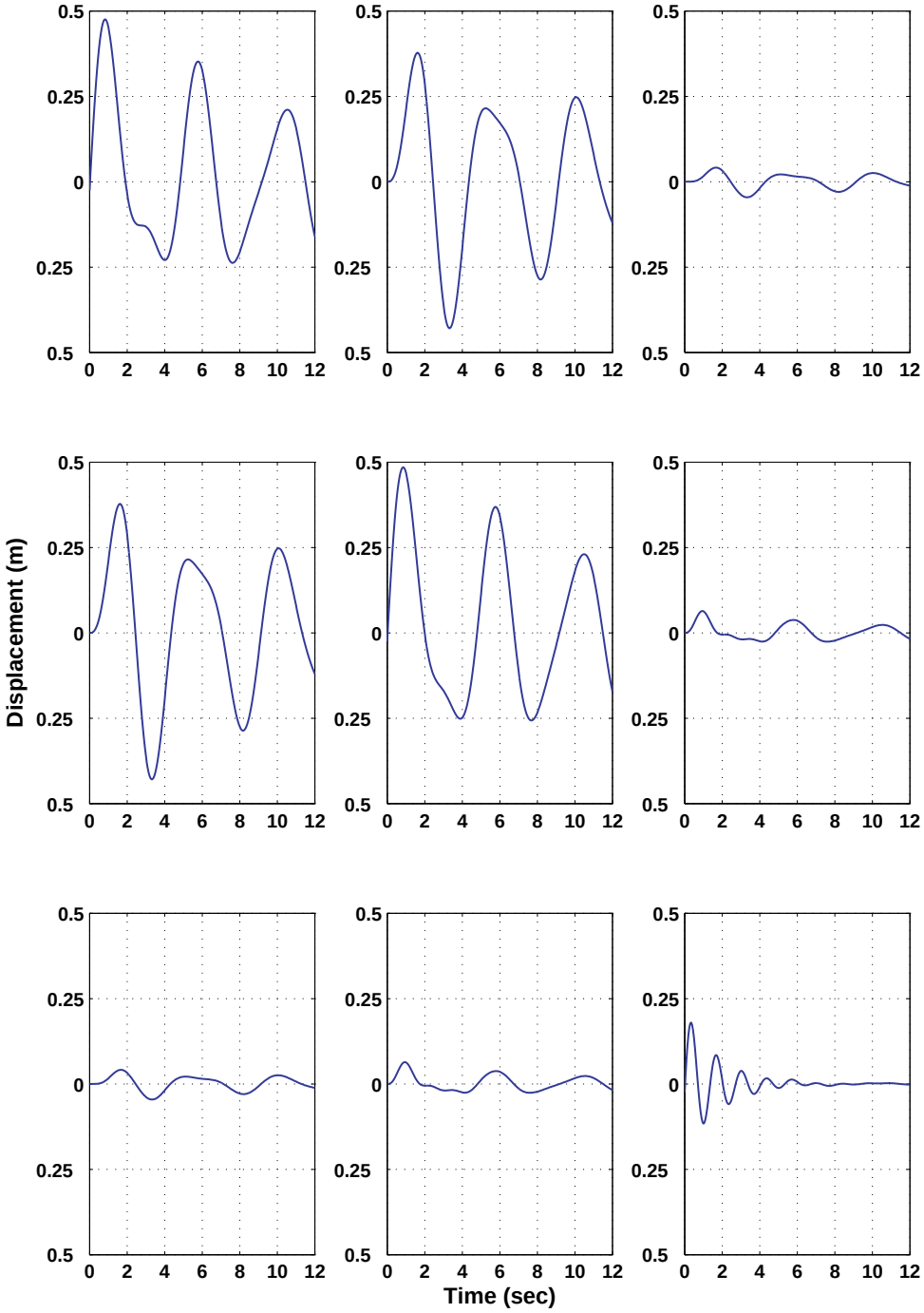

Figure 5.3: Identified system with ERA method

Extraction of Mass, Damping, and Stiffness matrices

For extracting $\mathbf{M}$, $\mathbf{D}_{amp}$, and $\mathbf{K}$ matrix from state-space matrix, equations (2.34) and (2.35) must be used as follows

```
MM = inv(Ce*Ae*Be)/Ts;
CCKK = -MM*Ce*Ae^2*inv([Ce;Ce*Ae]);
KK = CCKK(1:3,1:3);
CC = CCKK(1:3, 4:6);
```

According to above the codes, the derived mass, damper, and stiffness matrices can be obtained

```
>> era
===========
|   ERA   |
===========
MM =
    0.9958    0.0021    0.0000
    0.0021    0.9958    0.0021
    0.0000    0.0021    0.9774
KK =
    3.9792   -1.9834   -0.0041
   -1.9834    3.9751   -1.9465
   -0.0041   -1.9465   21.4978
CC =
    0.1990   -0.0992   -0.0002
   -0.0992    0.1988   -0.0973
   -0.0002   -0.0973    1.0749
>>
```

Most elements are almost same with those of the original matrices. The average of error percentages is less than 1.2%. The extraction method presented above is associated with

the singular values decomposition as introduced in [101]. Another extraction method which is presented in [3] is the following procedure. The eigenvalues of damping and stiffness matrices must be calculated using equation (2.42).

```
% To Change continuous system from exp(AcT), need similarity transformation
[Ve_Aie, Va_Aie]=eig(Aie);
% To Compare between (Eigenvalues of Damping and Stiffness Matrix)
%                 and (Eigenvalues from Matrix A)
lambda_ie = log(inv(Ve_Aie)*(Aie)*Ve_Aie)/Ts;

for i=1:chno
    % eigenvalues of Damping matrix
    D_eigvalue_ie(i) = 2*real(lambda_ie(i,i));
    % eigenvalues of Stiffness matrix
    K_eigvalue_ie(i) = -real(lambda_ie(i,i))^2 - imag(lambda_ie(i,i))^2;
end
```

The values of `D_eigvalue_ie(i)` and `K_eigvalue_ie(i)` mean eigenvalues of damping matrix and stiffness matrix, respectively. With those values and normalization mass matrix, damping and stiffness matrices can be obtained. Table 5.1 compares $\mathbf{D}_{amp}$ and $\mathbf{K}$ matrix

Table 5.1: Table for comparing $\mathbf{D}_{amp}$ and $\mathbf{K}$ matrix between derived and original values

Eigenvalues of	Original Values	Derived Values
Damping Matrix	0.0949, 0.2940, 1.1111	0.0949, 1.1111, 0.2940
Stiffness Matrix	1.8980, 5.8798, 22.2222	1.8980, 22.2222, 5.8798

between original values and derived values. Those values are exactly same.

5.2.3 The Fast ERA method

Identification

Because the matrix D_q is an inner product of the Hankel matrix, and the D_q^* is partition of D_q, we have to gather more output data than the ERA case for making the Hankel matrix. Then, make D_q , D_{q-1} and eigenvectors and eigenvalues of D_{q-1} with (3.9) and (3.10), i.e.,

```
% Hankel matrix
H0f=[yo(1:9,:)'
    yo(2:10,:)'
    yo(3:11,:)'];
D_q=H0f*H0f';                        % D_q
D_q1=H0f(1:6,1:9)*H0f(1:6,1:9)';     % D_(q-1)
[D_q1_v, D_q1_e]=eig(D_q1);          % Eigen decomposition
```

The fundamental matrices, D_q , D_{q-1} and eigenvalues, lead us to get the equations (3.22), (3.20), (3.23), and (3.26) as

```
P_q1pinv=D_q1_e^(-0.5)*D_q1_v';   % Pseudo inverse(P_{q}^1)
P_q1=D_q1_v*D_q1_e^(0.5);         % P_q^1
D_qs= D_q(4:9, 1:6);              % D_q*
P_q2 = D_qs*P_q1pinv';            % P_q^2
```

With same logic used in the ERA method, get discrete time identified state-space matrices $\mathbf{A}_{id}$, $\mathbf{B}_{id}$, $\mathbf{C}_{id}$, and $\mathbf{D}_{id}$.

```
%Identified system
Aif=P_q1pinv*P_q2;              % Discrete time basis
Bif=P_q1pinv*H0f(1:6,1);        % Discrete time basis
Cif=P_q1(1:3,1:6);              % Discrete time basis
sysif=ss(Aif,Bif,Cif,Do,Ts);    % Discrete time state-space formation
```

The transformation from a discrete time system to a continuous time system is necessary for comparing to the original data. The identified state-space model should be transformed with 'd2c' command in MATLAB.

```
SYSIf=d2c(sysif);                % Discrete to Continuous
[Af,Bf,Cf,Df]=ssdata(SYSIf);     % Continuous time state-space formation
```

Figure 5.4 shows the time responses of the identified system with the Fast ERA.

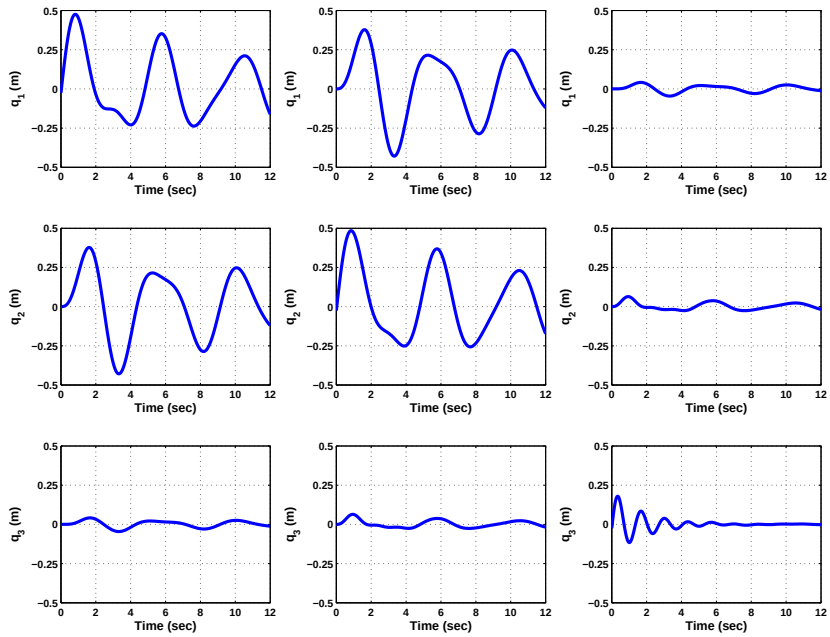

Figure 5.4: Identified System with Fast ERA

Observation and Extraction

The results of the Fast ERA are mostly same with those of the ERA method.

The procedure of extracting $\mathbf{M}$, $\mathbf{D}_{amp}$, and $\mathbf{K}$ matrix is starting on the continuous time state-space matrix. Then the rest of the procedures are the same as the ERA method. Also, the extracted mass, damper, and stiffness matrices with the Fast ERA are almost same with the ERA method. The Fast ERA method also has great result in terms of extraction. Table 5.2 shows the derived results - eingenvalues of the damping and stiffness matrix.

Table 5.2: Table for comparing $\mathbf{D}_{amp}$ and $\mathbf{K}$ matrix between derived and original values

Eigenvalues of	Original Values	Derived Values
Damping Matrix	0.0949, 0.2940, 1.1111	1.1111, 0.2940, 0.0949
Stiffness Matrix	1.8980, 5.8798, 22.2222	22.2222, 5.8798, 1.8980

Advantages and Disadvantages

There is no obvious advantages or disadvantages between the ERA and the Fast ERA method because they use same concepts for identifying systems. First of all, in terms of programming coding, the ERA method is easier than the Fast ERA because of complexity of establishing basic matrices. The ERA method just considers whether or not **U** and **V** of singular value decomposition are square for calculating easily. The Fast ERA case must be considered D_{q-1}.

5.2.4 The AR model

The ARX model can identify Multi-Inputs-Multi-Outputs system with a basic MATLAB command. The ARMAX model can be applied to only Single-Input-Single-Output systems. For using ARMAX command in MATLAB, therefore, Multi-Inputs-Multi-Outputs systems must be separated into Single-Input-Single-Output systems, that means ARMAX model must decouple real system because of noises effect and inputs. Because ARMAX can not be applied to MIMO system, ARX method will use for identifying example system in this section.

The ARX Model

After getting the impulse response of 3-DOF example system, data of system - inputs and outputs are required for using '*arx*' command in MATLAB. Therefore, '*iddata*' command will be used to get data of system. The '*pem*' command also will be used for estimate the parameters of general linear model. In addition, the sampling time will be 0.05 seconds

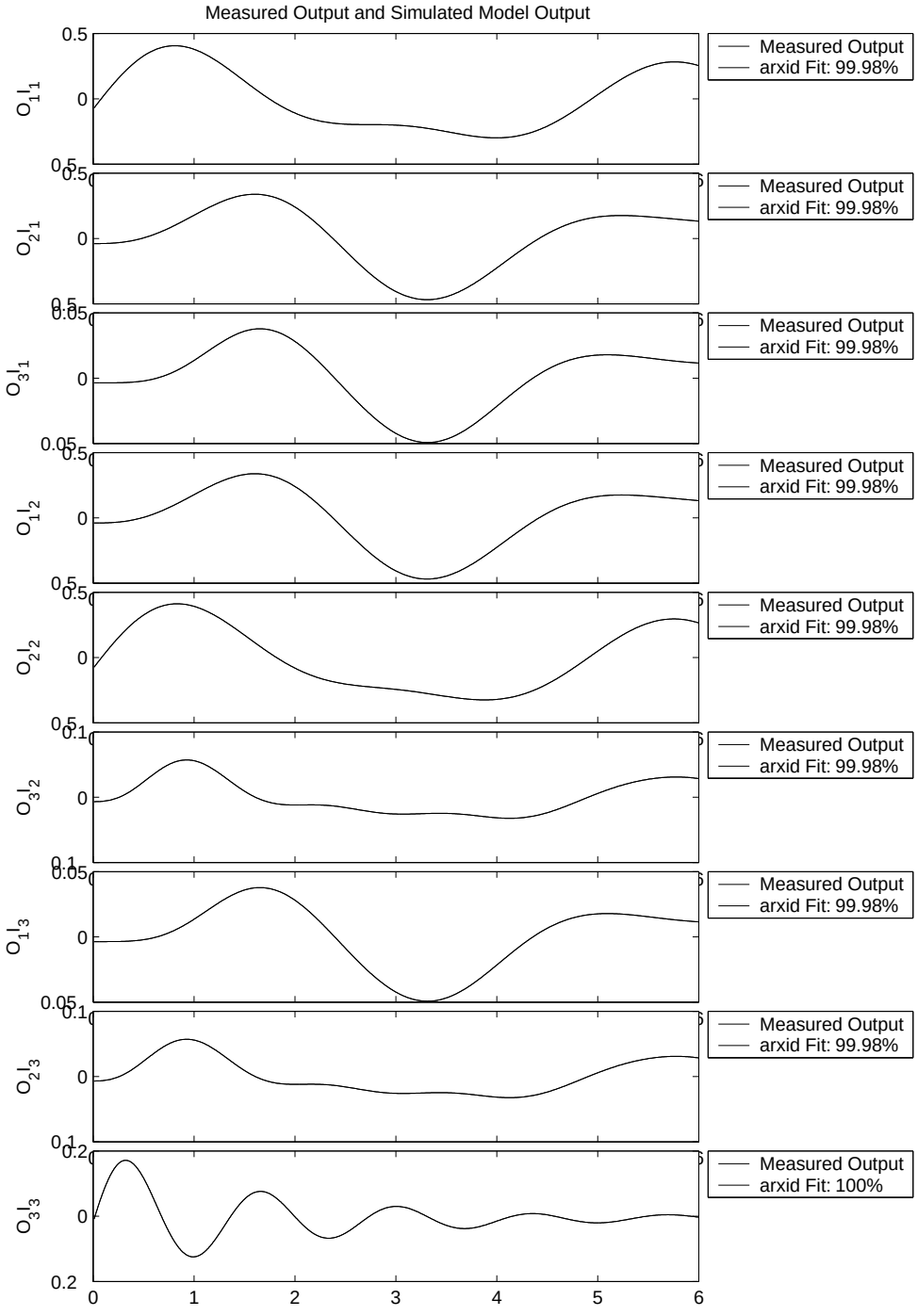

Figure 5.5: Identified System with ARX

which is same sampling time with the ERA or the Fast ERA method.

```
% sp_damp: name of data
sp_damp = iddata([yo(:,:,1) yo(:,:,2) yo(:,:,3)], [u u u], Ts);
noo=600;
ze = sp_damp(1:noo);

% Orders of System
na=6*ones(9,9);            % ny|ny size
nb=3*ones(9,3);            % ny|nu size
nk=1*ones(9,3);
orders=[na nb nk];         % No. of rows must be same with no. of outputs
```

Above codes are showing the way how to set up the fundamental matrix for using the ARX command. 'na', 'nb', and 'nk' mean orders of each polynomial corresponding to the A, B, and K respectively. With orders, ARX command will be used like the following codes.

```
arxid=arx(ze,orders);
figure
compare(ze, arxid)
```

The '*compare*' command is used for comparing identified response with original data. Figure 5.5 shows the result of the 'compare' command.

5.3 3-DOF System with noise

In the previous section, the three degree of freedom mechanical example does not include noise. Most sensors generally produce noisy signals. As a result, we need to consider measurement noises for evaluating the each identification method. In this section, white noise is added to all simulated responses. Figure 5.6 shows a block diagram for adding some white noises to system.

All procedures for identifying the system are exactly same with the previous example. However, some results - extraction of $\mathbf{M}$, $\mathbf{D}_{amp}$, and $\mathbf{K}$ matrix, are different due to measurement noises.

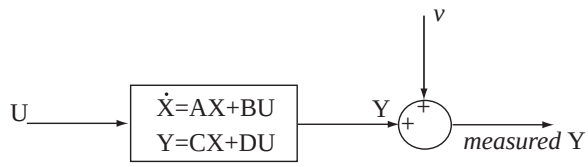

Figure 5.6: Block diagram for adding white noise to state-space model system.

Let us use the MATLAB command for a white noise. The white noises can be made with '*awgn*'[2] command in MATLAB and added to each response of the original 3-DOF example system. Basically the 'awgn' command can change a signal-to-noise ratio. By the trial and error, the best ratio for watching the effect of noises can be chosen around 25.

5.3.1 The ERA Method

Identification

The procedures of the ERA identification for noisy simulation system are exactly same with subsection 5.2.2. However, the size of state-space model must be larger than that of the previous example due to the effect of the white noises. As a result, more data, i.e., the bigger Hankel matrix, are required. Even though, however, many data are used to identify, the whole response of identified system can be unstable due to the noises. The unstable results can be explained by the singular value decomposition of the Hankel matrix[3]. Here, the Markov parameters and the Hankel matrix are as follow,

$$\begin{aligned}
\mathbf{M}_{arkov}(k) &= \begin{bmatrix} y_1(kT_s) & y_2(kT_s) & \cdots & y_{in}(kT_s) \end{bmatrix}, \\
\mathbf{H}_{qd}(0) &= \begin{bmatrix} \mathbf{M}_{arkov}(1) & \mathbf{M}_{arkov}(2) & \cdots & \mathbf{M}_{arkov}(d) \\ \mathbf{M}_{arkov}(2) & \mathbf{M}_{arkov}(3) & \cdots & \mathbf{M}_{arkov}(d+1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}_{arkov}(q) & \mathbf{M}_{arkov}(q+1) & \cdots & \mathbf{M}_{arkov}(q+d+1) \end{bmatrix}. \qquad (5.9)
\end{aligned}$$

where

[2]The 'awgn' command uses a white gaussian noise.

[3]In observation section, more detail explanation will be presented.

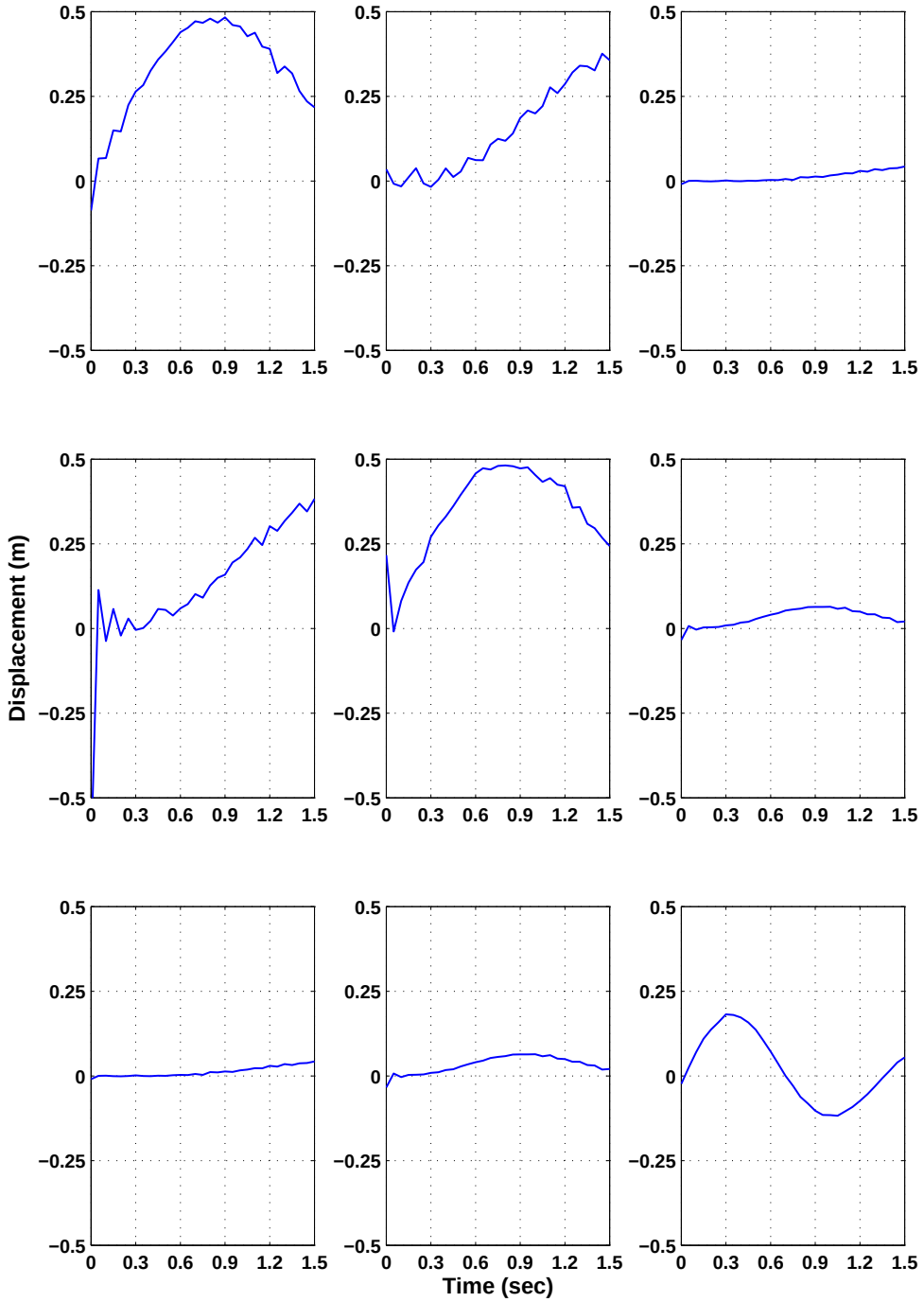

Figure 5.7: Identified System with ERA when size of Hankel matrix is 90 until 1.5 seconds.

$$k = 0,\ 1,\ 2,\ \cdots, \text{ and } T_s \text{ is the sampling time.}$$

The interval, in which identified responses are tracking the used data, determines the size of the Hankel matrix. Then the size of the state-space model is determined by the size of the Hanekel matrix, i.e., q and d in the equation (5.9) are determined. As many data are used, the tracking interval will increase. In this subsection, the size of the Hankel matrix is 90. This is determined by trial and error. The last time of given data can be calculated as

$$\begin{aligned} T_{final} &= T_s \times Size\ of\ Hankel\ Matrix/Number\ of\ Outputs \\ &= 0.05 \times 90/3 = 1.5 \end{aligned} \tag{5.10}$$

Until the calculated tracking last time (1.5 seconds), the response of the identified system will be obtained as Figure 5.7. In this interval, the response of the identified system looks like stable. However, the identified system is unstable as shown Figure 5.8.

The two figures 5.7 and 5.8 show that the ERA method can make identified system unstable due to the noise. According to Figure 5.7, however, the used data which are elements of the Hankel matrix are tracked very well in the time interval $[0, T_{final}]$. A technique for finding a stable identified model is to use a model order reduction technique as described below.

Extraction and Reduction

Without the model reduction, the extraction of the characteristics of the system with equations (2.34) and (2.35), i.e., the extraction of $\mathbf{M}$, $\mathbf{D}_{amp}$, $(\mathbf{G}(\omega))$ and $\mathbf{K}$ matrices, is meaningless because the number of the original states are not matching with the number of the identified states. In addition, from Figures 5.2 and 5.8, the response of the identified system is unstable unlike original data. As a result, the model reduction algorithm of the state number for extracting those matrices is required.

However, a proposed model reduction algorithm is not introduced yet. Let us assume that the system order is determined by a suitable model reduction algorithm. In Chapter 7, the new proposed model reduction algorithm will be introduced.

Figure 5.9 illustrates the singular values of the Hankel matrix. The dominant singular

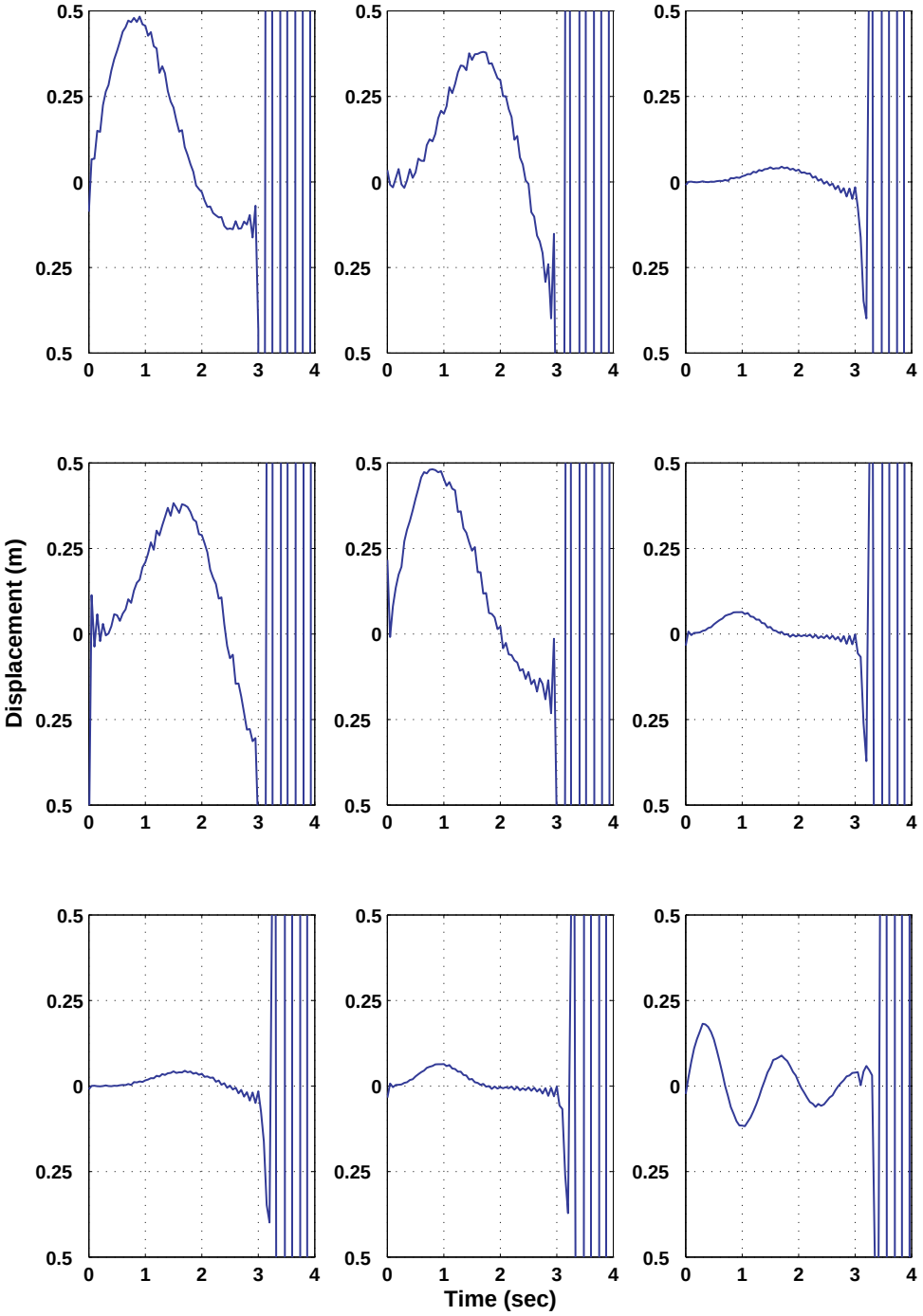

Figure 5.8: Identified System with ERA when size of Hankel matrix is 90 after 1.5 seconds.

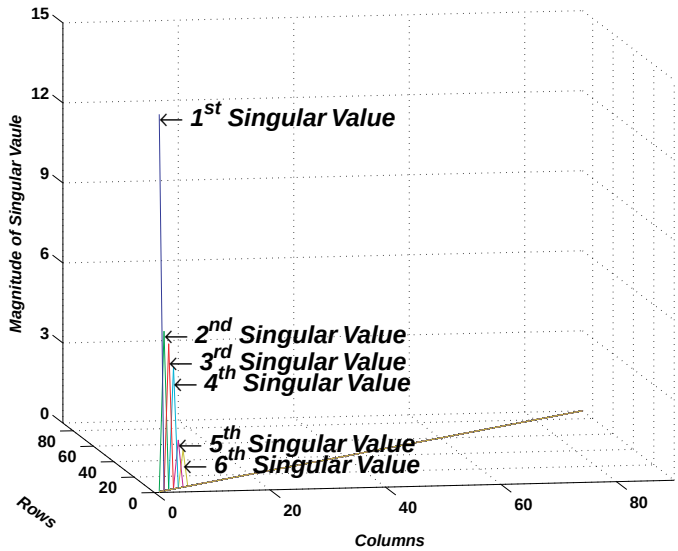

Figure 5.9: Singular Values of *Hankel matrix*

values are the first 6 singular values[4]. The others are relatively small and negligible. Thus we assume a model order, N_{order}, is 6.

The identified state-space $\hat{\mathbb{A}}$ matrix (2.22) can be expressed by the following equation[5].

$$\hat{\mathbb{A}}_{id} = \mathbf{\Sigma}_{qd}(1:6,1:6)^{(-0.5)}\mathbf{U}_q(:,1:6)^T\mathbf{H}_{qd}(1)\mathbf{V}_d(:,1:6)\mathbf{\Sigma}_{qd}(1:6,1:6)^{(-0.5)}. \quad (5.11)$$

Note that this matrix is only 6×6 composed to the original 90×90 Hankel matrix, moreover, the response obtained using this matrix $\hat{\mathbb{A}}_{id}$ matches the response of the original system as shown in Figure 5.10.

Also, the reduced matrix is stable while 90×90 matrix is unstable.

For extracting $\mathbf{M}$, $\mathbf{D}_{amp}$, and $\mathbf{K}$ matrix from the state-space matrix, equations (2.34) and (2.35) must be used. Then following results can be obtained easily. Note that the extraction results are acceptable as shown Table 5.3.

```
MM =
```

[4]Picking up the system order by the singular values of the Hankel matrix is a popular way.

[5]The detail explanation will be in Chapter 7

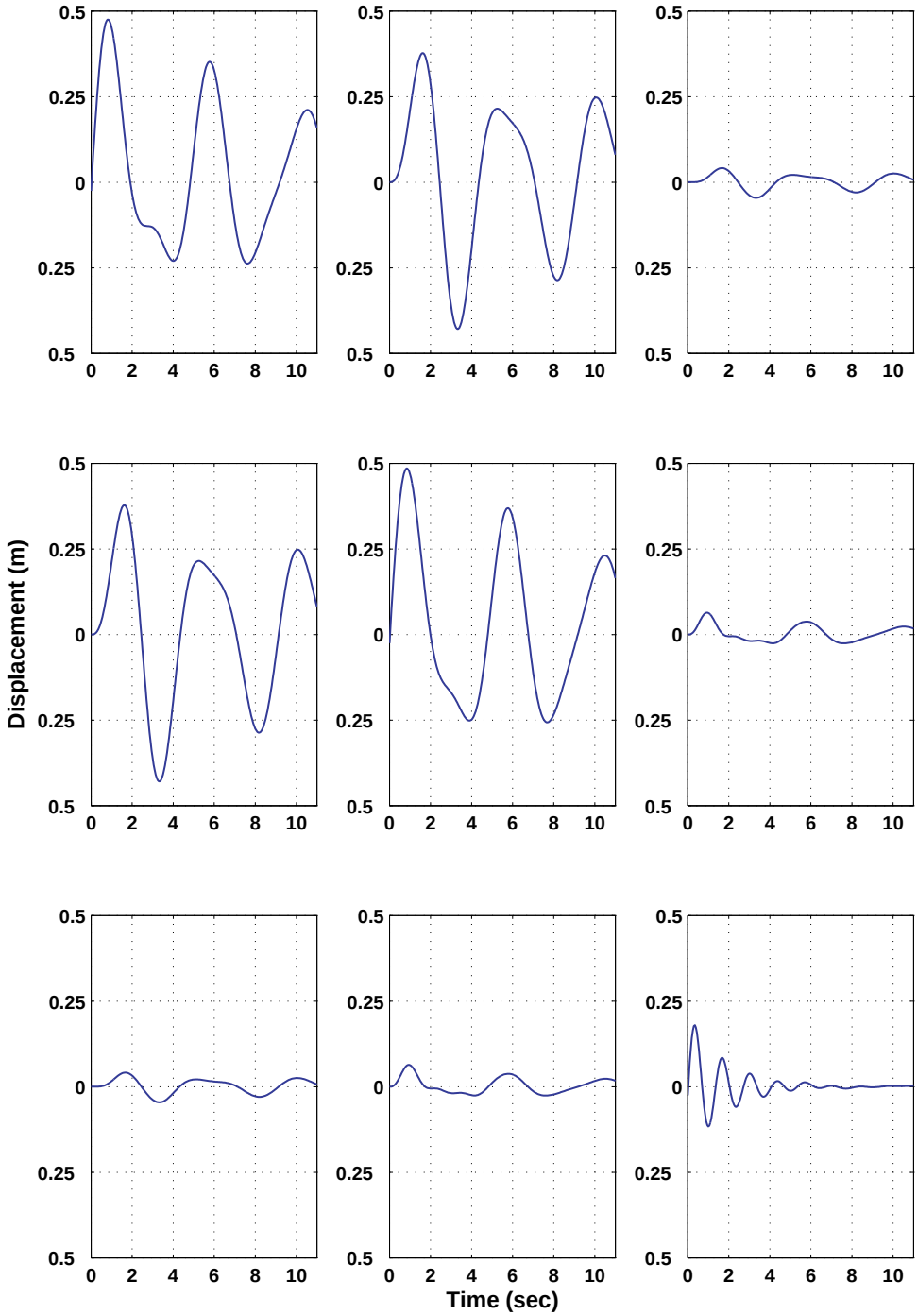

Figure 5.10: Identified System with ERA when size of state-space model (A) is 6 after reduction

Table 5.3: The extracted eigenvalues of damping and stiffness matrix with ERA identified and reduced state-space model.

Eigenvalues of	Damping Matrix	Stiffness Matrix
Original Values	0.0949, 0.2940, 1.1111	1.8980, 5.8798, 22.2222
Identified Values	0.0896, 0.2946, 1.1132	1.8909, 5.9259, 22.2425
Error Percentages	5.5%, 0.204%, 0.189%	0.374%, 0.78%, 0.091%

```
    0.9982    0.0029    0.0012
    0.0004    0.9959    0.0012
   -0.0002    0.0025    0.9765

KK =
    3.9845   -1.9859   -0.0012
   -1.9878    3.9780   -1.9491
   -0.0027   -1.9439   21.4836

CC =
    0.1973   -0.0994    0.0009
   -0.0987    0.1981   -0.0957
   -0.0009   -0.0983    1.0751
```

Also, when eigenvalues of the damping and stiffness matrices are extracted, the correct results can not be obtained with 90×90 Hankel matrix because the responses with 90×90 state-space model are unstable. However, if the reduced state-space model is used, the eigenvalues of damping and stiffness matrices are same with original damping and stiffness matrices. Table 5.3 shows the extracted eigenvalues.

5.3.2 The Fast ERA Method

Identification

For getting the result of the Fast ERA identification method, we must follow the procedure of subsection 5.2.3. However, the size of D_q must be larger than that of the subsection 5.2.3 due to noise effect same as the ERA method. Most conditions of the Fast ERA identification methods are similar with the ERA method. Figure 5.11 shows the response of tracking interval which can be calculated same with equation (5.10).

From Figure 5.11, the identified responses are tracking the original responses even with white noise. However, the ultimate identified system is unstable as in the case of the ERA method. The Fast ERA method also needs to be reduced to obtain a stable identified system.

Extraction and Reduction

In section 3.5, some differences and common factors between the ERA and the Fast ERA are explained in detail. The Fast ERA uses the eigenvalue decomposition instead of singular value decomposition. Most eigenvalue decomposition algorithms make eigenvalues ascending order, which means that the last eigenvalue must be used for reducing state-space model.

Figure 5.12 shows the eigenvalues of D_q. Unlike singular values, Figure 5.9, eigenvalues are sorted in ascending order. Therefore, when the size of state-space model is reduced, the largest several eigenvalues are used. The 85th, 86th, 87th, 88th, 89th, and the last eigenvalues[6] and eigenvectors which is related to largest eigenvalues, are used for reducing state-space model.

Then, with the equations (2.34) and (2.35), we can obtained the following mass, damping, and stiffness matrices.

```
MM =

    1.0127    0.0043    0.0019

   -0.0117    0.9975   -0.0089
```

[6]Those eigen values are already checked in Figure 5.12.

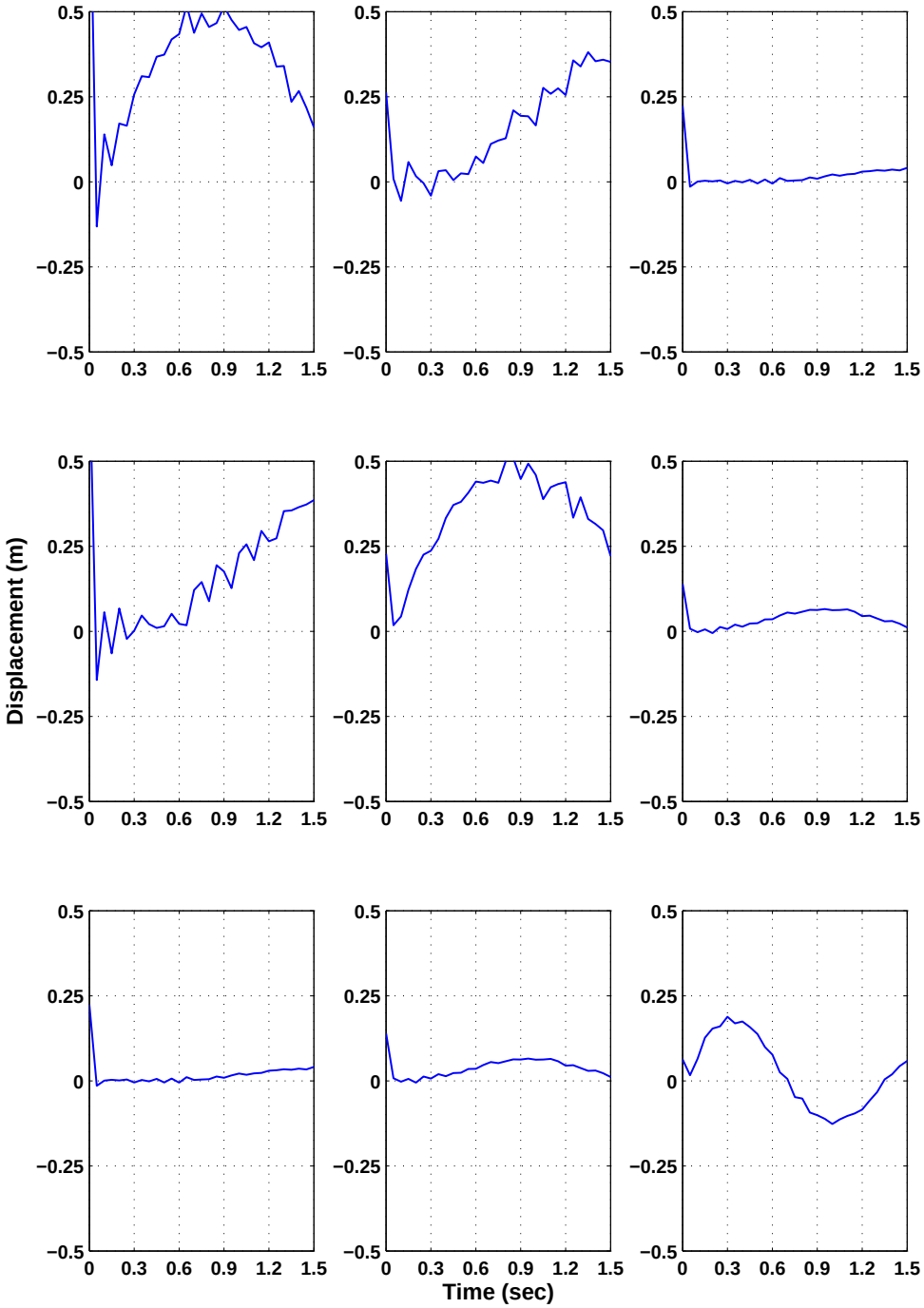

Figure 5.11: Identified System with Fast ERA when size of Hankel matrix is 90 until 1.5 seconds.

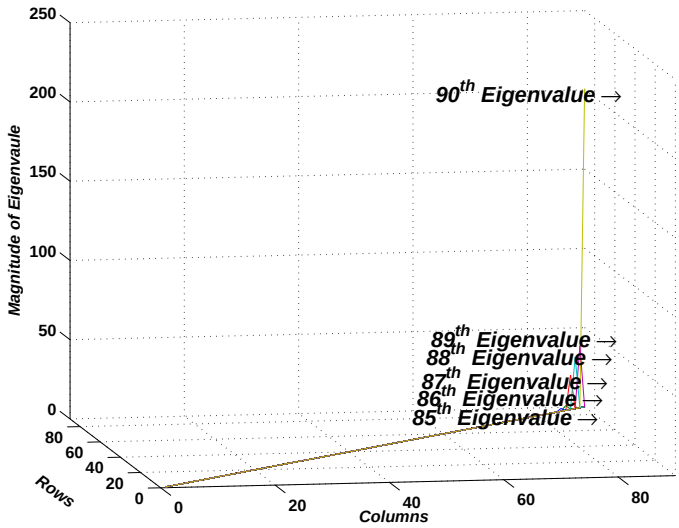

Figure 5.12: Singular Values of D_q

```
    0.0008    0.0076    0.9842
KK =
    4.0276   -1.9751    0.0695
   -2.0696    4.0242   -2.0027
    0.0144   -1.9228   21.5761
CC =
    0.1738   -0.1346   -0.0149
   -0.0694    0.1809   -0.0983
   -0.0170   -0.1246    1.0897
```

The reason why we can reduce the state-space model with the same logic with the ERA method, is that eigenvalue decomposition is similar to singular value decomposition.

Table 5.4 compares the derived eigenvalues of damping and stiffness matrices from reduced state-space model. Most eigenvalues of damping and stiffness matrix match the expected first eigenvalue of damping matrix, which means that the reduced state-space model should have the same response with the original model. In addition, Figure 5.13 illustrates the response of the reduced-identified state-space model with the Fast ERA method.

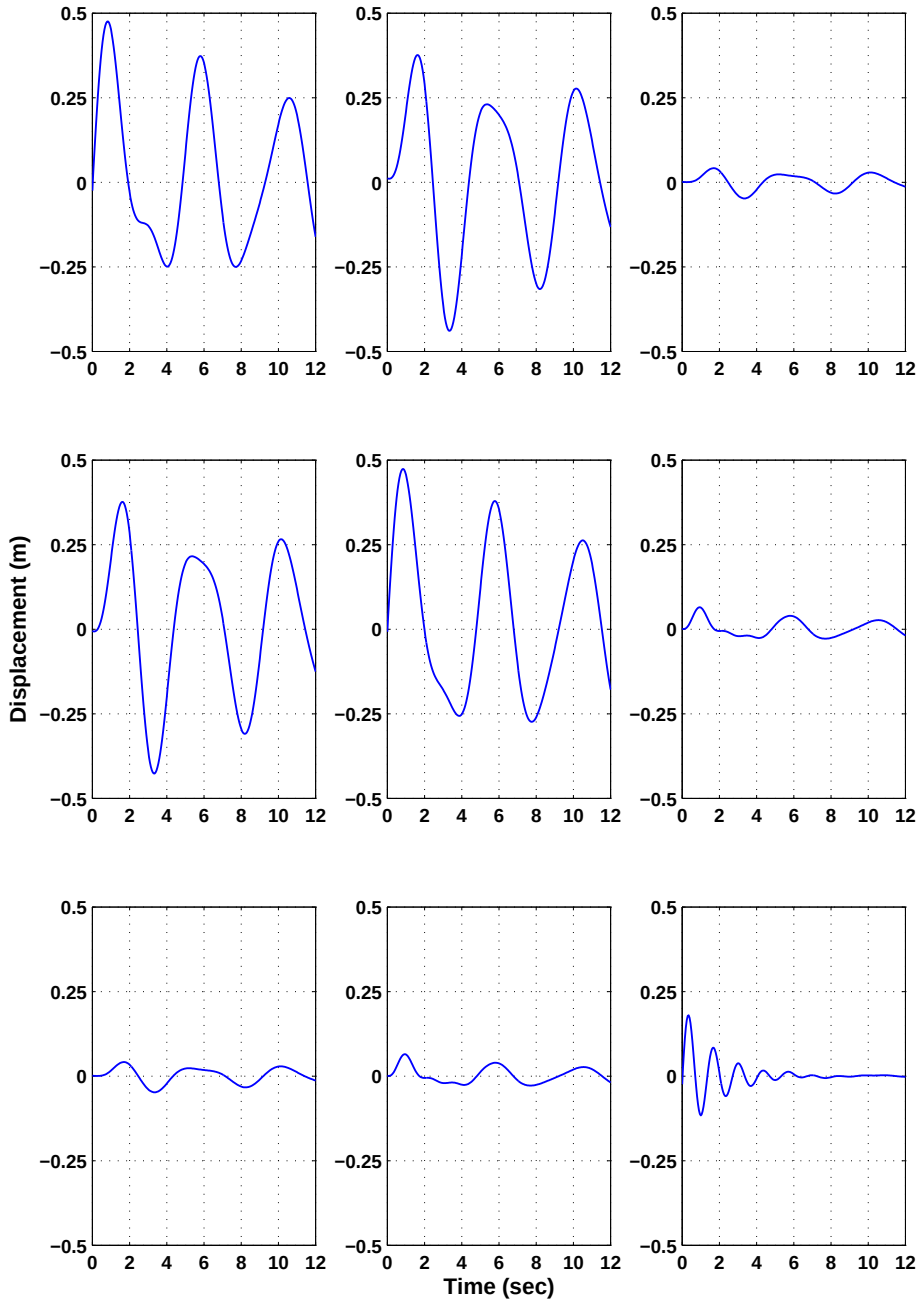

Figure 5.13: Identified System with Fast ERA when size of state-space model is 6 after reduction

Table 5.4: The extracted eigenvalues of damping and stiffness matrix with the Fast ERA identified and reduced state-space model.

Eigenvalues of	Damping Matrix	Stiffness Matrix
Original Values	0.0949, 0.2940, 1.1111	1.8980, 5.8798, 22.2222
Identified Values	0.1107, 0.2957, 1.1036	1.9354, 5.7962, 21.985
Error Percentages	16.65%, 0.578%, 0.675%	1.97%, 1.42%, 1.06%

5.3.3 The AR Model

Identification

The procedure of identifying 3-DOF system with the ARX are same way with 5.2.4. The reason why the '*arx*' command of MATLAB is not affected by white noise very much in this example, is that the AR models use the probability theory. In AR models, considering the white noise with the probability theory makes it possible to reject the white noise. Therefore, the number of poles and zeros is not changed. Figure 5.14 shows the response of the ARX identified model. Basically all responses of the system are with the same those of the AR identified system as explained in Section 5.2.4.

5.4 Gyroscopic System

For introducing the complex variable coordinate system, let us use a simple gyroscopic system developed by J.A. Wickert [97]. His original model has three degree of freedom, q_1, q_2, and q_3, however, a two degree of freedom for a model of a rotating shaft like Figure 5.15 can be derived. The gyroscope system has the equations of motion (5.1) and the matrix formulation.

$$\mathbf{M}\ \ddot{q} + (\mathbf{D}_{amp} + \mathbf{G}(\omega))\ \dot{q} + \mathbf{K}\ q = \mathbf{E}\bar{f}. \tag{5.12}$$

where

$$\mathbf{M} = M \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad (\mathbf{D}_{amp} + \mathbf{G}(\omega)) = 2\Omega \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{K} = \begin{bmatrix} \omega_0^2 - \Omega^2 & -\dot{\Omega} \\ \dot{\Omega} & \omega_0^2 - \Omega^2 \end{bmatrix}$$

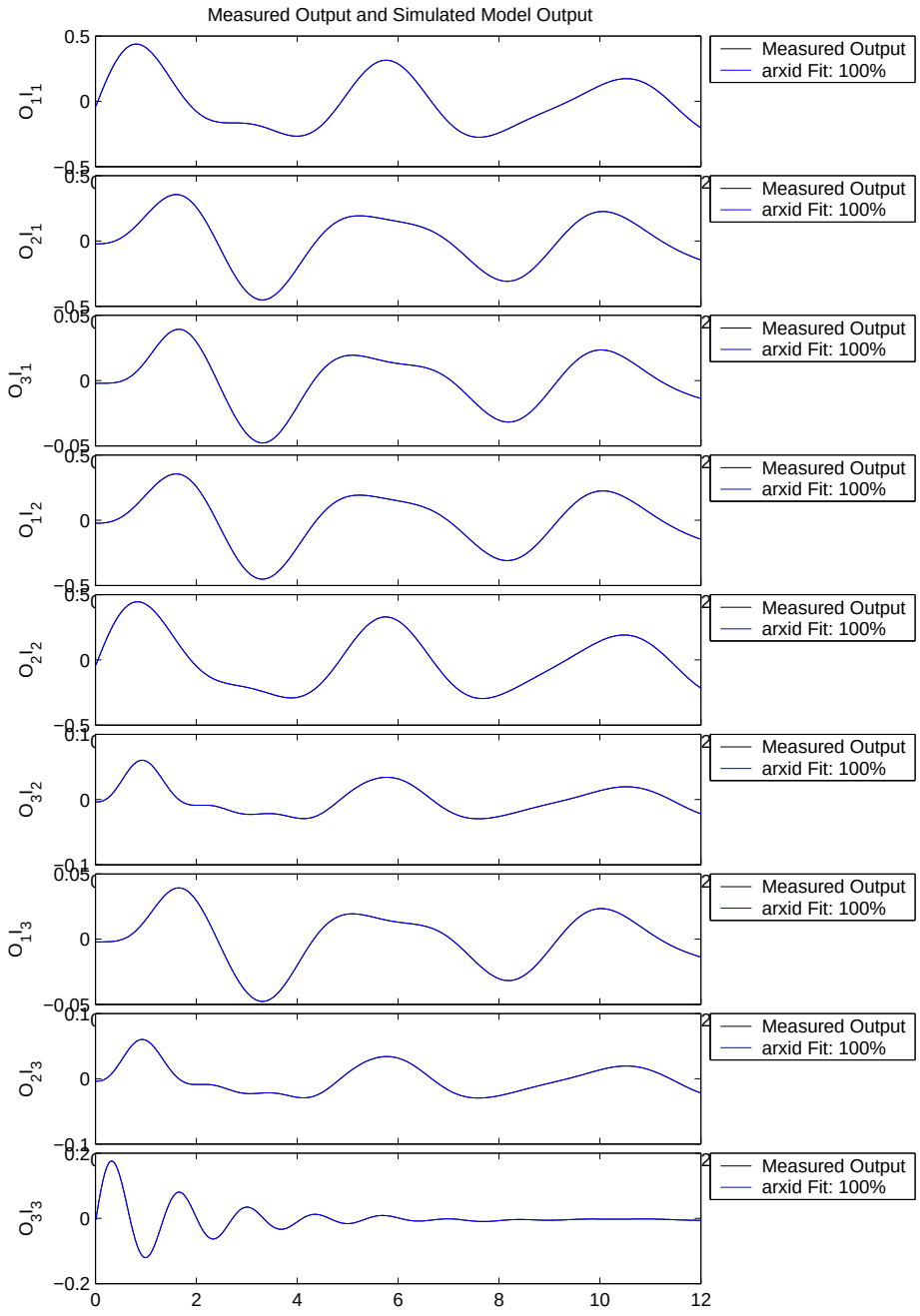

Figure 5.14: Comparison between ARX identified and original system with white noise

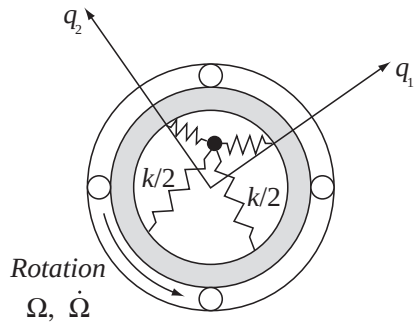

Figure 5.15: Two degree of freedom model for a shaft with non-constant angular velocity

This model will be used in Chapter 6 to explain a new approach to the identification of gyroscopic systems.

Chapter 6

COMPLEX VARIABLE COORDINATE SYSTEM WITH ERA

6.1 Introduction

This chapter presents a use of the complex coordinates ($z = x + i \cdot y$) for identifying a linear dynamic model of a Flywheel Energy Storage System (FESS) which contains gyroscopic terms. The complex coordinate system transformation eliminates the skew-symmetric gyroscopic elements in equations of motion, and produces a more compact representation of the equations of motion. As a result of this transformation, the non-square gyroscopic system, such as the FESS, can be identified easily with the complex coordinated state-space model.

This chapter presents some basic results from the linear system theory using state-space models whose state variables are represented by complex variables, i.e., the initial condition response and the forced response are derived. It is shown that concepts such as controllability and observability can be applied to this complex variable system representation. In addition, it is shown that the Eigensystem Realization Algorithm (ERA) can also be applied to these systems.

6.1.1 Idea of Complex Variable Transformation

Over the last ten years some research at the University of Washington has focused on the development of FESS [64], [91], and [63]. Throughout the research of the FESS, we have verified and identified our theoretical approaches and developed designs of FESS and control systems. Here we present a new approach to the identification of the FESS.

It should be noted that several studies have used complex variable coordinate systems to model rotor-dynamic systems. For example, Genta [27] and Goodwin [30] studied the rotor-dynamics focusing on the vibration of the structures. However, we are unaware of previous studies that have used complex variable coordinates for system identification. In addition, Kwok [56] showed some mathematical applications of the complex variables in

system analysis.

The identification of dynamic systems with gyroscopic elements have largely utilized an approach based on time series analysis, e.g., [39], [72], [87]. Here, we will consider a state-space, deterministic identification method based on the Eigensystem Realization Algorithm (ERA) [49].

The development proceeds as follows. Section 6.1.2 provides the linear equations of motion using real state variables, and complex state variables. Section 6.2 shows that many of the results from the linear systems theory, based on real state variables coordinates, can be applied to systems described by complex state variables. In particular, he following results for linear systems with a complex variable state-space representation are developed; (i) Initial condition response, (ii) Forced (impulse) response, (iii) Controllability and Observability matrices, and (iv) The Eigensystem Realization Algorithm. The technique developed here are applied in Chapter 8 via a numerical example and an experimental example that uses ERA with complex variable states to identify the FESS.

6.1.2 Complex Coordinates for Gyroscopic System

A gyroscopic system is introduced in subsection 5.4. The system will be used for verifying the control theory and a new identification technique with complex variable coordinates. The equations of motion of the gyroscopic system can be transformed into the complex variable coordinates (see [27]) in a more compact form with

$$x = q_1 + i \cdot q_2, \tag{6.1}$$

where $i = \sqrt{-1}$. Recall that q_1 and q_2 are the states of the gyroscopic example in Section 5.4.

To apply the transformation (6.1) to the equations of motion for the gyroscopic example, multiplying the second equation of (5.12) by the imaginary unit $i = \sqrt{-1}$ and adding it to the first equation of (5.12), reduces the equations of motion to the following

$$M\ddot{x} + (2\omega i)\dot{x} + [(\omega_0^2 - \Omega^2) + (\dot{\Omega} i)]x = F_{q_1} + i \cdot F_{q_2}. \tag{6.2}$$

The equations of motion, i.e., (5.12) and (6.2), will be used for comparing and verifying the computational results with a new proposed identification method throughout this chapter. For numerical evaluation, let us consider a specific system with

$$M = 10,\ k = 1000,\ \omega_0 = \sqrt{\frac{k}{M}} = 10,\ \Omega = 1,\ \dot{\Omega} = 0. \tag{6.3}$$

With the real coordinates system, the equations of motion (5.12) and numerical values (6.3), yield the state-space model with coefficients,

$$\begin{aligned}
\mathbf{M}_{real} &= \begin{bmatrix} M & 0 \\ 0 & M \end{bmatrix} = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}, \\
(\mathbf{G}(\Omega) + \mathbf{D}_{amp})_{real} &= \begin{bmatrix} 0 & -2\Omega \\ 2\Omega & 0 \end{bmatrix} = \begin{bmatrix} 0 & -2 \\ 2 & 0 \end{bmatrix}, \\
\mathbf{K}_{real} &= \begin{bmatrix} \omega_0^2 - \Omega^2 & -\dot{\Omega} \\ \dot{\Omega} & \omega_0^2 - \Omega^2 \end{bmatrix} = \begin{bmatrix} 99 & 0 \\ 0 & 99 \end{bmatrix}, \\
\mathbf{A}_{real} &= \begin{bmatrix} \mathbf{0}_{(2,2)} & \mathbf{I}_{(2,2)} \\ -\mathbf{M}_{real}^{-1}\mathbf{K}_{real} & -\mathbf{M}_{real}^{-1}\left(\mathbf{G}(\Omega) + \mathbf{D}_{amp}\right)_{real} \end{bmatrix}, \\
\mathbf{B}_{real} &= \begin{bmatrix} \mathbf{0}_{(2,2)} \\ \mathbf{I}_{(2,2)} \end{bmatrix},\ \mathbf{C}_{real} = \begin{bmatrix} \mathbf{I}_{(4,4)} \end{bmatrix},\ \mathbf{D}_{real} = \begin{bmatrix} \mathbf{0}_{(4,2)} \end{bmatrix}.
\end{aligned} \tag{6.4}$$

Also, with the complex coordinates system, the equations of motion (6.2) and numerical values (6.3) yield the coefficients,

$$\begin{aligned}
\mathbf{M}_{complex} &= \begin{bmatrix} M \end{bmatrix} = \begin{bmatrix} 10 \end{bmatrix}, \\
(\mathbf{G}(\Omega) + \mathbf{D}_{amp})_{complex} &= \begin{bmatrix} 2\Omega i \end{bmatrix} = \begin{bmatrix} 2i \end{bmatrix}, \\
\mathbf{K}_{complex} &= \begin{bmatrix} \omega_0^2 - \Omega^2 + \dot{\Omega} i \end{bmatrix} = \begin{bmatrix} 99 + 0i \end{bmatrix}, \\
\mathbf{A}_{complex} &= \begin{bmatrix} \mathbf{0}_{(1,1)} & \mathbf{I}_{(1,1)} \\ -\mathbf{M}_{complex}^{-1}\mathbf{K}_{complex} & -\mathbf{M}_{complex}^{-1}\left(\mathbf{G}(\Omega) + \mathbf{D}_{amp}\right)_{complex} \end{bmatrix}, \\
\mathbf{B}_{complex} &= \begin{bmatrix} \mathbf{0}_{(1,1)} \\ \mathbf{I}_{(1,1)} \end{bmatrix},\ \mathbf{C}_{complex} = \begin{bmatrix} \mathbf{I}_{(2,2)} \end{bmatrix},\ \mathbf{D}_{complex} = \begin{bmatrix} \mathbf{0}_{(2,1)} \end{bmatrix}.
\end{aligned} \tag{6.5}$$

Note that $(\mathbf{G}(\Omega) + \mathbf{D}_{amp})_{real}$, $\mathbf{K}_{real}$, and $\mathbf{A}_{real}$ are skew-symmetric.

For setting up the state-space model, we consider the equation (1.3). The system matrices are

$$
\begin{aligned}
\mathbf{A} &= \begin{bmatrix} \mathbf{0}_{N\times N} & \mathbf{I}_{N\times N} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}(\mathbf{D}_{amp}+\mathbf{G}(\omega)) \end{bmatrix}, \\
\mathbf{B} &= \begin{bmatrix} \mathbf{0}_{N\times N_{in}} \\ \mathbf{M}^{-1}\mathbf{E} \end{bmatrix}, \\
\mathbf{C} &= \begin{bmatrix} \mathbf{c} & \mathbf{0} \end{bmatrix}, \\
\mathbf{D} &= \mathbf{d},
\end{aligned}
$$

6.2 Applying Modern Control Theory

Because the ERA identification method uses the Markov parameter, the Hankel matrix, the controllability matrix, the observability matrix, and so on, we need to verify if we can use these properties [21] with a complex variable state-space model. In this section, most properties related to the ERA identification will be verified. Here, the bar ($^{-}$) denotes a complex number vector or matrix.

6.2.1 Response of State-Space Model - Markov parameter

Initial Condition Response

Consider the linear system

$$\dot{\bar{x}} = \bar{\mathbf{A}}\bar{x}, \tag{6.6}$$

where $(\dot{})$ denotes the derivative for time. $\bar{x} \in \mathfrak{C}_{omplex}^{N\times 1}$ and $\bar{\mathbf{A}} \in \mathfrak{C}_{omplex}^{N\times N}$ are the complex number vector and matrix, respectively. Let $\bar{x}(0)$ be an initial condition of the system.

We would like to find the solution to this problem. One approach is as follows. Let $\bar{\mathbf{\Lambda}}$ = diag$[\bar{\lambda}_1, \bar{\lambda}_2, \cdots, \bar{\lambda}_N]$ where $\bar{\lambda}_j$ are the eigenvalues of the $\bar{\mathbf{A}}$ and $\bar{X}$ be the corresponding eigenvectors then,

$$
\begin{aligned}
\bar{X}^{-1}\bar{\mathbf{A}}\bar{X} &= \bar{\mathbf{\Lambda}} \\
\bar{\mathbf{A}} &= \bar{X}\bar{\mathbf{\Lambda}}\bar{X}^{-1}.
\end{aligned} \tag{6.7}
$$

With the *similarity transformation matrix*, $\bar{x}$, a new state vector is given by $\bar{z} = \bar{X}^{-1}\bar{x}$. Multiplying (6.6) by $\bar{X}$ and using $\bar{x} = \bar{X}\bar{z}$ gives

$$\begin{aligned} \bar{X}\dot{\bar{z}} &= \bar{\mathbf{A}}\bar{X}\bar{z}, \\ \dot{\bar{z}} &= \bar{X}^{-1}\bar{\mathbf{A}}\bar{X}\bar{z}, \\ \dot{\bar{z}} &= \bar{\mathbf{\Lambda}}\bar{z}. \end{aligned} \tag{6.8}$$

Hence, assuming that all eigenvalues of the state-space matrix $\bar{\mathbf{A}}$ are distinct, we can use the diagonalized matrix $\bar{\mathbf{\Lambda}}$ with the *similarity transformation matrix* instead of the state-space matrix $\bar{\mathbf{A}}$. The simple differential equation (6.8) can be solved for $\bar{z}$ by using the exponential matrix of the matrix $\bar{\mathbf{\Lambda}}$, i.e.,

$$\begin{aligned} \bar{z}(t) = e^{\mathbf{\Lambda}t}\bar{z}(0) \leftarrow \bar{z} = X^{-1}\bar{x}, \\ \bar{x}(t) = Xe^{\mathbf{\Lambda}t}X^{-1}\bar{x}(0). \end{aligned} \tag{6.9}$$

Because the matrix $\bar{\mathbf{\Lambda}}$ is a diagonal matrix, the exponential matrix can be computed by the exponential of $\bar{\mathbf{\Lambda}}$ element by element. In addition, $e^{\bar{\mathbf{A}}t}$ can be calculated with

$$e^{\bar{\mathbf{A}}t} = \bar{X}e^{\mathbf{\Lambda}t}\bar{X}^{-1}. \tag{6.10}$$

With equations (6.9) and (6.10), we obtain the initial condition response of the system.

Example - Initial Condition Response

There are two coordinates systems to solve the initial condition response, i.e.

(a) Solve $\dot{x} = \mathbf{A}x$, $x(0) = x_0$ using real coordinates.

(b) Solve $\dot{\bar{x}} = \bar{\mathbf{A}}\bar{x}$, $\bar{x}(0) = \bar{x}_0$complex coordinates.

The solution of the initial condition response using the real coordinates is $x(t) = e^{A}tx(0)$ (see [9]). The solution using the complex coordinates is given equation (6.9). Using the MATLAB program, the initial response with real and complex coordinates are compared with same initial conditions, i.e., $x_{real}(0) = [2, 2, 1, 1]^T$ and $x_{complex}(0) = [2 + 2i, 2 + 2i]^T$. Figures 6.1 and 6.2 show that two computational results. As expected this simulation using real coordinates (6.4) and complex coordinate (6.5) give identical results.

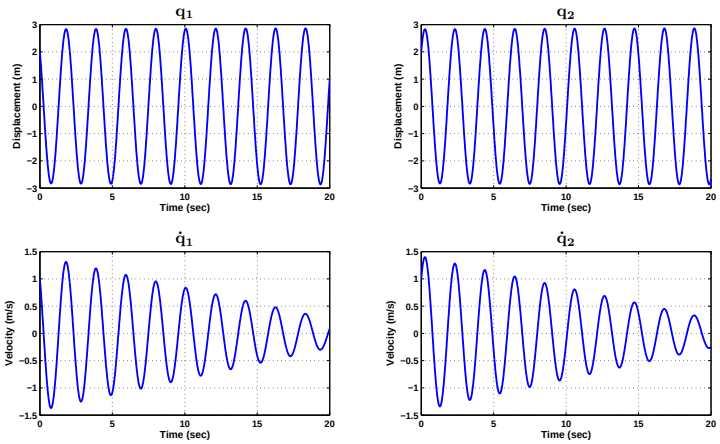

Figure 6.1: Initial responses of the gyroscopic example with real coordinates

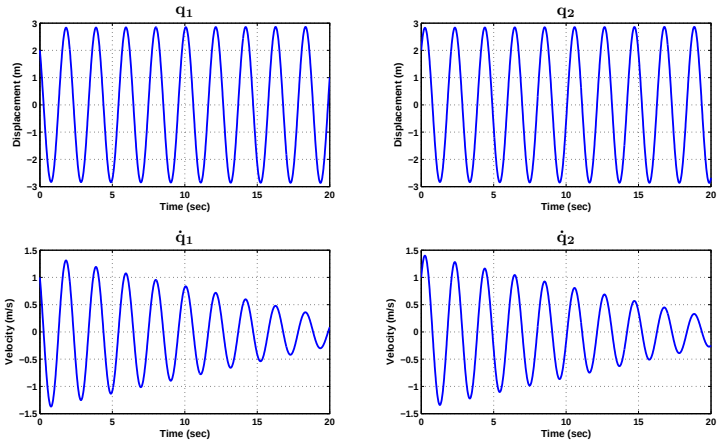

Figure 6.2: Initial response of gyroscopic example with complex coordinates

Forced Response

Consider the system.

$$\dot{\bar{x}}(t) = \bar{\mathbf{A}}\bar{x}(t) + \bar{v}(t), \tag{6.11}$$

with $\bar{x}(0) = \bar{x}_0$, where $\bar{v}(t)$ is a known integrable function of time. Regrouping the equation (6.11) and multiplying by the integrating factor $e^{-\bar{\mathbf{A}}t}$ gives

$$e^{-\bar{\mathbf{A}}t}\left[\dot{\bar{x}}(t) - \bar{\mathbf{A}}\bar{x}(t)\right] = e^{-\bar{\mathbf{A}}t}v(t). \tag{6.12}$$

Because the left hand side of the equation (6.12) is a perfect differential, the equation (6.12) can be expressed and integrated from 0 to t as

$$\begin{aligned} \int_0^t \frac{d}{dt}\left[e^{-\bar{\mathbf{A}}\tau}\bar{x}(\tau)\right]d\tau &= e^{-\bar{\mathbf{A}}t}\bar{x}(t) - e^{-\bar{\mathbf{A}}(0)}\bar{x}(0) \\ &= \int_0^t e^{-\bar{\mathbf{A}}\tau}v(\tau)d\tau. \end{aligned} \tag{6.13}$$

Finally, this can be rearranged and multiplied by $e^{\bar{\mathbf{A}}t}$ to get the solution of equation (6.11)

$$\bar{x}(t) = e^{\bar{\mathbf{A}}t}\bar{x}(0) + \int_0^t e^{\bar{\mathbf{A}}(t-\tau)}v(\tau)d\tau. \tag{6.14}$$

The equation (6.14) is the solution of the state-space model (6.11). This result (6.14) can be extended to the m-dimensional input vector $\bar{u}(t)$ and the n-dimensional output vector $\bar{y}(t)$ of the form .

$$\dot{\bar{x}}(t) = \bar{\mathbf{A}}\bar{x}(t) + \bar{v}(t), \tag{6.15}$$

$$\bar{v}(t) = \bar{\mathbf{B}}\bar{u}(t), \tag{6.16}$$

$$\bar{y}(t) = \bar{\mathbf{C}}\bar{x}(t). \tag{6.17}$$

Using the results above, it can be shown that the response of (6.15) $\sim$ (6.17) is given by

$$y(t) = \bar{\mathbf{C}}\left[e^{\bar{\mathbf{A}}t}x(0) + \int_0^t e^{\bar{\mathbf{A}}(t-\tau)}\bar{\mathbf{B}}u(\tau)d\tau\right]. \tag{6.18}$$

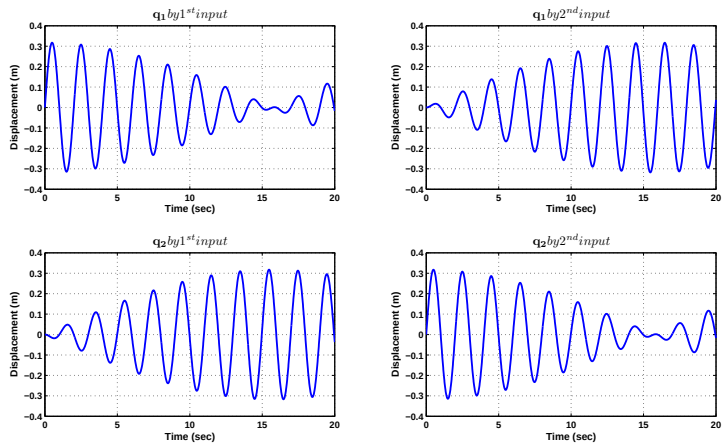

Figure 6.3: Excitation responses of the gyroscopic example with real coordinates

Example - Forced Response

Using the derived equations for the excitation response (6.18), e.g., impulse input, the responses of the gyroscopic example with the complex variable transformation are calculated as shown Figure 6.4. Using the 'impulse' command in MATLAB, the impulse responses of the gyroscopic example are illustrated on Figure 6.3. Two figures 6.3 and 6.4 show that there is no difference between two coordinates system - the real and the complex.

6.2.2 Response of Discrete State-Space Model

Chapter 2 has shown that the ERA method uses the discrete time state-space model. As a result, the responses must be obtained from the discrete time state-space model. Using equation (6.18), it can be shown that an equivalent discrete time system, is given by

$$\begin{aligned} \bar{x}(k+1) &= \bar{\mathbb{A}}\bar{x}(k) + \bar{\mathbb{B}}\bar{u}(k), \\ \bar{y}(k) &= \bar{\mathbb{C}}\bar{x}(k) + \bar{\mathbb{D}}\bar{u}(k), \end{aligned}$$

where

$$T_s \text{ is the sampling time, } k = 1,\ 2,\ \cdots,$$

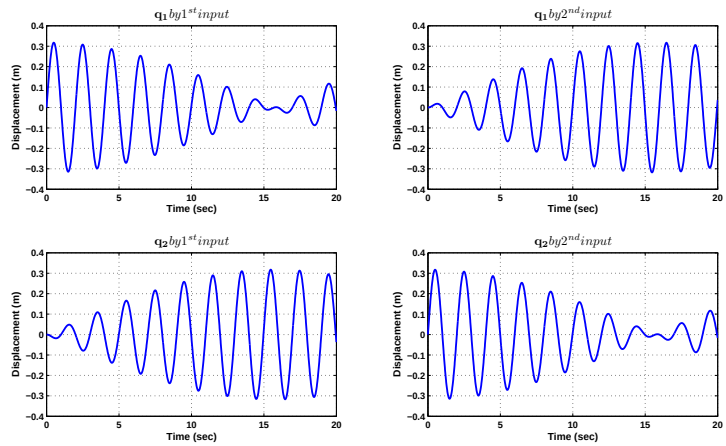

Figure 6.4: Excitation responses of the gyroscopic example with complex coordinates

$$\bar{\mathbb{A}} = e^{\bar{\mathbf{A}}T_s},\ \bar{\mathbb{B}} = \int_0^t e^{\bar{\mathbf{A}}(t-\tau')}d\tau'\bar{\mathbf{B}},$$
$$\bar{\mathbb{C}} = \bar{\mathbf{C}},\ \text{and}\ \bar{\mathbb{D}} = \bar{\mathbf{D}}.$$

The discrete time responses of the system due to an impulse input and zero initial condition are

$$\begin{aligned}\bar{y}\left((k+1)T_s\right) &= \bar{\mathbf{C}}\left[\bar{\mathbb{A}}^{(k+1)}\underbrace{\bar{x}(0)}_{=0} + \sum_{j=0}^{k}\bar{\mathbb{A}}^{(k+1-j)}\bar{\mathbb{B}}\bar{u}\left(jT_s\right)\right] \\ &= \bar{\mathbf{C}}\left[\sum_{j=0}^{k}\bar{\mathbb{A}}^{(k+1-j)}\bar{\mathbb{B}}\bar{u}\left(jT_s\right)\right]. \end{aligned} \tag{6.19}$$

The responses due to an impulse input are used for constructing the Markov parameter and the Hankel matrix. The impulse input can be generated by

$$\bar{u}\left(jT_s\right) = \begin{cases} \bar{P}\ \ for\ \ j=1 \\ 0\ \ for\ \ j \neq 1 \end{cases},$$

where $\bar{P}$ is the magnitude of an impulse input. The term in square brackets in (6.19) can be written as $\bar{\mathbb{A}}^k\bar{\mathbb{B}}\bar{P}$ where $\bar{P}$ is a magnitude of an impulse input. Therefore,

$$\bar{y}\left((k+1)T_s\right) = \bar{\mathbf{C}}\bar{\mathbb{A}}^k\bar{\mathbb{B}}\bar{P}, \tag{6.20}$$

where $k = 1, 2, \cdots$. The discrete time response (6.20) is used in the new identification technique with complex variable transformation.

6.2.3 Controllability and Observability Matrix

The controllability and observability matrices [21] are essential elements in the ERA [49]. This is because the controllability and observability matrices are factors of the Hankel matrix for real variable state-space models like equations (2.12) $\sim$ (2.17). In order to use the ERA method with the complex variable state-space models, we must show that the controllability and observability matrices in complex variable systems are factors of the complex variable Hankel matrix. Here, we develop the controllability and observability property for complex variable state-space linear systems. The analysis of this section follows an approach given in [90][1].

Observability

Definition 1 (Observable) *A particular state $\bar{x}$ of the system is said to be unobservable if for $\bar{u}(kT_s) \equiv 0$ and any time $k > 0$, the initial state $\bar{x}(0) = \bar{x}$ produces an output response $\bar{y}(kT_s) = 0$.*

A state $\bar{x}$ which makes the output $\bar{y}(kT_s)$ zero is also said to be unreconstructible. In this case, observability and reconstructibility are identical.

Theorem 1.1 (Observability Gramian) *A state $\bar{x}$ which is the complex variables vector is unobservable if and only if it belongs to the null space $\mathcal{N}_{ull}(W_{\mathcal{O}}(0, kT_s))$, where*

$$W_{\mathcal{O}}(0, kT_s) \triangleq \sum_{j=0}^{k-1} \left[\bar{\mathbb{A}}^j\right]^H \left[\bar{\mathbb{C}}\right]^H \bar{\mathbb{C}}\bar{\mathbb{A}}^j,$$

is a symmetric $N \times N$ matrix which is called the observability Gramian. Here, $(\cdot)^H$ is the Hermitian transpose matrix. Consequently, the system is completely observable if and only if $W_{\mathcal{O}}(0, kT_s)$ is nonsingular.

---

[1] Appendix A.1 provides the observability and the controllability for real state variables.

Proof - Observability Gramian

Assume that $\bar{x} \in \mathcal{N}_{ull}(W_{\mathcal{O}})$. This assumption means that

$$\begin{aligned} W_{\mathcal{O}}\bar{x} &= 0, \\ \bar{x}^H W_{\mathcal{O}}\bar{x} &= 0 = \bar{x}^H \left[\sum_{j=0}^{k-1} \left[\bar{\mathbb{A}}^j\right]^H \left[\bar{\mathbb{C}}\right]^H \bar{\mathbb{C}}\bar{\mathbb{A}}^j \right] \bar{x} \\ &= \sum_{j=0}^{k-1} \left\| \bar{\mathbb{C}}\bar{\mathbb{A}}^j \bar{x} \right\|^2 . \end{aligned} \tag{6.21}$$

Recall the response of the complex variable state-space model (6.18), when $\bar{u}(kT_s) = 0$ and $\bar{x}(0) = \bar{x}$, then

$$\begin{aligned} \bar{y}(kT_s) &= \bar{\mathbb{C}}\bar{\mathbb{A}}^k \bar{x} \qquad (k > 0) \\ &= 0. \end{aligned} \tag{6.22}$$

The equation (6.21) makes the above equation zero so that $\bar{x}$ is unobservable.

Conversely, if $\bar{x}$ is unobservable, then we can use the Definition 1 of the observability. Therefore,

$$\bar{y}(kT_s) = 0 = \bar{\mathbb{C}}\bar{\mathbb{A}}^k \bar{x} \qquad (k > 0). \tag{6.23}$$

If equation (6.23) is satisfied, then

$$W_{\mathcal{O}}\bar{x} = \sum_{j=0}^{k-1} \left[\bar{\mathbb{A}}^j\right]^H \left[\bar{\mathbb{C}}\right]^H \underbrace{\bar{\mathbb{C}}\bar{\mathbb{A}}^j \bar{x}}_{=0 \because (6.23)} = 0. \tag{6.24}$$

Therefore, the equation (6.24) shows that $\bar{x}$ is obviously the null space of the Observability Gramian - $W_{\mathcal{O}}$.

With equations (6.22) and (6.24), finally, we can conclude that the system is completely observable if and only if $\mathcal{N}_{ull}(W_{\mathcal{O}}) = 0$, i.e. $W_{\mathcal{O}}$ is nonsingular.

Theorem 1.2 (Observability Matrix) *If we define the $NN_{out} \times N$ observability matrix*

for the complex variables state-space system to be

$$\bar{\mathcal{O}} = \begin{bmatrix} \bar{\mathbb{C}} \\ \bar{\mathbb{C}}\bar{\mathbb{A}} \\ \bar{\mathbb{C}}\bar{\mathbb{A}}^2 \\ \vdots \\ \bar{\mathbb{C}}\bar{\mathbb{A}}^{N-1} \end{bmatrix},$$

then the subspace of unobservable states and the subspace of observable states are

$$\begin{aligned} \mathcal{N}_{ull}(W_{\mathcal{O}}(0, kT_s)) &= \mathcal{N}_{ull}(\bar{\mathcal{O}}), \\ R_{ange}(W_{\mathcal{O}}(0, kT_s)) &= R_{ange}(\bar{\mathcal{O}}) \\ &= R_{ange}\left(\begin{bmatrix} \bar{\mathbb{C}} \\ \bar{\mathbb{C}}\bar{\mathbb{A}} \\ \bar{\mathbb{C}}\bar{\mathbb{A}}^2 \\ \vdots \\ \bar{\mathbb{C}}\bar{\mathbb{A}}^{N-1} \end{bmatrix}\right), \end{aligned} \tag{6.25}$$

for any $k > 0$. Thus the system is completely observable if and only if the observability matrix has full rank.

Proof - Observability Matrix

Let us prove the equation (6.25) for verifying the theorem of the observability matrix. Assume $\mathcal{N}_{ull}(W_{\mathcal{O}}) \subseteq \mathcal{N}_{ull}(\bar{\mathcal{O}})$. If $\bar{z} \in \mathcal{N}_{ull}(W_{\mathcal{O}})$, then $W_{\mathcal{O}}\bar{z} = 0$ and

$$\begin{aligned} \bar{z}^H W_{\mathcal{O}} \bar{z} &= 0 \\ &= \sum_{j=0}^{k-1} \bar{z}^H \left[\bar{\mathbb{A}}^j\right]^H \bar{\mathbb{C}}^H \bar{\mathbb{C}}\bar{\mathbb{A}}^j \bar{z} = \sum_{j=0}^{k-1} \left\|\bar{\mathbb{C}}\bar{\mathbb{A}}^j \bar{z}\right\|^2. \end{aligned} \tag{6.26}$$

Since the each term of the summation equation (6.26) is independent each other, the each term must be zero, i.e.,

$$\bar{\mathbb{C}}\bar{\mathbb{A}}^j \bar{z} = 0, \tag{6.27}$$

where $j = 0, 1, \cdots, k-1$. The equation (6.27) can be put in the form

$$\begin{bmatrix} \bar{\mathbb{C}}\bar{z} \\ \bar{\mathbb{C}}\bar{\mathbb{A}}\bar{z} \\ \bar{\mathbb{C}}\bar{\mathbb{A}}^2\bar{z} \\ \vdots \\ \bar{\mathbb{C}}\bar{\mathbb{A}}^{N-1}\bar{z} \end{bmatrix} = 0 = \begin{bmatrix} \bar{\mathbb{C}} \\ \bar{\mathbb{C}}\bar{\mathbb{A}} \\ \bar{\mathbb{C}}\bar{\mathbb{A}}^2 \\ \vdots \\ \bar{\mathbb{C}}\bar{\mathbb{A}}^{N-1} \end{bmatrix} \bar{z} = \bar{\mathcal{O}}\bar{z}. \tag{6.28}$$

The fact that the multiplication $\bar{\mathcal{O}}$ and $\bar{z}$ is zero means that $\bar{z}$ must be in the null space of the observability matrix $\bar{\mathcal{O}}$.

Conversely, we shall establish that $\mathcal{N}_{ull}(W_{\mathcal{O}}) \supseteq \mathcal{N}_{ull}(\bar{\mathcal{O}})$. If $\bar{z}$ is in the null space of the observability matrix $\bar{\mathcal{O}}$, then the equations (6.27) and (6.28) must be hold.

$$\bar{z} \in \mathcal{N}_{ull}(\bar{\mathcal{O}}) \rightarrow \bar{\mathbb{C}}\bar{\mathbb{A}}^j\bar{z} = 0, \tag{6.29}$$

where $j = 0, 1, \cdots, k-1$.

Finally, let us put the equation (6.29) into the $W_{\mathcal{O}}\bar{z}$ to get,

$$W_{\mathcal{O}}\bar{z} = \sum_{j=0}^{k-1} \left[\bar{\mathbb{A}}^j\right]^H \left[\bar{\mathbb{C}}\right]^H \underbrace{\bar{\mathbb{C}}\bar{\mathbb{A}}^j\bar{z}}_{=0,\ \because (6.29)} \; d\tau = 0. \tag{6.30}$$

Therefore, the equations (6.28) and (6.30) verify $\mathcal{N}_{ull}(W_{\mathcal{O}}) \supseteq \mathcal{N}_{ull}(\bar{\mathcal{O}})$, and $\mathcal{N}_{ull}(W_{\mathcal{O}}) \subseteq \mathcal{N}_{ull}(\bar{\mathcal{O}})$. These two statements means that the null space of the observability Gramian is equal to the null space of the null space of the observability matrix.

Example

Let us check the observability with the example gyroscopic system. The observability matrix with real coordinates system is following.

$$\mathcal{O}_{real} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -9.9 & 0 & 0 & 0.2 \\ 0 & -9.9 & -0.2 & 0 \\ -9.9 & 0 & 0 & 0.2 \\ 0 & -9.9 & -0.2 & 0 \\ 0 & -1.98 & -9.94 & 0 \\ 1.98 & 0 & 0 & -9.94 \\ 0 & -1.98 & -9.94 & 0 \\ 1.98 & 0 & 0 & -9.94 \\ 98.406 & 0 & 0 & -3.968 \\ 0 & 98.406 & 3.968 & 0 \end{bmatrix} \tag{6.31}$$

The rank of $\mathcal{O}_{real}$ is 4, which means that the example gyroscopic system with real coordinates is observable. After the complex variable transformation, the observability matrix becomes

$$\mathcal{O}_{complex} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ -9.9 & 0 - 0.2i \end{bmatrix} \tag{6.32}$$

The rank of $\mathcal{O}_{complex}$ is 2, which means that the example gyroscopic system with complex variable coordinates also is observable. We can conclude that the complex variable coordinates gives us right results.

Finally, we can use the observability matrix instead of the observability Gramian. Let us move on to the definition of the controllability.

Controllability

Definition 2 (Controllable) *A particular state $\bar{x}$ of the system is said to be controllable if for every time $k > 0$, there exists a control function $\bar{u}(kT_s)$ which drives the system from the initial state $\bar{x}(0) = \bar{x}$ to $\bar{x}(kT_s) = 0$.*

The input $\bar{u}(kT_s)$ which satisfied with the definition 2 is not unique, i.e., other controls are able to drive the state to zero via different trajectories. A state $\bar{x}$ is also said to be '*reachable*' if a control exists which drives the system from $\bar{x}(0) = 0$ to $\bar{x}(kT_s) = \bar{x}$.

In this case, the controllability and reachability are identical.

Theorem 2.1 (Controllability Gramian) *A state $\bar{x}$ which is the complex number vector is controllable if and only if it belongs to the range space $\mathcal{R}_{ange}(W_{\mathcal{C}}(0, kT_s))$ where*

$$W_{\mathcal{C}}(0, kT_s) \triangleq \sum_{j=0}^{k-1} \bar{\mathbb{A}}^{-j}\bar{\mathbb{B}} \left[\bar{\mathbb{B}}\right]^H \left[\bar{\mathbb{A}}^{-j}\right]^H,$$

is a symmetric $N \times N$ matrix known as the controllability gramian. Consequently, the system is completely controllable if and only if $W_{\mathcal{C}}(0, kT_s)$ is nonsingular.

Proof - Controllability Gramian

Assume $\bar{x} \in \mathcal{R}_{ange}\{W_{\mathcal{C}}\}$. Then the equation

$$W_{\mathcal{C}}\bar{z} = \bar{x}, \tag{6.33}$$

has a solution $\bar{z}$ which is an n-dimensional vector of complex variables. According to the Theorem 2.1, if we choose the control input to be

$$\bar{u}(kT_s) = -\left[\bar{\mathbb{B}}\right]^H \left[\bar{\mathbb{A}}^{-k}\right]^H \bar{z}, \tag{6.34}$$

then, we find that

$$\begin{aligned}
\bar{x}(kT_s) &= \left[\bar{\mathbb{A}}\right]^k \bar{x} + \sum_{j=0}^{k-1} \left[\bar{\mathbb{A}}\right]^{(k-j)} \bar{\mathbb{B}}\bar{u}(jT_s) \\
&= \left[\bar{\mathbb{A}}\right]^k \bar{x} - \sum_{j=0}^{k-1} \left[\bar{\mathbb{A}}\right]^{k-j} \bar{\mathbb{B}} \left[\bar{\mathbb{B}}\right]^H \left[\bar{\mathbb{A}}^{-j}\right]^H \bar{z} \\
&= \left[\bar{\mathbb{A}}\right]^k \bar{x} - \left[\bar{\mathbb{A}}\right]^k \underbrace{\sum_{j=0}^{k-1} \left[\bar{\mathbb{A}}\right]^{-j} \bar{\mathbb{B}} \left[\bar{\mathbb{B}}\right]^H \left[\bar{\mathbb{A}}^{-j}\right]^H}_{=W_{\mathcal{C}}} \bar{z} \\
&= \left[\bar{\mathbb{A}}\right]^k \bar{x} - \left[\bar{\mathbb{A}}\right]^k W_{\mathcal{C}}\bar{z} = \left[\bar{\mathbb{A}}\right]^k \underbrace{\left[\bar{x} - W_{\mathcal{C}}\bar{z}\right]}_{=0 \because (6.33)} \\
&= 0. \qquad (6.35)
\end{aligned}$$

Therefore, a control input $\bar{u}(kT_s)$ exists which can drives the system from $\bar{x}(0) = 0$ to $\bar{x}(T) = \bar{x}$.

Theorem 2.2 (Controllability Matrix) *If we define the $N \times NN_{in}$ controllability matrix for the system to be*

$$\bar{\mathcal{C}} \triangleq \left[\bar{\mathbb{B}}, \bar{\mathbb{A}}\bar{\mathbb{B}}, \bar{\mathbb{A}}^2\bar{\mathbb{B}}, \cdots, \bar{\mathbb{A}}^{N-1}\bar{\mathbb{B}}\right], \qquad (6.36)$$

then the subspace of controllable states is

$$\mathcal{R}_{ange}\left(W_{\mathcal{C}}\left(0, kT_s\right)\right) = \mathcal{R}_{ange}\left(\bar{\mathcal{C}}\right),$$

and the subspace of uncontrollable state is

$$\mathcal{N}_{ull}(W_{\mathcal{C}}(0, kT_s)) = \mathcal{N}_{ull}(\bar{\mathcal{C}}^H) = \mathcal{N}_{ull} \begin{bmatrix} \bar{\mathbb{B}}^H \\ \bar{\mathbb{B}}^H \ \ \bar{\mathbb{A}}^H \\ \vdots \\ \bar{\mathbb{B}}^H \ \ \left[\bar{\mathbb{A}}^H\right]^{(N-1)} \end{bmatrix},$$

for any $k > 0$. Thus, the system is completely controllable if and only if the controllability matrix has full rank, that is,

$$\text{rank}(\bar{\mathcal{C}}) = N.$$

Proof - Controllability Matrix

First, let us assume that $\mathcal{N}_{ull}(W_{\mathcal{C}}) \subseteq \mathcal{N}_{ull}(\bar{\mathcal{C}}^H)$. If $\bar{z} \in \mathcal{N}_{ull}(W_{\mathcal{C}})$, then $W_{\mathcal{C}}\bar{z} = 0$, and

$$\begin{aligned} \bar{z}^H W_{\mathcal{C}} \bar{z} &= \sum_{j=0}^{k-1} \bar{z}^H \bar{\mathbb{A}}^{-j} \bar{\mathbb{B}} \left[\bar{\mathbb{B}}\right]^H \left[\bar{\mathbb{A}}^{-j}\right]^H \bar{z} \\ &= \sum_{j=0}^{k-1} \left\| \left[\bar{\mathbb{B}}\right]^H \left[\bar{\mathbb{A}}^{-j}\right]^H \bar{z} \right\|^2 = 0. \end{aligned} \tag{6.37}$$

In the equation (6.37), for this to hold for all $\bar{z}$ each term in the series must be zero. That is,

$$\bar{\mathbb{B}}^H = \bar{\mathbb{B}}^H \bar{\mathbb{A}}^H = \bar{\mathbb{B}}^H (\bar{\mathbb{A}}^H)^2 = \bar{\mathbb{B}}^H (\bar{\mathbb{A}}^H)^3 = \cdots = 0. \tag{6.38}$$

The equation (6.38) can be written as

$$\bar{\mathcal{C}}^H \bar{z} = \begin{bmatrix} \bar{\mathbb{B}}^H \bar{z} \\ \bar{\mathbb{B}}^H \ \bar{\mathbb{A}}^H \bar{z} \\ \vdots \\ \bar{\mathbb{B}}^H \ \left[\bar{\mathbb{A}}^H\right]^{(N-1)} \bar{z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Therefore, $\bar{z} \in \mathcal{N}_{ull}(\bar{\mathcal{C}}^H)$.

Conversely, let us assume that $\mathcal{N}_{ull}(W_{\mathcal{C}}) \supseteq \mathcal{N}_{ull}(\bar{\mathcal{C}}^H)$. If $\bar{z} \in \mathcal{N}_{ull}(\mathcal{C}^H)$, then the equation (6.38) is valid. If we choose the control input $\bar{u}(kT_s)$ as

$$\bar{u}(kT_s) = \sum_{j=0}^{k-1} \underbrace{\left[\bar{\mathbb{B}}^H \left[\bar{\mathbb{A}}^{-k}\right]^H \bar{z}\right]}_{=0 \because (6.38)} = 0,$$

then it follows that

$$W_{\mathcal{C}} \bar{z} = \sum_{j=0}^{k-1} \bar{\mathbb{A}}^{-j} \bar{\mathbb{B}} \left(\left[\bar{\mathbb{B}}\right]^H \left[\bar{\mathbb{A}}^{-j}\right]^H \bar{z} \right) = 0,$$

and we can conclude that $\bar{z} \in \mathcal{N}_{ull}(W_{\mathcal{C}})$.

Finally, having shown that each of spaces $\mathcal{N}_{ull}(W_{\mathcal{C}})$ and $\mathcal{N}_{ull}(\bar{\mathcal{C}}^H)$ is contained in the other, we can conclude that they must be identical.

Example

Now, let us apply the controllability matrix to the example of gyroscopic system. Then, the controllability matrix of the gyroscopic example is expressed numerically as

$$\mathcal{C}_{real} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0.2 & -9.94 & 0 \\ 0 & 0 & 0 & 1 & -0.2 & 0 & 0 & -9.94 \\ 1 & 0 & 0 & 0.2 & -9.94 & 0 & 0 & -3.968 \\ 0 & 1 & -0.2 & 0 & 0 & -9.94 & 3.968 & 0 \end{bmatrix}. \tag{6.39}$$

If we consider the complex variable transformation with the same example, the new controllability matrix which includes the complex numbers is followed

$$\mathcal{C}_{complex} = \begin{bmatrix} 0 & 0 & 1 & 0+1i \\ 1 & 0+1i & 0-0.2i & 0.2 \end{bmatrix}. \tag{6.40}$$

Both matrices (6.39) and (6.40) have full rank, i.e., the rank of the matrix (6.39) and the matrix (6.40) are four and two respectively. Therefore, the gyroscopic system is controllable regardless of the coordinate system, i.e., the real coordinate system and the complex coordinate system.

6.3 Applying the ERA Identification Method

Derivation of ERA identification method includes the Markov parameter, the Hankel matrix, the response of impulse input, the controllability and the observability matrices and a decomposition method of matrix. As a result, applying the ERA identification method to complex variables coordinates requires to verify those concepts for complex variable state-space model. Let us redefine and derive them.

6.3.1 Markov parameter and Hankel matrix

For the complex variable representation the Markov parameters are defined as

$$\bar{\mathbf{M}}_{arkov}(k) = [\bar{y}_1(kT_s) \quad \bar{y}_2(kT_s) \quad \cdots \quad \bar{y}_{N_{in}}(kT_s)], \tag{6.41}$$

where

$$k = 0,\ 1,\ 2,\ \cdots,\ \text{and } T_s \text{ is a sampling time.}$$

The Hankel matrix which is based upon the complex variable coordinates is written by

$$\bar{\mathbf{H}}_{qd}(k) = \begin{bmatrix} \bar{\mathbf{M}}_{arkov}(k+1) & \bar{\mathbf{M}}_{arkov}(k+2) & \cdots & \bar{\mathbf{M}}_{arkov}(k+d) \\ \bar{\mathbf{M}}_{arkov}(k+2) & \bar{\mathbf{M}}_{arkov}(k+3) & \cdots & \bar{\mathbf{M}}_{arkov}(k+d+1) \\ \vdots & \vdots & \ddots & \vdots \\ \bar{\mathbf{M}}_{arkov}(k+q) & \bar{\mathbf{M}}_{arkov}(k+q+1) & \cdots & \bar{\mathbf{M}}_{arkov}(k+q+d-1) \end{bmatrix} \tag{6.42}$$

where N_{out} is the number of outputs, q and d are chosen numbers like $N_{out} \times q = d \times$ number of inputs$(= N_{in})$.

6.3.2 Singular Value Decomposition

The singular value decomposition of a complex variable matrix $\bar{H}$ is

$$\bar{\mathbf{H}} = \bar{\mathbf{U}}\mathbf{\Sigma}\bar{\mathbf{V}}^H \tag{6.43}$$

where $\bar{\mathbf{U}}$ and $\bar{\mathbf{V}}$ are unitary matrices, the superscript of $(\cdot)^H$ is the complex conjugate transpose of $(\cdot)$, and $\mathbf{\Sigma}$ is a diagonal matrix whose elements are the singular values of the original matrix $\bar{\mathbf{H}}$.

In the book [28], G. H. Golub and C. F. Van Loan verified that there always exists such a decomposition with positive singular values, even when $\bar{\mathbf{H}}$ is a matrix which includes complex variables.

6.3.3 Controllability and Observability Matrix

The controllability and observability matrices which consist of complex variables have been derived from the controllability and observability gramians in the Section 6.2. Recall that the controllability matrix $\bar{\mathcal{C}}$ and the observability matrix $\bar{\mathcal{O}}$ are

$$\bar{\mathcal{C}} = \left[\bar{\mathbb{B}}, \bar{\mathbb{A}}\bar{\mathbb{B}}, \cdots, \bar{\mathbb{A}}^{(N-1)}\bar{\mathbb{B}}\right], \tag{6.44}$$

$$\mathcal{O} = \begin{bmatrix} \mathbb{C} \\ \mathbb{C}\mathbb{A} \\ \vdots \\ \mathbb{C}\mathbb{A}^{(N-1)} \end{bmatrix}, \tag{6.45}$$

where $\bar{\mathbb{A}} \in \mathfrak{C}_{omplex}^{N\times N}$, $\bar{\mathbb{B}} \in \mathfrak{C}_{omplex}^{N\times N_{in}}$, $\bar{\mathbb{C}} \in \mathfrak{C}_{omplex}^{N_{out}\times N}$, N is a number of the states, N_{in} is a number of inputs, and N_{out} is a number of outputs.

6.3.4 Identification

The Hankel Matrix at k=0 and $k = 1$ is

$$\bar{\mathbf{H}}_{qd}(0) = \begin{bmatrix} \bar{\mathbf{M}}_{arkov}(1) & \bar{\mathbf{M}}_{arkov}(2) & \cdots & \bar{\mathbf{M}}_{arkov}(d) \\ \bar{\mathbf{M}}_{arkov}(2) & \bar{\mathbf{M}}_{arkov}(3) & \cdots & \bar{\mathbf{M}}_{arkov}(d+1) \\ \vdots & \vdots & \ddots & \vdots \\ \bar{\mathbf{M}}_{arkov}(q) & \bar{\mathbf{M}}_{arkov}(q+1) & \cdots & \bar{\mathbf{M}}_{arkov}(q+d-1) \end{bmatrix}, \tag{6.46}$$

$$\bar{\mathbf{H}}_{qd}(1) = \begin{bmatrix} \bar{\mathbf{M}}_{arkov}(2) & \bar{\mathbf{M}}_{arkov}(3) & \cdots & \bar{\mathbf{M}}_{arkov}(d+1) \\ \bar{\mathbf{M}}_{arkov}(3) & \bar{\mathbf{M}}_{arkov}(4) & \cdots & \bar{\mathbf{M}}_{arkov}(d+2) \\ \vdots & \vdots & \ddots & \vdots \\ \bar{\mathbf{M}}_{arkov}(q+1) & \bar{\mathbf{M}}_{arkov}(q+2) & \cdots & \bar{\mathbf{M}}_{arkov}(q+d+1) \end{bmatrix}. \tag{6.47}$$

In equation (6.46), the Markov parameters can be substituted by equation (6.20) to give

$$\begin{aligned} \bar{\mathbf{H}}_{qd}(0) &= \begin{bmatrix} \bar{\mathbb{C}}\bar{\mathbb{B}} & \bar{\mathbb{C}}\bar{\mathbb{A}}\bar{\mathbb{B}} & \cdots & \bar{\mathbb{C}}\bar{\mathbb{A}}^{(d-1)}\bar{\mathbb{B}} \\ \bar{\mathbb{C}}\bar{\mathbb{A}}\bar{\mathbb{B}} & \bar{\mathbb{C}}\bar{\mathbb{A}}^{2}\bar{\mathbb{B}} & \cdots & \bar{\mathbb{C}}\bar{\mathbb{A}}^{d}\bar{\mathbb{B}} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{\mathbb{C}}\bar{\mathbb{A}}^{(q-1)}\bar{\mathbb{B}} & \bar{\mathbb{C}}\bar{\mathbb{A}}^{q}\bar{\mathbb{B}} & \cdots & \bar{\mathbb{C}}\bar{\mathbb{A}}^{(q+d-2)}\bar{\mathbb{B}} \end{bmatrix} \\ &= \begin{bmatrix} \mathbb{C} \\ \mathbb{C}\bar{\mathbb{A}} \\ \vdots \\ \mathbb{C}\bar{\mathbb{A}}^{(q-1)} \end{bmatrix} \left[\bar{\mathbb{B}}, \bar{\mathbb{A}}\bar{\mathbb{B}}, \cdots, \bar{\mathbb{A}}^{(d-1)}\bar{\mathbb{B}}\right] \\ &= \bar{\mathcal{O}}_q \bar{\mathcal{C}}_d. \end{aligned}$$

For getting some relationships from equation (2.14), apply the singular value decomposition to the Hankel matrix - $\bar{\mathbf{H}}_{qd}(0)$, i.e.,

$$\bar{\mathbf{H}}_{qd}(0) = \bar{\mathbf{U}}_q \mathbf{\Sigma}_{qd} \bar{\mathbf{V}}_d^H. \tag{6.48}$$

With the pseudo-inverse[2] of controllability and observability matrices and the singular value decomposition matrices, the equation (6.48) can be rewritten as

$$\bar{\mathcal{O}}_q = \bar{\mathbf{U}}_q \, \mathbf{\Sigma}_{qd}^{(1/2)} \rightarrow \bar{\mathcal{O}}_q^{\dagger} = \mathbf{\Sigma}_{qd}^{(-1/2)} \, \bar{\mathbf{U}}_q^H, \tag{6.49}$$

$$\bar{\mathcal{C}}_d = \mathbf{\Sigma}_{qd}^{(1/2)} \, \bar{\mathbf{V}}_d^H \rightarrow \bar{\mathcal{C}}_d^{\dagger} = \bar{\mathbf{V}}_d \, \mathbf{\Sigma}_{qd}^{(-1/2)}. \tag{6.50}$$

Multiplication of the observability, the controllability, and the complex variable state-space $\bar{\mathbb{A}}$ matrices is able to express the Hankel matrix. The equation makes it possible to solve a least square problem for $\bar{\mathbb{A}}$, i.e.,

$$\bar{\mathbb{A}} = \bar{\mathcal{O}}_q^{\dagger} \bar{\mathbf{H}}_{qd}(1) \bar{\mathcal{C}}_d^{\dagger}. \tag{6.51}$$

The pseudo-inverse of controllability and observability matrices have already been obtained. Put equations (6.50) and (6.49) into the equation (6.51), then the identified discrete state-space $\bar{\mathbb{A}}$ can be written as

$$\bar{\mathbb{A}}_{id} = \bar{\mathcal{O}}_q^{\dagger} \bar{\mathbf{H}}_{qd}(1) \bar{\mathcal{C}}_d^{\dagger} = \mathbf{\Sigma}_{qd}^{(-1/2)} \, \bar{\mathbf{U}}_q^H \bar{\mathbf{H}}_{qd}(1) \bar{\mathbf{V}}_d \, \mathbf{\Sigma}_{qd}^{(-1/2)}. \tag{6.52}$$

For identifying the $\bar{\mathbb{B}}$ matrix of the complex variable state-space model, use the controllability matrix $\bar{\mathcal{C}}$ (6.44). The $\hat{\bar{\mathbb{B}}}$ matrix must equal the first N_{in} columns of the controllability matrix because of the definition of the controllability matrix. Therefore, the $\hat{\bar{\mathbb{B}}}$ matrix can be obtained by multiplying calculated controllability matrix (6.50) to a matrix which eliminates the rest of the columns. Obviously, the number, N_{order}, must be chosen by a proper algorithm[3]. Hence we get that the identified $\bar{\mathbb{B}}$ is

$$\hat{\bar{\mathbb{B}}}_{id} = \bar{\mathcal{C}} \begin{bmatrix} \mathbf{I}_{N_{order} \times N_{in}} \\ \mathbf{0}_{(d-N_{order}) \times N_{in}} \end{bmatrix} = \mathbf{\Sigma}_{qd}^{(1/2)} \, \bar{\mathbf{V}}_d^H \begin{bmatrix} \mathbf{I}_{N_{order} \times N_{in}} \\ \mathbf{0}_{(d-N_{order}) \times N_{in}} \end{bmatrix}. \tag{6.53}$$

Using the same logic for identifying $\hat{\bar{\mathbb{B}}}$, the state-space identified $\bar{\mathbf{C}}$ matrix can be obtained, i.e., in the observability matrix $\bar{\mathcal{O}}$ (6.45), the state-space $\bar{\mathbf{C}}$ matrix have to equal the first N_{out} rows of the observability matrix. Multiplying the observability matrix (6.49) to a

[2]The pseudo-inverse is denoted by a superscript of †.

[3]Model reduction algorithm will be introduced in the Chapter 7.

matrix which will get rid of rest rows makes the state-space $\bar{\mathbf{C}}$ matrix,

$$\bar{\mathbf{C}}_{id} = \begin{bmatrix}\mathbf{I}_{N_{out}\times N_{order}} & \mathbf{0}_{N_{out}\times(q-N_{order})}\end{bmatrix} \bar{\mathcal{O}} = \begin{bmatrix}\mathbf{I}_{N_{out}\times N_{order}} & \mathbf{0}_{N_{out}\times(q-N_{order})}\end{bmatrix} \bar{\mathbf{U}}_q \, \mathbf{\Sigma}_{qd}^{(1/2)}. \tag{6.54}$$

The last step for the ERA is getting the state-space $\bar{\mathbf{D}}$ matrix. The state-space $\bar{\mathbf{D}}$ matrix must equal $\bar{\mathbf{M}}_{arkov}(0)$ because the $\bar{\mathbf{D}}$ matrix gives us information about the direct relationship between inputs and outputs.

For generalization of derivation, the complex variable Hankel matrix is $q \times d$ matrix. Usually the Hankel Matrix can be made by square matrix for getting a state-space matrix easily. Then, the complex variable state-space matrix $\hat{\bar{\mathbb{A}}}_{id}$, $\hat{\bar{\mathbb{B}}}_{id}$, $\bar{\mathbf{C}}_{id}$ and $\bar{\mathbf{D}}_{id}$ are obtained as

$$\begin{aligned}
\hat{\bar{\mathbb{A}}}_{id} &= \mathbf{\Sigma}^{(-1/2)}\bar{\mathbf{U}}^H\bar{\mathbf{H}}(1)\bar{\mathbf{V}}\mathbf{\Sigma}^{(-1/2)}, \\
\hat{\bar{\mathbb{B}}}_{id} &= \mathbf{\Sigma}^{(1/2)}\bar{\mathbf{V}}^H \begin{bmatrix} \mathbf{I}_{N_{order}\times N_{in}} \\ \mathbf{0}_{(d-N_{order})\times N_{in}} \end{bmatrix}, \\
\bar{\mathbf{C}}_{id} &= \begin{bmatrix}\mathbf{I}_{N_{out}\times N_{order}} & \mathbf{0}_{N_{out}\times(q-N_{order})}\end{bmatrix} \bar{\mathbf{U}}\mathbf{\Sigma}^{(1/2)}, \\
\bar{\mathbf{D}}_{id} &= \bar{\mathbf{M}}_{arkov}(0).
\end{aligned} \tag{6.55}$$

The equations (6.55) show the result of the complex variable ERA method. In addition, because the equations (6.55) include the imaginary numbers, the impulse responses of the identified system must be considered case by case.

6.3.5 Example

In the previous sections, the complex variable identification method are derived theoretically, i.e, responses of external excitation, observability matrix, controllability matrix, ERA method for complex variable coordinated system are verified in detail. Here, let us identify the gyroscopic system with the new identification technique (6.55). [4] Note that this example does not include a white noise.

After getting the responses due to the impulse input, we can construct the Hankel matrix which contains the complex number, i. New subroutine function '*response_hankel*' returns Hankel matrices, $\bar{\mathbf{H}}(0)$ and $\bar{\mathbf{H}}(1)$. With Hankel matrices, $\bar{\mathbf{H}}(0)$ and $\bar{\mathbf{H}}(1)$, the

[4] All subroutine functions for the new identification technique are in Appendix B.

number of inputs, N_{in}, the number of outputs, N_{out}, and the sampling time T_s, other function '*era_normal*' gives us the identified state-space model which is based upon the new identification method. Finally, the function '*complex_impulse*' can calculate the responses of impulse input.

Figure 6.5 shows that the new identification method gives us very good results.

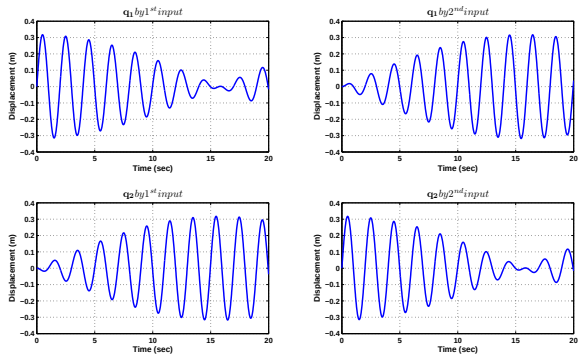

Figure 6.5: The time responses of the identified system.

We will apply this new identification technique to an actual FESS device in Chapter 8.

Chapter 7

MODEL ORDER DETERMINATION

7.1 Introduction

In many mechanical systems, the equations of motion are so complicated that a large amount data is required to identify the system. This is the case for the ERA, Fast ERA, and the new method proposed in this dissertation. An important consideration in the application of ERA type methods is the selection of the assumed model order. The assumed mode order determines the size of the Hankel matrix used in the analysis.

It has been found that selecting a model order that is too large leads to a very big Hankel matrix, which requires large computation times to determine this system order. Also, the resulting identified model can be unstable. This is shown in the Chapter 5. Thus, it is important that we develop a technique for finding the correct order of the system.

This chapter is organized as follows, in Section 7.2, we present some model order determination techniques that have been used by other researcher. In Section 7.3, we describe a new model order determination method that is applied to the complex variable ERA. In Section 7.4, we illustrate the performance of the method when applied to the FESS. In addition, we compare this new model order determination technique to other methods.

7.2 Model Order Selection Techniques

Before a new practical model order determination method is introduced, let us review several methods that have been developed by other researchers.

For ERA methods

Using a large amount of data can make the identified system unstable [79] and [81] for the ERA types identification methods. Model order selection techniques for the ERA methods are based on the relative size of the singular values of the Hankel matrix. The ratio between

the nth singular value and the first singular value determines the system order in [49] and [50]. However, the calculation of ratios can not exceed the precision limit of the computer tolerance.

For AR models

There are several model order selection methods for auto-regressive models [66] as follows.

- AIC technique
- MDL technique
- Combination of AIC and MDL
- SNR

In his paper [2], Akaike presented Akaike Information Criterion (AIC) that used an analysis of the Maximum Likelihood Estimate (MLE) [75]. AIC picks the model order that minimizes the log likelihood function, i.e., the logarithm of the joint conditional density of given data.

Rissanen's Minimum Descriptive Length (MDL) was developed in [85] and [86]. MDL method is that any regularity in data can be used to compress the data, i.e. to describe it using fewer symbols than needed to describe the data. As a result, he picked the model order by the minimum description length, i.e., calculating how many symbols are required to describe a parametric model.

For the balance between a better identified response and a computational requirement, Hannan and Quinn in [35] combined the AIC and the MDL criteria and found an optimum value. They penalized the model order selected as the model grows too large. All criteria for AR models as explained above, are based on the maximum likelihood principle.

Another practical criterion of model order selection for AR models used data sampling lengths and Signal-to-Noise Ratios (SNR) by [13]. Chen and Mechefske used the number of sample data points and the SNR.

Comparison Study

In [31], S. Gugercin and A.C. Antoulas have compared seven different model order reduction algorithms - three big catalogues ; balanced model reduction, Hankel norm approximation, and singular perturbation. None of previous model order reduction methods can select a proper order of an identified system using the complex variable transformation.

New Method

Let us briefly introduce the new practical model order reduction method. A key point of the proposed model reduction method is stabilizing an identified model as well as handling a complex variable state-space model. The stabilization problem can be solved by estimating locations of eigenvalues of an identified system. To do so, the *Gerschgorin's disk theorem* will be applied to the identified model which contains complex variables. H.T. Wu *et al.* in [100] used the Gerschgorin's disk theorem for high-resolution estimators and focused on the eignenvalue-based source number estimators not on the model reduction.

7.3 Model Reduction

The proposed model reduction method is developed by a very practical approach. Two necessary conditions for reducing system order of the identified system, i.e., stabilization and order selection of the identified system, must be considered at same time. In addition, the identified system which includes the complex variables can be applied to the proposed model reduction. For satisfying these objectives, the proposed model reduction method is starting from the identified state-space $\bar{\mathbb{A}}_{id}$ which contains the complex variables. For an example, let us use a theoretical modeling of the Flywheel Energy Storage System (FESS). The FESS is the representative of a gyroscopic system. After using simulated data for development of model order reduction method, we apply the proposed method to experimental data of FESS.

7.3.1 Simulation of FESS

The simulated data, based upon the equations of motion in [64], are created with spinning speed, $\omega = 1500$ rpm, sampling time, $T_s = 0.002$ sec, states vector, $q = [x, y, \theta_x, \theta_y]$ and impulse input by MATLAB. For finding the characteristics of the identified state-space $\bar{\mathbb{A}}_{id}$, calculate the magnitude of the each element as illustrated in Figure 7.1 using a Hankel matrix this is 500 by 500.

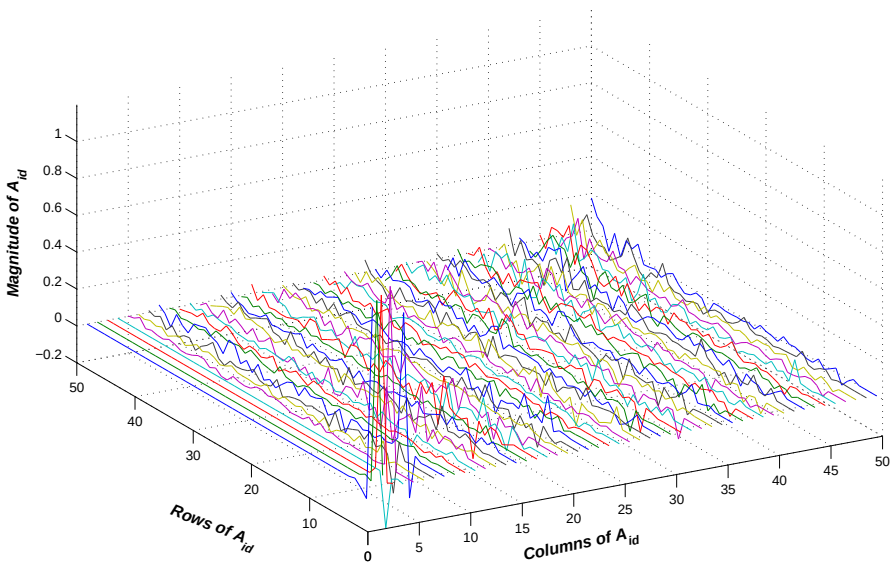

Figure 7.1: The Magintude of the identified state-space $\bar{\mathbb{A}}_{id}$.

As can be seen Figure 7.1 shows us two unique characteristics, i.e., (i) the dominant diagonal elements of the identified state-space $\bar{\mathbb{A}}_{id}$, and (ii) the trivial first several columns and rows of the identified state-space $\bar{\mathbb{A}}_{id}$. The mathematical expression of two characteristics is the Gerschgorin's disk theorem which uses the diagonal element and non-diagonal elements to estimate locations of eigenvalues.

Let us apply the Gerschgorin disk theorem to an arbitrary matrix before we use the Gerschgorin's disk theorem to select proper model order of an identified state-space model.

7.3.2 Gerschgorin's Disk Theorem

Consider a complex variable matrix $\bar{A}_{N\times N}$, $\bar{a}_{jk} \in \mathfrak{C}_{omplex}$, j and $k = 1, 2, \ldots, N$. Gerschgorin disks of the jth row of $\bar{A}$ consist of the center (cen_j) and the radius ($radii_j$), i.e.,

$$cen_j = \bar{a}_{jj}, \tag{7.1}$$

$$radii_j = \sum_{j=1,k\neq j}^{N} \mid \bar{a}_{jk} \mid . \tag{7.2}$$

Then the jth Gerschgorin's disk, $\mathfrak{D}_j$, is defined as the collection of points in the complex plane whose distance to cen_j is at most $radii_j$.

$$radii_j \geq \mid \bar{z} - cen_j \mid \tag{7.3}$$

where $\bar{z}$ is a complex variable. The Gerschgorin disk theorem proved that the eigenvalues of $\bar{A}$ are contained in the union of the disks (see section A.2).

Example

For a numerical example of Gerschgorin's disk theorem, a matrix, $\bar{A}_{3\times 3}$ is given by

$$\bar{A} = \begin{bmatrix} 10-i & 1+i & -i \\ -i & 4-i & 0.5i \\ i & -0.5i & 4+i \end{bmatrix}.$$

Gerschgorin disks can be found from (7.3), e.g., for the first Gerschgorin disk $\mathfrak{D}_1$,

$$\begin{aligned} cen_1 &= \bar{a}_{11} = 10 - i, \\ radii_1 &= \sum_{j=1,k\neq j}^{N} \mid \bar{a}_{jk} \mid \\ &= \mid \bar{a}_{12} \mid + \mid \bar{a}_{13} \mid \\ &= \sqrt{1+1} + \sqrt{1} = 1 + \sqrt{2}. \end{aligned}$$

The first Gerschogorin disk is $\mathfrak{D}_1 = \{\bar{z} : 1+\sqrt{2} \geq \mid \bar{z}-(10-i) \mid\}$. With same calculations, the second and the third Gerschogorin disks can be obtained as $\mathfrak{D}_2 = \{\bar{z} : 1.5 \geq \mid \bar{z} - (4-i) \mid\}$, and $\mathfrak{D}_3 = \{\bar{z} : 1.5 \geq \mid \bar{z} - (4+i) \mid\}$ respectively.

Because union of the second disk and the third disk does not intersect the first disk, there are two eigenvalues of $\bar{A}$ in $\mathfrak{D}_2 \cup \mathfrak{D}_3$ and another in $\mathfrak{D}_1$.

The numerical three eigenvalues of $\bar{A}$ are determined as $10.295 - 1.1121i$, $3.8397 - 0.62519i$, and $3.8652 + 0.73731i$. All Gerschgorin disks and eigenvalues of $\bar{A}$ as illustrated in Figure 7.2, verify the Gerschgorin disk theorem.

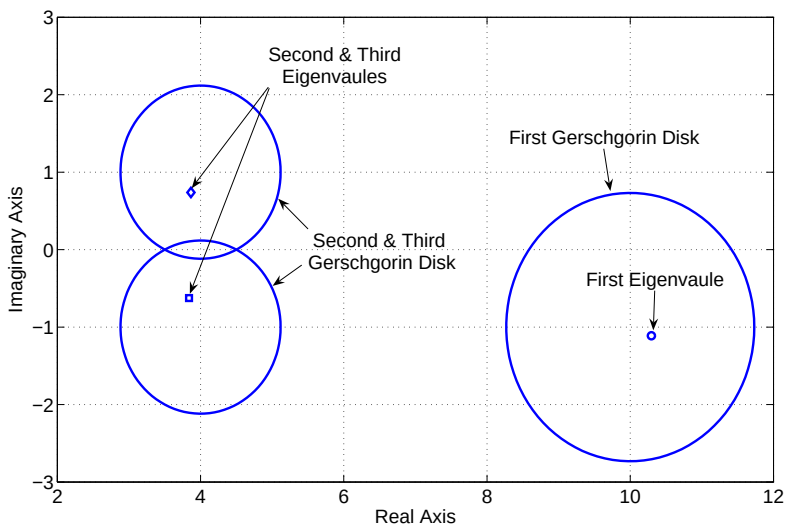

Figure 7.2: Gerschgorin's disks and eigenvalues of $\bar{A}$.

7.3.3 Proposed model reduction method

The Gerschgorin's disk theorem will be used for a model reduction method. Here, mathematical and systematical expression of reducing system order is required. To do so, review the singular value decomposition which the ERA identification method uses.

The singular value decomposition has several important properties in engineering fields. In the image processing, the singular value decomposition method is used for reducing the size of image matrix. As a matter fact, some singular values are significant and others are not. Using the fact, images matrix ($A \in \mathfrak{R}^{N \times N}$) can be simplified and manipulated by

following equations.

$$\begin{aligned} A = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T &= u_1\sigma_1 v_1^T + u_2\sigma_2 v_2^T + \cdots + u_r\sigma_r v_r^T + \cdots \\ &\quad \cdots + u_N\sigma_N v_N^T, \end{aligned} \tag{7.4}$$

where $\mathbf{\Sigma}$ is a singular value matrix, σ_m are singular values of a matrix, $\mathbf{U}$ and $\mathbf{V}$ are unitary matrix, u_m are columns of U, v_m^T are rows of V^T, $m = 1, 2, \cdots, r, \cdots, N$. N is the size of A matrix, r is an arbitrary number for reduced size, superscript of ^{T} denotes the transpose matrix.

The equation (7.4) denotes that any matrix can be expressed by adding r matrices of rank one. For an example, 1000 by 1000 matrix can be replaced by r by r matrix with (7.4), that will be helpful to manipulate a matrix.

With this property of the singular value decomposition, the identified state-space $\bar{\mathbb{A}}_{id}$ matrix can be expressed by just significant singular values and matrices as follows

$$\bar{\mathbb{A}}_r = \mathbf{\Sigma}_{sd}(1:r, 1:r)^{(-0.5)}\bar{\mathbf{U}}_s(:, 1:r)^H \cdot \bar{\mathbf{H}}_{sd}(1)\bar{\mathbf{V}}_d(:, 1:r)\mathbf{\Sigma}_{sd}(1:r, 1:r)^{(-0.5)}.$$

The $\bar{\mathbb{A}}_r$ has just $r \times r$ size because those $r \times r$ matrix can represent and replace the original $\bar{\mathbb{A}}_{id}$ matrix. The $\bar{\mathbb{A}}_r$ is same with the first $r \times r$ parts of $\bar{\mathbb{A}}_{id}$.

$$\bar{\mathbb{A}}_r = \bar{\mathbb{A}}_{id}(1:r, 1:r).$$

Figure 7.1, the Gerschgorin theorem and the property of singular value decomposition give us a way to select a system order, i.e., a sudden change of the Gerschgorin radii before and after r, causes the identified system unstable and unimportant.

After the $\bar{\mathbb{A}}_{id}$ state-space matrix is no longer than diagonal dominant, the Gerschgorin radius is increasing rapidly because the non-diagonal elements affect the Gerschgorin radius. As a result, a new criterion for model reduction can be presented by following equation.

$$10 \times radii(j, 1:j) > radii(j, :)$$

where $j = 1, 2, \cdots, r-1$. Figure 7.3 shows us the simple flowchart of model reduction method with the Gerschgorin radii.

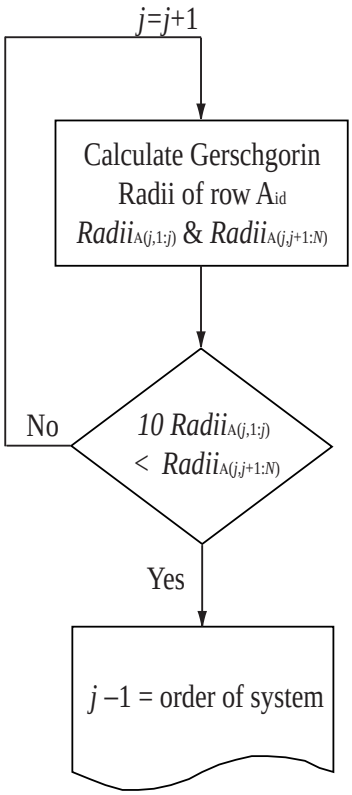

Figure 7.3: Flowchart of selecting system order with the Gerschgorin radii.

7.4 Example

For checking if the proposed model reduction method is working well, let us use an example which has been used in Chapters 5. According to the proposed model reduction method, the magnitude of the identified state-space matrix, $\bar{\mathbb{A}}_{id}$, is obtained as shown in Figure 7.4. The first 2 diagonal elements of the state-space $\bar{\mathbb{A}}_{id}$, matrix are relatively dominant. When the sizes of the third Gerschgorin disks are checked, the radius is increasing very much. Let us follow the detail procedure of the new model reduction method.

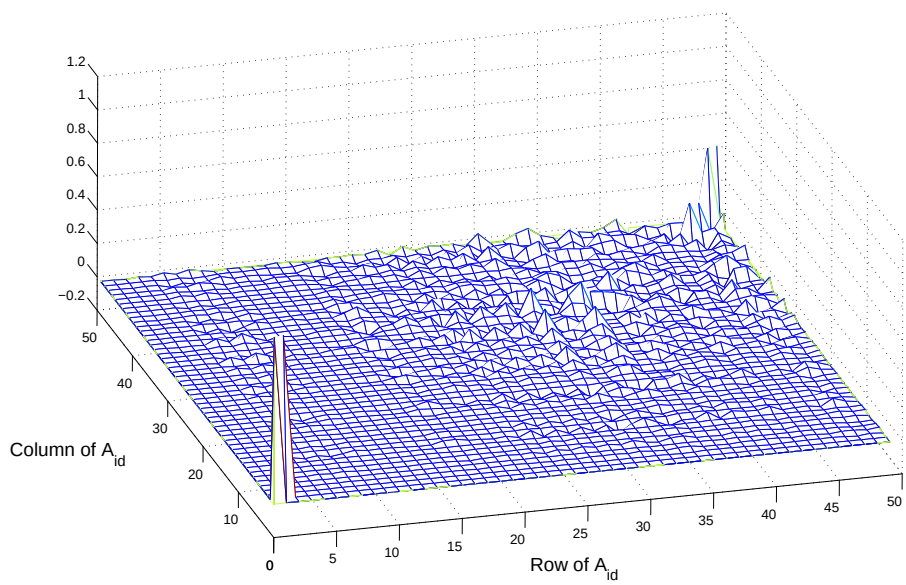

Figure 7.4: Parts of the magnitude of the identified state-space $\bar{\mathbb{A}}_{id}$, matrix.

Using the magnitude of the identified state-space $\bar{\mathbb{A}}_{id}$ matrix, the radii of the Gerschgorin disks can be obtained. Table 7.1 shows the radii of the Gerschgorin disk. According to Table 7.1, the third row of the identified state-space $\bar{\mathbb{A}}_{id}$ matrix has changed very much as one more column is added for calculating new radius of the Gerschgorin disk. Therefore, the proposed model reduction method will select 2 as a order of identified system.

Table 7.2 shows us the reduced order of the gyroscopic system which are obtained with

Table 7.1: Ratio of the Gerschgorin disk radius of the gyroscopic example

	Radius of pre-diagonal	Radius of post-diagonal	Ratio
$n = 2$	0.0013891	0.0013891	1
$n = 3$	1.03E-09	0.00020325	196966.7603
$n = 4$	0.00045812	0.04222	92.15925958
$n = 5$	0.0007404	0.10039	135.5888709
$n = 6$	0.10051	0.10051	1
$n = 7$	0.00083141	0.055249	66.45217161
$n = 8$	0.055417	0.055792	1.006766877
$n = 9$	0.0052246	0.022563	4.318608123
$n = 10$	0.022498	0.022541	1.001911281
$n = 11$	0.0012593	0.0012593	1
$n = 12$	2.27E-07	0.0071086	31278.21534

various model reduction methods such as the subspace method, black box method, singular value of the Hankel Matrix and the proposed method. The first two model reduction methods have already created in MATLAB programming. They choose 3 for the identified system order. The proposed model reduction and singular value model reduction method pick the same order with the theoretical system order. The singular value method and the new proposed method are made for subroutine functions (See Appendix A).

Table 7.2: The system order depending on the model reduction methods.

	Subspace	Black-Box	Singular Value	Proposed
System Order	3	3	2	2

Chapter 8 will provide real examples for verifying those methods and investigating some advantages in detail.

Chapter 8

FLYWHEEL ENERGY STORAGE SYSTEM

8.1 Introduction

In this chapter, both computer-simulated data and actual experimental data of the Flywheel Energy Storage System (FESS) will be used to determine a system model. Here, we will use the new proposed identification technique developed in 6. In addition, the ARX technique will also be used to identify the system model. Determination of the model order will be performed using the new method and the other techniques as presented in Chapter 7. The FESS is a perfect example to apply the new identification technique with complex variable transformation because the FESS includes the gyroscopic element. Let us review briefly the general properties and the history of the FESS before we identify the FESS model.

Development of FESS

In J. David Stienmier's dissertation [91], he has explained the advantages of the FESS and designed prototypes of the FESS. As explained in his dissertation, the FESS has two big important advantages, i.e., (i) quick charging/discharging rate and (ii) environment-friendly device. He used a non-contact magnetic bearing for improving the efficiency of charging and discharging.

The other former researcher, Rainer Leuschke, had illustrated studying the feasibility of the proposed actuation method and experimentally verified its functionality and performance. In addition, he showed several methods for adding external actuators to the FESS and implemented some disturbance rejection controllers [63]. In the paper [64], the new dynamics description of the FESS was derived with several additional terms, i.e., (i) the disturbance forces transmitted to the rotor via the magnetic bearings and (ii) the disturbance due to the unbalance of the FESS. This model gives us good results to identify the simulated data of the FESS.

By these significant developments, the system identification methods can be applied to the FESS for further its analysis. Subsection 8.2.1 describes the FESS based upon the equations of motion in [64] and we apply the complex variable transformation in subsection 8.2.2. Sections 8.2 and 8.3 will illustrate the following results.

- Results of the new identification technique.

- Selection of the model order by several methods.

- Comparison of model selection methods.

- Results of ARX model.

8.2 Simulated Data

8.2.1 Background - Equations of motion

The behavior of the gyroscopic system is governed by

$$\begin{aligned} \mathbf{M}\ddot{q} + \mathbf{G}(\omega)\dot{q} + \mathbf{D}_{amp}\dot{q} + \mathbf{K}q &= \mathbf{E}f, \\ y &= \mathbf{c}q + v, \end{aligned} \tag{8.1}$$

where $q \in \Re^{N\times 1}$ is a states vector of system, N_{out} is a number of outputs, N_{in} is a number of inputs, $\mathbf{M} \in \Re^{N\times N}$ is a symmetric inertia matrix, $\mathbf{G}(\omega) \in \Re^{N\times N}$ is a skew-symmetric gyroscopic matrix, ω is a spinning speed, $\mathbf{D}_{amp} \in \Re^{N\times N}$ is a symmetric damping matrix, $\mathbf{K} \in \Re^{N\times N}$ is a symmetric stiffness matrix, $\mathbf{E} \in \Re^{N\times N_{in}}$ is an input matrix, $f \in \Re^{N_{in}\times 1}$ is an input vector, $y \in \Re^{N_{out}\times 1}$ is an outputs vector, $v \in \Re^{N_{out}\times 1}$ is a disturbance vector, $\mathbf{c}$ $\in \Re^{N\times N}$ is an output matrix.

Typically, control system designers describe these systems by first order differential equations, i.e., in the state-space form. After transforming, we can obtain the following state-space form

$$\begin{aligned} \dot{x} &= \mathbf{A}x + \mathbf{B}f, \\ y &= \mathbf{C}x + v, \end{aligned} \tag{8.2}$$

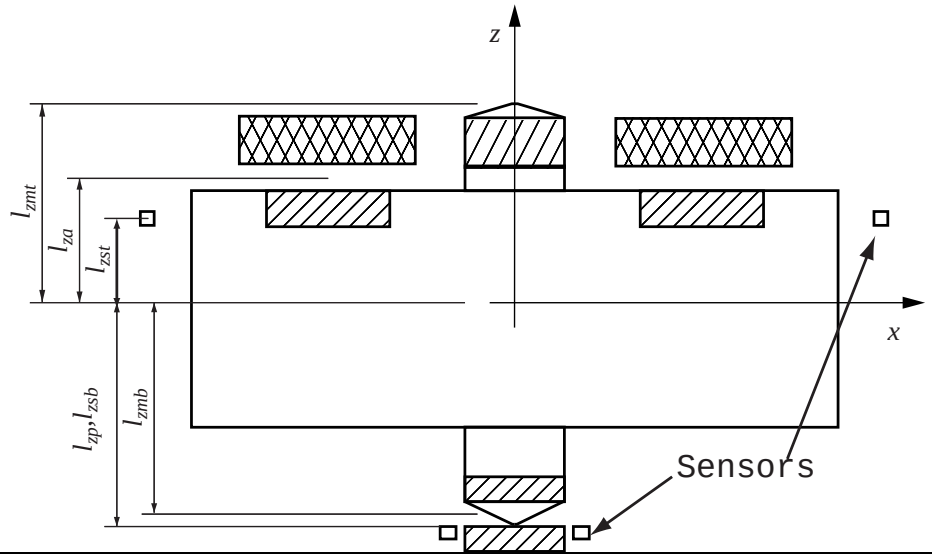

Figure 8.1: The flywheel energy storage system parameters

where

$$\bar{x} = \begin{bmatrix} q \\ \dot{q} \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}(\mathbf{G}(\omega) + \mathbf{D}) \end{bmatrix},$$
$$\mathbf{B} = \begin{bmatrix} \mathbf{0} \\ \mathbf{M}^{-1}\mathbf{E} \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} \mathbf{c} & \mathbf{0} \end{bmatrix}.$$

In the case of the Flywheel Energy Storage System (FESS) [64], we have

$$\begin{aligned} \mathbf{M}\ddot{q} + (\mathbf{G}(\omega) + \mathbf{D})\dot{q} + \mathbf{K}q &= \mathbf{E}f + \mathbf{D}v + d, \\ y &= \mathbf{C}q, \end{aligned} \tag{8.3}$$

where

$$
\begin{aligned}
\mathbf{M} &= \begin{bmatrix} m & 0 & 0 & 0 \\ 0 & m & 0 & 0 \\ 0 & 0 & I_x & 0 \\ 0 & 0 & 0 & I_y \end{bmatrix} \\
\mathbf{G}(\omega) &= \begin{bmatrix} c & 0 & 0 & -cl_{zp} \\ 0 & c & cl_{zp} & 0 \\ 0 & cl_{zp} & cl_{zp}^2 + c_\tau & I_z\omega \\ -cl_{zp} & 0 & -I_z\omega & cl_{zp}^2 + c_\tau \end{bmatrix}, \\
\mathbf{K} &= \begin{bmatrix} k_{11} & 0 & 0 & k_{14} \\ 0 & k_{11} & -k_{14} & 0 \\ 0 & -k_{14} & k_{33} & 0 \\ k_{14} & 0 & 0 & k_{33} \end{bmatrix}, \\
\mathbf{D} &= \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & -l_{zmt} & 0 & l_{zmb} \\ l_{zmt} & 0 & -l_{zmb} & 0 \end{bmatrix}, \\
\mathbf{E} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & -l_{za} \\ l_{za} & 0 \end{bmatrix}.
\end{aligned}
$$

Here, the generalized coordinates are $q = [x, y, \theta_x, \theta_y]^T$, with x and y for displacements of the flywheel center of mass. Also, θ_x and θ_y are the angular displacements about x and y axis respectively. ω denotes the angular velocity of the flywheel, and the radial actuator forces form the motor are $f^T = [f_x, f_y]$. The vector v and d represents the disturbance forces that are transmitted to the rotor via the magnetic bearings and the disturbance due to the unbalance of the flywheel, respectively. The FESS has same mass moment of inertia about x and y, i.e., $I_x = I_y$. In addition, elements of the stiffness matrix can be calculated

as follows

$$
\begin{aligned}
k_{11} &= k_b + k_t, \\
k_{14} &= -k_b l_{zmb} + k_t l_{zmt}, \\
k_{33} &= k_t l_{zmt}^2 + k_b l_{zmb} - mgl_{zp},
\end{aligned}
$$

where g denotes the gravity acceleration.

For making the FESS simple, we use the identity matrix instead of output matrix $\mathbf{C}$ and neglect the $\mathbf{D}$ matrix and d matrix. The FESS, however, is still a non-square plant, i.e, 2 inputs, 4 outputs.

The parameters of FESS are on Table 8.1.

Table 8.1: Parameters of the flywheel energy storage system.

Notation	Value	Notation	Value	Notation	Value
I_x	0.0030162 kg/m^2	I_y	0.0030162 kg/m^2	I_z	0.0044248 kg/m^2
m	2.094 kg	k_t	1490 N/m	k_b	1047 N/m
l_{zp}	0.04505 m	l_{zmt}	0.03995 m	l_{zmb}	0.04505 m
l_{zst}	0.01671 m	l_{zsb}	0.04505 m	l_{zct}	0.02495 m

Figure 8.2 shows us the impulse responses of the FESS which is based on the equations of motion (8.3).

8.2.2 Complex Coordinates

In Subsection 8.2.1, the equations of motion of FESS are based upon the $x - y - \theta_x - \theta_y$ coordinates. For using the complex variable representation and eliminating the skew-symmetric terms, we can use the complex variable coordinates as explained in Subsection 6.1.2,

$$
\begin{aligned}
\alpha &= x + yi, \\
\theta_\alpha &= \theta_y - \theta_x i,
\end{aligned}
\tag{8.4}
$$

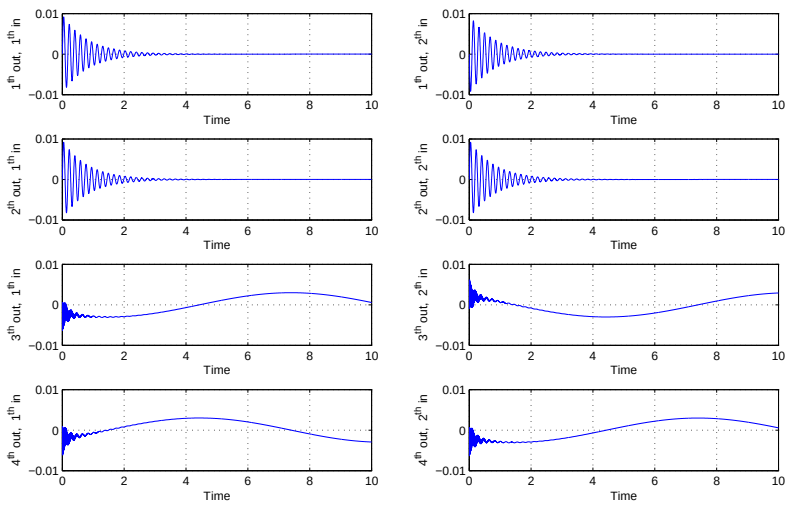

Figure 8.2: Impulse responses of the FESS with $\omega = 1500$RPM

where $i = \sqrt{-1}$. Using the transformation equation (8.4), we can rewrite the equations of motion of FESS (8.3) as follows

$$\begin{aligned} m\ddot{\alpha} + c\dot{\alpha} - cl_{zp}\dot{\theta}_\alpha + k_{11}\alpha + k_{14}\theta_\alpha &= F_x + iF_y, \\ I\ddot{\theta}_\alpha - cl_{zp}\dot{\alpha} + \left(cl_{zp}^2 + c_\tau - I_3\omega i\right)\dot{\theta}_\alpha + k_{14}\alpha + k_{33}\theta_\alpha &= l_{za}\left(-iF_y + F_x\right). \end{aligned} \tag{8.5}$$

The equations of motion with complex coordinates (8.5), show that the FESS is 2×2 system instead of 4×2.

8.2.3 Applying Complex Variables ERA

The identified complex variable state-space matrix $\bar{\mathbb{A}}_{id}$, $\bar{\mathbb{B}}_{id}$, $\bar{\mathbf{C}}_{id}$ and $\bar{\mathbf{D}}_{id}$ are following

$$\begin{aligned} \bar{\mathbb{A}}_{id} &= \mathbf{\Sigma}^{(-1/2)}\bar{\mathbf{U}}^H\bar{\mathbf{H}}(1)\bar{\mathbf{V}}\mathbf{\Sigma}^{(-1/2)}, \\ \bar{\mathbb{B}}_{id} &= \mathbf{\Sigma}^{(1/2)}\bar{\mathbf{V}}^H \begin{bmatrix} \mathbf{I}_{N_{order}\times N_{in}} \\ \mathbf{0}_{(d-N_{order})\times N_{in}} \end{bmatrix}, \\ \bar{\mathbf{C}}_{id} &= \begin{bmatrix} \mathbf{I}_{N_{out}\times N_{order}} & \mathbf{0}_{N_{out}\times(q-N_{order})} \end{bmatrix} \bar{\mathbf{U}}\mathbf{\Sigma}^{(1/2)}, \\ \bar{\mathbf{D}}_{id} &= \bar{\mathbf{M}}_{arkov}(0). \end{aligned} \tag{8.6}$$

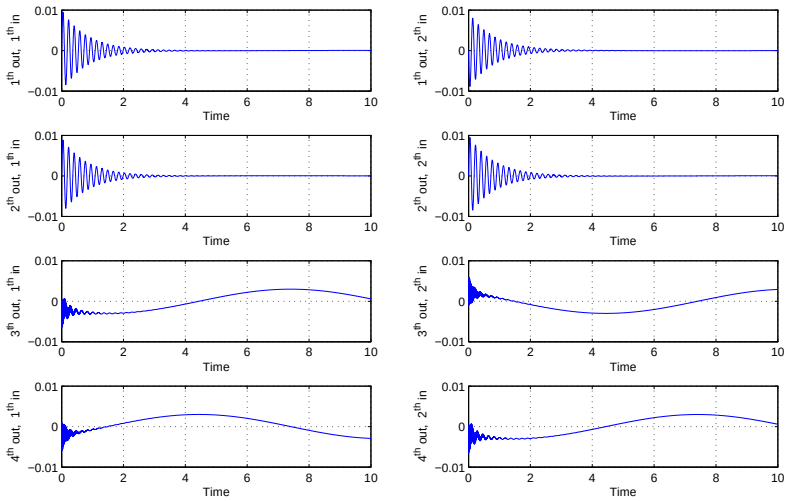

Figure 8.3: Complex variables Identified responses of the figure 8.2.

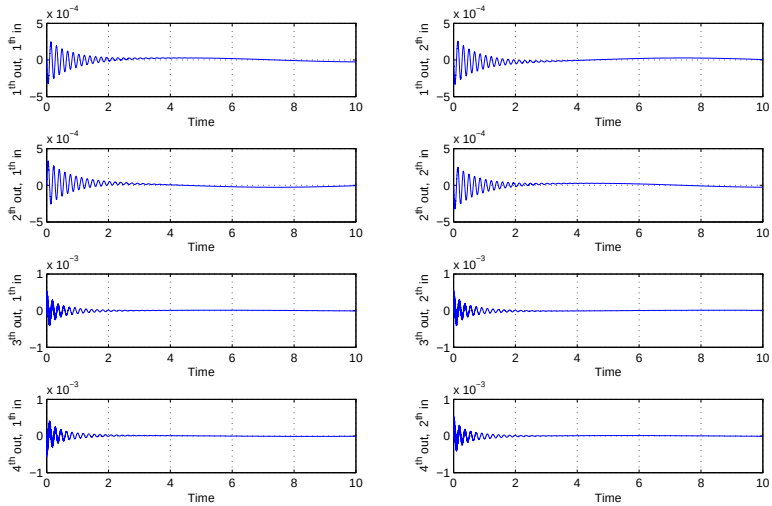

Figure 8.4: Errors between the figure 8.2 and the figure 8.3.

The complex variables ERA method (8.6) provides us Figure 8.3.

To evaluate the accuracy of the identified model we use the error measure;

$$Error\% = \frac{||y_{ori} - y_{id}||_2}{||y_{ori}||_2} \times 100, \tag{8.7}$$

where y_{id} is the response of the identified model, and y_{ori} is the experimentally measured response.

Table 8.2 shows us very good error percentages between the original data and the identified data.

Table 8.2: Error percentages table

	x direction	y direction	θ_x direction	θ_y direction
Error percentage	0.117%	0.1299%	0.0354 %	0.0479%

8.2.4 Applying Model Reduction

Here, we will apply the new model reduction technique to data of Figure 8.3. Before we use the new technique, all concepts of the new model reduction algorithm will be explained step by step.

Let us consider the magnitude of the elements of the identified state-space $\bar{\mathbb{A}}_{id}$ matrix. Figure 8.5 shows us the magnitude of the elements of the identified state-space $\bar{\mathbb{A}}_{id}$ matrix. Roughly analyzed Figure 8.5, the diagonal elements of the first four rows and columns of $\bar{\mathbb{A}}_{id}$ matrix are dominant, and the non-diagonal elements are relatively small. This observation can be adopted to the Gerschgorin disk theorem. In the theorem, the radius of the Gerschgorin disk can be divided by two parts - (i) pre-diagonal elements, and (ii) post-diagonal elements. For the fifth Gerschgorin disk, the post-diagonal elements after the fifth column make the Gerschgorin radius large. For the more detail analysis, let us use Table 8.3.

Table 8.3 shows us the ratios of the Gerschgorin disks radii. In Table 8.3, the second

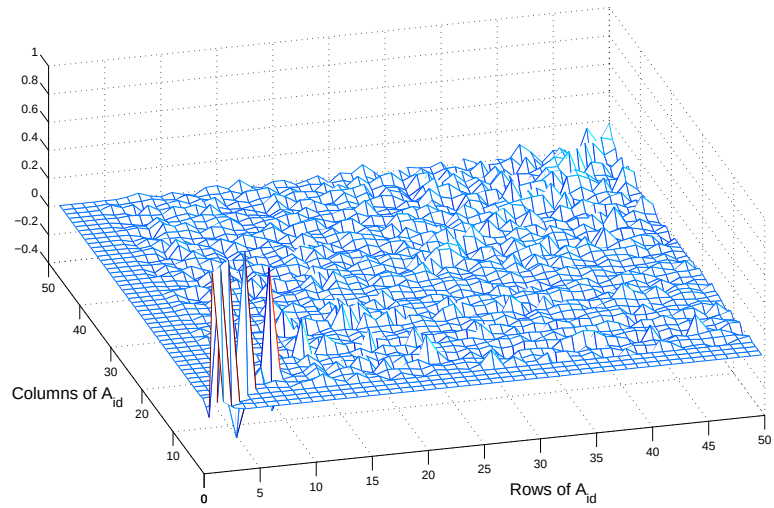

Figure 8.5: Magnitude of the identified state-space $\hat{\bar{\mathbb{A}}}_{id}$ matrix.

column and third column imply the radii of the Gerschgorin disk, i.e.,

$$
\begin{aligned}
radii_{n-1} &= \sum_{j=1}^{n-1} \mid \bar{a}_{jk} \mid, \\
radii_{n+1} &= \sum_{j=1, k \neq j}^{n-1} \mid \bar{a}_{jk} \mid .
\end{aligned}
$$

Table 8.3 shows that when $n = 5$ the ratio is greater than 10. Finally, after calculating the ratio of the radius of the Gerschgorin disks, the system order can be selected $N_{order} = 4$, which means that the system order of the experimental data for the FESS would be eight if we use the real coordinate system. The theoretical system order is matching with the order used our new model reduction technique. Table 8.4 shows the comparison of the system orders with several different model reduction methods. Note that the singular value method also used the complex variable Hankel matrix. In addition, a system order, $N_{order}^{complex} = 4$, for the complex variable coordinates implies that when using real coordinates the system order, $N_{order} = 8$. The proposed model determination technique gives us very good results.

Table 8.3: Ratio of the Gerschgorin disk radius for the simulated data of FESS

	Gerschgorin radius $radii_{n-1}$ pre-diagonal	Gerschgorin radius $radii_{n+1}$ post-diagonal	Ratio $\alpha = \frac{radii_{n+1}}{radii_{i-1}}$
$n = 2$	0.066007	0.06603	1.0003
$n = 3$	0.025086	0.034901	1.3912
$n = 4$	0.04917	0.04917	1
$n = 5$	4.3837e-009	0.031462	7.1771e+006
$n = 6$	0.0098197	0.0099607	1.0144
$n = 7$	0.025583	0.23213	9.0735
$n = 8$	0.22694	0.22962	1.0118
$n = 9$	0.0667	0.067728	1.0154
$n = 10$	0.028682	0.030358	1.0584
$n = 11$	0 0.034068	0.050163	1.4725
$n = 12$	0.11717	0.12714	1.0851
$n = 13$	0.036522	0.047087	1.2893

Table 8.4: The system order depending on the model reduction methods.

	Real number state		Complex number state	
	Subspace	Black-Box	Singular Value	Proposed
System Order	8	8	4	4

8.2.5 Applying the ARX Model

As explained in Chapter 5 the simulated data of the FESS using the real coordinates are applied to the ARX model. In ARX model, we use 5, 3, and 5 as orders of A, B, and K polynomials respectively. Increasing orders of the polynomials guarantee the better identified responses. Figure 8.6 shows us the responses of the simulated FESS using the ARX model.

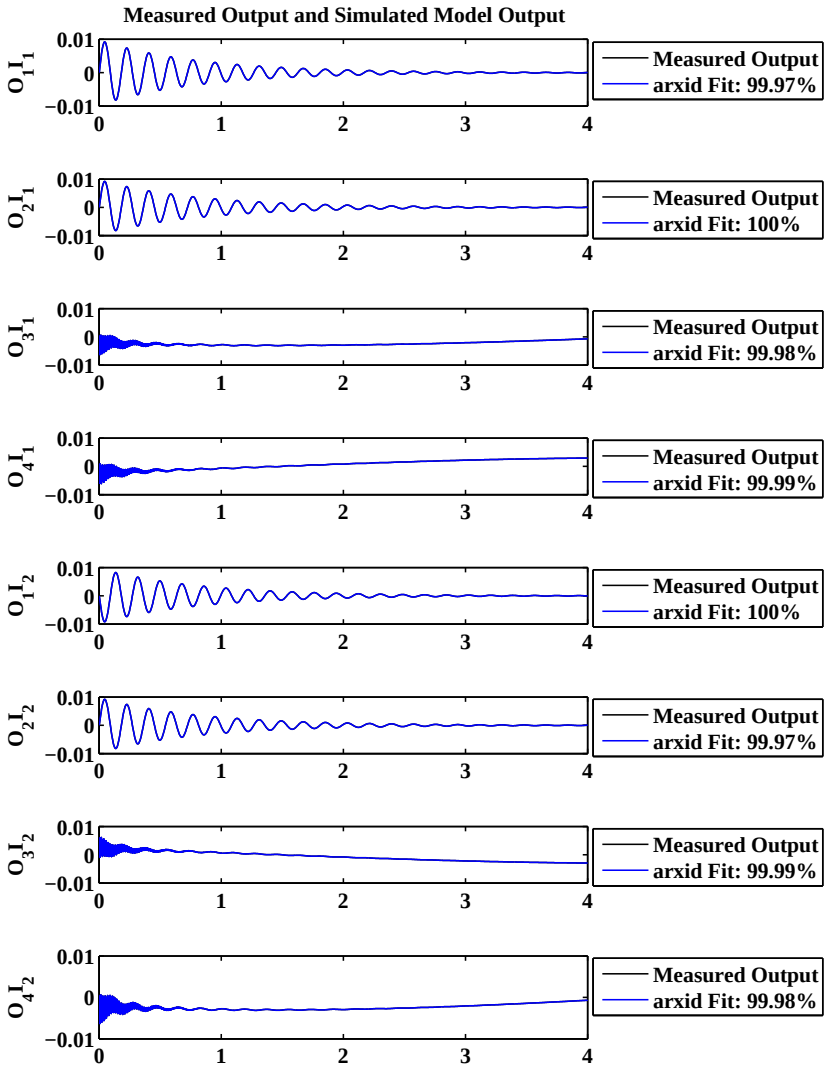

Figure 8.6: Identified responses of the simulated FESS with ARX model.

8.3 Experimental Data

For further application, let us consider the experimental data of the FESS. The sampling times are 2, 5, or 10 milliseconds written on top of a recorded text file. Those sampling times are very reliable [63]. We will use 5 milliseconds data. In addition, the observed data are the top displacement of x axis, x_{top}, the top displacement of y axis, y_{top}, the bottom displacement of x axis, x_{bottom}, and the bottom displacement of y axis, y_{bottom}. It is necessary to change the observed variables to the theoretical variables, i.e., x_{center}, y_{center}, θ_x, and θ_y, for comparing and analyzing data. The relationship equations between observed variables and theoretical variables are following.

$$\begin{aligned} x_{center} &= \frac{x_{top}l_{zst}+x_{bottom}l_{zsb}}{l_{zst}+l_{zsb}}, \\ y_{center} &= \frac{y_{top}l_{zst}+y_{bottom}l_{zsb}}{l_{zst}+l_{zsb}}, \\ \theta_x &= \frac{x_{top}-x_{bottom}}{l_{zst}+l_{zsb}}, \\ \theta_y &= \frac{y_{top}-y_{bottom}}{l_{zst}+l_{zsb}}. \end{aligned} \tag{8.8}$$

After the observed data are converted to theoretical variables, the ERA identification method with complex variable coordinates is applied.

Following two subsections illustrate the results of identifying non-spinning data and spinning data respectively. In the spinning data cases, we choose the data of 700 RPM because the 700 RPM is close to twice of the natural frequency of the FESS. In addition, using the same data, the model reduction method will be applied Subsection 8.3.3.

8.3.1 Non-Spinning Data

Let us consider the non-spinning experimental data of the FESS and analyze them systematically. If the FESS does not spin, a force of any radius direction makes only one directional motion along with the direction of the force. Therefore, we can theoretically describe a non-spinning FESS with only 2 coordinates (r and θ) instead of 4 coordinates (x_{center}, y_{center}, θ_x, and θ_y), i.e., non-spinning FESS has two degree of freedom.

As Figure 8.7 shows us, however, experimental data are not matching with the theoretical analysis, which means that the degree of freedom will be more than 2.

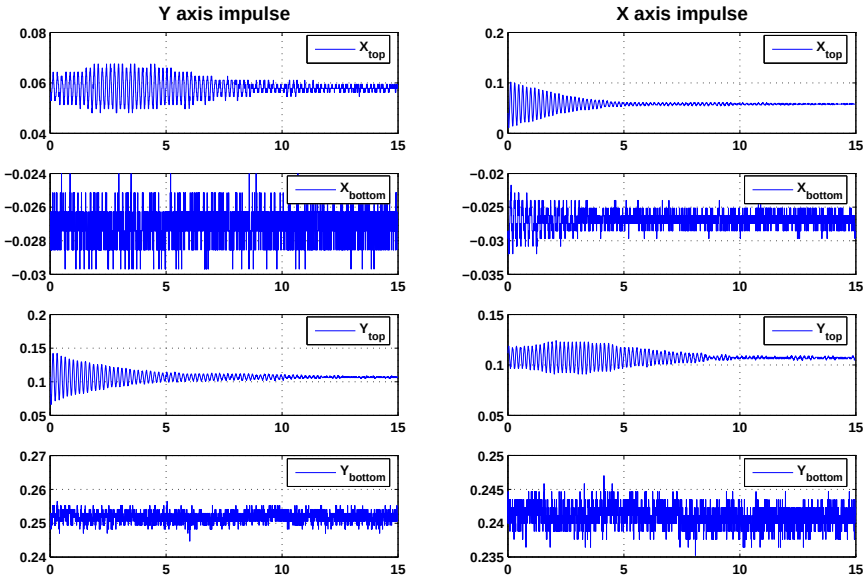

Figure 8.7: Non-spinning FESS experimental data (x_{top}, x_{bottom}, y_{top}, and y_{bottom}) with 0.005 sec sampling time.

Using many data of the FESS is required to identify experimental responses as explained in Section 5.3. In addition, the proposed model reduction technique will give us ideal system order later.

All observed data (x_{top}, x_{bottom}, y_{top}, and y_{bottom}) can be converted to the theoretical coordinates system (x_{center}, y_{center}, θ_x, and θ_y) using equations (8.8). The converted responses of Figure 8.8 are used for identifying the non-spinning FESS.

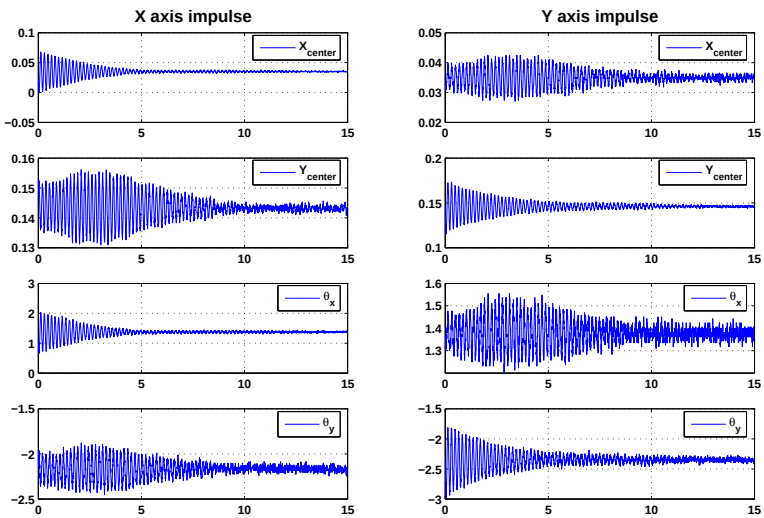

Figure 8.8: Converted non-spinning FESS experimental data (x_{center}, y_{center}, θ_x, and θ_y) with 0.005 sec sampling time.

The converted data - x_{center}, y_{center}, θ_x, and θ_y are applied to the complex variable coordinate (8.4), then go through the identification method as explained Chapter 5 and 6.

The responses of identified system can be affected by several important factors - sampling time, size of Hankel matrix, and system order. Let us consider these factors.

By Sampling Time

The observed data have 5 millisecond sampling time. By picking up data with an interval, the sampling time can be changed on purpose. However, without changing the size of Hankel

matrix and system order, the sampling time does not affect results of identifying system.

By Size of Hankel Matrix

As indicated Table 8.5, 300 × 300 Hankel Matrix has the best error percentages but there is no any tendency as the size of Hankel Matrix is increasing. Figure 8.9 shows us the responses of identified system with 300 × 300 Hankel Matrix, 5 milliseconds sampling time, and 4 system order which is assumed theoretically.

Table 8.5: Error percentages with 4-order system and various Hankel matrix.

	H_{150}		H_{300}		H_{600}		H_{1200}	
Direction	X	Y	X	Y	X	Y	X	Y
x_{center}	9.8974	8.8889	4.4813	2.4755	7.451	2.6761	10.477	4.2036
y_{center}	0.21652	0.30412	0.21756	0.358	0.33791	0.21884	0.13921	0.22509
θ_x	1.3984	2.5469	1.3987	0.73819	2.6575	0.68998	2.6008	1.3334
θ_y	0.28384	0.54799	0.33325	0.3713	0.36064	0.32056	0.15008	0.35245

By System Order

Choosing the system order of identified system is very important because responses of an improper system order are unstable. By the proposed model reduction algorithm, the identified system order should be 2, which we have assumed in previous section 8.3.1. The responses of 2-system order have good error percentages but they are so much simplified as shown Figure 8.10.

8.3.2 Spinning Data

In spinning data case, 700 RPM experimental data are used for identifying FESS. It should be noted that we filter the spinning data due to the unbalance rotation which is from manufacturing difficulty. As a result, the notch filter, which is capable to eliminate a specific frequency band, is designed. The design method of filter is based upon the Butterworth

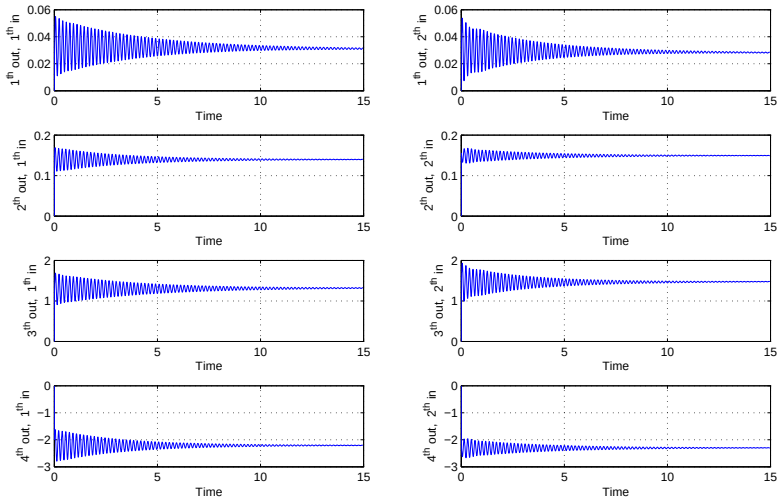

Figure 8.9: $H_{300\times300}$ and 4 order identified system responses

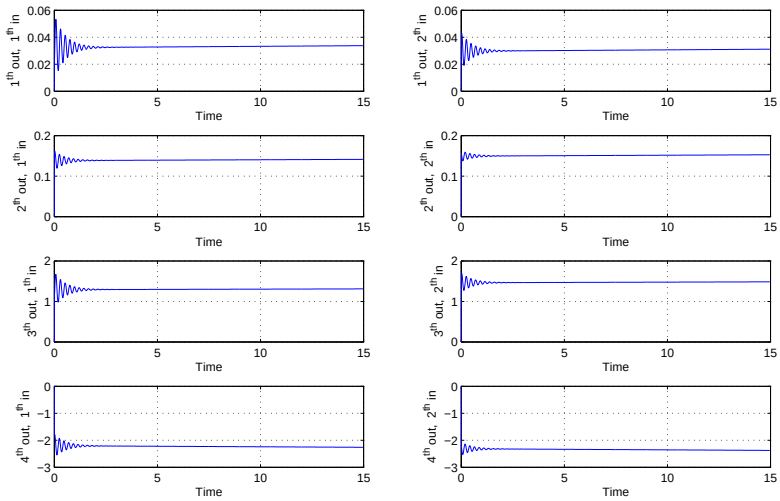

Figure 8.10: $H_{300\times300}$ and 2 order identified system responses

filter [11]. The width of stop band is range of 695 $\sim$ 705 rpm and sampling frequency is 2500 Hz. Figure 8.11 illustrates the frequency response and Bode plot of Butterworth notch filter. After filtering spinning FESS data through notch filter (Figure 8.12) and converting to x_{center}, y_{center}, θ_x, and θ_y coordinates, we can identify them.

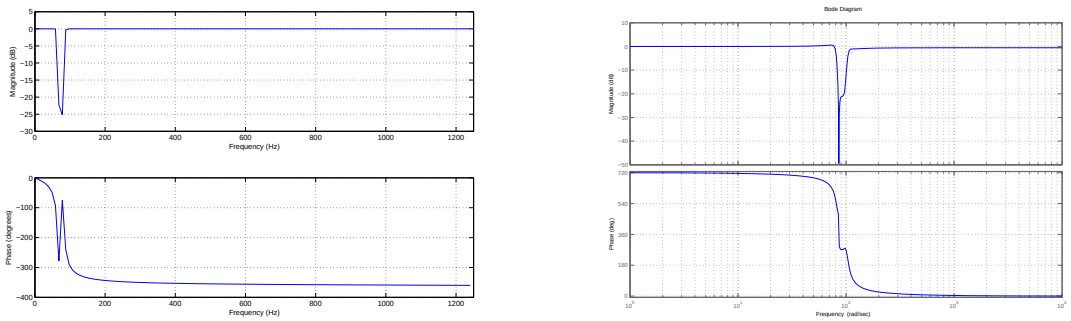

Figure 8.11: 2500Hz Filter response

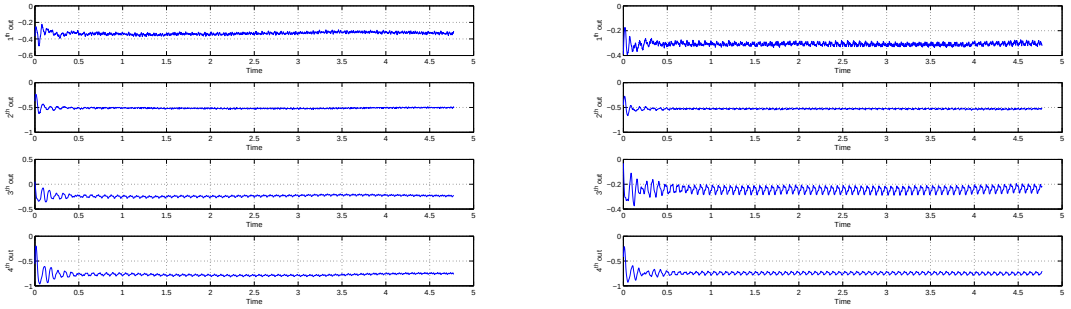

Figure 8.12: Time response of X- Y axis impulse after filtering

As explained in Subsection 8.3.1, the sampling time does not affect the identified results. For the spinning FESS example, 2 milliseconds sampling time is used. By size of Hankel matrix and system order, results of identifying spinning FESS data are analyzed as follows.

By Size of Hankel matrix

Table 8.6 reports the error measure for each of the (real) state variables for different Hankel matrix sizes, while Figure 8.13 plots the average error of all the state variables as a function of the Hankel matrix size.

As can be seen the accuracy of the identified model improves with Hankel matrix size. The best overall result is obtained with a Hankel matrix size of 2388 $\times$ 2388.

Table 8.6: Error percentages - 700 RPM, 2 system order, and various size of Hankel matrix

	100×100	200×200	300×300	400×400
x	1118.8	6820.9	0.79735	1.7255
y	804.27	193.58	64.284	8.1543
θ_x	5825.1	1695.7	162.38	10.952
θ_y	308.94	979.97	12.29	3.8391
	500×500	750×750	1000×1000	1250×1250
x	1.1443	0.53723	0.45758	0.44445
y	5.0394	2.8409	2.8222	2.6821
θ_x	7.5482	9.6684	7.4167	7.364
θ_y	3.0425	1.6853	1.4518	1.4336
	1500×1500	1750×1750	2000×2000	2380×2380
x	0.47443	0.56035	0.63082	0.80279
y	2.2765	1.822	1.4296	1.0854
θ_x	6.5998	6.0955	6.0834	6.6108
θ_y	1.418	1.3893	1.3619	1.3357

By System Order

The selection of the system order is another important factor in the ERA system identification technique. Here, we consider results obtained by fixing the Hankel matrix size at

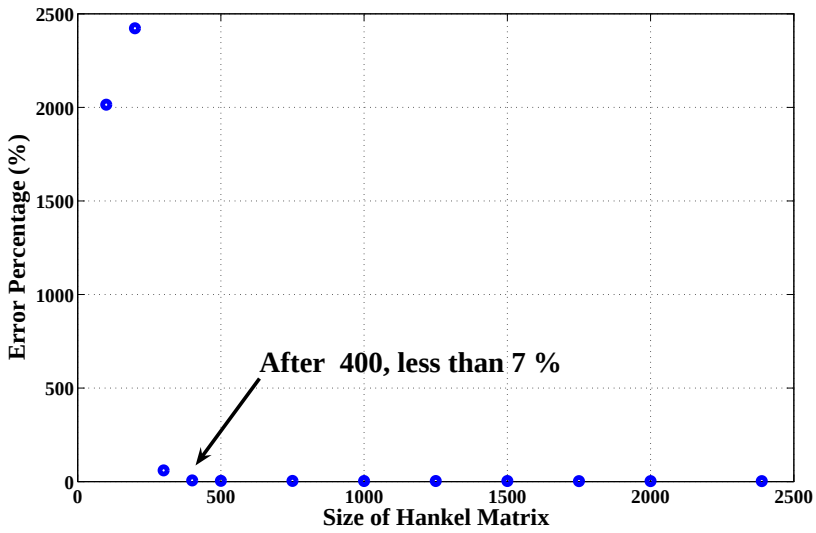

Figure 8.13: Average error percentages by size of Hankel matrix

2388 $\times$ 2388, and varying the assumed order of the identified complex variable system. The results obtained are shown in Table 8.7 and Figure 8.14.

As can be seen the error measure is the smallest for the complex variable system with assumed orders 1 and 2. Note that this corresponds to systems of order 2 and 4 if real state variables are used. Also, the analytical model of the FESS has order 4 with real state variables. As the assumed order of the complex variable system is increased the identified model becomes less accurate. Thus using actual experimental data we are able to identify a model for the FESS that produces small relative errors ($< 7\%$) in the state response.

Table 8.7: Error percentages - 700 RPM, various system order

	order 1	order 2	order 3	order 4	order 5	order 6
x	0.8017	0.8023	0.8495	0.7008	0.7085	0.7249
y	1.084	1.0848	1.1486	1.3182	1.3205	1.3235
θ_x	6.6151	6.6199	7.3662	6.9114	7.015	7.1942
θ_y	1.336	1.3359	1.6698	2.5548	2.5642	2.6229
	order 7	order 8	order 9	order 10	order 11	order 12
x	2.0546	2.6008	2.8402	2.8995	1.8712	1.8708
y	4.3137	3.5584	3.3674	3.3687	6.313	6.3408
θ_x	7.3555	9.6956	9.9547	10.614	12.06	12.043
θ_y	2.6634	3.2226	3.2192	3.2244	3.2495	3.2466

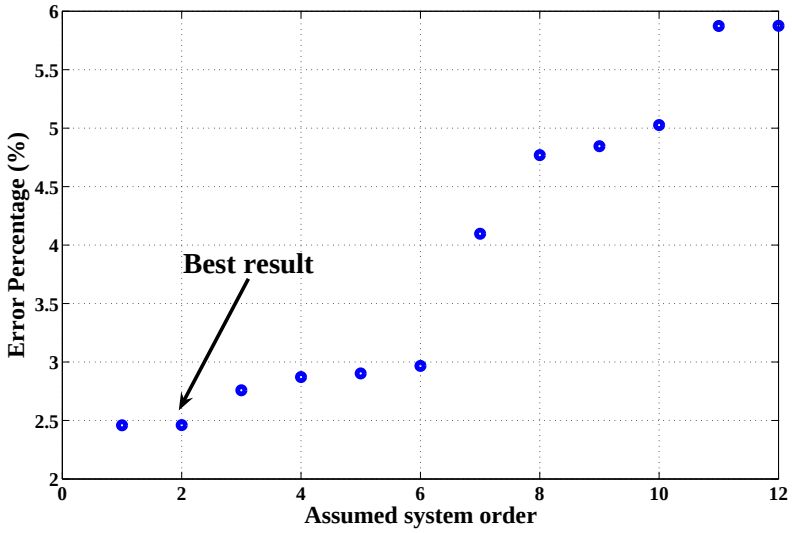

Figure 8.14: Average error percentages by system order

Using two as a system order, the identified system model in this case is

$$
\begin{aligned}
\bar{\mathbb{A}}_{id} &= \begin{bmatrix} 1 & 0 \\ 0 & 0.9829 + 0.18399i \end{bmatrix}, \\
\bar{\mathbb{B}}_{id} &= \begin{bmatrix} 1.8667 & 1.5114 - 0.0821i \\ 0.022185 & 0.3937 + 0.5774i \end{bmatrix}, \\
\bar{\mathbf{C}}_{id} &= \begin{bmatrix} -0.16683 - 0.3406i & 0.013966 - 0.012026i \\ 2.2327 + 0.8039i & 0.6920 + 0.1330i \end{bmatrix}, \\
\bar{\mathbf{D}}_{id} &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}.
\end{aligned}
$$

The identified system model has the following eigenvalues; $\lambda_1 = 1$ and $\lambda_2 = 0.9829 + 0.18399i$ which indicates that the system is stable.

8.3.3 Model Selection with Experimental Data

Figure 8.15 illustrates magnitudes of the first 50×50 of $\bar{\mathbf{A}}_{id}$ which originally has 2388×2388. Now, apply the proposed model reduction method.

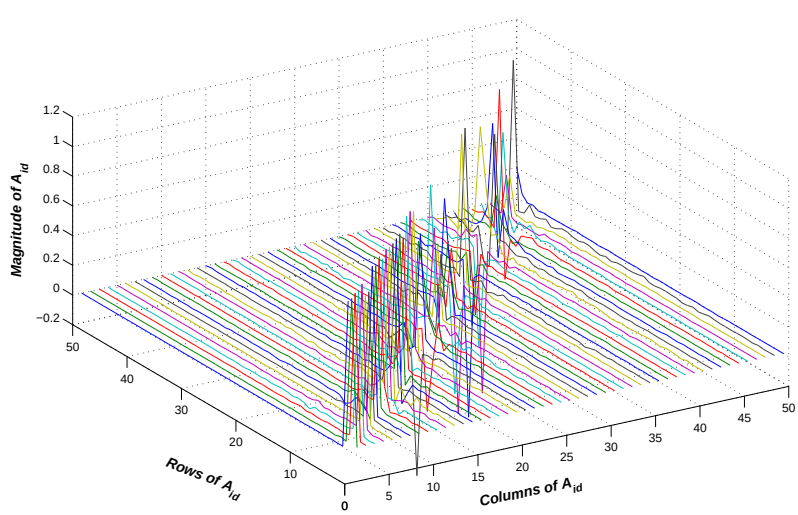

Figure 8.15: The Magintude of the identified state-space $\bar{\mathbf{A}}$.

The complex variable identified $\bar{\mathbf{A}}$ has following the Gerschgorin radii.

$$\begin{aligned} radii_{\bar{\mathbf{A}}_{id}(2,1:2)} &= 0.00054067, \\ radii_{\bar{\mathbf{A}}_{id}(2,:)} &= 0.054344, \\ \therefore \frac{radii_{\bar{\mathbf{A}}_{id}(2,:)}}{radii_{\bar{\bar{\mathbf{A}}}_{id}(2,1:2)}} &= 100.51. \end{aligned} \tag{8.9}$$

Equation (8.9) simply satisfies the proposed model reduction method with system order 2, which means the system order of the spinning FESS is 4 which is matching with the ideal-theoretical modeling. An ideal system order of spinning FESS is 4 with $[x, \theta_x, \dot{x}, \dot{\theta}_x]$. The error percentages of responses are very good - Table 8.8.

Table 8.8: Error percentages - 700 RPM

Input	x	y	θ_x	θ_y
1st	0.4576	2.8222	7.4167	1.4518

In Table 8.8, θ_x of the second input has a bigger error percentage because even filtered data of θ_x still oscillate.

The singular value method is unable to choose a proper system order with experimental data because singular values of Hankel matrix are changing very slightly. In addition, the AIC gives us 10 for system order of FESS. As a result, the proposed model reduction method illustrates good results for the complex variable identified state-space model.

8.3.4 ARX Model

For identifying data of the spinning FESS with ARX model, we use same orders in Subsection 8.2.5 first. As shown in Figure 8.16, the responses are so simplified. As a result, for better responses, it is necessary to increase orders of the polynomials.

For Figure 8.16, we use 25, 23, and 25 for orders of A, B, and K polynomials respectively. According to the '*compare*' command, the higher order gives us the better identified responses, however, the calculating time will be increasing very much.

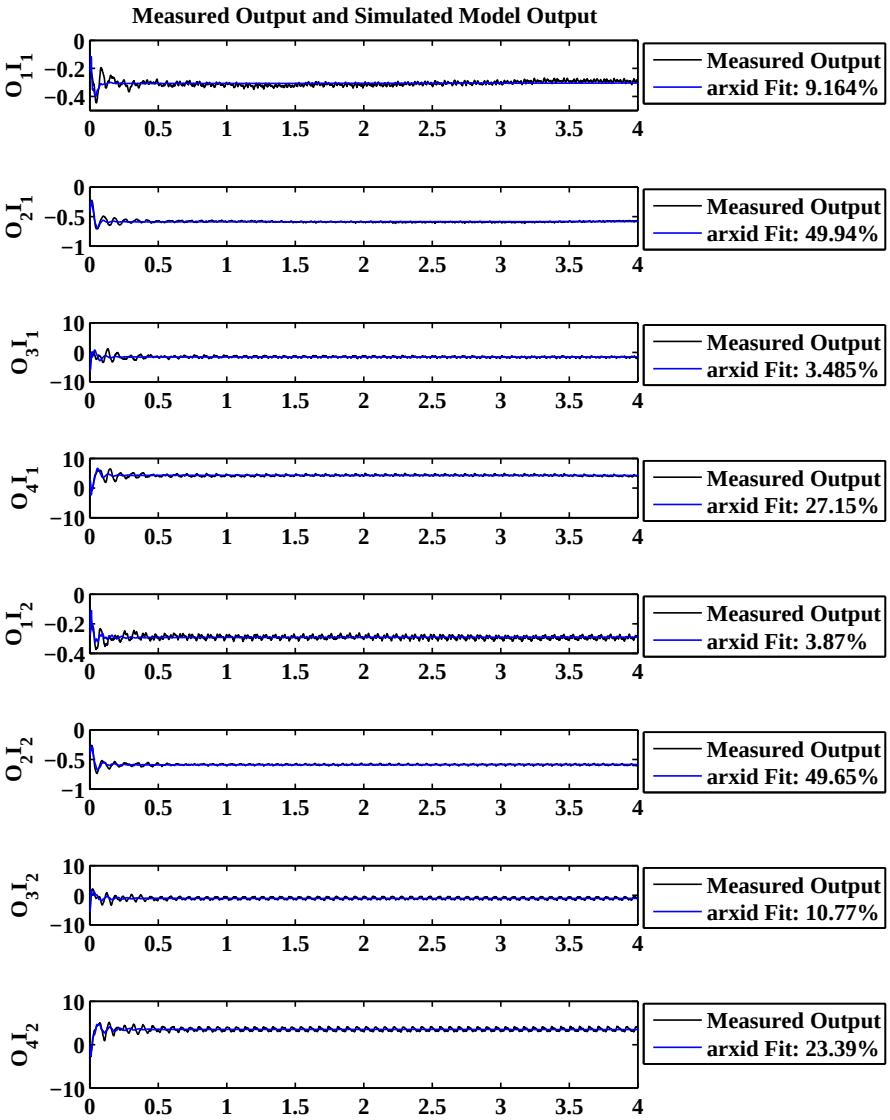

Figure 8.16: Identified responses of the experimental FESS data with ARX model.

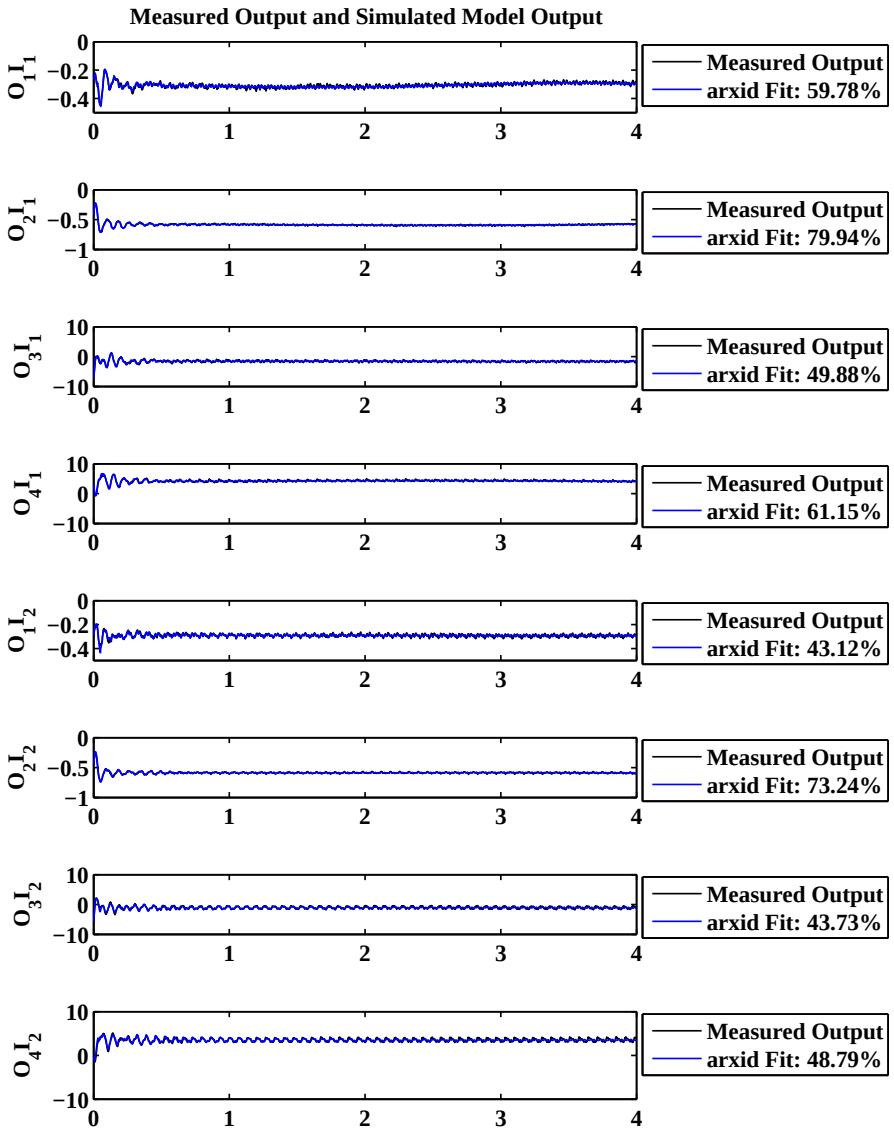

Figure 8.17: Identified responses of the experimental FESS data with higher order ARX model.

Chapter 9

TWO-LINKED-ARM ROBOT SYSTEM

9.1 Introduction

In this chapter, we will use the simulated data of a planar Two-Link-Arm (TLA) robot as another example of identifying the system with complex variables transformation. We apply the new proposed identification technique as derived in Chapter 6 and model selection method as shown in Chapter 7, to the simulated data of the TLA robot. The workspace of the TLA robot is planar, i.e., a x-y plane. The complex variable transformation ($z = x + i \cdot y$) can describe the model of the TLA robot compactly and identify the model. Let us review the development of TLA robot at University of Washington.

9.1.1 Study of TLA Robot

As the robot manipulators have been used in industrial manufacturing, the precise control of these systems is desirable. To achieve these demands, over a decade at University of Washington, the TLA robot has been developed for enhancing many aspects, i.e., the electromechanical system [62], the mathematical modeling [82], [69], stability problems [99], and the pulse width controller [10].

In Vatchara Lertpiriyasuwat's thesis [62], he had shown us the Two-Link-Arm robot as shown Figure 9.1. The direct-drive torque motors, which are the inputs of the system, are used in shoulder and elbow joints. The elbow and end point tip are supported by air cushions, which can make the shoulder and forearm links bend on only in a x-y plane. Therefore, the Two-Link-Arm robot can be configured with flexible links.

Greg Mason *et al* [69] had studied nonlinear friction, backlash, and flexibility of a test bed which is parts of the TLA robot. A complete mathematical model of the test bed was derived in [82]. This model will generate the simulated data for the new identification technique and model selection method.

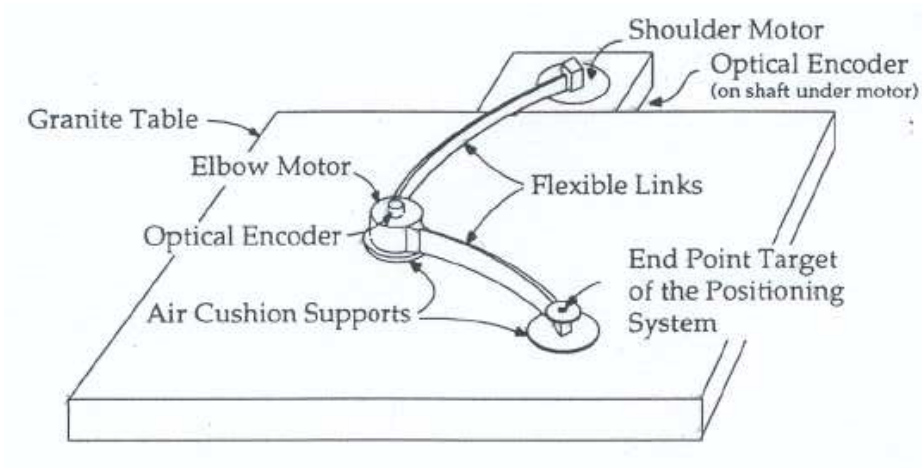

Figure 9.1: The flexible electromechanical system - Two-Link-Arm Robot.

9.2 Background

9.2.1 Basic System

The schematic model of the TLA robot as illustrated Figure 9.2, gives us a mathematical model of the TLA robot dynamics [82]. The following notations are used for describing the equations of motion.

m_{sub} : mass of subscript.

I_{sub} : mass moment of inertia[1] of subscript.

C_{sub} : distance from the shaft of Motor to the mass center of subscript.

O_{sub} : distance from the shaft of Motor to the point of attachment of Link i to subscript.

E_{sub} : Young's module of subscript.

L_i : length of Link i.

[1] All mass moment of inertia are calculated about an axis perpendicular to the plane of motion and passing through the mass center of themselves.

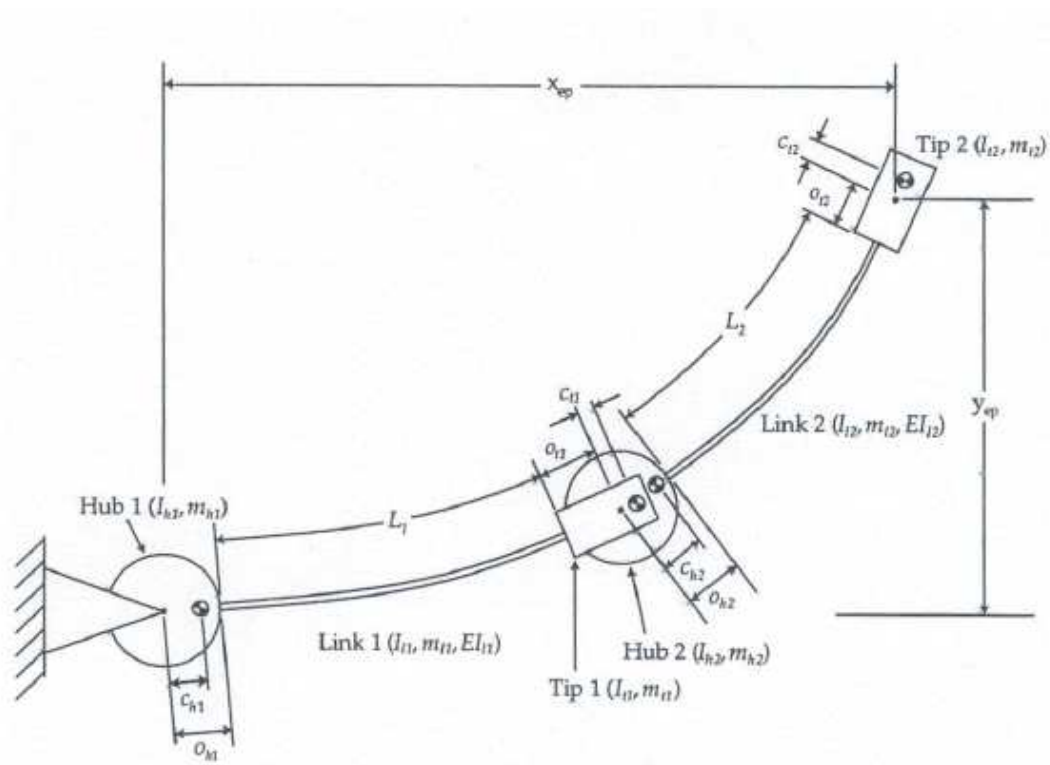

Figure 9.2: Schematic model of the test bed for mathematical model development.

h_i : Hub i.

l_i : Link i.

t_i : Tip i.

In Figure 9.2, θ_1 and θ_2 are the rigid body link rotations, τ_1 and τ_2 are the elbow and the shoulder motor torques, respectively. The positions, x_{ep} and y_{ep}, denote the position of the end point using x-y coordinates system. All numerical values of the TLA robot are in Table 9.1.

Table 9.1: Numerical values of Two-Link-Arm robot parameters

Parameter	$i = 1$	$i = 2$
m_{h_i}	2.2017 kg	0.5514 kg
I_{h_i}	0.0048 kg· m^2	0.0011 kg· m^2
C_{h_i}	0.0027 m	0.0258 m
O_{h_i}	0.0584 m	0.1118 m
L_i	0.4699 m	0.4699 m
m_{l_i}	0.0463 kg	0.0417 kg
I_{l_i}	0.0041 kg· m^2	0.0038 kg· m^2
E_{l_i}	58.7805 N/kg	69.4739 N/kg
m_{t_i}	2.3850 kg	0.4352 kg
I_{t_i}	0.0091 kg· m^2	0.0018 kg· m^2
C_{t_i}	-0.0027 m	-0.0107 m
O_{t_i}	0.1041 m	0.0191 m

The mathematical model had been developed with the assumed modes method [73]. By this approach, the transverse deflection ϖ_i of a point Q_i on Link i like Figure 9.2, at an arc length s_i from the point where Link i attaches to Hub i, at time t, is represented by

$$\varpi_i(s_i, t) = \sum_{j=1}^{\mu} q_j \psi_{ij}(s_i), \tag{9.1}$$

where μ is the number of assumed modes, and ψ_{ij} is the jth mode shape function for Link i. The nonlinear equations of motion can be expressed as

$$\mathbf{M}\ddot{z} + \mathbf{V}(z, \dot{z}) + \mathbf{K}z = T\tau, \tag{9.2}$$

where

$$\begin{aligned} z &= \begin{bmatrix} \theta_1 & \theta_2 & q_1 & \cdots & q_\mu & \theta_f \end{bmatrix}^T, \\ \tau &= \begin{bmatrix} \tau_1 & \tau_2 \end{bmatrix}^T, \\ \psi'_{1i}(L_1) &= \frac{d\psi'_{1i}(s_1)}{ds_1}, \Big|_{s_1=L_1}. \end{aligned}$$

Formulas for the elements of $\mathbf{M}$, $\mathbf{V}$, and $\mathbf{K}$ in terms of parameters of Table 9.1 and the mode shape function are given in [62].

In his thesis (see Chapter 3 and 6), Anawat Pongpunwattana showed a complete derivation for adding the model damping term, $\mathbf{C}$, as follows

$$\mathbf{M}\ddot{z} + \mathbf{V}(z, \dot{z}) + \mathbf{C}\dot{z} + \mathbf{K}z = T\tau. \tag{9.3}$$

9.2.2 Actuator and Sensor Characteristics

The actuator hardware at each joint has a digital-to-analog converter, a power supply/amplifier, and a direct-drive torque motor. Table 9.2 shows us the characteristics of the shoulder and elbow actuator of Two-Link-Arm robot.

Table 9.2: Characteristics of the shoulder and elbow actuator

	Shoulder Actuator	Elbow Actuator
Linear range	± 60 deg (around $\theta_1 = 0$ deg)	± 40 deg (around $\theta_2 = 90$ deg)
Saturation	± 2.603 N·m	± 0.6215 N·m
Quantization	0.001271 N·m	0.00030347 N·m

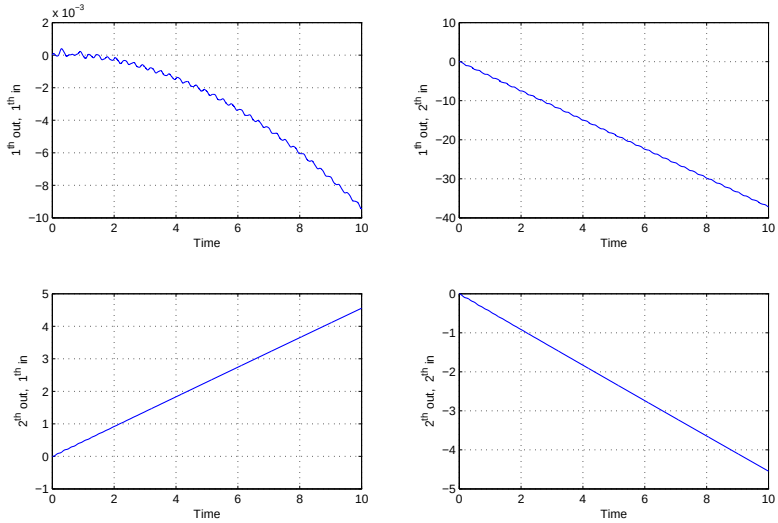

Figure 9.3: The impulse responses of Two-Link-Arm robot with x-y coordinates

9.2.3 Complex Coordinates

The impulse responses of Two-Link-Arm robot - x_{ep} and y_{ep} as shown Figure 9.3 can be expressed by z_{ep} with the complex coordinate. It should be noted that the responses in Figure 9.3 seem unstable because TLA robot has flexible link, i.e, flexible links are vibrating and the positions of end point are moving one direction. For using the complex variable coordinates, x and y axes are considered by the real and imaginary axes respectively. A transformed position, z_{ep}, becomes

$$z_{ep} = x_{ep} + i \cdot y_{ep}. \tag{9.4}$$

As equation (9.4) is used for the complex variables coordinate system, TLA robot can be presented by two inputs (τ_1 and τ_2) and one output (z_{ep}) instead of two inputs (τ_1 and τ_2) and two outputs (x_{ep} and y_{ep}). After we construct the complex variables Hankel matrix using the z_{ep}, the identified state-space model with new identification technique is obtained by the equation (6.55).

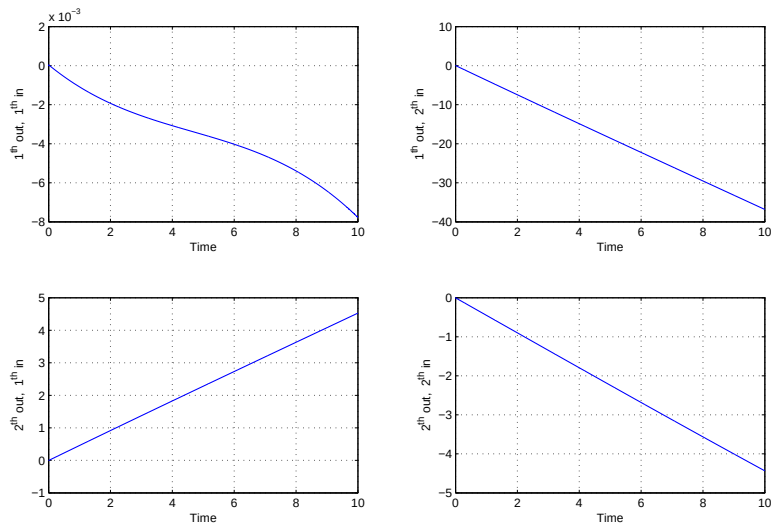

Figure 9.4: The impulse responses of the identified system with 1300×1300 Hankel matrix.

9.3 Result of Applying Complex Variables ERA

In subsection 9.2.3, all brief steps of identifying the TLA robot are explained. After getting the identified state-space model with the equations (6.55), we can obtain the impulse responses of the complex variables identified system as illustrated Figure 9.4. Note that the identified model is obtained with 1300×1300 Hankel matrix, $\theta_1 = 0$, $\theta_2 = 90°$, and 2 millisecond sampling time.

Table 9.3 shows us the error percentages which are used equation (8.7) for comparing the original data and the identified data. As the size of the Hankel matrix is increasing, the error percentages - especially x_{ep} of the shoulder input change irregularly because the end point does not move on x direction very much and there are nonlinearities in the system.

9.4 Applying Model Reduction

The proposed model reduction method for complex variables identified system gives us 2 as the order of the identified system, which means that 4 states are required for describing in the real variable coordinates system. Figure 9.5 shows the magnitude of the identified

Table 9.3: The error percentage table for the identified TLA robot system.

	1st input		2nd input	
	x_{ep} direction	y_{ep} direction	x_{ep} direction	y_{ep} direction
$H_{1100\times1100}$	924.05 %	0.22766 %	0.2829 %	0.078375 %
$H_{1200\times1200}$	166.09 %	0.0046376 %	0.003563 %	0.0017156 %
$H_{1300\times1300}$	8.635 %	0.0023575 %	0.006317 %	0.04808 %

state-space $\bar{\mathbb{A}}_{id}$ matrix. Table 9.4 shows the radii of the Gerschgorin discs and ratios of those radii.

Other model reduction methods give bigger system order than the proposed model reduction method does. The subspace and black-box model reduction methods give us 4 and the singular value of the Hankel matrix gives us 11.

Table 9.4: Ratio of the Gerschgorin disc radius for Two-Link-Arm robot

	Gerschgorin radius $radii_{n-1}$ pre-diagonal	Gerschgorin radius $radii_{n+1}$ post-diagonal	Ratio $\alpha = \frac{radii_{n+1}}{radii_{i-1}}$
$n = 2$	0.0013118	0.0014363	1.0948
$n = 3$	0.00034928	0.042034	120.34
$n = 4$	0.042215	0.04222	1.0001
$n = 5$	0.0007404	0.10039	135.6
$n = 6$	0.10051	0.10051	1.0001
$n = 7$	0.00083141	0.055249	66.452
$n = 8$	0.055417	0.055792	1.0068
$n = 9$	0.0052246	0.022563	4.3186
$n = 10$	0.022498	0.022541	1.0019
$n = 11$	0.0012593	0.0012593	1

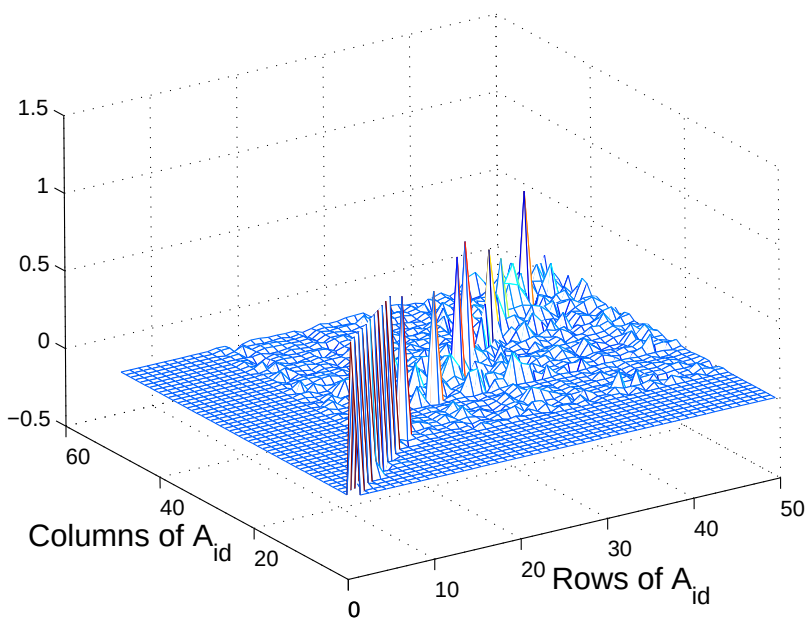

Figure 9.5: Magnitude of the identified state-space A matrix.

Table 9.5: The system order depending on the model reduction methods.

	Real number state		Complex number state	
	Subspace	Black-Box	Singular Value	Proposed
System Order	4	4	11	2

Chapter 10

CONCLUSION

10.1 Conclusion

This dissertation investigated some techniques for identifying linear dynamic systems. Here, we developed a new identification technique using a complex variable transformation, proposed a new model order selection method, and compared our techniques with other methods.

Specific applications included the identification of a flywheel energy storage device. It should be noted that this system contains a skew-symmetric gyroscopic element. Here, we summarize what we presented in this dissertation.

First, several identification methods were introduced and compared, specifically, Eigensystems Realization Algorithm, Fast Eigensystems Realization Algorithm, and Auto-Regressive Models. All identification methods are applied to an example - simple 3 masses, 4 springs and 4 dampers system, and compared each other.

Second, for compact and efficient representation of a gyroscopic systems, this dissertation introduced the complex variables transformation in Chapter 6. The complex variables transformation can be used to convert a non-square plant to a square plant and eliminate the skew-symmetric term in gyroscopic systems. In addition, the new identification technique using the complex variable transformation had been derived and applied to the simulated data, e.g., a simple vibrating shaft model. Before the identification methods were derived and applied to the example, proofs of the controllability matrix, the observability matrix, and responses of initial/forced condition using complex variable transformation were done.

For examples, as shown in Chapter 6, 8, and 9, the new identification technique can describe compactly the models, (e.g., vibrating shaft model, Flywheel Energy Storage System, and Two-Link-Arm robot). The technique has identified the models emphasizing on the time-efficiency. The new technique, in particular for FESS, has shown that the non-

square plant problem is easily solved without additional calculation or manipulation. In addition, all results using the new identification technique, shows us very good results in every example.

Third, new technique for model order determination is developed in Chapter 7. This method has been used in Chapter 8 and 9 in detail and compared with the other model selection methods. The new selection technique gives us very good results. Especially, in the Two-Link-Arm Robot example, the proposed method proved its' superiority.

10.2 Future Research

Even though there were several examples for the complex variables identification, still other examples may be required. Especially, very small scale system, e.g., Piezoactuator, is required. Piezoactuators have been used for noise-and-vibration control and aerospace positioning systems [5], [14]. Because most experimental data are based upon the frequency responses in small scale system, the identification in frequency domain using complex variable transformation will be future research.

The stability problem using the complex variable coordinates will be good topic, e.g., simple stability test of continuous time or discrete time system, Routh-Hurwitz Criterion, Jury's Stability Test, and Second Method of Lyapunov.

BIBLIOGRAPHY

[1] M. Ahrens, L. Kucera, and R. Larsonneur. Performance of a magnetically suspended flywheel energy storage device. *IEEE Transactions on Control System Technology*, 4(5):494–501, September 1996.

[2] A. Akaike. A new look at statistical model identification. *IEEE Transaction on Automatic Control*, AC-19:716–723, 1974.

[3] Kenneth F Alvin. *A Second-Order Structuaral Indentification Procedure via State Space-based System Identification.* Ph.d dissertation, University of Colorado, Boulder, 1993.

[4] I. Andoh, A. Moriyama, and I. Takahashi. Development of a new uninterruptible power supply using flywheel energy strorage tehcniques. *Transactions of the Institute of Electrical Engineers of Japan*, 116-D(11):1153–1158, November 1996.

[5] R. Barrett. Active plate and missile wing development using directionally attached piezoeletric elements. *AIAA Journal*, 32(3):601–609, 1994.

[6] Julius S. Bendat and Allan G. Piersol. *Random Data - Analysis and Measurement Procedures*, chapter Three. Wiley Series in Probability and Statistics, third edition, 2000.

[7] S. G. Braun. *Mechanical Signature Analysis: Theory and Application.* London: Academic Press., 1986.

[8] S. G. Braun, D. J. Ewins, and S. S. Rao. *Encyclopedia of Vibration*, chapter Parameteric Identification of Vibrating Systems, pages 673–685. London: Academic Press., 2001.

[9] William L. Brogan. *Modern Control Theory.* Prentice Hall, third edition, 1991.

[10] Keith W. Buffinon, Alexander D. Perkins, Craig E. Beal, and Martin C. Berg. Evaluation of pulse-width controllers for multi-link, revolute-jointed robotic arms. *Proceedings of the IASTED International Conference on Robotics and Application*, pages 131–139, 2005.

[11] Stephen Butterworth. On the theory of filter amplifiers. *Experimental Wireless and the Radio Engineer*, 7:536–541, 1930.

[12] S. Cerutti, D. Liberati, G. Avanzini, S. Franceschetti, and F. Panzica. Lassification of the electroencephalogram during neurosurgery parametric identification and kalman filtering compared. *Journal-of-Biomedical-Engineering*, 8(3):244–254, 1986.

[13] Z. Chen and C. K. Mechefske. Model order selection: A practical approach. *Mechanical System and Signal Processing*, 15(2):265–273, 2001.

[14] S. Choi, S. Cho, and Y. Park. Vibration and position tracking control of piezoceramic-based smart structures via qft. *ASME Journal of Dynamics Systems, Measurement and Control*, 121:27–33, 1999.

[15] D. Croft and S. Devasia. Vibration compensation for high speed scanning tunneling microscopy. *Scientific Instruments*, 70(12):4600–4605, December 1999.

[16] D. Croft, G. Shed, and S. Devasia. Creep, hysteresis, and vibration compensation for piezoactuators: Automic force microscopy application. *ASME, Journal of Dynamic System, Measurement, and Control*, 123:35–43, March 2001.

[17] D. H. Curtiss, P. P. Mongeau, and R. L. Puterbaugh. Advanced composite fly-wheel sturctual design for a pulsed disk alternator. *IEEE Transactions on Magnetics*, 31(1):26–31, January 1995.

[18] B.N. Datta and D. R. Sarkissian. Computational methods for feedback control in damped gyroscopic second-order systems. *Proceedings of the IEEE Conference on Decision and Control*, 4:4456–4461, 2002.

[19] S. D. Fassois. Mimo lms-armax identification of vibrating structures - part i: The method. *Mechanical Systems and Signal Processing*, 15(4):723–735, 2001.

[20] S. D. Fassois and A. Florakis. Mimo lms-armax identification of vibrating structures - part ii: A critical assessment. *Mechanical Systems and Signal Processing*, 15(4):737–758, 2001.

[21] Thomas E. Fortmann and Konrad L. Hitz. *An Introduction to Linear Control System.* Marcel Dkker, 1977.

[22] Gene F. Franklin, J. David Powell, and Micheal Workman. *Digital Control of Dynamic Systems.* Addison Wesley, 3 edition, 1997.

[23] William A. Gardner. *Introduction to Random Processes with Applications to Signals & Systems.* McGraw-Hill Publishing Company, second edition, 1990.

[24] William A. Gardner. *Introduction to Random Processes with Applications to Signals & Systems*, chapter 2, pages 29–55. McGraw-Hill Publishing Company, second edition, 1990.

[25] William A. Gardner. *Introduction to Random Processes with Applications to Signals & Systems*, chapter 9, pages 200–211. McGraw-Hill Publishing Company, second edition, 1990.

[26] William A. Gardner. *Introduction to Random Processes with Applications to Signals & Systems*, chapter 9, pages 189–200. McGraw-Hill Publishing Company, second edition, 1990.

[27] Giancarlo Genta. *Vibration of structures and machines: practical aspects*. Springer, 3rd edition, 1999.

[28] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Number PP 70-71, and 73. Johns Hopkins University Press, 3rd edition, 1996.

[29] G.H. Golub and C.F. Van Loan. *Matrix Computations*. Number PP 70-71. Johns Hopkins University Press, third edition, 1996.

[30] M.J. Goodwin. *Dynamics of Rotor-Bearing Systems*. Unwin Hyman, London, 1989.

[31] S. Guegercin and A.C. Antoulas. A comparative study of 7 algorithms for model reduction. *Proceedings of the 30th IEEE Conference on Decision and Control, Sydney, Australia*, pages 2367–2372, December 2000.

[32] E. J. Hannan. *Time Series Analysis*. Methuen, London, 1960.

[33] E. J. Hannan. The estimation of arma models. *Annals of Statistics*, 3:975–981, 1975.

[34] E. J. Hannan. The identification and parameterization of armax and state space form. *Econometrica*, 44:713–723, 1976.

[35] E. J. Hannan and B. G. Quinn. The determination of the order of an auto-regression. *Journal of Royal Statistical Society*, 41:190–195, 1979.

[36] M. A. Higgins, D. P. Plant, D. M. Riesand J. A. Kirk, and D. K. Anand. Flywhell energy storage for eletric utility load leveling. *Prodeedings of the 26th Intersociety Energy Conversion Engineering Conference*, 4:209–214.

[37] Nicholas J. Higham. *Accuracy and Stability of Numerical Algorithms*. SIAM, 2 edition, 2002.

[38] B.L. Ho and R.E. Kalman. Effective construction of linear state-variable models form input/output data. *Proceedings of the 3rd Annual Allerton Conference On Circuit and System Theory*, pages 449–459, 1965.

[39] Jiang Hong, Yang Wei-Qin, and Yang You-Tang. State space modeling of random drift rate in high-precision gryo. *IEEE Transactions on aerospace and electronic systems*, 32(3):1138–1143, July 1996.

[40] Yoon Shik Hong, Jong Hyun Lee, and Soo Hyun Kim. A laterally driven symmetric micro-resonator for gyroscopic application. *J. Micromech. Microeng.*, 10:452–458, 2000.

[41] Gary J. Hunter and Michael F. Goodchild. Modeling the uncertainty of slope and aspect estimates derived from spatial databases. *Geographical Analysis*, 29(1):35–49, 1997.

[42] A. H. Jazwinski. *Stochastic Processes and Filtering Theory.* New York: Academic Press, 1970.

[43] Allen John and Murray Alan. Modelling the relationship between peripheral blood pressure and blood volume pulses using linear and neural network system identification techniques. *Physiological-Measurement*, 20(3):287–301, August 1999.

[44] K. Jost. The patriot's hybrid-electric drivetrain. *Automotive Engineering*, pages 30–33, December 1994.

[45] Jer-Jan Juang. Mathematical correlation of modal-parameter-identification methods via system-realization theory. *Journal of Madal Analysis*, Jan 1987.

[46] Jer-Nan Juang. State-space system realization with input-and-output-data correlation. *NASA Technical Paper*, (3622), April 1997.

[47] Jer-Nan Juang and Richard W. Longman. A variance based confidence criterion for era identified modal parameters. *AAS/AIAA Astrodynamics Specialist Conference*, Paper AAs 87-454, 1987.

[48] Jer-Nan Juang and Richard S. Pappa. An eigensystem realization algorithm for modal parameter identification and model reduction. *Journal of Guidance, Control, and Dynamics*, 8(5):620–627, 1985.

[49] Jer-Nan Juang and Richard S. Pappa. An eigensystem realization algorithm for modal parameter identification and model reduction. *Journal of Guidance, Control, and Dynamics*, 8(5):620–627, 1985.

[50] Jer-Nan Juang and Richard S. Pappa. Effect of noise on madal parameter identified by the eigensystem realization algorithm. *Journal of Guidance, Control, and Dynamics*, 9(3):294–303, May-June 1986.

[51] Thomas Kailath, Ali H. Asyed, and Babak Hassibi. *Linear Estimation*. Prentice Hall, 2000.

[52] B. Kartanoglu, R. Soylu, and S. Oral. Mechanical energy storage using flywheel and design optimization. *Energy Storage Systems*, 167:619–648, 1988.

[53] Virginia C. Klema and Alan J. Laub. The singular value decomposition: Its computation and some applications. *IEEE Transaction on Automatic Control*, AC-25(2):164–176, April 1980.

[54] B.D. Kovacevic, M.M. Milosavljevic, and M. Dj. Veinovic. Robust recursive ar speech analysis. *Signal Processing*, 44(2):125–138, June 1995.

[55] Benjamin C. Kuo. *Digital Control System*. Saunders College Publishing, 1992.

[56] Yue Kuen Kwok. *Applied Complex Variables for Scientist and Engineers*. Cambridge University Press, 2002.

[57] C Lanczos. *Applied Analysis*. Prentice Hall, NJ., 1956.

[58] Alberto Leon-Garcia. *Probability and Random Processes for Electrical Engineering*. Addison Wesley Longman, second edition, 1994.

[59] Alberto Leon-Garcia. *Probability and Random Processes for Electrical Engineering*, chapter 4, pages 191–268. Addison Wesley Longman, second edition, 1994.

[60] Alberto Leon-Garcia. *Probability and Random Processes for Electrical Engineering*, chapter 6, pages 329–402. Addison Wesley Longman, second edition, 1994.

[61] Alberto Leon-Garcia. *Probability and Random Processes for Electrical Engineering*, chapter 7, pages 403–421. Addison Wesley Longman, second edition, 1994.

[62] Vatchara Lertpiriyasuwat. Development of mathematical model for the control flexible electromechanical systems test bed. Master's thesis, University of Washington, 1994.

[63] Rainer Leuschke. *Contributions to the Development of a Flywheel Energy Storages System*. PhD thesis, University of Washington, 2004.

[64] Rainer Leuschke and Brian C. Fabien. Disturbance attenuation using a dc motor for radial force actuation in a rotordynamic system. *Journal of Dynamic Systems, Measurement and Control*, V 129(n 6):804–812, November 2007.

[65] Lennart Ljung. *System Identification: Theory for the User*. Prentice Hall, Inc., 1987.

[66] Lennart Ljung. *System Identification Toolbox: for Use with MatLab*. Math Works, Inc, 2002.

[67] Lennart Ljung and T. Glad. *Modeling of Dynamics Systems*. Prentice Hall, 1994.

[68] Lonnie C. Ludeman. *Random Processes - Filtering, Estimation, and Detection*. Wiley, first edition, 2003.

[69] Greg Mason, Anawat Pongpunwattana, and Martin Berg. Design and modeling of a flexible test bed for use in control system analysis and verification. *Mechatronics*, (12):891–904, 2002.

[70] Robert T. M'Closkey, Steve Gibso, and Jason Hui. Input-output dynamics of the jpl microgyroscope. *Proceedings on the 37th IEEE Conference on Decision and Control - Tampa, Florida*, pages 4328–4333, 1998.

[71] Robert T. M'Closkey, Steve Gibson, and Jason Hui. Input-output dynamics of the jpl microgyroscope. *Proceedings on the 37th IEEE conference on decision and control*, pages 4328–4333, December 1998.

[72] Robert T. M'Closkey, Steve Gibson, and Jason Hui. System identification of a mems gyrosope. *Journal of dynamic systems, measurement, and control*, 123:201–210, JUNE 2001.

[73] Leonard Meirovitch. *Principles and Techniques of Vibrations*. Prentice Hall, 1997.

[74] Nirode Mohanty. *Random Signals Estimation and Identifiction Analysis and Applications*. Von Nostrand Reinhold Company Inc., first edition, 1986.

[75] Nirode Mohanty. *Signal Processing - Signals, Filtering, and Detection*. Von Nostrand Reinhold Company Inc., first edition, 1987.

[76] Wayne Nagata and N. Sri Namachchivaya. Bifurcations in gyroscopic systems with an application to rotating shafts. *Proc. R. Soc. Lond.*, 454:543–585, 1998.

[77] Gora C. Nandy and Xu Yangsheng. Dynamic model of a gyrosocpic wheel. *Proceeding of the 1998 IEEE International Conference on Robotics and Automation*, pages 2683–2688, May 1998.

[78] Frank Peeters, Rik Pintelon, Johan Schoukens, and Yves Rolain. Identification of rotor-bearing systems in the frequency domain part ii: Estimation of modal parameters. *Mechanical Systeme and Signal Processing*, 15(4):775–788, 2001.

[79] Frank Peeters, Rik Pintelon, Johan Schoukens, Yves Rolain, Enrique S. Gutierrez, and Patrick Guillaume. Identification of rotor-bearing systems in the frequency domain part i: Estimation of frequency response functions. *Mechanical Systeme and Signal Processing*, 15(4):759–773, 2001.

[80] Lee D. Peterson. Efficient computation of the eigensystem realization algorithm. *Journal of Guidance, Control, and Dynamics*, 18(3):395–403, May-June 1995.

[81] K. A. Petsounis and S.D. Fassois. Parametric time-domain methods for the identification of vibrating structures-a critical comparison and assessment. *Mechanical Systems and Signal Procsessing*, 15(6):1031–1060, 2001.

[82] Anawat Pongpunwattana. Modelling and control of a single-link flexible robotic arm under the influence of mechanical friction. Master's thesis, University of Washington, 1999.

[83] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brain P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing.* Cambridge University Press, 2 edition, 1992.

[84] Q.S. Ren and A.J. Willis. A maximum entropy approach to filtering and reconstructive imaging of the underwater environment. *Conference of Oceans*, 9-12 Octorber 1995.

[85] J. Rissanen. Modeling by shortest data description. *Automatica*, 14:465–471, 1983.

[86] Jorma Rissanen. Mdl donoising. *IEEE Trans. Information Theory*, 46(7):2537–2543.

[87] Sang Man Seong, Jang Gyu Lee, and Park Chan Gook. Equivalent arma model representation for rlg random errors. *IEEE transactions on aerospace and electronic systems*, 36(1):286–290, January 2000.

[88] K. Sam Shanmugan and A. M. Breipohl. *Random Signals - Detection, Estimation, and Data Analysis.* Wiley, frist edition, 1988.

[89] Yi-Jr Shen. Optimal control for the flywheel energy storage system with a hybrid magnetic bearing. Master's thesis, University of Washington, 1997.

[90] Robert E. Skelton. *Dynamic Systems Control Linear Systems Analysis and Synthesis.* John Wiley & Sons, 1988.

[91] J. David Stienmier. *Contributions to the Development of a Flywheel Energy Storages System.* PhD thesis, University of Washington, 1997.

[92] R. A. Stine. Model selection using information theory and the mdl principle. *Sociological Methods & Research*, 33:230–260, 2004.

[93] Gilbert Strang. *Linear Algebra and Its Applications*. Harcourt Brace & Company, 3rd edition, 1988.

[94] Söderström T. and P. Stoica. *System Identification*. Prentice Hall, 1989.

[95] S. Thielman and Brain C. Fabien. Experimantal control of a nonlinear electromagnetic suspension via feedback linearization. *1995 American Control Conference*, 5:3818–3819, 1995.

[96] Lloyd N. Trefethen and David Bau III. *Numerical Linear Algebra*. SIAM, 1997.

[97] J.A. Wickert. Transient vibration of gyroscopic systems with unsteady superposed motion. *Journal of Sound and Vibration*, 195(5):797–807, 1996.

[98] R. Wiesendanger. *Scanning Probe Microscopy and Spectroscopy*. Cambridge University Press, Cambridge UK, 1994.

[99] Gergory A. Wood. Analysis of stability problems of consistently linearized models of flexible systems. Master's thesis, University of Washington, 2001.

[100] Hsien-Tsai Wu, Jar-Ferr Yang, and Fwu-Kuen Chen. Source number estimators using transformed gerschgorin radii. *IEEE Transactions on Signal Processing*, 43(6):1325–1333, June 1995.

[101] Ciann-Dong Yang and Fang-Bo Yeh. Identification, reduction, and refinement of model parameters by the eigensystem realization algorithm. *Journal of Guidance*, 13(6):1051–1059, Nov-Dec 1990.

Appendix A

PROOFS

In this chapter, we provide fundamental proofs which are used in this dissertation.

A.1 Observability and Controllability Matrices for Continuous State-space Model

In Chapter 6, we proved the use of the observability matrix and the controllability matrix which include the complex variables for discrete time state-space model. Here, in complex variable domain, we will prove the observability matrix and the controllability matrix for the continuous time state-space model.

A.1.1 Observability

Definition 1 (Observable) *A particular state $\bar{x}$ of the system is said to be unobservable if for $u(t) \equiv 0$ and any $T > 0$, the initial state $\bar{x}(0) = \bar{x}$ produces an output response $y(t) = 0$.*

If the output $y(t)$ vanishes, a state $\bar{x}$ is also said to be unreconstructible. In this case, observability and reconstructibility are completely same.

Theorem 1.1 (Observability Gramian) *A state $\bar{x}$ which is the complex variables vector is unobservable if and only if it belongs to the null space $\mathcal{N}_{ull}(W_{\mathcal{O}}(0,T))$, where*

$$W_{\mathcal{O}}(0,T) \triangleq \int_0^T e^{\bar{\mathbf{A}}^H \tau} \bar{\mathbf{C}}^H \bar{\mathbf{C}} e^{\bar{\mathbf{A}}\tau} d\tau \tag{A.1}$$

is a symmetric $N \times N$ matrix known as the observability gramian. Consequently, the system is completely observable if and only if $W_{\mathcal{O}}(0,T)$ is nonsingular.

Proof - Observability Gramian

Assume that $\bar{x} \in \mathcal{N}_{ull}(W_{\mathcal{O}})$. This assumption means that

$$\begin{aligned} W_{\mathcal{O}}\bar{x} &= 0, \\ \bar{x}^H W_{\mathcal{O}}\bar{x} &= 0 = \int_0^T \bar{x}^H e^{\bar{\mathbf{A}}\tau}\bar{\mathbf{C}}^H\bar{\mathbf{C}}e^{\bar{\mathbf{A}}\tau}\bar{x}d\tau \\ &= \int_0^T \|\bar{\mathbf{C}}e^{\bar{\mathbf{A}}\tau}\bar{x}\|^2 d\tau. \end{aligned} \tag{A.2}$$

Recall the response of the complex variable state-space model, when $u(t) = 0$ and $\bar{x}(0) = \bar{x}$, then

$$\begin{aligned} y(\tau) &= \bar{\mathbf{C}}e^{\bar{\mathbf{A}}\tau}\bar{x}, \qquad (\tau > 0) \\ &= 0. \end{aligned} \tag{A.3}$$

The equation (A.2) makes the above equation zero so that $\bar{x}$ is unobservable.

Conversely, if $\bar{x}$ is unobservable, then we can use the definition of the observability [1]. Therefore,

$$y(\tau) = 0 = \bar{\mathbf{C}}e^{\bar{\mathbf{A}}\tau}\bar{x} \qquad (\tau > 0). \tag{A.4}$$

If the equation (A.4) is satisfied, then

$$W_{\mathcal{O}}\bar{x} = \int_0^T e^{\bar{\mathbf{A}}^H\tau}\bar{\mathbf{C}}^H \underbrace{\bar{\mathbf{C}}e^{\bar{\mathbf{A}}\tau}\bar{x}}_{=0} d\tau = 0. \tag{A.5}$$

Therefore, the equation (A.5) shows that $\bar{x}$ is obviously the null space of the Observability Gramian - $W_{\mathcal{O}}$.

With equations (A.3) and (A.5), finally, we can conclude that the system is completely observable if and only if $\mathcal{N}_{ull}(W_{\mathcal{O}}) = 0$, i.e. $W_{\mathcal{O}}$ is nonsingular.

Theorem 1.2 (Observability Matrix) *If we define the $N_{out} \times N$ observability matrix for the complex variables state-space system to be*

$$P_{\mathcal{O}} = \begin{bmatrix} \bar{\mathbf{C}} \\ \bar{\mathbf{C}}\bar{\mathbf{A}} \\ \bar{\mathbf{C}}\bar{\mathbf{A}}^2 \\ \vdots \\ \bar{\mathbf{C}}\bar{\mathbf{A}}^{N-1} \end{bmatrix}, \tag{A.6}$$

then the subspace of unobservable states and the subspace of observable states are

$$\begin{aligned} \mathcal{N}_{ull}(W_{\mathcal{O}}(0,T)) &= \mathcal{N}_{ull}(P_{\mathcal{O}}), & \text{(A.7)} \\ \mathcal{R}_{ange}(W_{\mathcal{O}}(0,T)) &= \mathcal{R}_{ange}(P_{\mathcal{O}}) \\ &= \mathcal{R}_{ange}\left[\begin{array}{cccc} \bar{\mathbf{C}}^H & \bar{\mathbf{A}}^H\bar{\mathbf{C}}^H & \cdots & (\bar{\mathbf{A}}^H)^{N-1}\bar{\mathbf{C}}^H, \end{array}\right] & \text{(A.8)} \end{aligned}$$

for any $T > 0$. Thus the system is completely observable if and only if the observability matrix has full rank.

Proof - Observability Matrix

Let us prove the equation (A.7) for verifying the theorem of the observability matrix. Assume $\mathcal{N}_{ull}(W_{\mathcal{O}}) \subseteq \mathcal{N}_{ull}(P_{\mathcal{O}})$. If $\bar{z} \in \mathcal{N}_{ull}(W_{\mathcal{O}})$, then $W_{\mathcal{O}}\bar{z} = 0$ and

$$\begin{aligned} \bar{z}^H W_{\mathcal{O}}\bar{z} &= 0 \\ &= \int_0^T \bar{z}^H e^{\bar{\mathbf{A}}^H\tau}\bar{\mathbf{C}}^H\bar{\mathbf{C}}e^{\bar{\mathbf{A}}\tau}\bar{z}d\tau = \int_0^T \|\bar{\mathbf{C}}e^{\bar{\mathbf{A}}\tau}\bar{z}\|^2 d\tau. & \text{(A.9)} \end{aligned}$$

Since the equation (A.9) is continuous, it must be identically zero, i.e.

$$\begin{aligned} \bar{\mathbf{C}}e^{\bar{\mathbf{A}}\tau}\bar{z} &= 0 \\ &= \bar{\mathbf{C}}[\mathbf{I} + \bar{\mathbf{A}}\tau + \frac{\bar{\mathbf{A}}\tau^2}{2!} + \cdots]\bar{z}. & \text{(A.10)} \end{aligned}$$

The each term of the power series (A.10) has to be zero because the power series is identically zero on a finite interval.

$$\bar{\mathbf{C}}\bar{z} = \bar{\mathbf{C}}\bar{\mathbf{A}}\bar{z} = \bar{\mathbf{C}}\bar{\mathbf{A}}^2\bar{z} = \cdots = 0. \qquad \text{(A.11)}$$

The equation (A.11) can be formulated by matrix formation as follows

$$\begin{bmatrix} \bar{\mathbf{C}}\bar{z} \\ \bar{\mathbf{C}}\bar{\mathbf{A}}\bar{z} \\ \bar{\mathbf{C}}\bar{\mathbf{A}}^2\bar{z} \\ \vdots \\ \bar{\mathbf{C}}\bar{\mathbf{A}}^{N-1}\bar{z} \end{bmatrix} = 0 = \begin{bmatrix} \bar{\mathbf{C}} \\ \bar{\mathbf{C}}\bar{\mathbf{A}} \\ \bar{\mathbf{C}}\bar{\mathbf{A}}^2 \\ \vdots \\ \bar{\mathbf{C}}\bar{\mathbf{A}}^{N-1} \end{bmatrix} \bar{z} = P_{\mathcal{O}}\bar{z}. \qquad \text{(A.12)}$$

The fact that the multiplication $P_{\mathcal{O}}$ and $\bar{z}$ is zero means that $\bar{z}$ must be in the null space of the observability matrix $P_{\mathcal{O}}$.

Conversely, we shall establish that $\mathcal{N}_{ull}(W_{\mathcal{O}}) \supseteq \mathcal{N}_{ull}(P_{\mathcal{O}})$. If $\bar{z}$ is in the null space of the observability matrix $P_{\mathcal{O}}$, then the equations (A.11) and (A.12) must be hold. Now, the another expression of exponential function (A.25) is applied $\bar{\mathbf{C}}e^{\bar{A}\tau}\bar{z}$

$$\begin{aligned} \bar{\mathbf{C}}e^{\bar{\mathbf{A}}\tau}\bar{z} &= \bar{\mathbf{C}}\left[\sum_{i=0}^{n-1}\bar{\mathbf{A}}^i\alpha_i(\tau)\right]\bar{z} \\ &= \sum_{i=0}^{n-1}\underbrace{\left(\bar{\mathbf{C}}\bar{\mathbf{A}}^i\bar{z}\right)}_{=0,\because(A.11)}\alpha_i(\tau) = 0. \end{aligned} \tag{A.13}$$

Finally, let us plug the equation (A.13) into the $W_{\mathcal{O}}\bar{z}$ as,

$$W_{\mathcal{O}}\bar{z} = \int_0^T e^{\bar{\mathbf{A}}^H\tau}\bar{\mathbf{C}}^H \underbrace{\bar{\mathbf{C}}e^{\bar{\mathbf{A}}\tau}\bar{z}}_{=0,\because(A.13)} d\tau = 0. \tag{A.14}$$

Therefore, the equations (A.12) and (A.14) verify $\mathcal{N}_{ull}(W_{\mathcal{O}}) \supseteq \mathcal{N}_{ull}(P_{\mathcal{O}})$, and $\mathcal{N}_{ull}(W_{\mathcal{O}}) \subseteq \mathcal{N}_{ull}(P_{\mathcal{O}})$. These two statements means that the null space of the observability gramian is equal to the null space of the null space of the observability matrix. Finally, we can use the observability matrix instead of the observability gramian. Let us move on to the definition of the controllability.

Controllability

Definition 2 (Controllable) *A particular state $\bar{x}$ of the system is said to be controllable if for every $T > 0$, there exists a control function $u(t)$ which drives the system from the initial state $\bar{x}(0) = \bar{x}$ to $\bar{x}(T) = 0$.*

If the definition of the controllability is true, the input $u(t)$ is not unique, i.e. other controls will also drive the state to zero via different trajectories. A state $\bar{x}$ is also said to be '*reachable*' if a control exists which drives the system from $\bar{x}(0) = 0$ to $\bar{x}(T) = \bar{x}$. In this case, the controllability and reachability are completely same.

Theorem 2.1 (Controllability Gramian) *A state $\bar{x}$ which is the complex number vector is controllable if and only if it belongs to the range space $\mathcal{R}_{ange}(W_{\mathcal{C}}(0,T))$ where*

$$W_{\mathcal{C}}(0,T) \triangleq \int_0^T e^{-\bar{\mathbf{A}}\tau}\bar{\mathbf{B}}\,\bar{\mathbf{B}}^H e^{-\bar{\mathbf{A}}^H\tau}d\tau \tag{A.15}$$

is a symmetric $N \times N$ matrix known as the controllability gramian. Consequently, the system is completely controllable if and only if $W_{\mathcal{C}}(0,T)$ is nonsingular.

Proof - Controllability Gramian

Assume $\bar{x} \in \mathcal{R}_{ange}\{W_{\mathcal{C}}\}$. Then the equation

$$W_{\mathcal{C}}\bar{z} = \bar{x}$$

has a solution $\bar{z} \in \mathcal{C}_{omplex}{}^{n}$. And according to the theorem 2.1, if we choose the control input like

$$u(t) = -\bar{\mathbf{B}}^H e^{-\bar{\mathbf{A}}^H t}\bar{z} \tag{A.16}$$

in the forced response formula (6.18) and (A.16), we find that

$$\begin{aligned} \bar{x}(T) &= e^{\bar{\mathbf{A}}(T)}\bar{x} + \int_0^T e^{\bar{\mathbf{A}}(T-\tau)}\bar{\mathbf{B}}u(\tau)d\tau \\ &= e^{\bar{\mathbf{A}}(T)}\bar{x} - e^{\bar{\mathbf{A}}(T)}\int_0^T e^{-\bar{\mathbf{A}}\tau}\bar{\mathbf{B}}\bar{\mathbf{B}}^H e^{-\bar{\mathbf{A}}^H\tau}\bar{z}d\tau \\ &= e^{\bar{\mathbf{A}}(T)}\bar{x} - e^{\bar{\mathbf{A}}(T)}W_{\mathcal{C}}\bar{z} = 0. \end{aligned} \tag{A.17}$$

Therefore, suppose a control input $u(t)$ exists which can drives $\bar{x} \neq 0$ to 0.

Theorem 2.2 (Controllability Matrix) *If we define the $N \times N_{in}$ controllability matrix for the system to be*

$$P_{\mathcal{C}} \triangleq \left[\bar{\mathbf{B}}, \bar{\mathbf{A}}\bar{\mathbf{B}}, \bar{\mathbf{A}}^2\bar{\mathbf{B}}, \cdots, \bar{\mathbf{A}}^{n-1}\bar{\mathbf{B}}\right], \tag{A.18}$$

then the subspace of controllable states is

$$\mathcal{R}_{ange}(W_{\mathcal{C}}(0,T)) = \mathcal{R}_{ange}(P_{\mathcal{C}}),$$

and the subspace of uncontrollable state is

$$\mathcal{N}_{ull}(W_{\mathcal{C}}(0,T)) = \mathcal{N}_{ull}(P_{\mathcal{C}}^H) = \mathcal{N}_{ull}\begin{bmatrix} \bar{\mathbf{B}}^H \\ \bar{\mathbf{B}}^H \ \ \bar{\mathbf{A}}^H \\ \vdots \\ \bar{\mathbf{B}}^H \ \ \bar{\mathbf{A}}^{T(n-1)} \end{bmatrix}, \tag{A.19}$$

for any $T > 0$. Thus the system is completely controllable if and only if the controllability matrix has full rank, that is,

$$\text{rank}(P_{\mathcal{C}}) = n. \tag{A.20}$$

Proof - Controllability Matrix

First, let us assume that $\mathcal{N}_{ull}(W_{\mathcal{C}}) \subseteq \mathcal{N}_{ull}(P_{\mathcal{C}}^H)$. If $\bar{z} \in \mathcal{N}_{ull}(W_{\mathcal{C}})$, then $W_{\mathcal{C}}\bar{z} = 0$ and

$$\bar{z}^H W_{\mathcal{C}} \bar{z} = \int_0^T (\bar{z}^H e^{-\bar{\mathbf{A}}\tau}\bar{\mathbf{B}})(\bar{\mathbf{B}}^H e^{-\bar{\mathbf{A}}^H\tau}\bar{z})d\tau = \int_0^T \|(\bar{\mathbf{B}}^H e^{-\bar{\mathbf{A}}^H\tau}\bar{z})\|^2 d\tau = 0. \tag{A.21}$$

Because the integral equation (A.21) is continuous and zero, the $(\bar{\mathbf{B}}^H e^{-\bar{A}^H\tau}\bar{z})$ must be zero. Also, we can expand the exponential function $e^{-\bar{\mathbf{A}}^H\tau}$ as follows

$$\bar{\mathbf{B}}^H e^{-\bar{\mathbf{A}}^H\tau}\bar{z} = \bar{\mathbf{B}}^H \left(\mathbf{I} - \bar{\mathbf{A}}^H\tau + \frac{(\bar{\mathbf{A}}^H)^2\tau^2}{2!} - \frac{(\bar{\mathbf{A}}^H)^3\tau^3}{3!} + \cdots \right) \bar{z} = 0. \tag{A.22}$$

Since the equation (A.22) having the power series is zero, each term of power series must be zero as

$$\bar{\mathbf{B}}^H\bar{z} = \bar{\mathbf{B}}^H\bar{\mathbf{A}}^H\bar{z} = \bar{\mathbf{B}}^H(\bar{\mathbf{A}}^H)^2\bar{z} = \bar{\mathbf{B}}^H(\bar{\mathbf{A}}^H)^3\bar{z} = \cdots = 0. \tag{A.23}$$

The equation (A.23) can be formed

$$P_{\mathcal{C}}^H\bar{z} = \begin{bmatrix} \bar{\mathbf{B}}^H\bar{z} \\ \bar{\mathbf{B}}^H \ \ \bar{\mathbf{A}}^H\bar{z} \\ \vdots \\ \bar{\mathbf{B}}^H \ \ \bar{\mathbf{A}}^{H(n-1)}\bar{z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{A.24}$$

Therefore, $\bar{z} \in \mathcal{N}_{ull}(P_{\mathcal{C}}^H)$.

Conversely, let us assume that $\mathcal{N}_{ull}(W_{\mathcal{C}}) \supseteq \mathcal{N}_{ull}(P_{\mathcal{C}}^H)$. If $\bar{z} \in \mathcal{N}_{ull}(P_{\mathcal{C}}^H)$, then the equation (A.23) can be valid. Then recall the expression of exponential function and with the equation (A.25), we have

$$e^{\bar{\mathbf{A}}t} = \sum_{j=0}^{n-1} \bar{\mathbf{A}}^j \alpha_j(t). \tag{A.25}$$

$$\begin{aligned} u(t) = \bar{\mathbf{B}}^H e^{-\bar{\mathbf{A}}^H\tau}\bar{z} &= \bar{\mathbf{B}}^H \left[\sum_{j=0}^{n-1} (\bar{\mathbf{A}}^H)^j \alpha_j(-\tau) \right] \bar{z} \\ &= \sum_{j=0}^{n-1} \underbrace{\left[\bar{\mathbf{B}}^H(\bar{\mathbf{A}}^H)^j\bar{z}\right]}_{=0, \because (A.23)} \alpha_j(-\tau) = 0, \end{aligned}$$

from which it follows that

$$W_{\mathcal{C}}\bar{z} = \int_0^T e^{-\bar{\mathbf{A}}\tau}\bar{\mathbf{B}}(\bar{\mathbf{B}}^H e^{-\bar{\mathbf{A}}^H\tau}\bar{z})d\tau = 0, \tag{A.26}$$

and we can conclude that $\bar{z} \in \mathcal{N}_{ull}(W_{\mathcal{C}})$.

Finally, having shown that each of spaces $\mathcal{N}_{ull}(W_{\mathcal{C}})$ and $\mathcal{N}_{ull}(P_{\mathcal{C}}^H)$ is contained in the other, we can conclude that they must be identical.

A.2 Gerschgorin Disks Theorem

This section prove the Gerschgorin disk which is used for model order determination in Chapter 7. *Proof* - Gerschgorin discs

Let $\bar{\lambda}$ be an eigenvalue of $\bar{H}$ and $\bar{x}$ its corresponding eigenvector. Choose i such that $|\bar{x}_i| = max_j\,|\bar{x}_j|$. Since $\bar{x}$ can't be 0, $|\bar{x}_i| > 0$. Now $\bar{H}\bar{x} = \bar{\lambda}\bar{x}$, or looking at the i-th component as

$$(\bar{\lambda} - \bar{h}_{ii})\bar{x}_i = \sum_{j\neq i} \bar{h}_{ij}\bar{x}_j. \tag{A.27}$$

Dividing equation (A.27) by $\bar{x}_i$, gives us

$$(\bar{\lambda} - \bar{h}_{ii}) = \sum_{j\neq i} \bar{h}_{ij}\frac{\bar{x}_j}{\bar{x}_i}. \tag{A.28}$$

Taking the norm on both sides of equation (A.28) and the given condition, $|\bar{x}_i| = max_j\,|\bar{x}_j|$, give us

$$\left|\bar{\lambda} - \bar{h}_{ii}\right| = \left|\sum_{j\neq i} \frac{\bar{h}_{ij}\bar{x}_j}{\bar{x}_i}\right| \leq \sum_{j\neq i} \left|\bar{h}_{ij}\right|. \tag{A.29}$$

In equation (A.29), the left side of the equation is the center of the Gerschgorin disk and the right side of the equation is the radius of the Gerschgorin disk.

Appendix B

M-FILES

B.1 Main Program

B.1.1 3 masses 4 springs and 4 damping System

```
% ERA with NOISE
% Last Modified Jan 24. 2005
% ERA for 3 DOF example ; Original identified response NO REUCTION!!!
% With 90 by 90 matrix!!!
% figure 1 -> 0 ~ 12 secs (Unstable)
% figure 2 -> 0 ~ 1.5 secs (Well Tracking)

close all;
clear all;

% Given Basic Matrix
K=[4 -2 0
    -2 4 -2
    0 -2 22];                       % Stiffness Matrix
D=[0.2 -0.1 0
    -0.1 0.2 -0.1
    0 -0.1 1.1];                   % Damping Matrix
M=[1 0 0
    0 1 0
    0 0 1];                          % Mass Matrix

% State-Space model for example - NO NOISE
```

```
Ao=[zeros(3,3), eye(3,3)
    -K, -D];
Bo=[zeros(3,3)
    eye(3,3)];
Co=[eye(3,3), zeros(3,3)];
Do=zeros(3,3);

% Basic parameter
Ts = 0.05;                                  % Sampling time
Tfinal=12;                                  % Final time
Twant=1.5;                                % Until good tracking time
T=0:Ts:Tfinal;                            % Time Duration
TT = 0:Ts:Twant;                        % for 90 by 90 Hankel matrix
SYSo = ss(Ao,Bo,Co,Do);         % Continuous System

% Response for Impulse input
[Yo,to,Xo] = impulse(SYSo,T);

yo(:,1,1)= awgn(Yo(:,1,1),10,'measured');
yo(:,1,2)= awgn(Yo(:,1,2),10,'measured');
yo(:,1,3)= awgn(Yo(:,1,3),10,'measured');
yo(:,2,1)= awgn(Yo(:,2,1),10,'measured');
yo(:,2,2)= awgn(Yo(:,2,2),10,'measured');
yo(:,2,3)= awgn(Yo(:,2,3),10,'measured');
yo(:,3,1)= awgn(Yo(:,3,1),10,'measured');
yo(:,3,2)= awgn(Yo(:,3,2),10,'measured');
yo(:,3,3)= awgn(Yo(:,3,3),10,'measured');

%=========================================%
%==================ERA===================%
```

```
%=========================================%
disp('===========')
disp('|       ERA       |')
disp('===========')
% Make the Hankel Matrix
no=90;                                  % Size fo Hankel matrix
bas=3;                                   % No. of Outputs
H0 = zeros(no,no);                 % Initialize
H1 = zeros(no,no);

for i = 1:no/3
    for j = 1:no/3
        for k = 1:3
            for l=1:3
                temp(k,l) = yo(i+j-1,k,l);
                %temp(k,l) = yo(i+j,k,l);
            end
        end
        H0( ((i-1)*bas+1):(i*bas),((j-1)*bas+1):(j*bas)) = temp;
    end
end

for i = 1:no/3
    for j = 1:no/3
        for k = 1:3
            for l=1:3
                temp(k,l) = yo(i+j,k,l);
                %temp(k,l) = yo(i+j+1,k,l);
            end
        end
```

```
        H1( ((i-1)*bas+1):(i*bas),((j-1)*bas+1):(j*bas)) = temp;
    end
end

% Singular Value Decomposition of Hankel Matrix
[U,S,V]=svd(H0);
E=[eye(3,3), zeros(3,no-3)]';
E1=[eye(3,3)
    zeros(no-3,3)];

% Identified Matrix using ERA
Aie=S^(-0.5)*U'*H1*V*S^(-0.5);
Bie=S^(0.5)*V'*E1;
Cie=E'*U*S^(0.5);
Die=zeros(3,3);

% no X no A
SYSe = ss(Aie,Bie,Cie,Die,Ts);          % Continuous System
SYSe = d2c(SYSe);

% Response for Identified system (ERA)
[ye,te,Xe] = impulse(SYSe,T);

% First 6 by 6 matrix of A for Comparison
A_sixsix=S(1:6,1:6)^(-0.5)*U(:,1:6)'*H1*V(:,1:6)*S(1:6,1:6)^(-0.5);

xmin=0;
xmax=Tfinal;
ymin=-0.5;
ymax=0.5;
```

```
% Plotting
figure
subplot(331)
plot(T,Ts*ye(:,1,1),'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:2:12])
grid on
subplot(332)
plot(T,Ts*ye(:,1,2),'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:2:12])
grid on
subplot(333)
plot(T,Ts*ye(:,1,3),'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:2:12])
grid on
subplot(334)
plot(T,Ts*ye(:,2,1),'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:2:12])
```

```
grid on
ylabel('Displacement (m)', 'FontSize', 12, 'FontWeight', 'Demi')
subplot(335)
plot(T,Ts*ye(:,2,2),'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:2:12])
grid on
subplot(336)
plot(T,Ts*ye(:,2,3), 'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:2:12])
grid on
subplot(337)
plot(T,Ts*ye(:,1,3),'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:2:12])
grid on
subplot(338)
plot(T,Ts*ye(:,2,3),'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:2:12])
grid on
```

```
xlabel('Time (sec)','FontSize', 12, 'FontWeight', 'Demi')
subplot(339)
plot(T,Ts*ye(:,3,3),'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:2:12])
grid on

xmax=Twant;
figure
subplot(331)
plot(TT,Ts*ye(1:xmax/Ts+1,1,1),'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:.3:1.5])
grid on
subplot(332)
plot(TT,Ts*ye(1:xmax/Ts+1,1,2),'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:.3:1.5])
grid on
subplot(333)
plot(TT,Ts*ye(1:xmax/Ts+1,1,3),'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
```

```
set(gca,'XTick',[0:.3:1.5])
grid on
subplot(334)
plot(TT,Ts*ye(1:xmax/Ts+1,2,1),'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:.3:1.5])
grid on
ylabel('Displacement (m)', 'FontSize', 12, 'FontWeight', 'Demi')
subplot(335)
plot(TT,Ts*ye(1:xmax/Ts+1,2,2),'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:.3:1.5])
grid on
subplot(336)
plot(TT,Ts*ye(1:xmax/Ts+1,2,3), 'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:.3:1.5])
grid on
subplot(337)
plot(TT,Ts*ye(1:xmax/Ts+1,1,3),'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:.3:1.5])
```

```
grid on
subplot(338)
plot(TT,Ts*ye(1:xmax/Ts+1,2,3),'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:.3:1.5])
grid on
xlabel('Time (sec)','FontSize', 12, 'FontWeight', 'Demi')
subplot(339)
plot(TT,Ts*ye(1:xmax/Ts+1,3,3),'LineWidth',1)
set(gca, 'FontWeight', 'Demi')
axis([xmin xmax ymin ymax])
set(gca,'YTick',[-0.5:0.25:0.5])
set(gca,'XTick',[0:.3:1.5])
grid on
```

B.1.2 Simplest Gimbal Gyroscopic System

```
% Final Version : May 8 2007 10:00PM
% This example will be example parts of dissertation.
clear all;
close all;

disp(' ')
disp('**************************************************')
disp('**    The Identification Example                                **')
disp('**                     with Complex Variables Coordinates    **')
disp('**                                                    - Gimbal       **')
disp('************* ***********************************')
```

```
% Parameters
I1 = 1;
I3 = 1;
b = 0.75;                                                   % Damping Coefficient
omega = 100;                                              % Spinning Speed

% Mass Matrix
m = [I1 0
    0 I1];
% Damping and Gyroscopic Matrix
d = [b, I3*omega
    -I3*omega, b];
% Stiffness Matrix
k = zeros(2,2);
% Input Angle
theta=45*pi/180;

% State-Space Model
A = [ -inv(m)*d -inv(m)*k
    eye(2,2) zeros(2,2)];
B = [ cos(theta)  cos(theta+90*pi/180)
    sin(theta) sin(theta+90*pi/180)
    zeros(2,2)];
C = [zeros(2,2), eye(2,2)];
D = zeros(2,2);

% Parameters for getting responses of Identified System
P = 500;                                       % Magnitude of Impulse
Ts = 0.002;                                   % Sampling Time
```

```
T = 0:Ts:10;                                % Time Interval

% Responses and Graphics
sys = ss(A,B,C,D);
[y,t,x] = impulse(sys,T);
Jgraphic(y,t)

% Create the Complex Variable Responses
for ii = 1:3000
    z(ii,1) = y(ii,1,1)+i*y(ii,2,1);
    z(ii,2) = y(ii,1,2)+i*y(ii,2,2);
end

% Complex Variable Coordinate ERA Identification Method
size = 10;
[H0,H1] = response_hankel(z,size);
[Aie, Bie,Cie,Die] =era_normal(H0,H1,2,1,Ts);

% Model Reduction - System Order
order = id_sys_order(Aie)
svd_order = svd_sys_order(H0)

% Gershgorin Disks Drawing
gershgorindisk(Aie)
chosen = order;

% Impulse responses of complex variables state-space model.
[yyy,ttt] = complex_impulse(Aie,Bie,Cie,Die, t, chosen);

for k=1:length(t)
```

```
    yid(k,1,1) = real(yyy(k,1,1));
    yid(k,2,1) = -imag(yyy(k,1,1));
    yid(k,1,2) = imag(yyy(k,1,2));
    yid(k,2,2) = real(yyy(k,1,2));
end


% Check the error percentage of the Identified System.
errorper = error_table(y,yid,Ts);

% Graphs of the identified system.
Jgraphic(yid,t)
% Graphs of the Errors between original and identified responses
Jgraphic(y-yid,t)

u(1,1) = P;
u(2:5001,1)= zeros(5000,1);
data = iddata([yid(:,:,1) yid(:,:,2)],[u u],Ts);
Sub_order1 = n4sid(data,'best')
Sub_order2 = pem(data,'best')
% Last Line
```

B.1.3 Flywheel Energy Storage System

```
% This FESS system identification method will be parts of the dissertation.
% Therefore, I have to protect this file.

% Last Version May 11 2007
% Made by Jae Hyeuk Suk
```

```
clear all
close all

disp(' ')
disp('**************************************************')
disp('**    The Identification Example                                  **')
disp('**                  with Complex Variables Coordinates       **')
disp('**                       - Fly Wheel Energy Storage System   **')
disp('************* ***********************************')

% Basic Parameters
Ts = 0.002 ;                                        % Sampling Time
T = 0:Ts:10;                                        % Time Interval
size=150;                                           % Size of Hankel
omega = 1500;                                     % Spinning Speed
theta =45*pi/180;                                 % impulse angle

% Get the Mass, Gyroscopic, Stiffness matrices of FESS
[M,G,K,E,D5,L] = fly_model(omega);     %[x,y, theta_x, theta_y]

% State-Space Model
A = [ zeros(4,4) eye(4,4);
    -inv(M)*K, -inv(M)*G];
B = [0  0
0   0
0   0
0   0
0.47755*cos(theta)  0.47755*cos(theta+90*pi/180)
0.47755*sin(theta)  0.47755*sin(theta+90*pi/180)
-8.2885*sin(theta)  -8.2885*sin(theta+90*pi/180)
```

```
8.2885*cos(theta)   8.2885*cos(theta+90*pi/180)];
C = [eye(4,4), zeros(4,4)];
D = zeros(4,2);
syso=ss(A,B,C,D);

% Getting Impulse Responses FESS
[yo,to,xo] = impulse(syso,T);
Jgraphic(yo,to)

% Create the
for ii=1:length(T)
    z(ii,1,1)=yo(ii,1,1)+yo(ii,2,1)*i;           % z=x+i*y
    z(ii,2,1) = yo(ii,4,1)-yo(ii,3,1)*i;          % theta_z = theta_y-i*theta_x
    z(ii,1,2)=yo(ii,1,2)+yo(ii,2,2)*i;           % z=x+i*y
    z(ii,2,2) = yo(ii,4,2)-yo(ii,3,2)*i;          % theta_z = theta_y-i*theta_x
end

% Eigensystem Realization Algorithm Identification Method
% with the Complex Variables Coordinates
[H0,H1] = response_hankel(z,size);
[Aie, Bie,Cie,Die] =era_normal(H0,H1,2,2,Ts);

% Model Reduction
svd_order = svd_sys_order(H0);
order = id_sys_order(Aie);
gershgorindisk(Aie)

% % To get the impulse responses of identified system
chosen=order;
[yyy,ttt] = complex_impulse(Aie,Bie,Cie,Die,T, chosen);
```

```
for iii=1:1:length(T)
    yid(iii,1,1) = -real(yyy(iii,1,1));
    yid(iii,2,1) = imag(yyy(iii,1,1));
    yid(iii,3,1) = imag(yyy(iii,2,1));
    yid(iii,4,1) = real(yyy(iii,2,1));
    yid(iii,1,2) = -real(yyy(iii,1,2));
    yid(iii,2,2) = imag(yyy(iii,1,2));
    yid(iii,3,2) = imag(yyy(iii,2,2));
    yid(iii,4,2) = real(yyy(iii,2,2));
end

errorper1 = error_table(yo, yid,Ts);

Jgraphic(yid,T')
Jgraphic(yo-yid,T')

u(1,1) = 500;
u(2:5001,1)= zeros(5000,1);
data = iddata([yid(:,:,1) yid(:,:,2)],[u u],Ts);
Sub_order1 = n4sid(data,'best')
Sub_order2 = pem(data,'best')

% Last Line
```

B.1.4 Two-Link-Arm Robot

```
% ME 582
% Loads linear state model of small-perturbation TLA dynamics
```

```
% Version: 5May00
close all
clear all


disp(' ')
disp('****************************************************')
disp('** The System Identification                                    **')
disp('**           with Complex Variables Coordinates                 **')
disp('**                                    - 2 by 2 Two Link Arm Robot   **')
disp('****************************************************')

% Basic Parameters
theta1=0;
theta2=90;
theta10 = theta1*pi/180;
theta20 = theta2*pi/180;
span=(0.25:0.25:1.00)';

% State-Space Model
[Ac,Bc,Cc,Dc]=tla_setup(theta10,theta20,span);
plantc=ss(Ac,Bc,Cc,Dc);
T = 0:0.002:10;
Ts = 0.002;
[y,t,x] = impulse(plantc,T);

%%%%%%%
%
% xep           :   end point of x direction
% yep           :   end point of y direction
```

```
% xhp           :   hub point of x direction        :   need to get
% yhp           :   hub point of y direction        :   need to get
% theta1       :   angle of 1st hub
% theta2       :   angle of 2nd hub
% w14          :    deflection of 4th span of 1st arm
% w24          :    deflection of 4th span of 2nd arm
%
% xhp           :   =

xep1=y(:,13,1);
yep1=y(:,14,1);
% for iii=1:length(xep1)
%     xhp1(iii,1)=y(iii,18,1)*cos(y(iii,1,1)) + y(iii,22,1)*cos(y(iii,1,1)+y(iii,2,1));
%     yhp1(iii,1)=y(iii,18,1)*cos(y(iii,1,1)) + y(iii,22,1)*sin(y(iii,1,1)+y(iii,2,1));
% end

xep2=y(:,13,2);
yep2=y(:,14,2);

for ii = 1:length(xep1)
    zep(ii,1,1) = xep1(ii)+i*yep1(ii);
    zep(ii,1,2) = xep2(ii)+i*yep2(ii);
end
yo=y(:,13:14,1:2);
Jgraphic(yo,t);

size =1300;
[H0,H1] = response_hankel(zep,size);
[Aie, Bie,Cie,Die] =era_normal(H0,H1,2,1,Ts);
```

```
% newA = triu(Aie,1)+tril(Aie,-1);
% radius(2,2) = norm(newA(2,1:2),2);
% radius(2,3) = norm(newA(2,1:3),2);
% radius(3,3) = norm(newA(3,1:3),2);
% r=3;
% ratio = radius(2,3) - 10*radius(2,2);
% while ratio < 0
%      r=r+1;
%      for j=r-1:r
%         radius(j,r) = norm(newA(j,1:r),2);
%      end
%      ratio = radius(r-1,r) - 500*radius(r-1,r-1);
% end

% break
% Model Reduction
svd_order = svd_sys_order(H0);
order = id_sys_order(Aie);
gershgorindisk(Aie)

% % To get the impulse responses of identified system
chosen=svd_order;
[yyy,ttt] = complex_impulse(Aie,Bie,Cie,Die,T, chosen);


% Only one input therefore, we don't need to multiply (-i).
%We need to convert again. So, 1st output will be imaginary part of id.

for iii=1:1:length(T)
    yid(iii,1,1) = imag(yyy(iii,1,1));
```

```
    yid(iii,2,1) = -real(yyy(iii,1,1));
%    yid(iii,3,1) = imag(yyy(iii,2,1));
 %   yid(iii,4,1) = real(yyy(iii,2,1));
     yid(iii,1,2) = -imag(yyy(iii,1,2));
     yid(iii,2,2) = -real(yyy(iii,1,2));
% %    yid(iii,3,2) = imag(yyy(iii,2,2));
  %  yid(iii,4,2) = real(yyy(iii,2,2));
end



errorper1 = error_table(yo, yid,Ts);


Jgraphic(yid,T')
Jgraphic(yo-yid,T')
```

B.2 Subroutine Functions

B.2.1 ERA

Construction of Hankel Matrix

```
function [H0 , H1] = reponse_hankel(response, hsize)
% [H0, H1] = response_hankel(response, size_hankel)
% makes hankel matrix (H(0) and H(1)) of MIMO system.
%
% If Y (response of system) has NI inputs and NO outputs
% and your desired size of hankel matris is SH, then
% Hankel matrix H(0) and H(1) can be calculated.
%
% Basically, Y (response) has 3 dimensional matrix
% like Y(:, NO, NI).
% Need to modify for using any response....
```

```
% Modified Version Jan. 06 2006 (from jhankel)
% Made by Jae Hyeuk Suk

[no_response, no_output, no_input] = size(response);

H = zeros(hsize,hsize+no_input);

if no_input==1
    for i = 1:(hsize)/no_output
        for j = 1:hsize+no_input
            for k = 1:no_output
                temp(k,1) = response(i+j-1,k);
            end
            H( ((i-1)*no_output+1):(i*no_output),...
                ((j-1)*no_input+1):(j*no_input)) = temp;
        end
    end
else
    for i = 1:hsize/no_output
        for j = 1:(hsize+no_input)/no_input
            for k = 1:no_output
                for l=1:no_input
                    temp(k,l) = response(i+j-1,k,l);
                end
            end
            H( ((i-1)*no_output+1):(i*no_output),...
                ((j-1)*no_input+1):(j*no_input)) = temp;
        end
    end
```

```
end
H0=H(1:hsize,1:hsize);
H1=H(1:hsize,1+no_input:hsize+no_input);
```

ERA

```
function [Ai,Bi,Ci,Di]= era_normal(H0,H1,in, out,Ts)
% [Ai,Bi,Ci,Di]= era_normal(H0,H1,in, out,Ts) will give
% us the identified state-space model with Eigensystem
% Realization Algorithm(ERA) identification method.
%
% For getting the identified state-space model, you
% need the Hankel matrix H(0), H(1), number of inputs,
% number of outputs, and sampling time of data.
%

% Version Jan. 09 2008
% Made by Jae Hyeuk Suk

[no, no]=size(H0);
[U,S,V]=svd(H0);

E=[eye(out,out), zeros(out,no-out)];
E1=[eye(in,in)
    zeros(no-in,in)];

% Identified Matrix using ERA
Ai=S^(-0.5)*U'*H1*V*S^(-0.5);
Bi=S^(0.5)*V'*E1;
Ci=E*U*S^(0.5);
```

```
Di=zeros(out,in);
```

B.2.2 After ERA

Response of Complex Variables State-space Model

```
function [y,t] = complex_impulse(A,B,C,D, t, chosen)
% This function gives us the impulse responses of
% complex variable state-space model. Basically, this
% function is created for complex variables Eigensystem
% Realization Algorithm Identification method.
% Therefore, you need to understand all processes of
% complex variables ERA identification method to use
% this function well.

% Version May 11 2008
% Made by Jae Hyeuk Suk

Ts = t(2)-t(1); P = 1/Ts;
[out, in] = size(D);

for jj = 1:1:length(t)
    for kk = 1:in
        yidcom(jj+1,:,kk) = A(1:chosen, 1:chosen)^(jj)*...
            B(1:chosen,kk)*P*(-i);
    end
end
for jj = 1:1:length(t)
    for kk = 1:in
        yiidcom(jj,:,kk) = C(:,1:chosen)*yidcom(jj,:,kk)';
```

```
    end
end
y = yiidcom*Ts;
% Last Line
```

Graphics

```
function Jgraphic(response,x)
% This function will give us the graphs of multi-input
% multi-output system. To use this function, you must
% have x axis date and responses whose size must
% be matching with the x axis data.

% First Version April 27 2008
% Made by Jae Hyeuk Suk

[r,out,in] = size(response);
[no,d] = size(x);
tot=1;

if no ==r
    figure
    if in==1
        for oo=1:out
            subplot(out,1,oo)
            plot(x,response(:,oo))
            xlabel('Time')
            ylabel([num2str(oo),'^{th} out'])
            grid on
        end
```

```
    else
        for ooo=1:out
            for ii=1:in
                subplot(out,in,tot)
                plot(x,response(:,ooo,ii))
                xlabel('Time')
                ylabel([num2str(ooo),'^{th} out,  ',...
                    num2str(ii),'^{th} in'])
                grid on
                tot=tot+1;
            end
        end
    end
else
end

clear r in ooo tot out oo
% Last Line
```

B.2.3 Model Reduction

Model Reduction Method - Singular Value of the Hankel matrix

```
function order = svd_sys_order(H0)
% order = svd_sys_order(H0) select the order of system
% with the Hankel matrix (H(0)).
%
% This method uses ratios of the signular values.
% If the ratio is very very big, the order can be selected.
%
```

```
% Version May 9 2008
% Last Verstion May 9 2008
% Made by Jae Hyeuk Suk

[s,v,d] = svd(H0);
h_size = length(H0);
for ii=1:h_size-1
    vratio(ii) =  v(ii,ii)/v(ii+1,ii+1);
end

orderP = find(vratio > 100);
order = min(orderP);
% Last Line
```

Proposed Model Reduction Method

```
function order = id_sys_order(Aid)
% order = id_sys_order(Aid) select the order of identified
% system with the identified state-space A matrix which
% must be square.
%
% The ERA identified state-space A matrix of very
% noisy system have very large system order.
% This system order reduction model consider the
% 'Gershgorin' disk theorem which is related to the
% estimation of eigenvalue location.
%

% First Version Sept. 23 2004
```

```
% Last Verstion April 27 2007
% Made by Jae Hyeuk Suk

[m,n] = size(Aid);
radius = zeros(m,m);

newA = triu(Aid,1)+tril(Aid,-1);
radius(2,2) = norm(newA(2,1:2),2);
radius(2,3) = norm(newA(2,1:3),2);
radius(3,3) = norm(newA(3,1:3),2);
r=3;
ratio = radius(2,3) - 10*radius(2,2);

while ratio < 0
     r=r+1;
     for j=r-1:r
        radius(j,r) = norm(newA(j,1:r),2);
     end
     ratio = radius(r-1,r) - 10*radius(r-1,r-1);
end
order=r-2;

% Last Line
```

Gershgorin Discs

```
function gershgorindisk(A)
% gershgorindisk(A) will draw the graphs of the discs
% of Gershgorin theorems which can be related to the
% model reduction method. You can observe  changes
```

```
% of disks' size then you can select the model order
% with the change.

% Version May 3 2008
% Made by Jae Hyeuk Suk

% get rid of diagonal elements
newA = triu(A,1)+tril(A,-1);
% get the reduced order of identified system
order = id_sys_order(A);

figure
radius1 = norm(newA(order+1,1:order+1),2);
cx1=real(A(order+1,order+1));
cy1=imag(A(order+1,order+1));
radius2 = norm(newA(order+1,1:order+2),2);
cx2=real(A(order+1,order+1));
cy2=imag(A(order+1,order+1));
mycircle(cx1, cy1, radius1);
mycircle(cx2, cy2, radius2);
title(['D_{(',num2str(order+1),',' num2str(order+2),...
    ')}/ D_{(',num2str(order+1),',' num2str(order+1),')} = ',...
    num2str(radius2/radius1)])
grid on
xlabel('Real Axis')
ylabel('Imaginary Axis')

% Last Line
```

VITA

Jae Hyeuk Suk was born in Seoul, Korea on March 30, 1973. He finished his bachelor degree of Mechanical Engineering at Kyung-Hee University in Korea. After he served a military service for his country, he decided to study abroad. He moved to Seattle, Washington to attend the University of Washington. He obtained a Master of Science in Mechanical Engineering with research of symbolic computational algorithm in 2001. He then continued his Ph.D. in Mechanical Engineering under Prof. Brian Fabien and received a Ph.D. in Mechanical Engineering in 2008.