PARAMETER ESTIMATION FOR LINEAR DYNAMICAL SYSTEMS WITH APPLICATIONS TO EXPERIMENTAL MODAL ANALYSIS

A Thesis Submitted to
the Graduate School of Engineering and Sciences of
İzmir Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE

in Electrical and Electronics Engineering

by
İlker Tanyer

August 2008
İZMİR

We approve the thesis of **İlker Tanyer**

Date of Signature

. . . . . **15 July 2008**
Assist. Prof. Dr. Serdar ÖZEN
Supervisor
Department of Electrical and Electronics Engineering
İzmir Institute of Technology

. . . . . **15 July 2008**
Assist. Prof. Dr. Cemalettin DÖNMEZ
Department of Civil Engineering
İzmir Institute of Technology

. . . . . **15 July 2008**
Assoc. Prof. Dr. Bilge KARAÇALI
Department of Electrical and Electronics Engineering
İzmir Institute of Technology

. . . . . **15 July 2008**
Assist. Prof. Dr. Mustafa Aziz ALTINKAYA
Department of Electrical and Electronics Engineering
İzmir Institute of Technology

. . . . . **15 July 2008**
Assoc. Prof. Dr. Serhan ÖZDEMİR
Department of Electrical and Electronics Engineering
İzmir Institute of Technology

. . . . . **15 July 2008**

Prof. Dr. F.Acar SAVACI
Head of Department
Department of Electrical and Electronics Engineering
İzmir Institute of Technology

. . . . . .
Prof. Dr. Hasan BÖKE
Head of the Graduate School

ACKNOWLEDGEMENTS

First of all, i would like to thank my supervisors, Assist. Prof. Dr. Serdar ÖZEN and Assist. Prof. Dr. Cemalettin DÖNMEZ for spending their time to help me in this project.

I thank Assist. Prof. Dr. Cemalettin DÖNMEZ and R.A. Eyyub KARAKAN for experimental measurements, giving knowledge about vibration theory and analytical solutions.

Also I would like to thank Assist. Prof. Dr. Mustafa Aziz ALTINKAYA for helping me about pre-processing and simulations of ERA.

In addition, I would like to thank Assist. Prof. Dr. Serdar ÖZEN, Assist. Prof. Dr. Cemalettin DÖNMEZ, Assoc. Prof. Dr. Bilge KARAÇALI, Assist. Prof.Dr. Mustafa Aziz ALTINKAYA and Assoc.Prof.Dr. Serhan ÖZDEMİR for giving me corrections and suggestions which helped me a lot in revising the thesis.

This project is funded by TÜBİTAK with project number 104I107, therefore i would like to thank TÜBİTAK for their support.

Finally I would like to thank my family, İlkay, Köksal-Devlet TANYER for their help and support. This project is also dedicated to my family as my other projects.

ABSTRACT

PARAMETER ESTIMATION FOR LINEAR DYNAMICAL SYSTEMS WITH APPLICATIONS TO EXPERIMENTAL MODAL ANALYSIS

In this study the fundamentals of structural dynamics and system identification have been studied. Then some fundamental parameter estimation algorithms in the literature are provided. These algorithms will be applied to an experimental and an artificial system to extract their structural properties. Consequently, the main objective of this study is constructing the mathematical model of a structure by using only the measurement data.

To process measurement data, three fundamental modal analysis algorithms are examined. Least-Squares Complex Exponential(LSCE), Eigensystem Realization Algorithm(ERA) and Polyreference Frequency Domain(PFD) algorithms are implemented in MATLAB environment. We applied these algorithms to artificial and experimental data, then we compared the performance of these algorithms. State estimation for linear dynamical systems have also been studied, and details of the Kalman filter as a state estimator are provided. Kalman filter as a state estimator has been integrated with the ERA algorithm and the performance of the Kalman-ERA is provided.

ÖZET

DOĞRUSAL DİNAMİK SİSTEMLER İÇİN DEĞİŞKEN KESTİRİMİ VE DENEYSEL MODAL ANALİZ UYGULAMALARI

Bu çalışmada yapı dinamiği ve sistem tanılamanın temelleri gözden geçirilecektir. Ardından literatürdeki bazı temel parametre kestirim algoritmaları anlatılacaktır. Bu algoritmalar yapay ve deneysel sistemlere uygulanacak ve bu sistemlerin yapısal özellikleri bulunmaya çalışılacaktır. Kısaca çalışmadaki asıl amaç sadece yapıdan alınan ölçümleri kullanarak yapının matematiksel modelini oluşturmaktır.

Ölçüm verilerini işleyebilmek için üç temel Modal Analiz algoritması incelenmiştir. En Küçük Kareler Karmaşık Üstel Metodu, Özsistem Gerçekleştirme Algoritması ve Çoklu Referans Frekans Bölgesi algoritmaları MATLAB ortamında gerçeklenmiştir. Bu algoritmalar yapay ve deneysel verilere uygulanmış ve algoritmaların performansları karşılaştırılmıştır. Ayrıca Kalman süzgeci bir durum kestiricisi olarak kullanılarak doğrusal dinamik sistemler için durum kestirimi işlemi incelenmiş ve daha doğru bir parametre kestirimi yapmak adına Kalman süzgeci Özsistem Gerçekleştirme Algoritması ile birleştirilmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

FRF	Frequency response function
IRF	Impulse response function
LTI	Linear time invariant
EM	Expectation-maximization
FFT	Fast Fourier transform
LSCE	Least squares complex exponential
PFD	Polyreference frequency domain
SDOF	Single degree of freedom
MDOF	Multi degree of freedom
MIMO	Multiple-input multiple-output
ML	Maximum-likelihood
MMSE	Minimum mean squared error
ERA	Eigensystem realization algorithm
MLE	Maximum likelihood estimation
SNR	Signal to noise ratio

CHAPTER 1

INTRODUCTION

1.1. Overview

Mechanical , aeronautical or civil structures need to be lighter, stronger and more flexible because of the demands of safety and reliability. Furthermore predicting the response of a structure to an excitation is so critical. Because of these facts, vibration analysis of structures become a popular subject for engineers. Making experiments on a structure, constructing a mathematical model, controlling the structure or designing strong and stabilized structures are the main studies of vibration analysis. As an example, the vibration properties of Transamerica Building in Figure 1.1 were determined by forced vibration tests. The fundamental natural vibration periods of this 60-story steel building are 2.90 sec for north-south vibration and also for east-west vibration (Chopra 1995). By using these vibration tests, the health of a building can be judged or the response of a building to an earthquake can be predicted. Before making experiments on a complex

Figure 1.1. Transamerica Pyramid Building in San Francisco

structure as Transamerica Building, learning the mathematical background of vibrational analysis and applying them to an experimental setup is necessary.

In this study, civil engineers and electronics engineers worked together to investigate the vibrational characteristics of an experimental system. The experimental setup is

shown in Figure 1.2 . There is a steel beam which is connected to an accelerometer. The impact hammer is used to create an impulse on the beam and the acceleration response is measured by using the accelerometer on the beam. The data acquisition device is used for transferring the acceleration data from accelerometer to the computer. Then measurement data can be processed to find the characteristics of the system. At the end of the experiment, we have the acceleration response of the beam and the excitation data . We will use this data to identify the system and construct a mathematical model of the beam.

Figure 1.2. Experimental Setup

Constructing mathematical model of a real structure is not an easy procedure. Mathematical model of a system includes important system characteristics by sets of differential equations. These differential equations can be written easily by applying physical laws to the system. Mathematical modelling procedure is a 3-step process. First step is drawing a schematic diagram of the system and defining the variables as in Figure 1.3. Second step is writing the differential equations for each independent component by using physical laws and constructing the model. The last step is verifying the model by comparing with experimental results. The last step is for performance prediction of the system. Experiment is the only way to verify the mathematical model. If the experimental results

are different from the prediction, then a modification must be done for the mathematical model. The process must be repeated until a satisfactory agreement is obtained between experimental results and prediction.

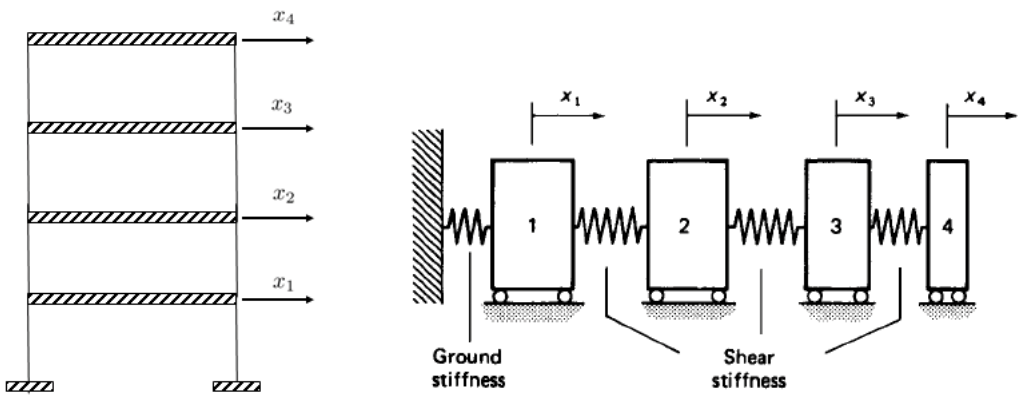


Figure 1.3. Schematic Diagram for a 4 story frame

Mathematical modelling of dynamical systems and analyzing the dynamical behavior of the structures are the subjects of *System Dynamics*. A system is called dynamical if its present output depends on past input. If the system's current output depends only on the current input, system is called as static. As said before, dynamical systems can be modelled by differential equations. Classifying differential equations is important for the classification of the system. According to this classification, the analysis for modelling can be changed. For example if a system has nonlinear differential equations, its mathematical model will be a nonlinear model and nonlinear analysis must be done for this system. Mathematically linearity means that the relationship between the input and the output of the system satisfies the superposition property. According to superposition property, if the input to the system is the sum of two component signals as

$$x(t) = ax_1(t) + bx_2(t) \,, \tag{1.1}$$

then the output of the system will be,

$$y(t) = ay_1(t) + by_2(t) \,, \tag{1.2}$$

where $y_k(t)$ for $k = 1, 2$, are the output signals resulting from the input signals $x_k(t)$ for $k = 1, 2$, and the coefficients a, b are complex valued scalars.

Time invariance is another important system property for dynamical system analysis. Time invariance means that if the input is affected by a time delay, the output should be affected by the same time delay. Consider a linear system with impulse response $h(t)$ where the input-output relationship is given by

$$y(t) = x(t) * h(t) \,, \tag{1.3}$$

where $x(t)$ is the input, $y(t)$ is the output and $*$ is the convolution operator. If a time shift in the input signal results in an identical time shift in the output signal, as

$$y(t - t_0) = x(t - t_0) * h(t) \,, \tag{1.4}$$

then $h(t)$ is said to be a linear time-invariant system. If a system satisfies both the linearity and the time invariance properties, this system is called Linear Time Invariant (LTI) system.

In this study we will deal with the mathematical models of LTI dynamical systems and we will try to find the characteristics of the system by using *System Identification*, *Structural Dynamics* and *Modal Analysis* concepts.

System identification is the process of building dynamical models from measured data by using several mathematical tools and algorithms. In this study, some system identification techniques will be investigated and these techniques will be verified by MATLAB simulations.

Structural dynamics is the analytical study of the structures which covers the behavior of structures when subjected to dynamical loading. Buildings, bridges, satellites, aircrafts, vehicles can be considered as examples of structures. Dynamical loading can be wind, wave, earthquake, blasts. Dynamical loading means that the load on the structure changes with time.

Since the problem of mathematical modelling is so complex, engineers generally use *Finite Element Analysis* as a design tool to solve this complexity. Finite element analysis is a computer modelling approach based on numerical analysis. Modal analysis is an important part of dynamical finite element analysis and it is a powerful tool for civil engineers. The main objective of modal analysis can be defined as determining, improving and optimizing the dynamical characteristics of structures by utilizing modal analysis and system identification procedures. Modal analysis is a 3-step process. First the dynamical properties of systems are investigated under vibrational excitation. Then the

dynamical characteristics of the system are determined in the form of natural frequencies and damping ratios. Consequently, these modal parameters can be used to formulate a mathematical model.

Modal analysis is based upon the fact that the vibration response of a LTI dynamical system can be expressed as the linear combination of a set of harmonic motions called the natural modes of vibration. Actually it is a complicated waveform which can be represented as a combination of sine and cosine waves. Natural modes of vibration are special characteristics to dynamical systems and they can be determined by its physical properties where the physical properties are *mass*, *stiffness*, and *damping*. Natural modes can be described in terms of *natural frequency*, *modal damping factor* and *mode shape*. These are also called *modal parameters*.

In the last 20 years, data acquisition systems and processing capabilities of computers have developed rapidly. This technological development makes the experimental technique of modal analysis useful. The experimental part of modal analysis is called *modal testing*. Modal testing is an experimental procedure which is used to find the dynamical characteristics of the structure. In practice measuring the impulse response of a structure is first accomplished and then the Frequency Response Function (FRF) and other desired parameters, such as natural frequency and damping ratio, of the system are obtained.

In this study, fundamentals of system dynamics will be used to define the problem of mathematical modelling, and then modal analysis techniques will be used to identify the parameters of the dynamical system and estimating the model.

1.2. Background

Vibration is a motion that repeats itself. It can be regarded as the transfer between the kinetic energy and potential energy, so it can be said that there must be storing and releasing energy in a vibratory system. This type of motion can be modelled easily by a mass and a spring. Figure 1.4 shows a typical vibratory system. In this system mass is responsible for kinetic energy and the spring is responsible for potential energy. There are several types of vibration. The list of vibration types are shown in Table 1.1 . Our study will begin from the free vibration of a system. Systems could be discretised to inertia, spring and damper elements to define the vibrational models. In Figure 1.5, the

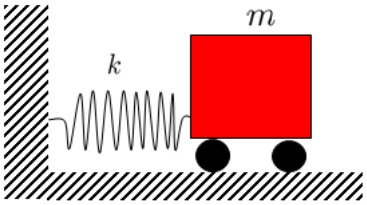


Figure 1.4. A Vibratory System

Table 1.1. Vibration Types

Reference Terms	Vibration Type	Description
External Excitation	Free Vibration	Vibration induced by initial input(s) only.
	Forced Vibration	Vibration subjected to one or more continuous external inputs.
Presence of Damping	Undamped Vibration	Vibration with no energy loss or dissipation.
	Damped Vibration	Vibration with energy loss.
Linearity of Vibration	Linear Vibration	Vibration for which superposition principle holds.
	Nonlinear Vibration	Vibration that violates superposition principle.
Predictability	Deterministic Vibration	The value of vibration is known at any given time.
	Random Vibration	Only the statistical properties of vibration are known.

basic mechanical elements and their force equations are shown. In these force equations x represents the displacement, dots on the top of variable x represents derivatives with respect to time; therefore $\dot{x}$ and $\ddot{x}$ becomes the velocity and acceleration. After defining the system by using these mechanical elements, mathematical model of the system in Figure 1.4 can be developed by applying physical laws to the system. In vibrational

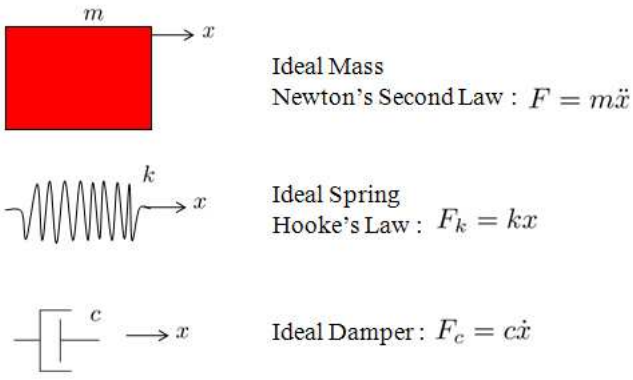


Figure 1.5. Mechanical Elements

systems, inertia elements are the masses. Mass is the property of a body that gives inertia to the body, whereas the inertia is commonly known as the resistance to starting motion

and resistance arriving to a full stop while in motion. Newton's second law, known as *law of acceleration*, is used to define the equations of motion of the masses. Newton's second law says that (Cohen and Whitman 1999):

Law 1.1 *The rate of change of momentum of a body is proportional to the resultant force acting on the body and is in the same direction.*

Mathematical representation of this law is

$$F = ma \ , \tag{1.5}$$

where F is the force, m is the mass and a is the acceleration.
A spring element is a flexible elastic object which is used to store mechanical energy. It can be deformed by an external force such that the deformation is directly proportional to the force applied to it. Hooke's law is used to define the model of a spring mathematically. Hooke's law says that (Ugural and Fenster 2003):

Law 1.2 *As the extension, so the force.*

Mathematical representation of this law is

$$F_k = kx \ , \tag{1.6}$$

where F_k is the force on the spring, k is the spring constant and x is the distance that the spring has been stretched.

Damping is any effect that reduces the amplitude of oscillations of an oscillatory system. Damper elements shows a damping effect in an oscillatory system. Generally damper element absorbs energy and the absorbed energy is dissipated as heat. Viscous damping is a common form of damping which is inherently found in many engineering systems. As a result, the characteristic of damping is generally modelled as viscous damping for civil structures. In physics, viscous damping is mathematically defined as a force synchronous with the velocity of the object but opposite direction to it. The force equation of a viscous damper can be written as

$$F_c = cv \ , \tag{1.7}$$

where F_c is the force seen on damper, c is the viscous friction constant and v is the velocity.

In a vibratory system, the force equations that are acting on the masses can be combined in an equation by using D'Alembert's principle. D'Alembert's principle is a statement of the fundamental classical laws of motion. The principle states that since the sum of the forces acting on a DOF 'i 'results in its acceleration a_i, the application of a fictitious force $-m_i\ddot{r}_i$ would produce a state of equilibrium (Jimin and Zhi-Fang 2001). This explanation can be written mathematically as

$$\sum_i (F_i - m_i a_i)\delta_i r_i = 0\,, \tag{1.8}$$

where F_i are the applied forces, $\delta_i r_i$ is the virtual displacement of the system, m_i are the masses of the particles in the system, a_i are the accelerations of the particles in the system. $m_i a_i$ represents the time derivatives of the system momenta. By using Newton's Laws, Hooke's Law and D'Alembert's Principle, one can write the equations of motion of any vibratory system.

1.2.1. Analysis of Single Degree of Freedom Systems

In vibration studies, "degree of freedom" number is a critical point. Before the analyzing procedure, degree of freedom number must be known. This term is defined as the minimum number of independent coordinates required to determine completely the motion of all parts of the system at any instant of time. A structure has as many natural frequencies as its degrees of freedom. If it is excited at any of these natural frequencies, a state of resonance exists, so that a large amplitude vibration response occurs. For each natural frequency, the structure has a particular way of vibrating, so that it has a mode of vibration at each natural frequency. Many real structures can be represented by a single degree of freedom (SDOF) model. Besides, there are many real structures that have several bodies and therefore several degrees of freedom. For example the system in Figure 1.6 can be represented by a single coordinate x mathematically , so this is a SDOF system. Now by using the physical laws that are mentioned, the free vibration behavior of this system will be investigated. Free vibration is the motion of a structure without any dynamical excitation or external forces. Actually all of the three schemes in Figure 1.6 represents an undamped SDOF vibrational system with a single mass and a single spring. But Civil engineers use the right one for the civil structures and mechanical engineers use the middle one. If the degree of freedom in Figure 1.6 is represented as $x(t)$, the following

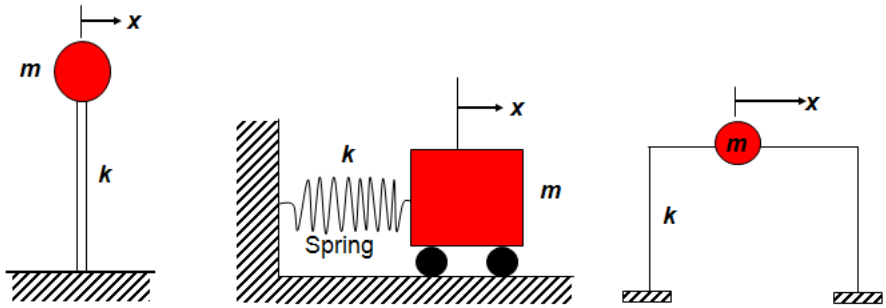


Figure 1.6. Mass-Spring System

equation can be derived by using physical laws that are mentioned before:

$$m\ddot{x} = -kx \tag{1.9}$$

$$m\ddot{x} + kx = 0 \tag{1.10}$$

Equation 1.10 can also be derived from energy approach. To derive this equation, first the kinetic energy must be written as $T = \frac{1}{2}m\dot{x}^2$ and potential energy as $U = \frac{1}{2}kx^2$ by choosing x as an equilibrium point. Lagrangian function for this vibrational system can be defined as,

$$L = T - U = \frac{1}{2}m\dot{x}^2 - \frac{1}{2}kx^2 \,. \tag{1.11}$$

The compact form of Lagrange's equation for a conservative system is given by,

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{x}} - \frac{\partial L}{\partial x} = 0 \,. \tag{1.12}$$

By taking derivatives of Equation 1.11, terms that are necessary for Equation 1.12 can be found as

$$\begin{aligned} \frac{\partial L}{\partial \dot{x}} &= m\dot{x} \,, \\ \frac{d}{dt}\frac{\partial L}{\partial \dot{x}} &= m\ddot{x} \,, \\ \frac{\partial L}{\partial x} &= -kx \,. \end{aligned} \tag{1.13}$$

By using Equations 1.12 and 1.13, the general equation for SDOF undamped vibration system can be written as

$$m\ddot{x} + kx = 0 \,. \tag{1.14}$$

Equation 1.10 is a linear, homogeneous 2^{nd} order differential equation with constant coefficients. The solution of this equation for $x(t)$ begins with assuming $x(t) = e^{st}$. By writing the Equation 1.10 with this assumption,

$$m\frac{\partial^2 e^{st}}{\partial t^2} + ke^{st} = 0\,, \tag{1.15}$$

$$(ms^2 + k)e^{st} = 0\,. \tag{1.16}$$

Since e^{st} cannot be equal to zero, $(ms^2 + k)$ must be equal to zero. The value of s can be found as

$$[ms^2 + k] = 0\,, \tag{1.17}$$

$$s_{1,2} = \sqrt{\frac{-k}{m}} = \pm j\omega_n\,, \tag{1.18}$$

$$\omega_n = \sqrt{\frac{k}{m}}\,, \tag{1.19}$$

where ω_n is the natural frequency of the system in *rad/s*. General solution of differential equation is,

$$x(t) = A_1 e^{s_1 t} + A_2 e^{s_2 t} = A_1 e^{jw_n t} + A_2 e^{-jw_n t}\,. \tag{1.20}$$

By using the relations

$$\cos x = \frac{e^{jx} + e^{-jx}}{2}\,, \tag{1.21}$$

$$\sin x = \frac{e^{jx} - e^{-jx}}{2j}\,, \tag{1.22}$$

Equation 1.20 can be written by means of cosine and sine functions,

$$x(t) = A\cos\omega_n t + B\sin\omega_n t\,. \tag{1.23}$$

To find the coefficients A and B, the initial displacement and initial velocity of the mass body must be known.

$$x(0) = A\cos 0 + B\sin 0 = A \tag{1.24}$$

$$\dot{x}(t) = -\omega_n A\sin\omega_n t + \omega_n B\cos\omega_n t \tag{1.25}$$

$$\dot{x}(0) = \omega_n B \tag{1.26}$$

From Equations 1.24 and 1.26 , the coefficients A and B can be written by means of the initial displacement and initial velocity.

$$A = x(0) \tag{1.27}$$

$$B = \frac{\dot{x}(0)}{\omega_n} \tag{1.28}$$

The solution of Equation 1.10 will be,

$$\begin{aligned} x(t) &= A\cos\omega_n t + B\sin\omega_n t \\ &= x(0)\cos\sqrt{\frac{k}{m}}t + \frac{\dot{x}(0)}{\sqrt{\frac{k}{m}}}\sin\sqrt{\frac{k}{m}}t\,. \end{aligned} \tag{1.29}$$

The graphical illustration of Equation 1.29 is shown in Figure 1.7. As shown in this

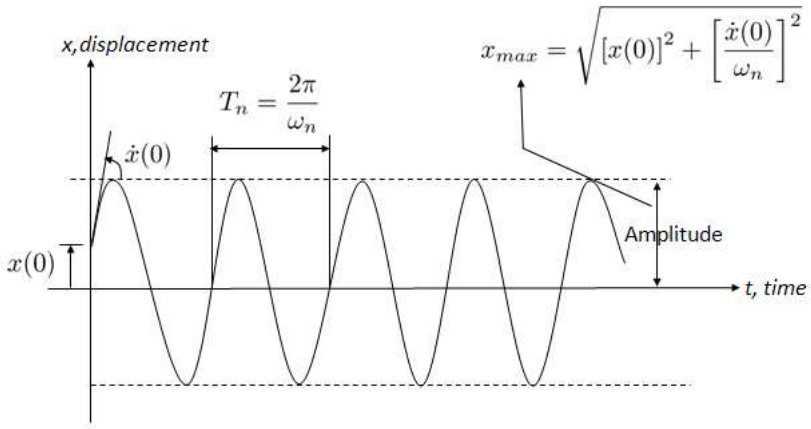


Figure 1.7. Free Vibration of SDOF Undamped System

figure, the amplitude of the free vibration response of an undamped system depends on the initial displacement and velocity. Amplitude remains the same cycle after cycle and motion does not decay because of the absence of damping.

The former equations describe the free vibration of the structure, however they don't explain why the system oscillates. The reason for oscillation is "conservation of energy". The conversion of potential energy in the spring and kinetic energy in the mass creates these oscillations. In our model in Figure 1.6, the mass will continue to oscillate forever, but in a real system there is always damping that dissipates the energy and therefore the system eventually comes to rest. Now assume that system also has a viscous damper with damping value C as in Figure 1.8. C is equal to the damping force for a unit velocity. Schematic diagrams in Figure 1.8 represents a damped SDOF vibrational system with a single mass, a single spring and a single damper. The equation of motion can be written as

$$F + F_c + F_k = 0\,, \tag{1.30}$$

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = 0\,. \tag{1.31}$$

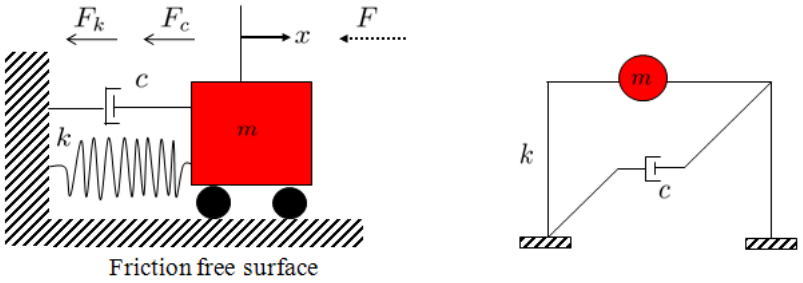


Figure 1.8. Mass-Spring-Damper System

The solution of the damped system can be found by using the solution method of Equation 1.10. By assuming $x(t) = e^{st}$,

$$m\frac{\partial^2 e^{st}}{\partial t^2} + c\frac{\partial e^{st}}{\partial t} + ke^{st} = 0\,, \tag{1.32}$$

$$(ms^2 + cs + k)e^{st} = 0\,. \tag{1.33}$$

Since e^{st} cannot be equal to zero, it follows that,

$$ms^2 + cs + k = 0\,. \tag{1.34}$$

Equation 1.34 is also called as characteristic equation. The roots of this equation can be found as,

$$s_{1,2} = \frac{-c \pm \sqrt{c^2 - 4mk}}{2m}\,. \tag{1.35}$$

Now there are three cases according to discriminant in Equation 1.35,

$$\Delta = c^2 - 4mk\,. \tag{1.36}$$

When Δ is positive, s_1 and s_2 will be negative real number, then $x(t)$ will become an exponentially decaying function. These systems are called overdamped systems. When Δ is negative, s_1 and s_2 will be complex conjugate numbers with negative real part. In that case, $x(t)$ will oscillate while it decays exponentially. When $\Delta = 0$, s_1 and s_2 will be equal to $-c/2m$, so $x(t)$ decays exponentially, and this is a critically damped system. The graphical illustration of these three cases is shown in Figure 1.9. Structural systems usually have small dampings. Because of this, we only consider the case $\Delta < 0$, underdamped systems. The damping value that makes discriminant zero is called critical damping coefficient. It is given by,

$$c_c = 2\sqrt{mk}\,. \tag{1.37}$$

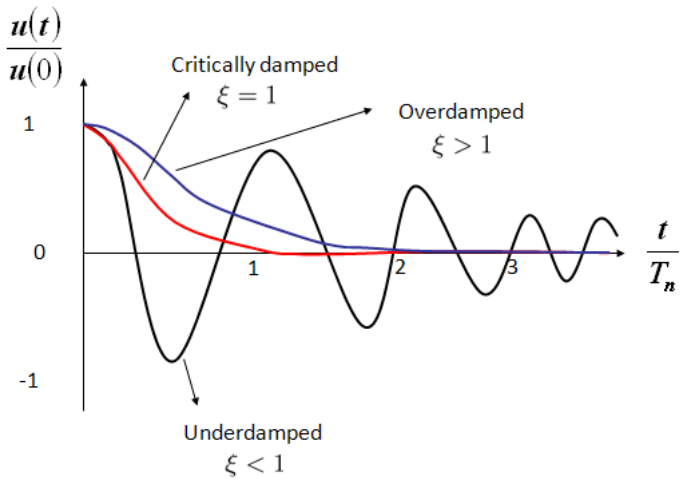


Figure 1.9. Free Vibration of SDOF Damped System

Damping ratio of the system is defined as the ratio of damping value and critical damping coefficient.

$$\xi = \frac{c}{c_c} = \frac{c}{2\sqrt{mk}} \tag{1.38}$$

Let us write Equation 1.35 as

$$s_{1,2} = \frac{-c}{2m} \pm j\sqrt{\frac{k}{m} - \frac{c^2}{4m^2}}\,. \tag{1.39}$$

In this equation the imaginary part of the root is usually referred as damped natural frequency, given as,

$$\omega_d = \sqrt{\frac{k}{m} - \frac{c^2}{4m^2}}\,. \tag{1.40}$$

By using the damping ratio, the mathematical definition of damped natural frequency can be written as,

$$\omega_d = \omega_n\sqrt{1-\xi^2}\,. \tag{1.41}$$

The real part of the roots can be written by means of natural frequency and damping factor as,

$$\xi\omega_n = \frac{c}{2m}\,. \tag{1.42}$$

Roots of the Equation 1.34 can be written as,

$$s_{1,2} = \sigma \pm j\omega_d = -\xi\omega_n \pm j\omega_d\,, \tag{1.43}$$

where σ is the decay rate. The solution of Equation 1.31 is,

$$x(t) = e^{\sigma t} \left[A \cos \omega_d t + B \sin \omega_d t \right] \ . \tag{1.44}$$

To find the coefficients A and B, the initial displacement and initial velocity of the mass body must be known. From Equation 1.44:

$$\begin{aligned} x(0) &= A \cos 0 + B \sin 0 = A \\ \dot{x}(0) &= -\omega_n A + \omega_d B \ . \end{aligned} \tag{1.45}$$

From Equation 1.45, the coefficients A and B can be written by means of the initial displacement and initial velocity as,

$$\begin{aligned} A &= x(0) \\ B &= \frac{\dot{x}(0) - \sigma x(0)}{\omega_d} \ . \end{aligned} \tag{1.46}$$

Therefore, the solution of the Equation 1.31 can be written as,

$$x(t) = e^{\sigma t} \left[x(0) \cos \omega_d t + \frac{\dot{x}(0) - \sigma x(0)}{\omega_d} \sin \omega_d t \right] \ . \tag{1.47}$$

The graphical illustration of Equation 1.47 is shown in Figure 1.10. The comparison between damped and undamped system can be seen from that figure. Coefficient p in

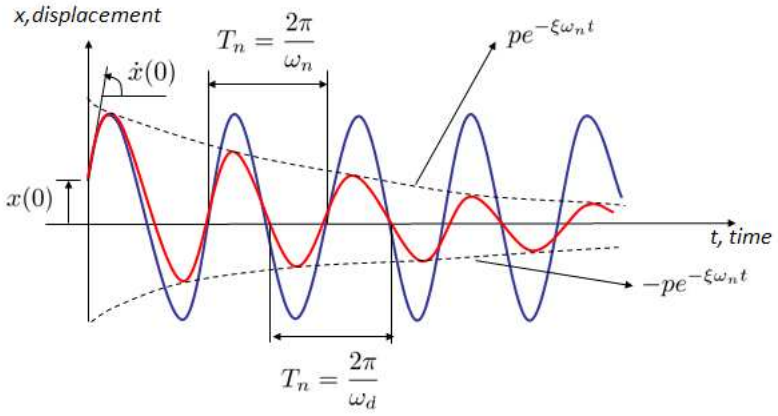


Figure 1.10. Comparison of Undamped System and Underdamped System

Figure 1.10 is magnitude of the solution. It can be derived as,

$$p = \sqrt{[x(0)]^2 + \left[\frac{\dot{x}(0) - \sigma x(0)}{\omega_d} \right]^2} \ . \tag{1.48}$$

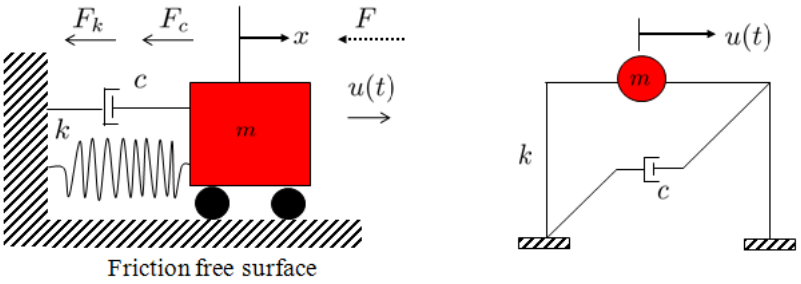


Figure 1.11. Mass-Spring-Damper System with Input

Free vibration behavior of SDOF systems is investigated until now. When any continuous input disturbs the system, it is called forced vibration. As shown in Figure 1.11, external input can be represented by $u(t)$ in system schemes and equations. By using physical laws again, the vibrational system in Figure 1.11 can be mathematically defined as,

$$F + F_c + F_k = u \,, \tag{1.49}$$

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = u(t) \,. \tag{1.50}$$

In that case, the response of the system to any external input can be found by writing the Impulse Response Function (IRF) of the system. Assuming initial conditions are zero, the transfer function of the system can be found easily by applying Laplace transformation to Equation 1.50.

$$s^2 mX(s) + scX(s) + kX(s) = U(s) \tag{1.51}$$

$$H(s) = \frac{X(s)}{U(s)} = \frac{1}{s^2 m + sc + k} \tag{1.52}$$

The transfer function can be shown as,

$$H(s) = \frac{1}{s^2 m + sc + K} = \frac{1/m}{s^2 + s(c/m) + (k/m)} \,. \tag{1.53}$$

The roots of the denominator of Equation 1.53 is evaluated as s_1 and s_1^* in Equation 1.43. Since these roots are the poles of the transfer function, λ_1 and λ_1^* symbols can be used instead of s_1 and s_1^*.

$$H(s) = \frac{1/m}{(s-\lambda_1)(s-\lambda_1^*)} = \frac{A_1}{s-\lambda_1} + \frac{A_2}{s-\lambda_1^*} \tag{1.54}$$

The IRF of the system can be found by taking inverse Laplace transform of Equation 1.54,

$$h(t) = A_1 e^{\lambda_1 t} + A_2 e^{\lambda_1^* t} = e^{\sigma t}(A_1 e^{jw_d t} + A_2 e^{-jw_d t}) \,, \tag{1.55}$$

$$\lambda_{1,2} = -\sigma \pm jw_d \,, \tag{1.56}$$

where σ is the decay rate and w_d is the damped natural frequency.
The main objective of this study is to estimate the values of w_d and σ from the measurements. Besides that Civil engineers are also interested in the values of m, c, and k.

1.2.2. Analysis of Multi Degree of Freedom Systems

For Multi Degree of Freedom (MDOF) systems, system parameters are M, C and K matrices. There are N natural frequencies and N damping ratios for a N-degree of freedom system. For example in Figure 1.12, there are 2 mass blocks and they can move

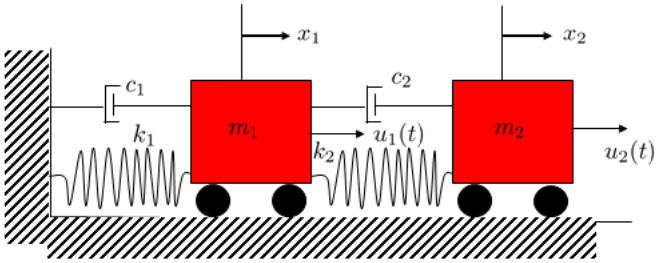


Figure 1.12. 2-DOF System

independently. If any of the masses is excited by an external force, the other one can move by the affect of this force. This system is a two degree of freedom system (2-DOF).
While constructing the mathematical model of the system, the degree of freedom number must be known. Equations of motion can be written according to degree of freedoms as shown in Equations 1.57 and 1.58.

$$-m_1\ddot{x}_1 \; -(c_1+c_2)\dot{x}_1 \; +c_2\dot{x}_2 \; -(k_1+k_2)x_1 + k_2x_2 + u_1(t) = 0 \tag{1.57}$$

$$-m_2\ddot{x}_2 \; +c_2\dot{u}_1 \; -c_2\dot{x}_2 \; +k_2x_1 - k_2x_2 + u_2(t) = 0 \tag{1.58}$$

Equations 1.57 and 1.58 can be combined in a matrix equation as,

$$\begin{bmatrix} m_1 & 0 \\ 0 & m_2 \end{bmatrix} \begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{bmatrix} + \begin{bmatrix} c_1+c_2 & -c_2 \\ -c_2 & c_2 \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} + \begin{bmatrix} k_1+k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix} .$$

(1.59)

Equation 1.59 can be written as,

$$M\ddot{\boldsymbol{x}} + C\dot{\boldsymbol{x}} + K\boldsymbol{x} = \boldsymbol{u}\,. \tag{1.60}$$

MDOF damped vibrational systems can be defined by Equation 1.60. The transfer function of this 2-input, 2-output system can be written in a matrix form as,

$$H(s) = \begin{bmatrix} H_{11}(s) & H_{12}(s) \\ H_{21}(s) & H_{22}(s) \end{bmatrix}\,, \tag{1.61}$$

where $H_{ij}(s)$ for $i, j = 1, 2$, is the transfer function from i^{th} input to j^{th} output.
2-DOF system has 2 natural frequency and 2 modal damping factor as w_{d1}, w_{d2} and σ_1, σ_2. IRF of 2-DOF system can be written as,

$$h(t) = \sum_{i=1}^{2} e^{\sigma_i t}(a_i \sin w_{di}t + b_i \cos w_{di}t)\,. \tag{1.62}$$

Transfer function and IRF of MIMO vibrational systems can be generalized in a form such that,

$$h(t) = \sum_{i=1}^{N} e^{\sigma_i t}(a_i \sin w_{di}t + b_i \cos w_{di}t)\,, \tag{1.63}$$

$$H_{ij}(s) = \sum_{r=1}^{N} \left[\frac{A_{ij(r)}}{s - s_r} + \frac{A^*_{ij(r)}}{s - s^*_r}\right]\,, \tag{1.64}$$

where N is the number of degree of freedom in the system. Mathematically, our aim is to find system parameters as w_{di}, σ_i in Equation 1.63 by using the measurement of $h(t)$.

As mentioned before, natural frequency values can be easily calculated from differential equations for SDOF systems. The solution for MDOF systems begins with undamped case. Equation 1.65 is a general equation for undamped vibrational systems. All mass-spring systems can be defined by this equation.

$$M\ddot{\boldsymbol{x}}(t) + K\boldsymbol{x}(t) = 0 \tag{1.65}$$

The displacement response based on the n^{th} mode can be given as,

$$\boldsymbol{x}(t) = q_n(t)\boldsymbol{\phi_n}\,, \tag{1.66}$$

where $q_n(t)$ is the time variation of displacement and $\boldsymbol{\phi_n}$ is the mode shape which does not vary with time. $q_n(t)$ and $\boldsymbol{\phi_n}$ can be written as,

$$q_n(t) = A_n \cos \omega_n t + B_n \sin \omega_n t \text{ for n=1,2,..,N ,} \tag{1.67}$$

$$\boldsymbol{\phi_n} = \begin{bmatrix} \phi_{1n} \\ \phi_{2n} \\ . \\ . \\ \phi_{Nn} \end{bmatrix} . \tag{1.68}$$

The displacement function can be written as,

$$\boldsymbol{x}(t) = \boldsymbol{\phi_n}[A_n \cos \omega_n t + B_n \sin \omega_n t] . \tag{1.69}$$

The second derivative of $\boldsymbol{x}(t)$, acceleration function, can be found easily from Equation 1.69.

$$\begin{aligned} \ddot{\boldsymbol{x}}(t) &= \frac{\partial \left[-\boldsymbol{\phi_n} A_n \omega_n \sin \omega_n t + \boldsymbol{\phi_n} B_n \omega_n \cos \omega_n t\right]}{\partial t} \\ &= -\boldsymbol{\phi_n} A_n \omega_n^2 \cos \omega_n t - \boldsymbol{\phi_n} B_n \omega_n^2 \sin \omega_n t \\ &= -\boldsymbol{\phi_n} \omega_n^2 q_n(t) \end{aligned} \tag{1.70}$$

By using Equation 1.65, 1.66 and 1.70,

$$M(-\boldsymbol{\phi_n} \omega_n^2 q_n(t)) + K \boldsymbol{\phi_n} q_n(t) = 0 , \tag{1.71}$$

$$q_n(t) \left[-\omega_n^2 M \boldsymbol{\phi_n} + K \boldsymbol{\phi_n}\right] = 0 . \tag{1.72}$$

Since $q_n(t)$ is the time variation of displacement, it can't be zero, so $[-\omega_n^2 M \boldsymbol{\phi_n} + K \boldsymbol{\phi_n}]$ will be equal to zero.

$$\left[-\omega_n^2 M \boldsymbol{\phi_n} + K \boldsymbol{\phi_n}\right] = 0 \tag{1.73}$$

$$K \boldsymbol{\phi_n} = \omega_n^2 M \boldsymbol{\phi_n} \tag{1.74}$$

Equation 1.74 is a matrix eigenvalue problem and it can be written as,

$$\left[K - \omega_n^2 M\right] \boldsymbol{\phi_n} = 0 . \tag{1.75}$$

By using Equation 1.75, N equations for ϕ_{jN} (j=1,2,..,N) can be written. A homogeneous system of N equations in N unknowns has a solution different from the obvious one, if

and only if the determinant of the coefficient matrix is zero. The solution of the Equation 1.75 exists if and only if

$$\det\left[K - \omega_n^2 M\right] = 0\,. \tag{1.76}$$

This equation is called frequency equation, because natural frequencies can be found from this equation. There are N real and positive roots for ω_n^2. Since the natural frequencies are found, mode shapes, $\boldsymbol{\phi_n}$, can be found from Equation 1.75.

1.3. Main Goal

Our main goal is to find the structural parameters of the system in our experiment. We want to find the natural frequencies, damping ratios and the mode shapes of the system in Figure 1.2. To do this, first the simple estimation methods that can be applied to SDOF systems will be examined in the next chapter. After that advanced estimation algorithms will be investigated and we will try to identify our experimental system mathematically by using these algorithms.

Table 1.2. Most Common Modal Analysis Algorithms

Algorithm	**Domain**
Complex Exponential Algorithm	Time
Least-Squares Complex Exponential Algorithm	Time
Polyreference Time Domain Algorithm	Time
Presence of Damping	Time
Ibrahim Time Domain Algorithm	Time
Multi-reference Ibrahim Time Domain Algorithm	Time
Eigensystem Realization Algorithm	Time
Polyreference Frequency Domain Algorithm	Frequency
Simultaneous Frequency Domain	Frequency
Multi-reference Frequency Domain	Frequency
Rational Fraction Polynomial	Frequency
Orthogonal Polynomial	Frequency
Complex Mode Indicator Function	Frequency

There are many advanced algorithms about system identification. In this study algorithms in the literature is reviewed and the most common ones for our experiment is listed in the Table 1.2.

Three fundamental and reliable algorithms are chosen from this list. We tried to use two time-domain algorithms and one frequency-domain algorithm for the experimental data. These are the bold ones in the Table 1.2, “Least Squares Complex Exponential (LSCE) Method ” , “Polyreference Frequency Domain Algorithm ” and “Eigensystem Realization Algorithm ” . Initially, we will try to understand the theoretical background of these methods. These methods will be verified for an artificial N degree of freedom systems by using MATLAB simulations. Then these MATLAB scripts will be applied to real measurement data and we will try to find the characteristic parameters of the system in our experiment.

CHAPTER 2

SIMPLE ESTIMATION METHODS

2.1. Frequency Domain Estimation

In the first chapter, it was shown that the transfer function of a damped vibrational system can be written as

$$\begin{aligned} H(s) &= \frac{1/m}{s^2 + s(c/m) + (k/m)} \\ &= \frac{1/m}{(s-\lambda_1)(s-\lambda_1^*)} = \frac{A_1}{s-\lambda_1} + \frac{A_2}{s-\lambda_1^*} \,, \end{aligned} \tag{2.1}$$

where A_1 and A_2 are the residues of the transfer function. The residues of the transfer function directly related to the amplitude of the IRF. By multiplying both sides of the transfer function expression by $s-\lambda_1$ and evaluating the result at $s=\lambda_1$, residue A_1 can be found as

$$\begin{aligned} [\,(s-\lambda_1)H(s)]\,|_{s-\lambda_1} &= [\,A_1 + \frac{(s-\lambda_1)A_2}{(s-\lambda_1^*)}\,]\,|_{s-\lambda_1} \,, \\ A_1 &= \frac{1/m}{\lambda_1 - \lambda_1^*} = \frac{1/m}{j2\omega_1} \,. \end{aligned} \tag{2.2}$$

By the same way, A_2 can be found easily,

$$A_2 = \frac{1/m}{-j2\omega_1} \,. \tag{2.3}$$

As shown in Equations 2.2 and 2.3, A_1 and A_2 are complex conjugates of each other, so A_1^* can be written instead of A_2. Transfer function can be written as

$$H(s) = \frac{A_1}{(s-\lambda_1)} + \frac{A_1^*}{(s-\lambda_1^*)} \,. \tag{2.4}$$

By evaluating the transfer function along the $j\omega$ axis, the frequency response of the system can be found.

$$H(j\omega) = \frac{A_1}{(j\omega-\lambda_1)} + \frac{A_1^*}{(j\omega-\lambda_1^*)} \tag{2.5}$$

Experimentally when somebody is talking about measuring the transfer function, actually the FRF is measured. At damped frequency, transfer function is such that,

$$H(j\omega_1) = \frac{-A_1}{\sigma_1} + \frac{A_1^*}{(j2\omega_1 - \sigma_1)} \,. \tag{2.6}$$

Second term in Equation 2.6 approaches zero when ω_1 gets large. $H(\omega_1)$ can be represented as

$$H(\omega_1) = \frac{-A_1}{\sigma_1} \ . \tag{2.7}$$

Frequency response of SDOF system can be represented as

$$H(\omega) = \frac{A_1}{j\omega - \lambda_1} \ . \tag{2.8}$$

Assuming that our system is a lightly damped SDOF system, parameters needed for a partial fraction model can be estimated directly from the measured FRF. While this approach is based upon a SDOF system, as long as the modal frequencies are not too close together, the method can be used for multiple degree of freedom (MDOF) systems as well.

As shown in Equation 2.5, A_1 and λ_1 must be estimated to identify the FRF. Since $\lambda_1 = \sigma_1 + j\omega_1$, decay rate and the natural frequency of the system must be estimated to find the pole of the transfer function.

The estimation process begins with estimating the damped natural frequency, ω_1. Damped natural frequency could be estimated in one of three ways :

1)Damped natural frequency is the frequency where magnitude of FRF reaches maximum.

2)Damped natural frequency is the frequency where the real part of FRF crosses zero.

3)Damped natural frequency is the frequency where imaginary part of FRF reaches a relative minima or maxima.

The last approach generally gives the most reliable results. The estimation process can be shown by an example simulation on MATLAB environment. Assume that we have a SDOF vibrational system which has a mass of 1 kg, 100 N/m spring constant and 1 Ns/m damping coefficient value. The frequency response of this system is shown in Figure 2.1 and 2.2. As shown in Figure 2.2, imaginary part of FRF reaches a relative minima at nearly 10 rad/s. We can say that damped natural frequency of this system is equal to 10 rad/s from the frequency response of the system. Analytically, the real damped natural frequency can be calculated as 9.9875 rad/s from Equation 1.40. Error is caused from the sampling rate of the measurement device. Actually the damped natural frequency of the system is 9.9875 rad/s but we saw that value as 10 rad/s.

Once the damped natural frequency w_1 has been estimated, then the damping ratio ξ_1 can be estimated from the magnitude of the FRF. Damping ratio ξ_1 can be estimated by using

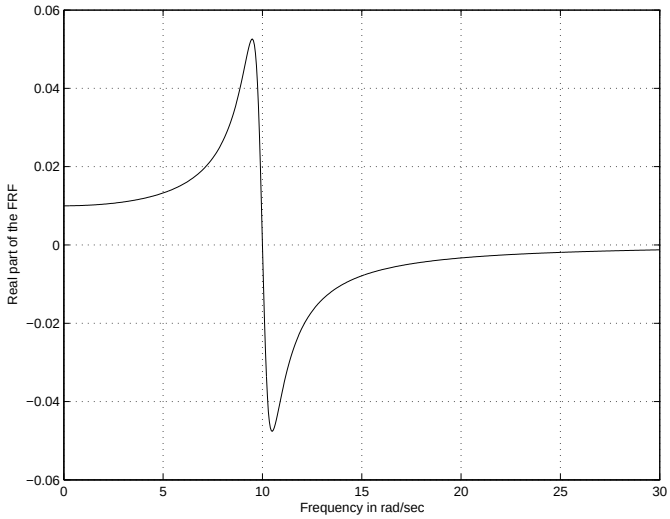


Figure 2.1. Real part of the FRF

the half-power bandwidth method. This method uses the data from the FRF in the region of the resonance frequency to estimate the fraction of critical damping from the formula:

$$\xi_1 = \frac{\omega_b - \omega_a}{2\omega_1} \ , \tag{2.9}$$

where ω_1 is the damped natural frequency as previously estimated. ω_a is the frequency, below ω_1, where the magnitude is 0.707 of the peak magnitude of the FRF. This corresponds to a half power point. ω_b is the frequency, above ω_1, where the magnitude is 0.707 of the peak magnitude of the FRF. This is also a half power point. For our example, we can see that the half power frequency values are $\omega_a = 9.46$ rad/s and $\omega_b = 10.47$ rad/s. Therefore the fraction of critical damping can be calculated from Equation 2.9 as 0.0506. Once ξ_1 is estimated, the decay rate, σ_1 can be estimated as,

$$\sigma_1 = -\xi_1\omega_1 = 0.505 \ . \tag{2.10}$$

The pole of the transfer function is $\lambda_{1,2} = -0.505 \pm j9.9875$. Once $\lambda_{1,2}$ has been estimated, the residue A_1 can be estimated by evaluating the partial fraction model at a specific frequency. If the specific frequency is chosen to be ω_1, the following result is obtained.

$$H(j\omega_1) = \frac{A_1}{j\omega - (\sigma_1 + j\omega_1)} + \frac{A_1^*}{j\omega - (\sigma_1 - j\omega_1)} \tag{2.11}$$

As long as ω_1 is not too small, the above equation could be approximated as,

$$H(\omega_1) = \frac{-A_1}{\sigma_1} \ \rightarrow A_1 \approx (-\sigma_1)H(\omega_1) \ . \tag{2.12}$$

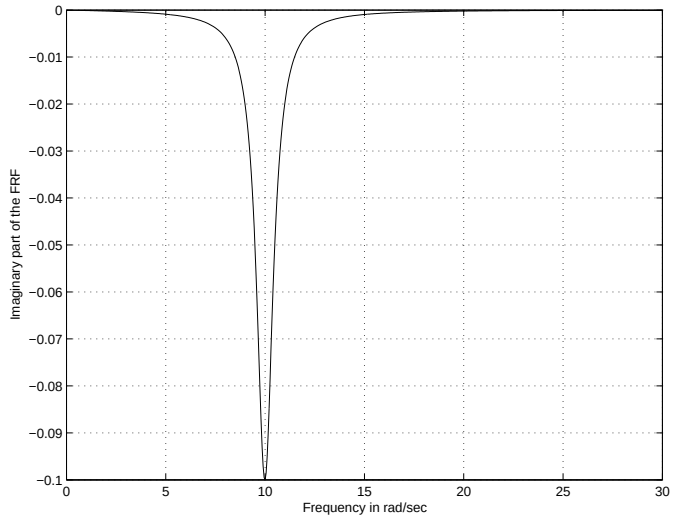


Figure 2.2. Imaginary part of the FRF

In our example $H(\omega_1) = 0.0025 - 0.1001j$, so the residue A_1 can be calculated as $0.0013 - 0.0506j$ from Equation 2.12. Now the transfer function can be written as shown in Equation 2.13. We showed that the frequency response can be mathematically defined from measurements for a SDOF damped vibrational system.

$$H(j\omega) = \frac{0.0013 - 0.0506j}{(j\omega - (-0.505 + j9.9875))} + \frac{0.0013 + 0.0506j}{(j\omega - (-0.505 - j9.9875))} \tag{2.13}$$

Estimated transfer function can be written as,

$$H(s) = \frac{0.026s + 1.023}{s^2 + 1.01s + 100.005} \ . \tag{2.14}$$

From Equation 1.53, the exact transfer function can be written as

$$H(s) = \frac{1}{s^2 + s + 100} \ . \tag{2.15}$$

We can say that estimated transfer function is satisfactory, the natural frequencies approach the exact value. But this is a simple estimation method, so it will fail for noisy measurement data. As a summary, clear frequency response measurement can be used to estimate the parameters of the system. MATLAB code in Appendix-A1 is used to obtain the frequency response figures and values for our example.

2.2. Time Domain Estimation

A simple time domain method will be investigated for parameter estimation. Free vibration test of a structure with M,C,K is used to explain this method. The relationship

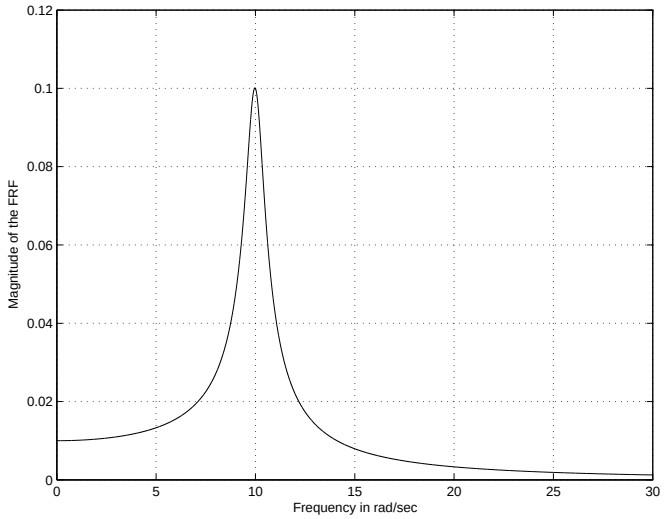


Figure 2.3. Magnitude of the FRF

between the damped and undamped natural frequency is known. Since $T_D = \frac{2\pi}{\omega_d}$ From Equation 1.41, the relationship between the damped and undamped natural period can be written as,

$$T_D = \frac{T_n}{\sqrt{1-\xi^2}} \, . \tag{2.16}$$

Two consecutive peaks from the acceleration response data of the structure will be used for estimation. Assume that the first peak is at time t, then the next peak must be at time $t + T_D$. By using Equation 1.44, the ratio of the acceleration function at these two time instants can be taken and the Equation 2.17 can be derived easily.

$$\frac{x(t)}{x(t+T_D)} = exp(\xi\omega_n T_D) = exp(\frac{2\pi\xi}{\sqrt{1-\xi^2}}) \tag{2.17}$$

If the peaks in the time domain response is numbered as i^{th}, $i+1^{th}$ peak, one can write that,

$$\frac{x_i}{x_{i+1}} = exp(\frac{2\pi\xi}{\sqrt{1-\xi^2}}) \, . \tag{2.18}$$

The natural logarithm of the ratio in Equation 2.18 is called the logarithmic decrement which is denoted by δ.

$$\delta = \ln \frac{x_i}{x_{i+1}} = \frac{2\pi\xi}{\sqrt{1-\xi^2}} \tag{2.19}$$

Since the damping ratio is so small, the term of $\sqrt{1-\xi^2}$ will be approximately one. The logarithmic decrement, δ will be,

$$\delta = 2\pi\xi \, . \tag{2.20}$$

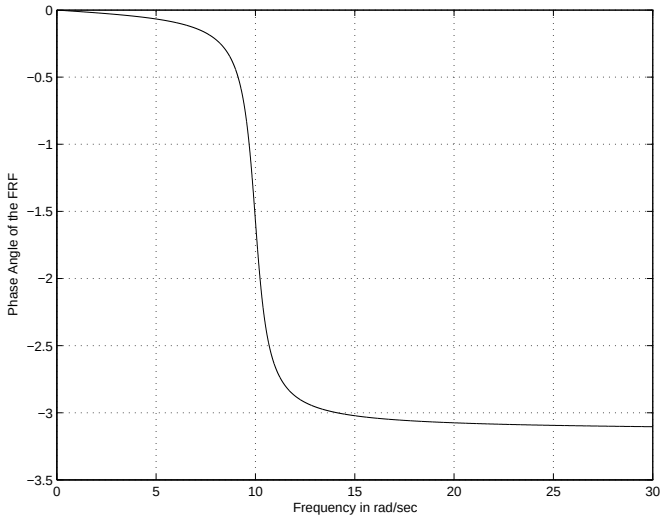


Figure 2.4. Phase of the FRF

By using Equation 2.18 and 2.19, logarithmic decrement coefficient can be written by means of x_i's.

$$\frac{x_1}{x_{j+1}} = \frac{x_1}{x_2}\frac{x_2}{x_3}\frac{x_3}{x_4}...\frac{x_k}{x_{k+1}} = e^{k\delta} \tag{2.21}$$

A relationship between the damping coefficient and the peak values can be written as,

$$\delta = \frac{1}{j}\ln\frac{x_1}{x_{j+1}} = 2\pi\xi\ , \tag{2.22}$$

$$\xi = \frac{1}{2\pi j}\ln\frac{x_i}{x_{i+j}}\ . \tag{2.23}$$

We showed that the damping ratio of a vibrational structure can be determined from acceleration measurements. This method will be verified with an example computer simulation. Assume that we have a SDOF vibrational system again which has a mass of 1 kg, 100 N/m spring constant and 1 Ns/m damping coefficient value. The impulse response of this system is shown in Figure 2.5. Assume that we measure the impulse response data at the first peak and the eleventh peak. MATLAB code in Appendix-A1 is used to obtain the impulse response figure. Natural vibration period and frequency can be found as,

$$T_d = \frac{t_{11} - t_1}{11 - 1} = \frac{6.44 - 0.16}{10} = 0.628 \text{ sec}\ , \tag{2.24}$$

$$\omega_d = \frac{2\pi}{T_d} \approx 10.0051 \text{ rad/s}\ . \tag{2.25}$$

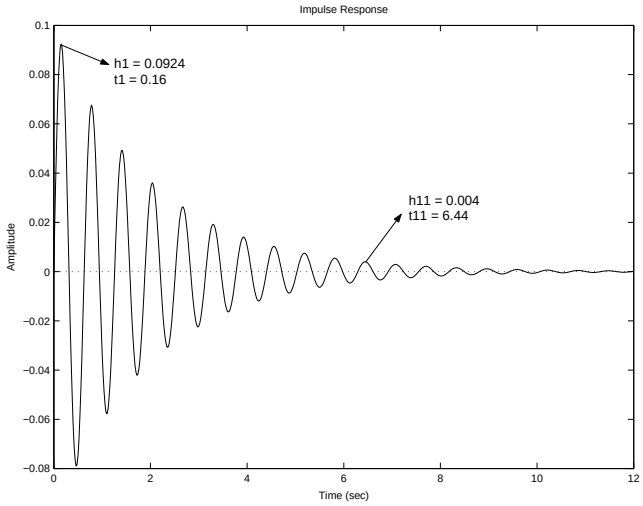


Figure 2.5. Impulse Response Function

Damping ratio can be determined by using Equation 2.23.

$$\xi = \frac{1}{2\pi 10} \ln \frac{0.0924}{0.004} = 0.05 \tag{2.26}$$

Exact damping factor and damped natural frequency can be found by using Equations 1.38 and 1.40.

$$\xi = \frac{c}{2\sqrt{mk}} = \frac{1}{2\sqrt{100}} = 0.05 \tag{2.27}$$

$$\omega_d = \sqrt{\frac{k}{m} - \frac{c^2}{4m^2}} = \sqrt{\frac{100}{1} - \frac{1}{4}} \approx 9.9875 \text{ rad/s} \tag{2.28}$$

We showed that structural parameters of a SDOF vibrational system can be found from time-domain measurements. Results of the time domain method is satisfactory. However these simple estimation methods can't be applied to a real system because of the measurement noise. Advanced algorithms must be used for SDOF and MDOF systems to counteract the noise.

CHAPTER 3

PRE-PROCESSING TECHNIQUES

Before applying system identification algorithms, some pre-processing techniques must be applied to experimental measurements. By using these techniques, noise in measurements can be reduced, then the IRF or FRF data of the system can be found clearly. Windowing, averaging, filtering are the main pre-processing techniques.
The experimental system in this study is a single-input multi-output(SIMO) system. We repeated the experiment 5 times and we obtained 5 input-output data pair for each reference point. Acceleration measurements are taken from 10 reference point. At the end of the experiment, we had 50 measurement files. By using all of these data files and pre-processing techniques, IRF or FRF estimate of each data pair can be found.

3.1. Frequency Domain Division Method

To eliminate the noise from input data, we assumed that the noise signal is equal to the input signal measurement after first 100 data points, as shown in Figure 3.1. By

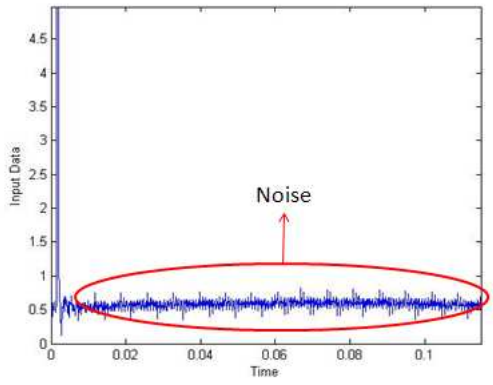


Figure 3.1. Noise Estimation from Input Signal

taking the d.c. component of the noise signal and subtracting it from the input data, the input data can be cleared from noise approximately.

After making a noise reduction, windowing process can be applied to input/output data. Windowing is simply defined as multiplying the signal by a typical window func-

tion. This process is used to minimize edge effects that results in spectral leakage in the FFT spectrum. Exponential window is used for measurement data in our experiment. Mathematically this process can be given as,

$$x_w(t) = x(t)e^{-(t/\tau)} \,, \tag{3.1}$$

where τ is called the exponential time constant. Also $\sigma = 1/\tau$ is called as damping or decay rate. Generally decay rate is selected as,

$$\sigma = -\frac{ln(w(T)}{T} \,, \tag{3.2}$$

where $w(T)$ is the value of exponential window at the end of analyzer's time record and T is analyzer's time record length.

To find the FRF's of each input-output pair, FFT of the input and the output data can be taken by using MATLAB. Then FRF of each pair can be found by,

$$H(f) = \frac{Re[X(f)X^*(f)]}{Y(f)X^*(f)} \,. \tag{3.3}$$

By averaging all FRF's of one reference point, the error in the estimate of the FRF's can be reduced. Then impulse response data of the system can be obtained by using inverse FFT transform.

$$\tilde{h}(t) = \mathcal{IFT}\{H(f)\} \tag{3.4}$$

Pre-processing part of the study can be shown in a block diagram as in Figure 3.2.

Instead of division in frequency domain, an alternative method can be used to estimate the FRF. The autocorrelation and cross-correlation functions of input and output data can be used to obtain a FRF estimation. The autocorrelation function and cross correlation function are the inverse Fourier transforms of power spectral density and cross spectral density of the input/output signals. The relation of correlation and spectral density functions can be given as,

$$S_{XY}(f) = \mathcal{F}\{R_{XY}(\tau)\} \,, \tag{3.5}$$

$$S_{YY}(f) = \mathcal{F}\{R_{YY}(\tau)\} \,. \tag{3.6}$$

By dividing the power spectral density of output signal by cross power spectral density of input and output, the estimate of FRF can be obtained.

$$S_{XY}(f) = H^*(f)S_X(f) \quad S_{YY}(f) = H^*(f)H(f)S_X(f) \tag{3.7}$$

$$\hat{H}(f) = \frac{S_{YY}(f)}{S_{XY}(f)} \tag{3.8}$$

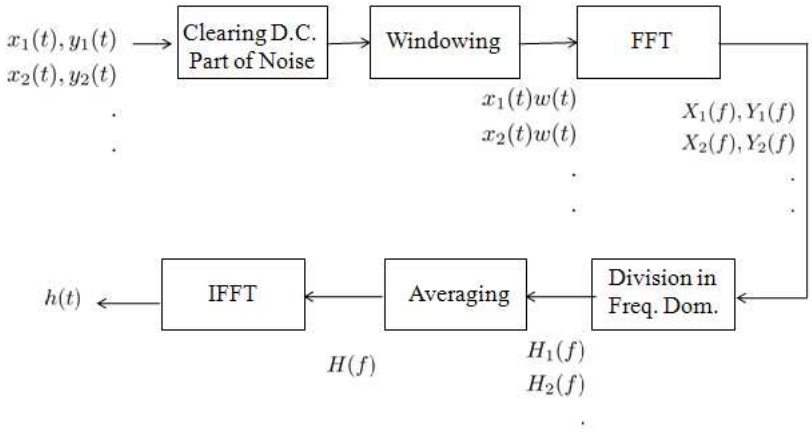


Figure 3.2. Pre-processing

Finally we can take the IFFT of FRF estimate to obtain the IRF estimate of the experimental system.

3.2. Time Domain IRF Estimation Method

For a successive modal parameter estimation, it is important to obtain the IRF after making a well-arranged experiment. The output of a continuous time LTI system can be obtained by taking the convolution integral of the input signal and the IRF of the system.

$$y(t) = x(t) * h(t) = \int_{-\infty}^{+\infty} x(\tau)h(t-\tau)d\tau \tag{3.9}$$

It was shown that, FRF of an LTI system can be found by dividing the Fourier transform of output signal by the Fourier transform of input signal as,

$$H(w) = \frac{Y(w)}{X(w)} \, . \tag{3.10}$$

Once FRF is found, IRF can be found easily by taking the inverse Fourier transform of the FRF. But there are two inconvenient cases for Equation 3.10.

1) At some frequencies the value of $X(w)$ can be approximately zero and the result of the division in Equation 3.10 can be extremely large.

2) The measurement noise can't be modelled as a gaussian white or random gaussian noise.

Because of these 2 serious problem, we searched a new method for estimating the IRF. Consequently, we developed a correlation based IRF estimation algorithm.

Assume that the sampling period of the continuous time LTI system is T seconds and the duration of the IRF is LT seconds, so the sampled IRF can be written as,

$$h[n] = \begin{cases} 0 & n < 0 \\ h(t)|_{t=nT}, & n = 0, \cdots, L-1, \\ 0, & n \geq L. \end{cases} . \tag{3.11}$$

By assuming the duration of the input signal is $M << L$, the discrete time input signal can be written as,

$$x[n] = \left\{ \begin{array}{ll} 0 & n < 0 \\ x(t)|_{t=nT}, & n = 0, \cdots, M-1, \\ 0, & n \geq M. \end{array} \right\} . \tag{3.12}$$

Since the input and output signals of the system is sampled, this system is a discrete time system. The output of the system can be found by using convolution sum instead of convolution integral.

$$y[n] = x[n] * h[n] = \sum_{m=-\infty}^{+\infty} x[m]h[n-m] \tag{3.13}$$

By using Equations 3.11 , 3.12 and 3.13, the output of the system can be written as,

$$y[n] = \sum_{k=0}^{M-1} x[k]h[n-k], \quad n = 0, \cdots, M+L-2 \, . \tag{3.14}$$

By making the matrix definitions below,

$$\mathbf{h} = [h[0], h[1], \cdots, h[L-1]]^T \, , \tag{3.15}$$

$$\mathbf{x} = [x[0], x[1], \cdots, x[M-1]]^T \, , \tag{3.16}$$

$$\mathbf{y} = [y[0], y[1], \cdots, y[M+L-2]]^T \, , \tag{3.17}$$

and the convolution operator X,

$$\begin{aligned} \mathbf{X} &= \text{Toeplitz}\{[\mathbf{x}; \mathbf{0}_{(L-1,1)}], [x[0], \mathbf{0}_{(1,L-1)}]\} \\ &= \begin{bmatrix} x[1] & 0 & \dots & \dots & \dots & \dots & 0 \\ x[2] & x[1] & \dots & \dots & \dots & \dots & 0 \\ \vdots & \vdots & \ddots & \dots & \dots & \dots & \vdots \\ x[M] & x[M-1] & \dots & x[1] & \vdots & \dots & 0 \\ 0 & x[M] & x[M-1] & \dots & \ddots & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & x[M] & \dots & \dots & x[1] \end{bmatrix} , \end{aligned} \tag{3.18}$$

Equation 3.14 can be written as a matrix equation,

$$\mathbf{y} = \mathbf{X}\mathbf{h} \, , \tag{3.19}$$

where $\mathbf{y}$ is the sampled response of the system, h is the impulse response vector and X is the convolution matrix. The important point at Equation 3.19 is only $\mathbf{h}$ is unknown, output and the input signal can be measured.

To estimate the impulse response vector of the system, the observation of the input can be used. For this reason, the correlation of the input and output signal can be written as,

$$\text{xcorr}\{x[n], y[n]\} = \mathbf{X}^T\mathbf{y} = \mathbf{X}^T\mathbf{X}\mathbf{h} \, . \tag{3.20}$$

From Equation 3.20, estimate of IRF can be given as,

$$\bar{\mathbf{h}} = (\mathbf{X}^\mathbf{T}\mathbf{X})^{-\mathbf{1}}\mathbf{X}^\mathbf{T}\mathbf{y} \, , \tag{3.21}$$

which is known as the least squares estimate of the system impulse response (Louis L. Scharf 2001).

Input signal in our experiment is taken as a dirac delta function. Thus the rightmost term in the Equation 3.20, $\mathbf{X}^\mathbf{T}\mathbf{X}\mathbf{h}$ can be taken as a multiplication of impulse response vector by a scalar. The Equation 3.20 can be written as,

$$\mathbf{X}^T\mathbf{y} = \mathbf{X}^T\mathbf{X}\mathbf{h} \approx \alpha\mathbf{I}\mathbf{h} = \alpha\mathbf{h} \, . \tag{3.22}$$

The approximation in Equation 3.22 is acceptable when the input signal is obtained by "hitting a hammer "which closely resembles a Dirac Delta function. Note that the approximation in Equation 3.22 becomes an equality when the input signal is a perfect Dirac Delta function. The term of $\mathbf{X}^\mathbf{T}\mathbf{X}$ is the autocorrelation of the input signal $x[n]$, and this can be represented by a diagonal matrix $\alpha\mathbf{I}$ where α is the total energy of the input signal $x[n]$. This relation can be shown as,

$$\alpha = \mathbf{X}^T\mathbf{X} = \sum_{k=0}^{M-1} x^2[k] \, . \tag{3.23}$$

By using Equation 3.22, an estimate of impulse response of the system can be written as,

$$\hat{\mathbf{h}} = \frac{1}{\alpha} \, \mathbf{X}^T\mathbf{y} \, . \tag{3.24}$$

In each case, Equation 3.21 gives more reliable results than Equation 3.24, because there must be an approximation error from Equation 3.22. We compared the results of these two equations in the next section.

By using the estimates of the impulse response in Equation 3.24 and Equation 3.21, inconvenient cases in Equation 3.10 are solved. Now any parameter estimation algorithm can be applied to estimated impulse response data to find the parameters of the system.

3.3. Comparison of IRF Estimation Methods

IRF of the structure which is obtained from experimental data by using IRF estimation methods explained in former sections. A MATLAB simulation is performed to compare these methods for one reference point. In Figures 3.3 and 3.5, estimated IRF's from frequency division method and time domain estimation methods are compared. In Figures 3.4 and 3.6, the absolute relative error graphs are given for each method. Error function in Figure 3.4 can be given as,

$$\text{Absolute Relative Error} = \frac{|\tilde{\mathbf{h}} - \hat{\mathbf{h}}|}{|\tilde{\mathbf{h}}|} \, . \tag{3.25}$$

Also the error function in Figure 3.6 can be given as,

$$\text{Absolute Relative Error} = \frac{|\tilde{\mathbf{h}} - \bar{\mathbf{h}}|}{|\tilde{\mathbf{h}}|} \, . \tag{3.26}$$

From error graphics, we can say that IRF estimation which is obtained from Equation 3.21, $\bar{\mathbf{h}}$, is more reliable than $\hat{\mathbf{h}}$. We expect this result because approximation in Equation 3.22 makes this error. $\bar{\mathbf{h}}$ or $\tilde{\mathbf{h}}$ can be used for system identification algorithms. Figures are obtained by using the MATLAB routine in Appendix-3. Also we will use the results of this routine for system identification algorithms in next chapters.

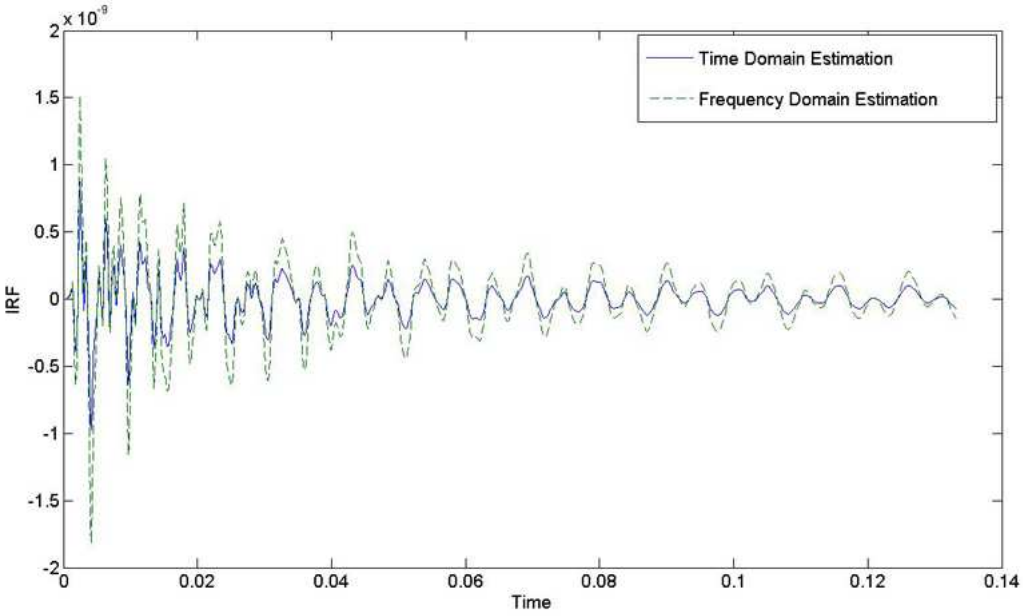


Figure 3.3. Comparison of $\tilde{\mathbf{h}}$ and $\hat{\mathbf{h}}$

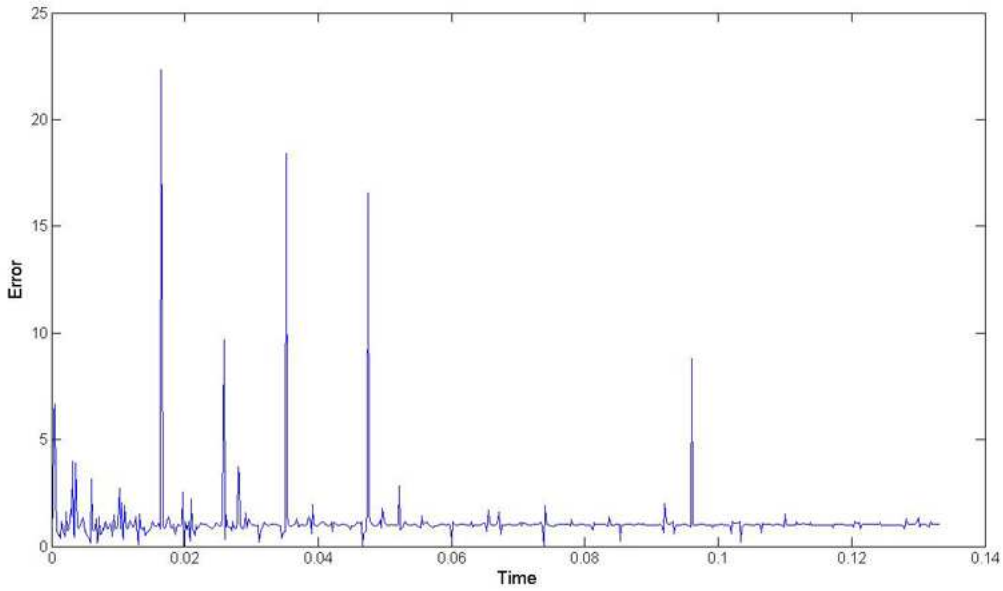


Figure 3.4. Error Graph for $\tilde{\mathbf{h}}$ and $\hat{\mathbf{h}}$

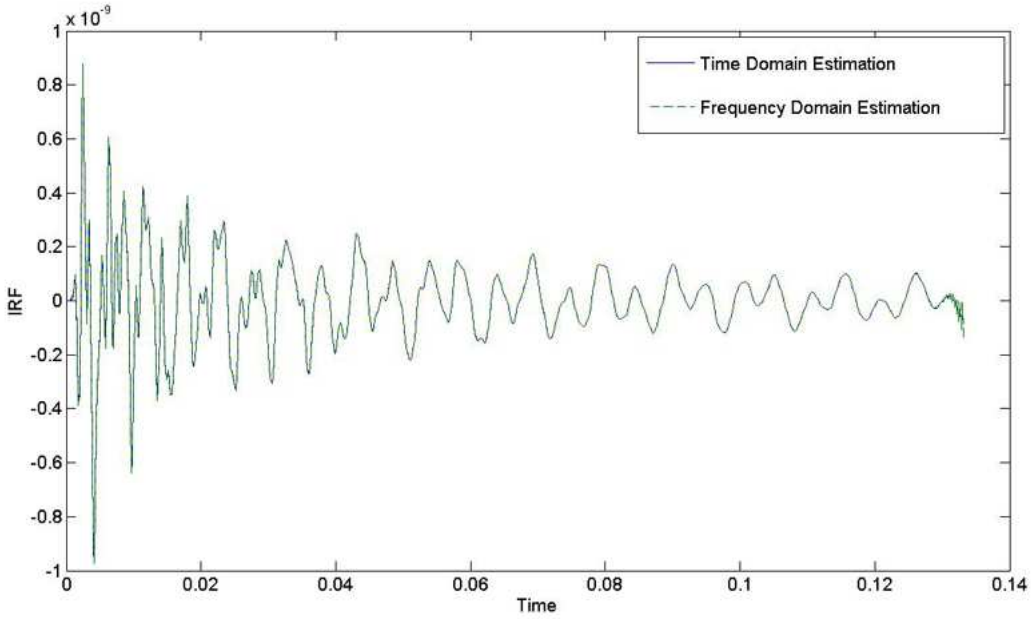


Figure 3.5. Comparison of $\tilde{\mathbf{h}}$ and $\bar{\mathbf{h}}$

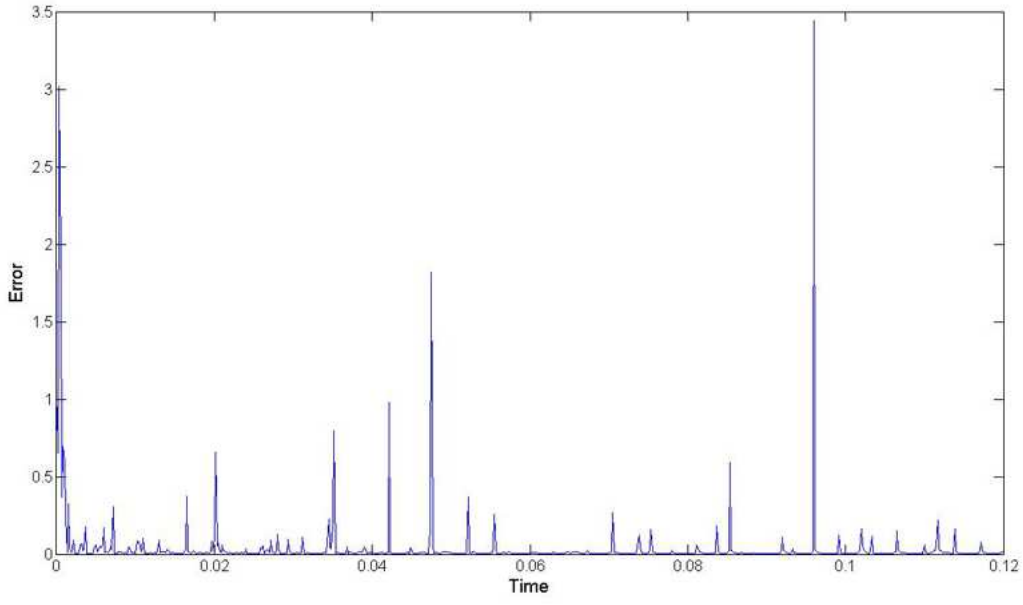


Figure 3.6. Error Graph for $\tilde{\mathbf{h}}$ and $\bar{\mathbf{h}}$

CHAPTER 4

POLYREFERENCE FREQUENCY DOMAIN METHOD

4.1. State Space Representation

In order to understand the polyreference frequency domain method, state-space concept and its applications to vibrational systems must be known. In this section the state-space representation will be introduced and an example about a vibrational system will be given.

State space representation is a mathematical description of a physical model. General equations of this representation are,

$$\dot{\bar{\boldsymbol{x}}} = A\bar{\boldsymbol{x}} + B\boldsymbol{u}\,, \tag{4.1}$$

$$\boldsymbol{y} = C\bar{\boldsymbol{x}} + D\boldsymbol{u}\,, \tag{4.2}$$

where,

$$\bar{\boldsymbol{x}} = \begin{bmatrix} x_1 \\ \cdot \\ \cdot \\ x_n \end{bmatrix} \tag{4.3}$$

is the state vector which includes state variables as velocity or acceleration of a cart,

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ \cdot \\ \cdot \\ y_q \end{bmatrix} \tag{4.4}$$

is the output vector which includes outputs of the system, it can be the measurements in an experiment,

$$\boldsymbol{u} = \begin{bmatrix} u_1 \\ \cdot \\ \cdot \\ u_p \end{bmatrix} \tag{4.5}$$

is the input vector, which contains the inputs that are applied to the system. In this representation A is called state matrix, B is called input matrix, C is called output matrix and D is called feedforward matrix . By using Laplace transform, one can describe the whole system with a rational transfer function. Derivation of the transfer function begins with taking the Laplace transform of the state equation,

$$\dot{\bar{\boldsymbol{x}}} = A\bar{\boldsymbol{x}} + Bu \,, \tag{4.6}$$

$$s\bar{X}(s) = A\bar{X}(s) + BU(s) \,, \tag{4.7}$$

$$s\bar{X}(s) - A\bar{X}(s) = BU(s) \,, \tag{4.8}$$

$$(sI - A)\bar{X}(s) = BU(s) \,, \tag{4.9}$$

$$\bar{X}(s) = (sI - A)^{-1}BU(s) \,. \tag{4.10}$$

Equation 4.10 represents a relation between the input and the state. Another relation between the input and the output can be written as,

$$\boldsymbol{y} = C\bar{\boldsymbol{x}} + D\boldsymbol{u} \,, \tag{4.11}$$

$$Y(s) = C\bar{X}(s) + DU(s) \,, \tag{4.12}$$

$$Y(s) = C(sI - A)^{-1}BU(s) + DU(s) \,, \tag{4.13}$$

$$Y(s) = [C(sI - A)^{-1}B + D]U(s) \,. \tag{4.14}$$

From Equation 4.14, the transfer function of the system can be written as,

$$H(s) = \frac{Y(s)}{U(s)} \,, \tag{4.15}$$

$$= [C(sI - A)^{-1}B + D] \,. \tag{4.16}$$

From Equation 4.16, it can be said that if A, B, C and D matrices is known, the transfer function and IRF can be easily found. To understand the usage of state space representation for vibration systems, we will make an example for 2-DOF system in Figure 4.1. By choosing the system states as the displacement and velocity of each cart,

$$\bar{\boldsymbol{x}} = \begin{bmatrix} x_1 \\ \dot{x}_1 \\ x_2 \\ \dot{x}_2 \end{bmatrix}$$

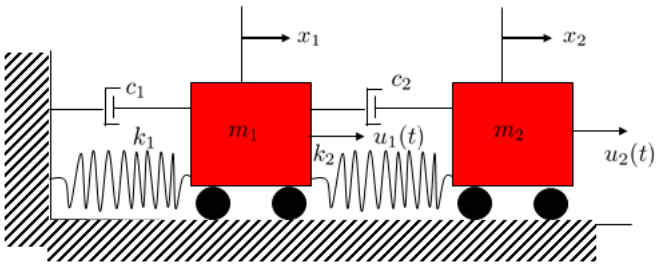


Figure 4.1. 2-DOF System

becomes the state vector and

$$\boldsymbol{u} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

becomes the input vector. Now a state-space representation can be defined by using differential equations of the system. From Figure 4.1, the equations of motion can be written. From the first cart;

$$-m_1\ddot{x}_1 \; -(c_1+c_2)\dot{x}_1 \; +c_2\dot{x}_2 \; -(k_1+k_2)x_1 + k_2x_2 + u_1(t) = 0\,, \tag{4.17}$$

$$\ddot{x}_1 \; = -\frac{(k_1+k_2)}{m_1}x_1 + \frac{(k_2)}{m_1}x_2 - \frac{(c_1+c_2)}{m_1}\dot{x}_1 \; + \frac{c_2}{m_1}\dot{x}_2 \; + \frac{1}{m_1}u_1\,, \tag{4.18}$$

and from the second cart;

$$-m_2\ddot{x}_2 \; + c_2\dot{u}_1 \; - c_2\dot{x}_2 \; + k_2x_1 - k_2x_2 + u_2(t) = 0\,, \tag{4.19}$$

$$\ddot{x}_2 \; = \frac{k_2}{m_2}x_1 - \frac{k_2}{m_2}x_2 + \frac{c_2}{m_2}\dot{x}_1 \; - \frac{c_2}{m_2}\dot{x}_2 \; + \frac{1}{m_2}u_2\,. \tag{4.20}$$

Equations of motion can be combined in state equation as,

$$\dot{\bar{\boldsymbol{x}}} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \ddot{x}_1 \\ \ddot{x}_2 \end{bmatrix} = A\bar{\boldsymbol{x}} + B\boldsymbol{u} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{(k_1+k_2)}{m_1} & \frac{(k_2)}{m_1} & -\frac{(c_1+c_2)}{m_1} & \frac{c_2}{m_1} \\ \frac{k_2}{m_2} & -\frac{k_2}{m_2} & \frac{c_2}{m_2} & -\frac{c_2}{m_2} \end{bmatrix} \bar{\boldsymbol{x}} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{1}{m_1} & 0 \\ 0 & \frac{1}{m_2} \end{bmatrix} \boldsymbol{u}\,. \tag{4.21}$$

The output equation will be written according to the measurements. In our experiment the acceleration is measured, so we must write the output equation for acceleration measurement.

$$\boldsymbol{y} = \begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{bmatrix} = C\bar{\boldsymbol{x}} + D\boldsymbol{u} = \begin{bmatrix} -\frac{(k_1+k_2)}{m_1} & \frac{(k_2)}{m_1} & -\frac{(c_1+c_2)}{m_1} & \frac{c_2}{m_1} \\ \frac{k_2}{m_2} & -\frac{k_2}{m_2} & \frac{c_2}{m_2} & -\frac{c_2}{m_2} \end{bmatrix} \bar{\boldsymbol{x}} + \begin{bmatrix} \frac{1}{m_1} & 0 \\ 0 & \frac{1}{m_2} \end{bmatrix} \boldsymbol{u}$$

(4.22)

If the displacement response is measured instead of acceleration, output equation could be written as,

$$\boldsymbol{y} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = C\bar{\boldsymbol{x}} + D\boldsymbol{u} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \bar{\boldsymbol{x}} \,. \tag{4.23}$$

Many advantages of state space representation is used in this study. This representation will be used frequently in the Frequency Domain Algorithm section. Once the A,B, C and D matrices is known, it is easy to find transfer function and IRF of the system. By using these matrices, system's response to any input can be found in time domain or frequency domain. Furthermore, this representation is useful for constructing the mathematical model in MATLAB environment. As seen from the MATLAB programs in the Appendices, the artificial mathematical models is created easily and by using these models the modal analysis algorithm is verified easily.

4.2. Frequency Domain Algorithm

In first chapter, it was shown that a linear mechanical system can be defined by

$$M\ddot{\boldsymbol{x}}(t) \; + C\dot{\boldsymbol{x}}(t) \; + K\boldsymbol{x}(t) = \boldsymbol{u}(t) \,, \tag{4.24}$$

where M, C and K are the mass, damping and stiffness matrices. For a time invariant system , these matrices are constant and real. Vector x represents the displacement Then $\dot{x}(t)$ and $\ddot{x}(t)$ becomes velocity and acceleration. Transfer function of the system can be written as,

$$H(s) = \frac{X(s)}{U(s)} = \frac{output}{input} \,. \tag{4.25}$$

If the external input, u, is a dirac delta function, laplace transform of input function will be unity,

$$U(s) = 1 \,, \tag{4.26}$$

then displacement response of the cart will be equal to the IRF.

$$H(jw) = X(jw) \Longleftrightarrow H(s) = X(s) \Longleftrightarrow \boldsymbol{h}(t) = \boldsymbol{x}(t) \tag{4.27}$$

The Equation 4.24 can be written as,

$$M\ddot{\boldsymbol{h}}(t) + C\dot{\boldsymbol{h}}(t) + K\boldsymbol{h}(t) = \boldsymbol{u}_\delta(t)\,. \tag{4.28}$$

Since M is a non-singular matrix, by multiplying with M^{-1} each side of the Equation 4.28,

$$M^{-1}M\ddot{\boldsymbol{h}}(t) + M^{-1}C\dot{\boldsymbol{h}}(t) + M^{-1}K\boldsymbol{h}(t) = M^{-1}\boldsymbol{u}_\delta(t)\,, \tag{4.29}$$

$$\ddot{\boldsymbol{h}}(t) + M^{-1}C\dot{\boldsymbol{h}}(t) + M^{-1}K\boldsymbol{h}(t) = M^{-1}\boldsymbol{u}_\delta(t)\,. \tag{4.30}$$

By setting $A_0 = M^{-1}K$, $A_1 = M^{-1}C$ and $B_0 = M^{-1}$, Equation 4.30 can be written as,

$$\ddot{\boldsymbol{h}}(t) = -A_1\dot{\boldsymbol{h}}(t) - A_0\boldsymbol{h}(t) + B_0\boldsymbol{u}_\delta(t)\,. \tag{4.31}$$

If the states of the system is chosen as

$$\bar{\boldsymbol{x}} = \begin{bmatrix} \dot{\boldsymbol{h}}(t) \\ \boldsymbol{h}(t) \end{bmatrix}\,, \tag{4.32}$$

then a state-space representation can be written for our system. It is important to note that $\bar{\boldsymbol{x}}$ represents the state vector and x represents the displacement response, they are different from each other.

$$\dot{\bar{\boldsymbol{x}}} = \begin{bmatrix} \ddot{\boldsymbol{h}}(t) \\ \dot{\boldsymbol{h}}(t) \end{bmatrix} = \begin{bmatrix} -A_1 & -A_0 \\ I & 0 \end{bmatrix}\bar{\boldsymbol{x}} + \begin{bmatrix} B_0 \\ 0 \end{bmatrix}\boldsymbol{u} \tag{4.33}$$

Equation 4.33 is the state equation. In this equation, vector x represents the states and vector u represents the inputs. To complete the state space representation, output must be written. The output can be taken as the displacement of the structure.

$$\boldsymbol{y} = \boldsymbol{h(t)} = \begin{bmatrix} 0 & I \end{bmatrix}\bar{\boldsymbol{x}} \tag{4.34}$$

Equation 4.34 is also called the observation equation. The state space representation of the system defined in Equation 4.24 is obtained as

$$\begin{aligned} \dot{\bar{\boldsymbol{x}}} &= A\bar{\boldsymbol{x}} + B\boldsymbol{u} \\ \boldsymbol{y} &= C\bar{\boldsymbol{x}}\,, \end{aligned} \tag{4.35}$$

where A, B and C are given in Equations 4.33 and 4.34. Transfer function for the mechanical system can be written by using Equation 4.35 ,

$$H(s) = C[sI - A]^{-1}B \Longleftrightarrow h(t) = Ce^{At}B\,. \tag{4.36}$$

The input-output relationship for a linear time-invariant system can be written as,

$$Y(s) = H(s)U(s)\ , \tag{4.37}$$

$$\boldsymbol{y(t)} = \boldsymbol{h(t)} * \boldsymbol{u(t)}\ , \tag{4.38}$$

where $y(t)$ is output, $u(t)$ is input and $h(t)$ is the impulse response, note that $*$ is the convolution operator and $Y(s) = \mathcal{L}\{\boldsymbol{y}(t)\}$, $H(s) = \mathcal{L}\{\boldsymbol{h}(t)\}$ and $U(s) = \mathcal{L}\{\boldsymbol{u}(t)\}$. Output of the system can be written as,

$$\boldsymbol{y(t)} = Ce^{At}B\boldsymbol{u(t)}\ . \tag{4.39}$$

By taking the derivative of Equation 4.39, $\dot{\boldsymbol{y}}(t)$ and $\ddot{\boldsymbol{y}}(t)$ can be derived easily.

$$\dot{\boldsymbol{y}}(t) = CAe^{At}B\boldsymbol{u}(t) + Ce^{At}B\dot{\boldsymbol{u}}(t) \tag{4.40}$$

$$\mathcal{L}\{\dot{y}(t)\} = s\boldsymbol{Y}(s) - \boldsymbol{y}(0) \tag{4.41}$$

$$\ddot{\boldsymbol{y}}(t) = CA^2e^{At}B\boldsymbol{u}(t) + 2CAe^{At}B\dot{\boldsymbol{u}}(t) + Ce^{At}B\ddot{\boldsymbol{u}}(t) \tag{4.42}$$

$$\mathcal{L}\{\ddot{\boldsymbol{y}}(t)\} = s^2Y(s) - s\boldsymbol{y}(0) - \dot{\boldsymbol{y}}(0) \tag{4.43}$$

By applying Laplace transform to the Equation 4.31,

$$\mathcal{L}\{\ddot{\boldsymbol{h}}(t)\} + A_1\mathcal{L}\{\dot{\boldsymbol{h}}(t)\} + A_0\mathcal{L}\{\boldsymbol{h}(t)\} = M^{-1}\ . \tag{4.44}$$

To find $\mathcal{L}\{\dot{h}(t)\}$ and $\mathcal{L}\{\ddot{h}(t)\}$ in Equation 4.44, the differentiation properties of Laplace transform can be used.

$$\frac{df(t)}{dt} \iff sF(s) - f(0^-) \tag{4.45}$$

$$\frac{d^2f(t)}{dt^2} \iff s^2F(s) - sf(0^-) - \dot{f}(0^-) \tag{4.46}$$

By setting $\mathcal{L}\{h(t)\} = H(s)$ and applying the Laplace properties in Equation 4.45 and 4.46 to Equation 4.36,

$$\mathcal{L}\{\dot{\boldsymbol{h}}(t)\} = sH(s) - \boldsymbol{h}(0^-) = sH(s) - CB\ , \tag{4.47}$$

$$\mathcal{L}\{\ddot{\boldsymbol{h}}(t)\} = s^2H(s) - s\boldsymbol{h}(0^-) - \dot{\boldsymbol{h}}(0^-) = s^2H(s) - sCB - CAB\ . \tag{4.48}$$

Lets consider the input-output relation of the mechanical system,

$$\ddot{\boldsymbol{y}}(t) + A_1\dot{\boldsymbol{y}}(t) + A_0\boldsymbol{y}(t) = M^{-1}\boldsymbol{u}(t) \tag{4.49}$$

$$[\ s^2Y(s) - s\boldsymbol{y}(0) - \dot{\boldsymbol{y}}(0)\ ] + A_1[\ sY(s) - \boldsymbol{y}(0)] + A_0Y(s) = M^{-1}\boldsymbol{U}(s) \tag{4.50}$$

$y(0)$ and $\dot{y}(0)$ must be evaluated to simplify the Equation 4.50 and to obtain the *identification equation*.

$$\boldsymbol{y}(0) = [Ce^{At}B\boldsymbol{u}(t)]_{t=0} = CB\boldsymbol{u}(0) = 0\,, \tag{4.51}$$

since $CB = 0$.

$$\dot{\boldsymbol{y}}(0) \;= [\; CAe^{At}B\boldsymbol{u}(t) + Ce^{At}B\dot{\boldsymbol{u}}(t)\;]_{t=0} \tag{4.52}$$

In Equation 4.52 the term $[\; Ce^{At}B\dot{\boldsymbol{u}}(t)\;]_{t=0}$ is taken as zero in the original article (Lembregts and Leuridan 1990). But we found that that term is not zero, this error affects the *identification equation*. To find the value of this term, a property and its proof is given,

Property :

$$e^{at}\delta(t)' = -a\delta(t) + \delta(t)' \tag{4.53}$$

$$e^{At}\delta(t)' = -A\delta(t) + I\delta(t)' \tag{4.54}$$

Proof :

$$\int_{-\infty}^{+\infty} f(t)\delta'(t)dt = -\int_{-\infty}^{+\infty} f'(t)\delta(t)dt = -f'(0) \tag{4.55}$$

$$\begin{aligned}
\int_{-\infty}^{+\infty} \phi(t)[e^{at}\delta'(t)]dt &= \int_{-\infty}^{+\infty} [\phi(t)e^{at}]\delta'(t)dt && (4.56)\\
&= -[\phi(t)ae^{at} + \phi(t)'e^{at}]_{t=0} && (4.57)\\
&= -[\phi(0)a + \phi(0)'] && (4.58)\\
&= -\int a\phi(t)\delta(t)dt + \int \phi(t)\delta(t)'dt && (4.59)\\
&= \int \phi(t)[-a\delta(t) + \delta(t)']dt && (4.60)
\end{aligned}$$

The wrong term can be written as,

$$\begin{aligned}
[\; Ce^{At}B\dot{\boldsymbol{u}}(t)\;]_{t=0} &= [\; -CAB\boldsymbol{u}(0) + CB\dot{\boldsymbol{u}}(t)\;]_{t=0}\\
&= -CAB\boldsymbol{u}(0)\,, && (4.61)
\end{aligned}$$

since $CB = 0$. Then by putting this result to Equation 4.52,

$$\begin{aligned}
\dot{\boldsymbol{y}}(0) &= CAB\boldsymbol{u}(0) - CAB\boldsymbol{u}(0)\\
&= 0\,. && (4.62)
\end{aligned}$$

Equation 4.50 can be written as,

$$s^2Y(s) + A_1sY(s) + A_0Y(s) = M^{-1}U(s)\,, \tag{4.63}$$

$$[\,s^2I + A_1s + A_0]Y(s) = M^{-1}U(s)\,. \tag{4.64}$$

When input is dirac delta function, $u(t) = \delta(t)$, its laplace transform becomes $U(s) = 1$ and output will be the impulse response of the linear system.

$$[\,s^2I + A_1s + A_0]H(s) = M^{-1} \tag{4.65}$$

In our experiment, the acceleration response of the system can be measured only, so the system must be identified according to the acceleration measurements. Equation 4.65 must be written according to the acceleration transfer function, $H_a(s)$. It was shown that $\ddot{h}(t)$ is the acceleration response, since h(t) is the displacement response. Let $\mathcal{L}\{\ddot{h}(t)\} = H_a(s) = s^2H(s)$, then

$$H(s) = H_a(s)/s^2\,, \tag{4.66}$$

$$\mathcal{L}\{\dot{h}(t)\} = sH(s) = H_a(s)/s\,. \tag{4.67}$$

By putting Equations 4.66 and 4.67 into equation 4.65,

$$H_a(s) + A_1\frac{H_a(s)}{s} + A_0\frac{H_a(s)}{s^2} = M^{-1}\,, \tag{4.68}$$

$$s^2H_a(s) + A_1sH_a(s) + A_0H_a(s) = s^2M^{-1}\,, \tag{4.69}$$

$$[\,s^2 + A_1s + A_0]H_a(s) = s^2M^{-1}\,. \tag{4.70}$$

By setting $B_0 = M^{-1}$, the equation for identification can be completed as,

$$[s^2I + A_1s + A_0]H_a(s) = s^2B_0\,. \tag{4.71}$$

There are 3 unknowns in Equation 4.71, A_0, A_1 and B_0. To find these unknowns, least square solution methods will be used. The Equation 4.71 can be written in frequency domain, by setting $s = jw$,

$$[(jw)^2I + A_1jw + A_0]H_a(jw) = (jw)^2B_0\,. \tag{4.72}$$

where j is the square root of -1 and w is the angular frequency. For $s = jw_1, jw_2, ..., jw_m$ one can write that,

$$\begin{aligned}
[\,(jw_1)^2 + A_1 jw_1 + A_0]H_a(jw_1) &= (jw_1)^2 B_0 \\
[\,(jw_2)^2 + A_1 jw_2 + A_0]H_a(jw_2) &= (jw_2)^2 B_0 \\
[\,(jw_3)^2 + A_1 jw_3 + A_0]H_a(jw_3) &= (jw_3)^2 B_0 \\
&\vdots
\end{aligned} \tag{4.73}$$

Equation set 4.73 can be combined in a matrix notation.

$$\begin{bmatrix} jwH_a(jw) & H_a(jw) & -(jw)^2 \end{bmatrix} \begin{bmatrix} A_1 \\ A_0 \\ B_0 \end{bmatrix} = -(jw)^2 H_a(jw) \tag{4.74}$$

Consequently, the matrix Equation 4.74 can be written as,

$$FG = H\ . \tag{4.75}$$

Equation 4.74 is the last part of the identification. The exact value of F and H matrices can be written easily from acceleration measurements. We want to find G matrix which includes A_1 , A_0 and B_0 . According to acceleration measurements, the row number of F is much more than the column number. These systems are called as overdetermined systems, equation number is much more than the number of unknowns. Generally these systems can be solved by using least-square techniques like QR decomposition or Singular value decomposition. After solving G vector by least-square techniques, M , C and K matrices and other system parameters can be easily calculated.

Derivation of the least-square approximation of the matrix Equation 4.75 is given below. F is a $m \times n$ matrix and $m \geq n$, so the inverse of this matrix can't be taken. By multiplying each side of the Equation 4.75 with F^T,

$$F^T FG = F^T H\ . \tag{4.76}$$

Since F is a $m \times n$ matrix, $F^T F$ is a $m \times m$ matrix, and the inverse of this matrix can be taken, then finding G matrix becomes possible.

$$[F^T F]^{-1}[F^T F]G = [F^T F]^{-1} F^T H \tag{4.77}$$

$$G = [F^T F]^{-1} F^T H \tag{4.78}$$

The term $[F^T F]^{-1} F^T$ is called the "pseudo inverse ". Once matrix G is found, A_1 , A_0 and B_0 can be written easily. It was shown that $B_0 = M^{-1}$, so mass matrix can be found. If mass matrix is known, stiffness and damping matrices can be found from equations $A_0 = M^{-1}K$ and $A_1 = M^{-1}C$.

To verify this method in a computer simulation, a MATLAB script is written. The MATLAB routine in Appendix-A2 takes the Mass, Stiffness, Damping matrices and the model order from user, then it creates an artificial N-degree of freedom vibrational system and its frequency domain response functions by using the state-space representation in Equation 4.33. By using only the FRF's, polyreference frequency domain identification algorithm is used to obtain the damped natural frequencies and the damping ratios of the system.

For an example simulation, mass, stiffness and damping matrices are taken as below.

$$M = \begin{bmatrix} 10 & 0 \\ 0 & 5 \end{bmatrix} \; C = \begin{bmatrix} 12 & -6 \\ -6 & 6 \end{bmatrix} \; K = \begin{bmatrix} 1200 & -600 \\ -600 & 600 \end{bmatrix} . \tag{4.79}$$

FRF's of this system can be obtained as in Figure 4.2 and 4.3.

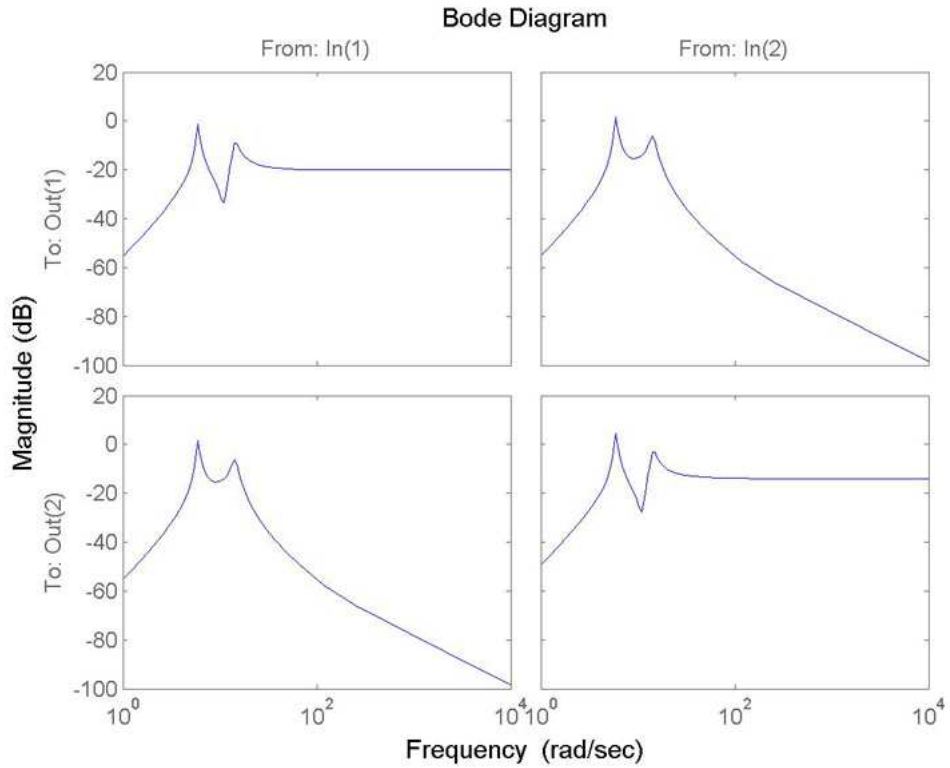


Figure 4.2. Magnitude of the FRF's of 2-DOF System

By using the frequency response datas in Figure 4.2 and 4.3, the parameters of the system can be extracted from PFD algorithm. The results of the MATLAB routine is in the Table 4.1. Relative error in this table can be defined as,

$$\text{Absolute Relative Error} = \frac{|f_{exact} - f_{estimated}|}{f_{exact}} \, . \tag{4.80}$$

It can be said that from any FRF, the natural frequencies of the system can be extracted. These results shows that, if the frequency response data of the system is measured clearly, modal parameters of the system can be extracted by using PFD algorithm.

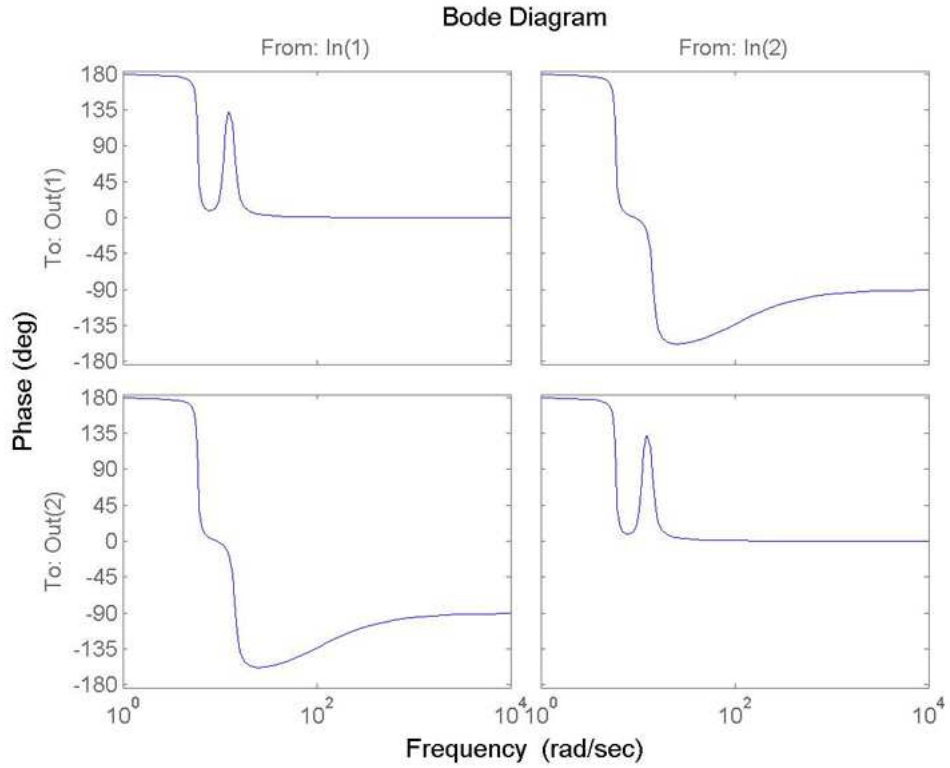


Figure 4.3. Phase of the FRF's of 2-DOF System

Table 4.1. Results of PFD Simulation

Theoretical Natural Frequency	Estimated Natural Frequency	Relative Error
14.31268017327243	14.27598321867646	$0,0025$
5.92850624167794	5.925901040354	$0,00044$

After this verification, we tried to find the modal parameters of our experimental material. First the time domain response and the input of the system is measured, then we took the Fast Fourier Transform (FFT) of them. By dividing the frequency response of the output and input, we found the FRF of the whole system. Then we applied the PFD algorithm for this data. Estimated natural frequencies and damping ratios were not satisfactory. Because of the noise in the measurement, we couldn't successfully implement the PFD algorithm for real measurement data. We decided to search any other advanced algorithm for experimental data.

CHAPTER 5

LEAST SQUARES COMPLEX EXPONENTIAL METHOD

As shown in Equation 1.64, a N-DOF MIMO system's transfer function from i^{th} node to j^{th} node can be written as,

$$H_{ij}(s) = \sum_{r=1}^{N} \left[\frac{A_{ij(r)}}{s - s_r} + \frac{A^*_{ij(r)}}{s - s^*_r} \right] . \tag{5.1}$$

For $A_{ij(r)} = A^*_{ij(r)}$ and $s_r = s^*_r$, transfer function can be written as,

$$H_{ij}(s) = \sum_{r=1}^{2N} \frac{A_{ij(r)}}{s - s_r} . \tag{5.2}$$

Then by using inverse Laplace transform, IRF of the system can be found as,

$$h_{ij}(t) = \sum_{r=1}^{2N} A_{ij(r)} e^{s_r t} \ . \tag{5.3}$$

If this IRF is sampled by equally spaced time intervals $k\Delta$, we will have a discrete data about the impulse response of the system. Actually at the end of the experiment, we have only this discrete data. This data will be used to find the parameters of the system.

$$h(k\Delta) = \sum_{r=1}^{2N} A_{ij(r)} e^{s_r k\Delta} \quad (k = 0, 1, ..., 2N) \tag{5.4}$$

By setting $z_r = e^{s_r \Delta}$, the IRF can be written as,

$$h[k] = h(k\Delta) = \sum_{r=1}^{2N} A_{ij(r)} z_r^k \quad (k = 0, 1, ..., 2N) \ . \tag{5.5}$$

At this point a fitting method called Prony's method will be used for the discrete impulse response data. This method was developed by Gaspard Riche de Prony in 1795. Prony's method extracts valuable information from a uniformly sampled signal and builds a series of damped complex exponentials or sinusoids. This allows for the estimation of frequency, amplitude, phase and damping components of a signal. By using this method the poles of the system can be found. First assume that z_r is the solution of the polynomial,

$$\beta_0 + \beta_1 z_r + \beta_2 z_r^2 + ... + \beta_{2N-1} z_r^{2N-1} + \beta_{2N} z_r^{2N} = 0 \quad \text{(Prony Equation)} . \tag{5.6}$$

The aim is to find all β's and by using them z_r values can be found. By multiplying h_k with β_k in Equation 5.5 and take a sum from $k = 0$ to $2N$,

$$\sum_{k=0}^{2N} \beta_k h[k] = \sum_{k=0}^{2N} \beta_k \sum_{r=1}^{2N} A_{ij(r)} z_r^k \ , \tag{5.7}$$

$$\sum_{k=0}^{2N} \beta_k h[k] = \sum_{r=1}^{2N} A_{ij(r)} \sum_{k_0}^{2N} \beta_k z_r^k \ . \tag{5.8}$$

From the assumption in the Equation 5.6, the term $\sum_{k=0}^{2N} \beta_k z_r^k$ is zero.

$$\sum_{k=0}^{2N} \beta_k[k] = 0 \tag{5.9}$$

$$\beta_0 h[0] + \beta_1 h[1] + \beta_2 h[2] + ... + \beta_{2N-1} h[2N-1] = -\beta_{2N} h[2N] \tag{5.10}$$

By setting $\beta_{2N} = -1$, Equation 5.10 can be written as,

$$\beta_0 h[0] + \beta_1 h[1] + \beta_2 h[2] + ... + \beta_{2N-1} h[2N-1] = h[2N] \ . \tag{5.11}$$

The values of $h[0], h[1], ...$ are known from IRF data. If $4N$ samples are taken from the IRF and the hankel matrix is constructed as,

$$\begin{bmatrix} h[0] & h[1] & h[2] & . & . & . & h[2N-1] \\ h[1] & h[2] & h[3] & . & . & . & h[2N] \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ h[2N-1] & h[2N] & h[2N+1] & . & . & . & h[4N-2] \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ . \\ . \\ . \\ \beta_{2N-1} \end{bmatrix} = \begin{bmatrix} h[2N] \\ h[2N+1] \\ . \\ . \\ . \\ h[4N-1] \end{bmatrix} , \tag{5.12}$$

β values can be found easily by using matrix inversion. Also the number of rows in Equation 5.12 can be increased for a least-square solution. Last matrix equation can be renamed as $H\boldsymbol{\beta} = \boldsymbol{h}$. From least-square approximation, β vector can be found easily. Then z_r values can be found from the Equation 5.6. After that, natural frequencies and damping ratios can be found by using z_r values from,

$$\boldsymbol{\beta} = (H^T H)^{-1} H^T \boldsymbol{h} \ , \tag{5.13}$$

$$w_r = \frac{1}{\Delta} \sqrt{\ln z_r \ln z_r^*} \ , \tag{5.14}$$

$$\zeta_r = \frac{-\ln(z_r z_r^*)}{2 w_r \Delta} \ . \tag{5.15}$$

Consequently, the identification matrix equation must be constructed initially and then this equation must be solved for all β values. By using the polynomial assumption and β values z_r values can be found. Then by using z_r's, natural frequencies and damping factors can be found.

To verify this method in a computer simulation, a MATLAB script is written. The MATLAB routine in Appendix-A3 takes the Mass, Stiffness, Damping matrices and the model order from user, then it creates an artificial N-degree of freedom vibrational system and its time domain IRF's by using the state-space representation in Equation 4.33. By using only the IRF's, the time domain identification algorithm is used to obtain the damped natural frequencies and the damping ratios of the system.

For an example simulation mass, stiffness and damping matrices are taken as,

$$M = \begin{bmatrix} 10 & 0 \\ 0 & 5 \end{bmatrix} \; C = \begin{bmatrix} 12 & -6 \\ -6 & 6 \end{bmatrix} \; K = \begin{bmatrix} 1200 & -600 \\ -600 & 600 \end{bmatrix} . \tag{5.16}$$

Then the IRF's of this 2-DOF system can be obtained as in Figure 5.1. By using the data sets in Figure 5.1, the parameters of the system can be extracted from LSCE algorithm. The results of the MATLAB routine is in the Table 5.1. Relative error in this table can be defined as,

$$\text{Absolute Relative Error} = \frac{|f_{exact} - f_{estimated}|}{f_{exact}} . \tag{5.17}$$

These results shows that, if the impulse response data of the system is measured clearly, the modal parameters of the system can be estimated by using LSCE algorithm.

After this verification, we tried to find the modal parameters of our experimental material. The MATLAB routine in Appendix-5 performs the LSCE algorithm for the data which is obtained from pre-processing part. An overdetermined system is built and a block of natural frequency is found from LSCE algorithm. Natural frequencies which are close to X-Modal Analysis and SAP2000 results are selected as estimated natural frequencies, these are given in Tables 5.2 and 5.3. SAP2000 is a software which makes a finite element analysis for a given structural material's dimensions and parameters. For our experiment, the dimension of the beam is given to this software, then SAP2000 program finds the natural frequency of the experimental structure by utilizing a modal analysis algorithm. X-Modal program applies a modal analysis algorithm and a complex mode indicator function for real measurement data. By using X-Modal program, any measurement

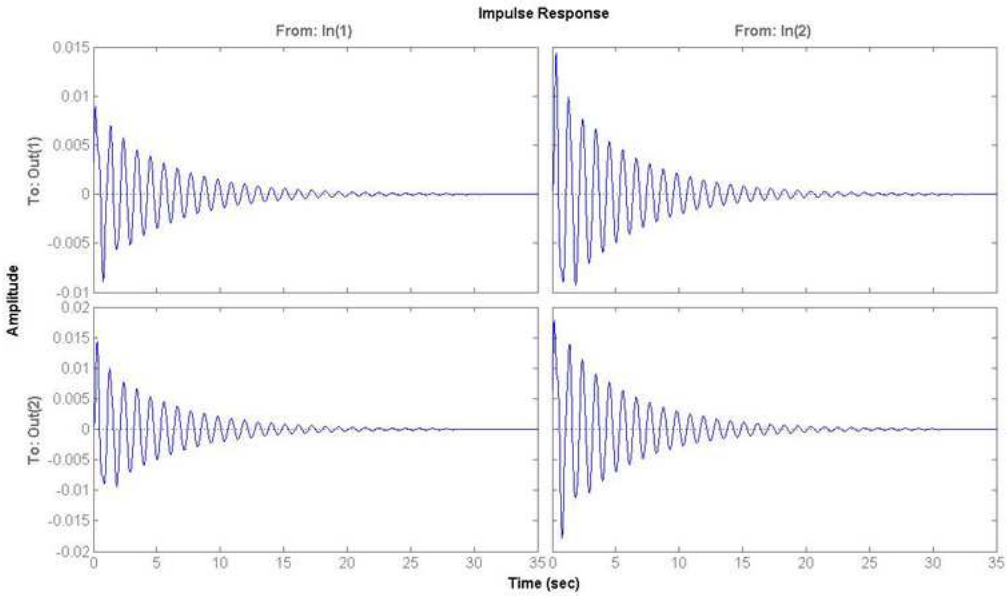


Figure 5.1. IRF's of 2-DOF System

data set can be processed and the natural frequency values of the experimental structure can be found. All X-Modal Analysis and SAP Model results are taken from (Karakan 2008). Three of the estimated natural frequencies is very close to analytical results, it can be said that three of them is estimated succesfully. Lowest frequency couldn't be detected, its relative error is so large. As shown in Tables Tables 5.2 and 5.3, we couldn't successfully run the LSCE algorithm for real measurement data because of the noise. We decided to search any other advanced algorithm for experimental data.

Table 5.1. Results of LSCE Simulation for Artificial Data

IRF	Theoretical Natural Frequency	Estimated Natural Frequency	Relative Error
h_{11}	14.31268017327243	14.31268017326813	3.10^{-13}
h_{11}	5.92850624167794	5.92850624166910	$1,5.10^{-12}$
h_{12}	14.31268017327243	14.31268017326593	4.10^{-13}
h_{12}	5.92850624167794	5.92850624169114	2.10^{-12}
h_{21}	14.31268017327243	14.31268017328763	1.10^{-12}
h_{21}	5.92850624167794	5.92850624166432	2.10^{-12}
h_{22}	14.31268017327243	14.31268017326908	2.10^{-13}
h_{22}	5.92850624167794	5.92850624168014	$3,7.10^{-13}$

Table 5.2. Comparison of X-modal Analysis and LSCE Simulation Results

Modal Analysis Frequency in Hz	Estimated Frequency in Hz	Relative Error(%)
24	30.58	27
85.1	87.78	3.1
192.3	192.37	0.03
341.7	360.07	5.3

Table 5.3. Comparison of SAP2000 Analysis and LSCE Simulation Results

Sap Model Frequency in Hz	Estimated Frequency in Hz	Relative Error(%)
23.5	30.58	30.1
87.7	87.78	0.1
194.5	192.37	1.1
343.7	360.07	4.7

CHAPTER 6

EIGENSYSTEM REALIZATION ALGORITHM

One of the popular modal analysis techniques for civil structures is the eigensystem realization algorithm. In this chapter a summary of this algorithm will be given and after that the verification of the algorithm will be made by a MATLAB simulation. Jer-Nan Juang and Richard Pappa developed ERA at the NASA Langley Research Center in 1985. ERA is a minimum order realization technique and it is an extension of the Ho-Kalman algorithm that uses singular value decomposition technique for denoising. ERA begins with the definition of the Markov parameter of a state-space model. Consider a discrete time state space model such that,

$$\begin{aligned} x[k+1] &= Ax[k] + Bu[k] \\ y[k] &= Cx[k] + Du[k] \,. \end{aligned} \tag{6.1}$$

To obtain the impulse response of the system, the input must be an impulse, it can be a dirac delta. If $u[k] = \delta[k]$, the output $y[k]$ will be the impulse response of the system $h[k]$. If we start to write the $h[k]$'s beginning from the zero-state by assuming the system is initially at rest,

$$\begin{aligned} h[0] &= Cx[0] + D\delta[0] = D \\ x[1] &= Ax[0] + B\delta[0] = B \\ h[1] &= Cx[1] + D\delta[1] = CB \\ x[2] &= Ax[1] + B\delta[1] = AB \\ h[2] &= Cx[2] + D\delta[2] = CAB \\ h[3] &= CA^2B \\ &\vdots \\ h[n] &= CA^{n-1}B \end{aligned} \tag{6.2}$$

The impulse response of the system can be written as,

$$h[n] = \begin{cases} D, & n = 0 \\ CA^{n-1}B, & n > 0. \end{cases} . \tag{6.3}$$

The term of $CA^{n-1}B$ is called the Markov parameter of the system. By using these parameters one can define the impulse response of the system.

$$Y[k] = CA^{k-1}B \tag{6.4}$$

Consequently, the identification problem is: Given values of $Y[k]$'s, construct the constant matrices A, B and C to identify the system. The algorithm begins by constructing a rxs generalized Hankel matrix,

$$H_{rs}[k] = \begin{bmatrix} Y[k] & Y[k+t_1] & . & . & . & . & Y[k+t_{s-1}] \\ Y[j_1+k] & Y[j_1+k+t_1] & . & . & . & . & Y[j_1+k+t_{s-1}] \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ Y[j_{r-1}+k] & Y[j_{r-1}+k+t_1] & . & . & . & . & Y[j_{r-1}+k+t_{s-1}] \end{bmatrix} . \tag{6.5}$$

where $j_i (i = 1, ..., r-1)$ and $t_i (i = 1, ..., s-1)$ are arbitrary integers. Controllability and observability matrices of a state-space system can be written by using A, B and C matrices as in,

$$V_r = \begin{bmatrix} C \\ CA^{j_1} \\ CA^{j_2} \\ . \\ . \\ CA^{j_{r-1}} \end{bmatrix} , \tag{6.6}$$

$$W_s = [B, A^{t_1}B, A^{t_2}B, ..., A^{t_{s-1}}B] , \tag{6.7}$$

where V_r is the observability matrix and W_s is the controllability matrix. The Hankel matrix in Equation 6.5 can be written by means of observability and controllability matrices as,

$$H_{rs}[k] = V_r A^k W_s . \tag{6.8}$$

Assume that there exists a matrix $H^{\#}$ satisfying the relation,

$$W_s H^{\#} V_r = I_n \ , \tag{6.9}$$

where I_n is an identity matrix of order n. From Equations 6.8 and 6.9,

$$H_{rs}[0] H^{\#} H_{rs}[0] = V_r W_s H^{\#} V_r W_s = V_r W_s = H_{rs}[0] \ . \tag{6.10}$$

It can be said that $H^{\#}$ is the pseudoinverse of the matrix $H_{rs}[0]$.

$$H^{\#} = [[H_{rs}[0]]^T [H_{rs}[0]]]^{-1} [H_{rs}[0]]^T \tag{6.11}$$

Now a general solution for $H^{\#}$ will be given. By writing the singular value decomposition of $H_{rs}[0]$ as,

$$H_{rs}[0] = PDQ^T \ , \tag{6.12}$$

where D is a diagonal matrix which consists the singular values of $H_{rs}[0]$. By combining the matrices P and D,

$$H_{rs}[0] = PDQ^T = [PD][Q^T] = P_d Q^T \ . \tag{6.13}$$

The matrices P_d, Q^T, W_s and V_r^T has rank and row number n. By using Equation 6.8 with $k = 0$,

$$V_r W_s = H_{rs}[0] = P_d Q^T \ . \tag{6.14}$$

Multiplying left side by P_d^T and solving for Q^T,

$$TW_s = (P_d^T P_d)^{-1} P_d^T V_r W_s = Q^T \ . \tag{6.15}$$

Consider a U matrix such as,

$$U = W_s Q (Q^T Q)^{-1} = W_s Q \ . \tag{6.16}$$

It can be seen that $TU = I$ from Equations 6.15 and 6.16. T is nonsingular. By using the relation $TU = I$ and Equations 6.15 and 6.16,

$$W_s [Q (P_d^T P_d)^{-1} P_d^T] V_r = I_n \ . \tag{6.17}$$

Now matrix $H^{\#}$ can be extracted as,

$$H^{\#} = [Q][(P_d^T P_d)^{-1} P_d^T] = [Q][D^{-1} P^T] = Q P_d^{\#} \ . \tag{6.18}$$

By defining $E_p^T = [I_p, 0_p, ..., 0_p]$ and $E_m^T = [I_m, 0_m, ..., 0_m]$ where 0_p is a null matrix of order p and I_p is an identity matrix of order p and by using Equations 6.8, 6.9 and 6.18, a minimum order realization can be obtained.

$$\begin{aligned} Y[k+1] &= E_p^T H_{rs}[k] E_m = E_p^T V_r A^k W_s E_m \\ &= E_p^T V_r W_s H^{\#} V_r A^k W_s H^{\#} V_r W_s E_m \\ &= E_p^T H_{rs}[0] Q P_d^{\#} V_r A^k W_s Q P_d^{\#} H_{rs}[0] E_m \\ &= E_p^T H_{rs}[0] Q [P_d^{\#} H_{rs}[1] Q]^k P_d^{\#} H_{rs}[0] E_m \\ &= E_p^T P_d [P_d^{\#} H_{rs}[1] Q]^k Q^T E_m \\ &= E_p^T P D^{1/2} [D^{-1/2} P^T H_{rs}[1] Q D^{-1/2}]^k D^{1/2} Q^T E_m \end{aligned}$$

The minimum order realization of the matrices $[A, B, C]$ is $[D^{-1/2} P^T H_{rs}[1] Q D^{-1/2}, D^{1/2} Q^T E_m, E_p^T P D^{1/2}]$. The system in Equation 6.1 can be written with these parameters,

$$\begin{aligned} x[k+1] &= D^{-1/2} P^T H_{rs}[1] Q D^{-1/2} x[k] + D^{1/2} Q^T E_m u[k] \\ y[k] &= E_p^T P D^{1/2} x[k] \,, \end{aligned} \tag{6.19}$$

where

$$x[k] = W_s Q D^{-1/2} \overline{x}[k] \,. \tag{6.20}$$

Consequently, we can say that a discrete time LTI dynamical system can be identified from the measurement functions if the system is controllable and observable. Matrices V_r and W_s must be full rank for controllability and observability of the system. Natural frequencies and the damping ratios of the vibrational system can be found from the eigenvalues of the realization of the matrix A. Remember that the imaginary part of the eigenvalues of matrix A gives us the natural frequencies of the system.

To verify ERA in a computer simulation, a MATLAB script is written. The MATLAB routine in Appendix-A4 takes the Mass, Stiffness, Damping matrices and the model order from user, then it creates an artificial N-degree of freedom vibrational system and its time domain IRF's by using the state-space representation in Equation 4.33. By using only the impulse response functions, ERA is used to obtain the damped natural frequencies and the damping ratios of the system.

For an example simulation mass, stiffness and damping matrices are taken as,

$$M = \begin{bmatrix} 10 & 0 \\ 0 & 5 \end{bmatrix} \; C = \begin{bmatrix} 12 & -6 \\ -6 & 6 \end{bmatrix} \; K = \begin{bmatrix} 1200 & -600 \\ -600 & 600 \end{bmatrix} . \tag{6.21}$$

IRF's of this 2-DOF system is given in Figure 5.1. By using the data sets in Figure 5.1 only, the parameters of the system can be extracted from ERA. The results of the MATLAB routine is in the Table 6.1. Relative error in this table can be defined as,

$$\text{Absolute Relative Error} = \frac{|f_{exact} - f_{estimated}|}{f_{exact}} . \tag{6.22}$$

As shown, the natural frequencies and damping ratios of any system can be extracted from any IRF. These results shows that, if we can measure the impulse response data of the system clearly, we can extract the modal parameters of the system by using ERA algorithm.

Table 6.1. Results of ERA Simulation for Artificial Data

IRF	Theoretical Natural Frequency	Estimated Natural Frequency	Relative Error
h_{11}	14.31268017327243	14.27598321866659	25.10^{-4}
h_{11}	5.92850624167794	5.92590104035906	4.10^{-4}

Table 6.2. Comparison of X-modal Analysis and ERA Simulation Results

Modal Analysis Frequency in Hz	Estimated Frequency in Hz	Relative Error(%)
24	25.11	4.6
85.1	85.55	0.5
192.3	194.33	0.03
341.7	346.65	1.0

After this verification, we apply the ERA algorithm for experimental data in a MATLAB simulation. An overdetermined system is built and a block of natural frequency is found from LSCE algorithm. Some frequencies are eliminated by using a consistent mode indicator function(Pappa and Elliott 1992), finally 4 natural frequency is found. As shown in Tables 6.2 and 6.3, estimated natural frequencies were so close to analytical results. In spite of the measurement noise, performance of the ERA is satisfactory. It can be said that ERA is more effective than LSCE or PFD for noisy systems.

Table 6.3. Comparison of SAP2000 Analysis and ERA Simulation Results

Sap Model Frequency in Hz	Estimated Frequency in Hz	Relative Error(%)
23.5	25.11	6.8
87.7	85.55	2.4
194.5	194.33	0.08
343.7	346.65	0.86

CHAPTER 7

STATE ESTIMATION FOR DYNAMICAL SYSTEMS

In this chapter, we study the state estimation problem of a linear dynamical system with a Kalman filter. In addition, we integrated ERA with Kalman filter to find more accurate estimates. As shown in Figure 7.1, ERA supplies an initial estimate for this method. The initial estimate of state space matrices found from ERA will be used in this algorithm to construct a successful mathematical model. Now lets begin with introducing the state estimation with Kalman filter.

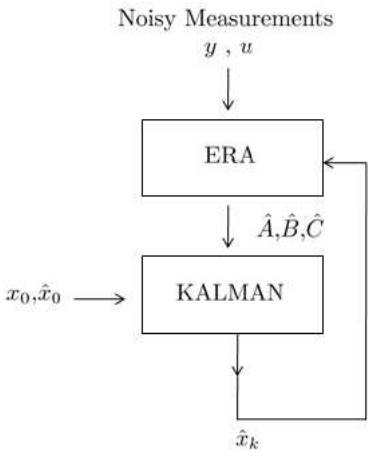


Figure 7.1. Usage of ERA and Kalman Filter

A discrete time LTI system can be defined by state space equations as,

$$x[k] = Ax[k-1] + Bu[k-1] + w[k-1]$$
$$y[k] = Cx[k] + v[k]\,, \tag{7.1}$$

where process noise, w, is normally distributed with zero-mean and covariance Q, the measurement noise, v, is normally distributed with zero-mean and covariance R. During the estimation process the term of $\hat{x}[n|m]$ represents the estimate of x at time n, given observations up to and including time m. The filter can be defined mathematically by using two variables, first one is the estimate of state at time k , $\hat{x}[k|k]$, and the second one is the error covariance matrix, $P[k|k]$ which gives us the accuracy of the state estimate

(Bar-Shalom and Kirubarajan 2001). There are 2 steps for filtering process; Prediction and Update. In prediction step, a state from previous time instant is taken and it is used to obtain a present-time state estimate. The mathematical definition of the prediction step is such as,

$$\hat{x}[k|k-1] = A\hat{x}[k-1|k-1] + Bu[k-1]\,, \tag{7.2}$$

$$P[k|k-1] = AP[k-1|k-1]A^T + Q[k-1]\,, \tag{7.3}$$

where $\hat{x}[k|k-1]$ is the predicted state and $P[k|k-1]$ is the predicted state covariance. After the prediction, the estimate of the state must be improved by using the measurement. This step is called as update and it can be defined as,

$$\text{Inn}[k] = y[k] - C\hat{x}[k|k-1]\,, \tag{7.4}$$

$$S[k] = CP[k|k-1]C^T + R[k]\,, \tag{7.5}$$

$$K[k] = P[k|k-1]C[k]^T S[k]^{-1}\,, \tag{7.6}$$

where $\text{Inn}[k]$ is the innovation or measurement residual, $S[k]$ is the innovation covariance and $K[k]$ is the optimal kalman gain. State and the error covariance can be updated by using these parameters.

$$\hat{x}[k|k] = \hat{x}[k|k-1] + K[k]\hat{y}[k] \tag{7.7}$$

$$P[k|k] = [I - K[k]C]P[k|k-1] \tag{7.8}$$

$\hat{x}[k|k]$ is the updated state estimate and $P[k|k]$ is the updated estimate covariance. By using these steps iteratively, the states of a linear dynamical system can be estimated successfully.

An example is done for state estimation with Kalman filter. Consider a SDOF vibrational system which has a mass of $5\,$kg, $80\,$N/m spring constant and $4\,$Ns/m damping coefficient value. By assuming that the system has no process noise, state space equations can be written as

$$\begin{aligned} x[k] &= Ax[k-1] + Bu[k-1] \\ y[k] &= Cx[k] + v[k]\,. \end{aligned} \tag{7.9}$$

To verify Kalman filter in a computer simulation, a MATLAB script is written. The MATLAB routine in Appendix-A5 takes the Mass, Stiffness, Damping matrices and the

model order from user, then it creates an artificial N-degree of freedom vibrational system by using the state-space representation in Equation 4.33. The input, u, is taken as a dirac delta function and the measurement noise, v, is taken as a zero mean, unity variance white noise. By using only the measurement data and initial state values, state estimation of the system is found by Kalman filter.

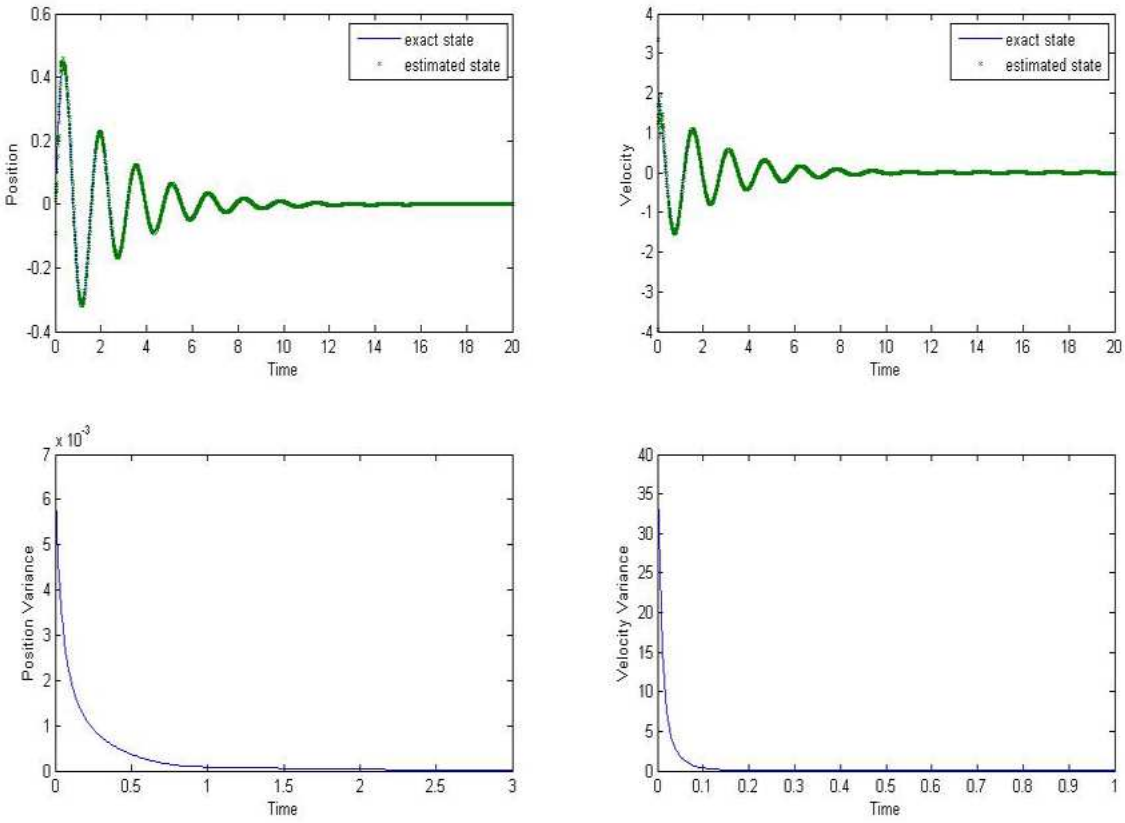


Figure 7.2. State Estimation and Error with SNR=0dB

Position and velocity states of the structure and the estimation error variances is shown in Figure 7.2, 7.3 and 7.4 for different signal-to-noise ratio (SNR) values. SNR is a term that compares the level of a desired signal to the level of background noise. Mathematically, SNR is defined as the power ratio between a signal and the noise,

$$SNR(dB) = 10 \log \left(\frac{P_{signal}}{P_{noise}} \right) , \tag{7.10}$$

where dB is decibel. As shown in Figures 7.2, 7.3 and 7.4, when SNR value gets larger, so that the signal power gets larger, error variances gets smaller. Exact states and estimated states are so close to each other.

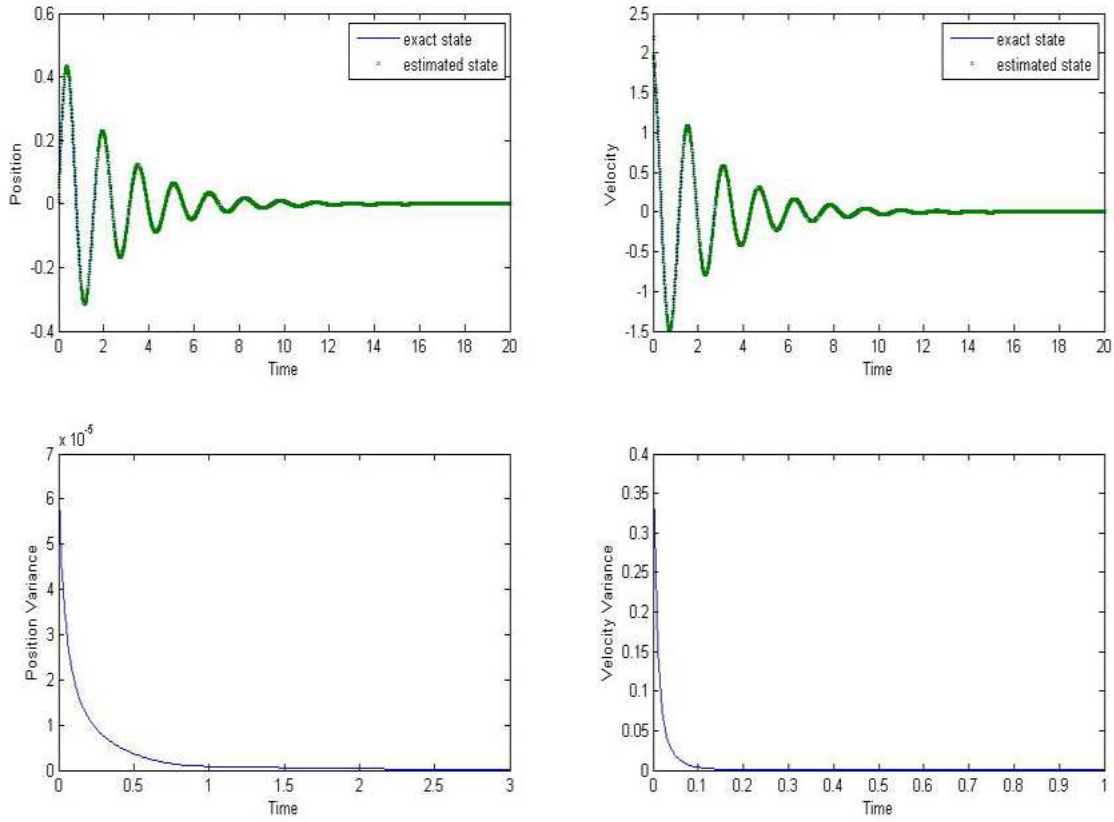


Figure 7.3. State Estimation and Error with SNR=20dB

As comparison graphs of the estimated states and exact states, consistency of the filter is also important. Consistency of a filter is the convergence of the estimate to the true value. Consistency in the mean square sense can be defined as,

$$\lim_{k\to\infty} E[[\hat{x}(k, Y^k) - x_0]^2] = 0\,. \tag{7.11}$$

To decide the consistency of our filter, we made a statistical test. The error function is taken as,

$$\epsilon(k) = \tilde{x}[k|k]^T P[k|k]^{-1} \tilde{x}[k|k], \quad \tilde{x}[k|k] = x[k|k] - \hat{x}[k|k]\,. \tag{7.12}$$

$\epsilon(k)$ is also called Normalized State Estimation Error Squared (NEES) and this variable can be taken as a chi-square random variable. Chi-square Distribution is defined as

$$\epsilon \sim \chi^2_{n_x} \rightarrow \epsilon = \sum_{i=1}^{k} x_i^2\,, \tag{7.13}$$

where x_i's are k independent, normally distributed random variables with mean 0 and

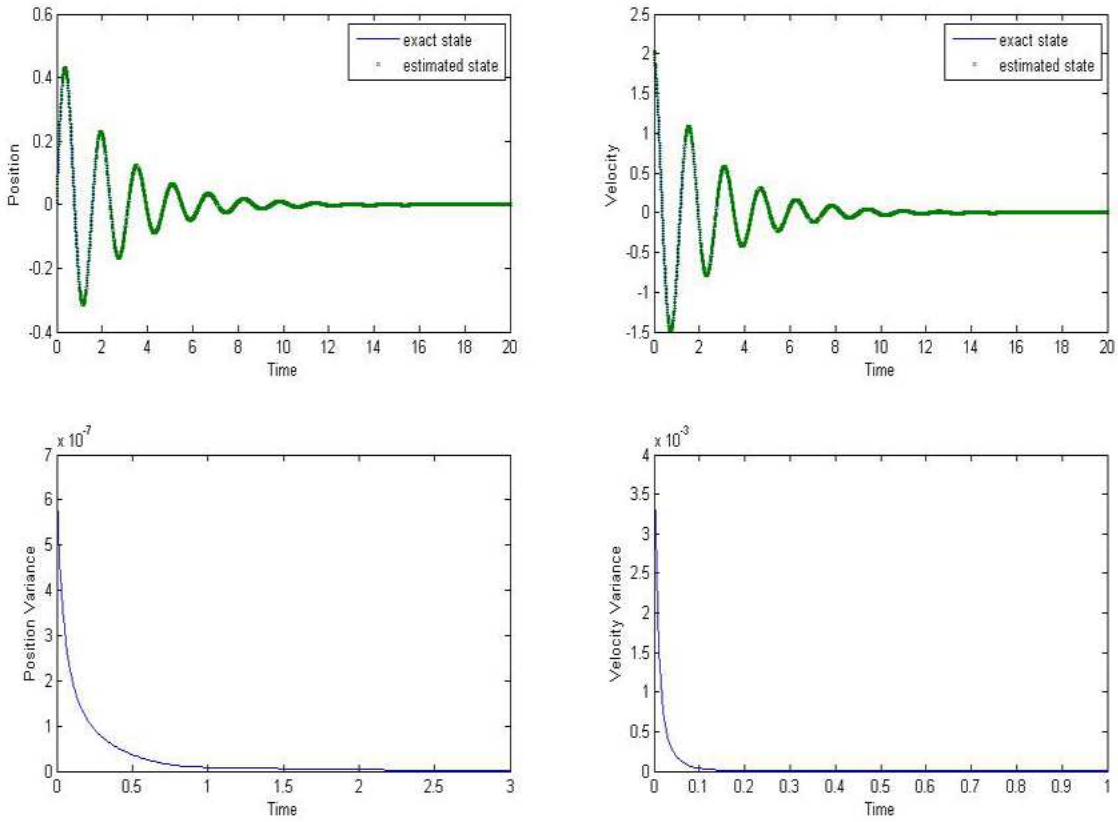


Figure 7.4. State Estimation and Error with SNR=40dB

variance 1. The statistical properties of a chi-square random variable can be shown as,

$$f(x;k) = \left\{ \begin{array}{cc} \frac{1}{2^{k/2}\Gamma(k/2)}\, x^{(k/2)-1} e^{-x/2} & x > 0 \\ 0 & x \leq 0 \end{array} \right\}$$

$$E(X) = k$$

$$Var(X) = 2k\,, \tag{7.14}$$

where $f(x;k)$ is the probability density function, $E(X)$ is the expected value, $Var(X)$ is the variance and k is the degree of freedom of the random variable X. Also its probability density function is shown as in Figure 7.5. Tail probability values for chi-square PDF can be found from a chi-square table. An example table is shown in Figure 7.6. For a single Kalman filter run, given probability value, Q, the one-sided probability region of a 2-DOF chi-square random variable is $[0, \chi_2^2(1-Q)]$, and the two-sided probability region of a 2-DOF chi-square random variable is $[\chi_2^2(\frac{Q}{2}), \chi_2^2(\frac{1-Q}{2})]$. These regions can be used to test the consistency of the filter. If NEES function stays in these regions with probability Q, we can say that filter is consistent. In filtering process, random noise is used for obtaining

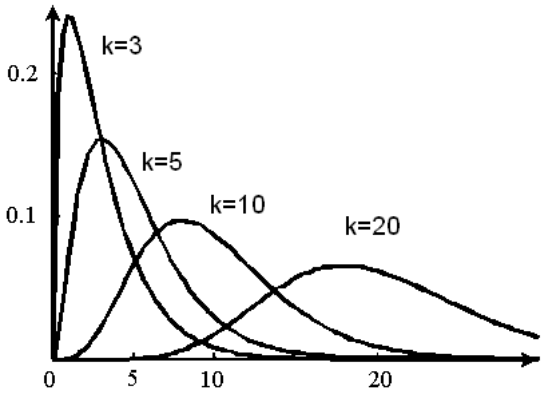
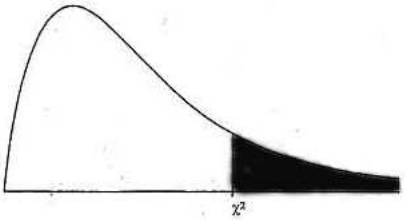


Figure 7.5. Chi-square PDF



	Tail probability p										
df	.25	.20	.15	.10	.05	.025	.02	.01	.005	.0025	.001
1	1.32	1.64	2.07	2.71	3.84	5.02	5.41	6.63	7.88	9.14	10.83
2	2.77	3.22	3.79	4.61	5.99	7.38	7.82	9.21	10.60	11.98	13.82
3	4.11	4.64	5.32	6.25	7.81	9.35	9.84	11.34	12.84	14.32	16.27
4	5.39	5.99	6.74	7.78	9.49	11.14	11.67	13.28	14.86	16.42	18.47
5	6.63	7.29	8.12	9.24	11.07	12.83	13.39	15.09	16.75	18.39	20.51
6	7.84	8.56	9.45	10.64	12.59	14.45	15.03	16.81	18.55	20.25	22.46
7	9.04	9.80	10.75	12.02	14.07	16.01	16.62	18.48	20.28	22.04	24.32
8	10.22	11.03	12.03	13.36	15.51	17.53	18.17	20.09	21.95	23.77	26.12
9	11.39	12.24	13.29	14.68	16.92	19.02	19.68	21.67	23.59	25.46	27.88
10	12.55	13.44	14.53	15.99	18.31	20.48	21.16	23.21	25.19	27.11	29.59

Figure 7.6. Tabulated values for Chi-Square PDF

noisy output data, so noise data changes for every single run. To reduce this error, Monte Carlo simulation method is used to test the consistency again. Monte Carlo Simulation is repeating random sampling to compute any algorithm. It is generally used to examine the performance of an estimation algorithm. For N Monte Carlo simulations, the error function gets smaller and our interval for statistical test gets smaller.

$$\overline{\epsilon}_x = \frac{1}{N}\sum_{i=1}^{N} \epsilon_x^i \quad \rightarrow \quad N\overline{\epsilon}_x \sim \chi^2_{Nn_x} \tag{7.15}$$

where $\overline{\epsilon}_x$ are the sample average NEES from N independent runs. $P = 1 - Q$ two-sided

probability region for $N\overline{\epsilon}_x$ is $[\epsilon_1^{'}, \epsilon_2^{'}]$ where,

$$\epsilon_1^{'} = \chi^2_{Nn_x}\left(\frac{Q}{2}\right) ,$$
$$\epsilon_2^{'} = \chi^2_{Nn_x}\left(1-\frac{Q}{2}\right) . \tag{7.16}$$

So probability region for $\overline{\epsilon}_x$ is,

$$[\epsilon_1, \epsilon_2] = [\epsilon_1^{'}/N, \epsilon_2^{'}/N] . \tag{7.17}$$

To verify the consistency of our Kalman filter in a computer simulation, a MATLAB script is written. In this routine, we run the Kalman filter algorithm for a single-run, 2, 5, 10, 25 and 50 Monte Carlo runs with probability values $Q = 0.05$, $Q = 0.02$ and $Q = 0.01$. For example, $Q = 0.05$ means that, NEES, error function, must be in probability region with 95% probability. NEES graphs are shown in Figure 7.7, 7.8 and 7.9. As shown in these figures, NEES function always stays in the probability regions, so it can be said that our Kalman filter is consistent and we can use it to estimate the states of our system from noisy output data sets.

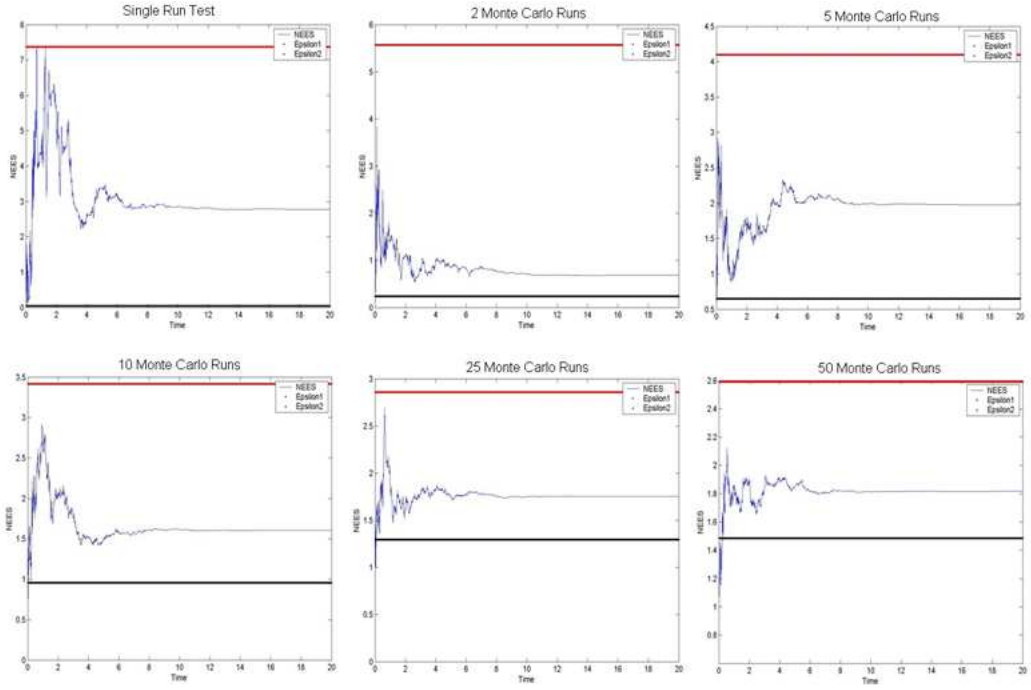


Figure 7.7. NEES with probability Q=0.05

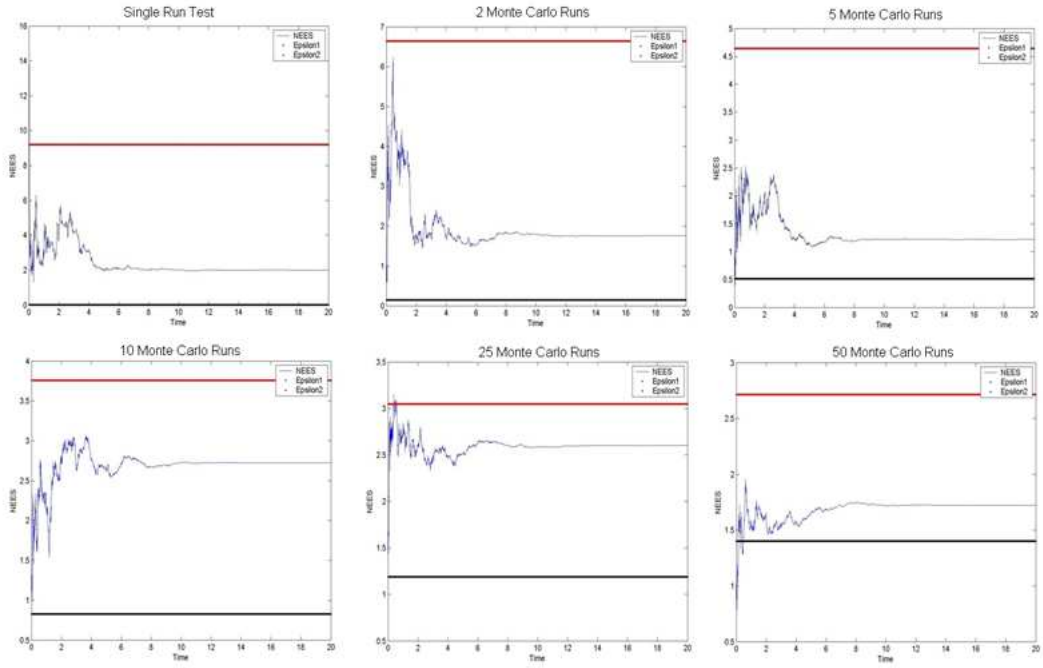


Figure 7.8. NEES with probability Q=0.02

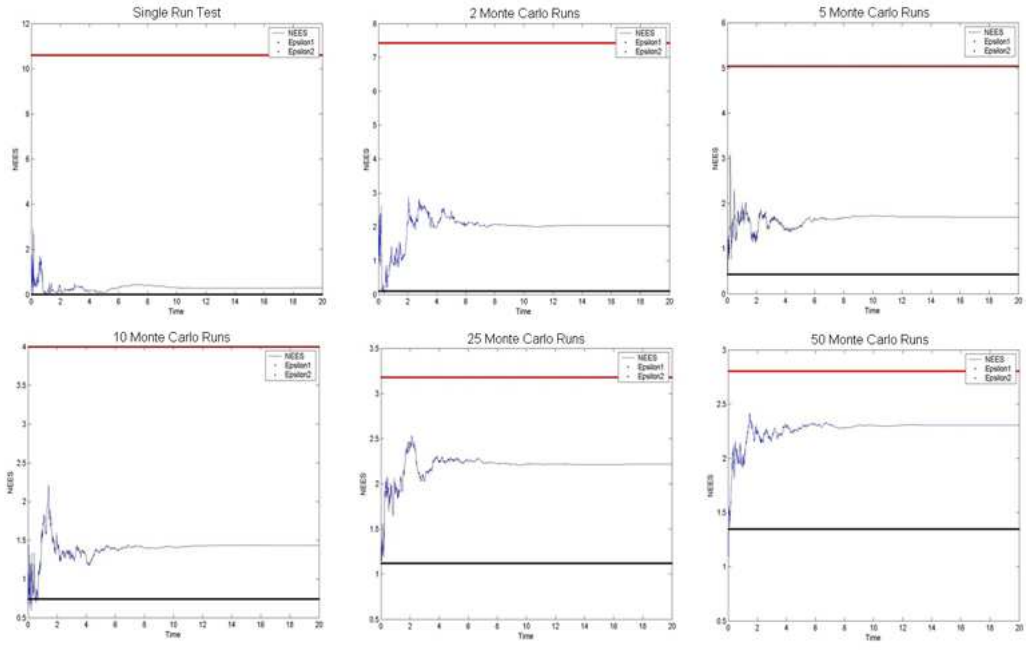


Figure 7.9. NEES with probability Q=0.01

A MATLAB routine is written to implement the block diagram in Figure 7.1. Analytical results are shown in Tables 7.1 and 7.2, they are so close to the results of ERA and Kalman filter. We can say that Kalman filter is working successfully with ERA. We showed that the states of a LTI dynamical system can be estimated from noisy observations by using a consistent Kalman filter.

Table 7.1. Comparison of X-modal Analysis and ERA&Kalman Simulation Results

Modal Analysis Frequency in Hz	Estimated Frequency in Hz	Relative Error(%)
24	24.91	3.8
85.1	85.64	0.6
192.3	194.34	1.06
341.7	346.5	1.4

Table 7.2. Comparison of SAP2000 Analysis and ERA&Kalman Simulation Results

Sap Model Frequency in Hz	Estimated Frequency in Hz	Relative Error(%)
23.5	24.91	6
87.7	85.64	2.3
194.5	194.34	0.08
343.7	346.5	0.81

CHAPTER 8

COMPARISON OF SIMULATION RESULTS

In this chapter, we made a noise analysis for each modal analysis algorithm. An artificial system is created for mass, stiffness and damping matrices which is given as,

$$M = \begin{bmatrix} 10 & 0 \\ 0 & 5 \end{bmatrix} \; C = \begin{bmatrix} 12 & -6 \\ -6 & 6 \end{bmatrix} \; K = \begin{bmatrix} 1200 & -600 \\ -600 & 600 \end{bmatrix} . \tag{8.1}$$

Artificial system is created by using the state-space representation in Equation 4.33. Then white gaussian noise is added to the impulse response of this artificial system for different noise levels. Our SNR definition for noise levels can be given as,

$$SNR(dB) = 10 \log \left(\frac{P_{signal}}{P_{noise}} \right) , \tag{8.2}$$

where dB is decibel. We tried to estimate natural frequencies from noisy data by using ERA, PFD and LSCE algorithms. These algorithms are simulated 10 times and the averages of estimated frequencies and absolute relative errors are taken. Results are shown in Table 8.1. In this table, w_{ex} is the exact natural frequency values, w_{est} is the averages of estimated natural frequency values and σ_w is the standard deviation of estimated frequencies. Relative error in this table can be defined as,

$$\text{Absolute Relative Error} = \frac{|w_{exact} - w_{estimated}|}{w_{exact}} . \tag{8.3}$$

As shown in results, when SNR value gets larger, LSCE and PFD algorithms fails, but the performance of ERA is good. For example, when SNR is zero, this means that noise power is equal to signal power, the average relative errors for ERA algorithm are acceptable.

Table 8.1. Comparison of Modal Analysis Algorithms for Different Noise Levels

Algorithm	SNR (dB)	w_{ex} (rad/s)	w_{est} (rad/s)	σ_w	Average Absolute Relative Error (%)
	80	14.31	14.36	0.0416	0.37
	80	5.93	6.04	0.1073	1.92
	60	14.31	14.27	0.004	0.28
	60	5.93	0.78	0.0532	86.2
LSCE	40	14.31	14.32	0.0254	0.17
	40	5.93	31.32	4.4171	422.32
	20	14.31	212.52	26.48	1384.9
	20	5.93	12.60	0.0869	112.68
	50	14.31	14.30	0.0243	0.14
	50	5.93	5.92	0.0138	0.21
PFD	30	14.31	13.24	0.3084	7.4
	30	5.93	5.12	0.1930	13.5
	10	14.31	0.78	0.6163	94.53
	10	5.93	0.49	0.2753	91.72
	10	14.31	14.25	0.0143	0.18
ERA	10	5.93	5.93	0.1483	0.89
	0	14.31	14.45	0.0430	0.6
	0	5.93	5.91	0.5571	3.2

CHAPTER 9

CONCLUSION AND FUTURE WORK

9.1. Conclusion

Structural health monitoring(SHM) became an important problem and developing technology in civil, mechanical, and aerospace engineering. It is used to examine the damage analysis of any building. SHM can be defined as the analysis of the dynamical behavior of civil structures to observe the reliability of them. The challenge of SHM is finding a mathematical model of a structure by which the health of a structure can be judged. The process of SHM is like in Figure 9.1. Modal analysis techniques are generally used to obtain the mathematical model of the structures. In this study fundamental modal analysis techniques and algorithms are examined. During the study, three fundamental system identification algorithm is reviewed; LSCE algorithm, PFD algorithm and ERA. After examining the theoretical background of these algorithms, application and verification of these algorithms is done by using MATLAB. Simulations shows that

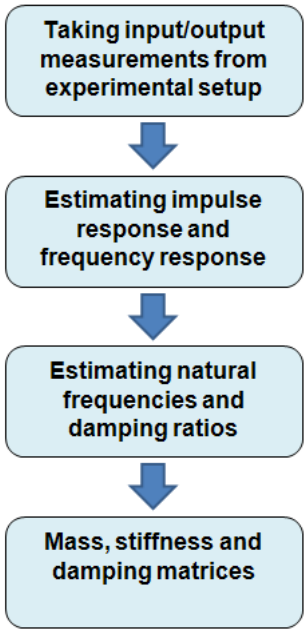


Figure 9.1. The process of SHM

the parameters of a structure can be identified correctly from clear measurements. The response of artificial 1-DOF and 2-DOF systems are used in MATLAB simulations to verify the methods.

Although the verification of the PFD method for artificial models is implemented successfully, the parameters of experimental system can not be identified from experimental measurements because of the measurement noise. PFD algorithm is so sensitive to noise. Also performance of LSCE is good when there isn't noise in measurement, but when noise is present in the data, this method performs poorly.

ERA incorporates singular value decomposition to counteract inherent noise and it gives satisfactory results for the experimental measurement data. We can say that ERA is a more robust algorithm than LSCE or PFD. We used ERA and Kalman filter together to update the system parameters and we have found good results.

9.2. Future Work

Since EM is an efficient algorithm for communication systems, it can be used for linear dynamical systems as vibrational systems. To use EM for vibrational systems, the states of the system must be estimated. Kalman filter estimates the state of a dynamical system from a series of incomplete and noisy measurements. By using the estimated state at each time instant, the parameters of the system can be estimated by using EM algorithm.

Because of the measurement noise and the nonlinearity of the structures, a new advanced method for identification of a structure can be established. The basis of this new method will be upon the variants of Kalman filter and EM algorithm. To set up this new method successfully, we need an initial estimate of system parameters. We can use the result of the ERA algorithm as an initial guess of parameters and try to estimate the real parameters iteratively by using EM algorithm. Also Extended Kalman filter can be used with EM algorithm for nonlinear models. This will be a new solution approach for SHM problem. Now an introductory knowledge about EM algorithm will be given.

The main objective of EM algorithm is to build a probabilistic model of a real-life structure which can be defined by some parameters. By using EM algorithm, the noise in observation data can be eliminated during the identification procedure, so the identified system parameters becomes more realistic, this is the advantage of the EM. To understand

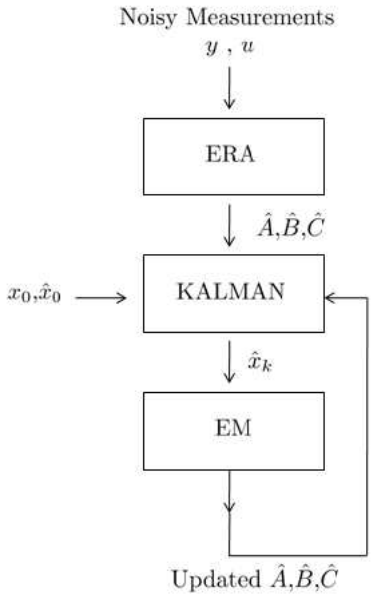


Figure 9.2. Usage of ERA, Kalman Filter and EM Algorithm

EM, *Maximum Likelihood Estimation* (MLE) concept must be known. MLE is a popular probabilistic method for fitting a mathematical model to observed data. As it can be understood from its name, the objective is maximizing the likelihood of the data. It can be said that some terms must be defined like likelihood, to define MLE mathematically. Equation 9.1 and 9.2 represents the likelihood and logarithmic likelihood of the data,

$$P_{\Theta}(Y_i) = P(Y_i = y_i|\Theta)\,, \tag{9.1}$$

$$\log P_{\Theta}(y_i|\Theta) = L(\Theta)\,, \tag{9.2}$$

where Θ is a random vector that consists the parameters of the system and y_i's are the observation data samples. The best model that fits the observation data will be found by using these terms. Actually the best model is the model that assigns the highest probability to the observation for a given parameter vector. To find the best model, $\prod_i P_{\Theta}(Y_i)$ or $\sum_i P_{\Theta}(Y_i)$ must be maximized by using optimization methods. A typical example to understand MLE can be a coin tossing experiment. Assume that a coin is tossed N times and probability of ôa Head comes is p. The result is a set of Heads and Tails and we know that Head comes m times. To find the probability of a head come according to MLE, first the log-likelihood function of the problem must be written as,

$$L(\Theta) = \log P_{\Theta}(y_i|\Theta) = \log p^m (1-p)^{N-m}\,. \tag{9.3}$$

Then log-likelihood function must be maximized. By taking the derivative of the Equation 9.3 and equalize it to zero, parameter p can be found.

$$\frac{dL(\Theta)}{d\Theta} = \frac{m}{p} - \frac{N-m}{1-p} = 0 \tag{9.4}$$

$$p = \frac{m}{N} \tag{9.5}$$

In this example problem, there was no unobserved data or any noisy measurement, MLE worked successfully for this case, but what happens if any unobserved data or noisy measurement data set occurs. The answer is an intuitive and iterative algorithm, EM. There are 3 steps for EM algorithm;

$\Rightarrow$ Create an initial parameter vector Θ_0

$\Rightarrow$ Use Θ_0 to obtain new model , with Θ_1, that provides $L(\Theta_1) > L(\Theta_0)$

$\Rightarrow$ Obtain a sequence like $L(\Theta_0) < L(\Theta_1) < .... < L(\Theta_N)$

Θ_N is the parameter vector of our system and we can define the mathematical model of the structure by using it. For example Θ can be defined as $[A, B, C]^T$ for the system in Equation 7.3. If the states that are estimated from Kalman filter used together with EM algorithm, it can be possible that the parameters of the state-space system in Equation 7.3 can be estimated iteratively.

As shown in Figure 9.2, theoretical underwork of this method is ready but it must be verified by a computer simulation. We run the ERA and Kalman filter together, but we haven't used the EM algorithm yet. The simulation of EM algorithm is left as a future work for this study.

REFERENCES

Isaac, Newton. 1687. *Philosophiae Naturalis Principia Mathematica*.

I.B. Cohen and A. Whitman. 1999. *The Principia : A new translation*. University of California press, Berkeley.

A.C. Ugural and S.K. Fenster. 2003. *Advanced Strength and Applied Elasticity*. New Jersey: Prentice Hall.

Chopra, Anil K. 1995. *Dynamics of Structures, Theory and Applications to Earthquake Engineering*. New Jersey: Prentice Hall.

Ogata, Katsuhiko. 1995. *System Dynamics*. New Jersey: Prentice Hall.

Allemang, Randall. 1999. *Vibrations: Analytical and Experimental Modal Analysis*.

Jimin He and Zhi-Fang Fu. 2001. *Modal Analysis*. Oxford: Butterworth-Heinemann.

Eyyub Karakan. 2008. *Estimation of Frequency Response Function for Experimental Modal Analysis*. Department of Civil Engineering. İzmir Institute of Technology.

Zhang, L. and H. Kanda. 1985. A Polyreference Frequency Domain Method for Modal Parameter Identification. *Mechanical Systems and Signal Processing* Volume 4, Issue 1, p. 65-75.

F. Lembregts and J. Leuridan. 1990. Frequency Domain Direct Parameter Identification for Modal Analysis: State Space Formulation. *ASME Design Engineering Conference, Cincinnati, Ohio*.

Ho B.L. and Kalman R.E. 1963. Effective Construction of Linear State Variable Models from Input/Output Data. *Regelungstechnik* 14 545-548.

Juang J.N. and Pappa R.S. 1985. An eigensystem realization algorithm for modal parameter identification and model reduction. *J. Guidance, Control and Dynamics* 9(3): 294-303.

Juang J.N. and Cooper J.E. 1987. An Eigensystem Realization Algorithm Using Data Correlations For Modal Parameter Identification. *J. Guidance, Control and Dynamics* 9(3): 294-303.

Pappa R.S. and Elliott K.B. 1992. A Consistent Mode Indicator For The Eigensystem Realization Algorithm *NASA Technical Memorandum 107607*.

Marlon D. Hill. 2003. An Experimental Verification of the Eigensystem Realization Algorithm For Vibration Parameter Identification *Florida A&M University*.

Yaakov Bar-Shalom and Thiagalingam Kirubarajan, eds. 2001. *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons Inc.

Scharf, Louis L. 2001. *Statistical Signal Processing*. Addison-Wesley.

Todd K. Moon and Wynn C. Stirling. 2000. *Mathematical Methods and Algorithms for Signal Processing*.

Ninness B. and Gibson S. 1987. The EM Algorithm for Multivariable Dynamic System Estimation. *J. Guidance, Control and Dynamics* 9(3): 294-303.

APPENDIX A

MATLAB SCRIPTS

A.1 Time and Frequency Response of a SDOF Vibrational System

```
clear
M=1;
C=1;
K=100;
num=[1/M];
den=[1 C/M K/M];
sys = tf(num,den);
w = 0:0.0025:30;
H = freqresp(sys,w);
M=real(H);
M1=squeeze(M);
plot(w,M1) \\ Sketches the real part of the frequency response function
N=imag(H);
N1=squeeze(N);
figure plot(w,N1) \\ Sketches the imaginary part of the frequency response function
L=abs(H);
L1=squeeze(L);
figure
plot(w,L1) \\ Sketches the magnitude of the frequency response function
G=angle(H);
G1=squeeze(G);
figure
plot(w,G1) \\ Sketches the phase of the frequency response function
figure
impulse(sys) \\ Sketches the impulse response function
```

A.2 Pre-processing of Experimental Data

This routine begins from defining “data ”variable, this variable consists all measurement data and it has 3 dimension. First dimension, $D1$, is the number of the experiment, second dimension, $D2$, is the number of samples and the third dimension, $D3$, is the number of column in the measurement file. In measurement files, first column is the time, second column is the input data, and other three columns are acceleration outputs.

[D1,D2,D3]=size(data); **All data for 1 reference point is collected in ”data” variable**

data_orig=data;

for i=1:D1

noise=data(i,100:18010,2);

noise_dc=mean(noise);

data(i,:,2)=data(i,:,2)-noise_dc; **Eliminating noise from input data**

end

a=zeros((D3-2),size(data,2),D1); **acceleration data**

v=zeros((D3-2),size(data,2)-1,D1); **velocity data**

d=zeros((D3-2),size(data,2)-2,D1); **displacement data**

for d3=1:D3-2

for d1=1:D1

for d2=1:D2

a(d3,d2,d1) = a(d3,d2,d1) + data(d1,d2,d3+2); **taking acceleration from ”data”**

end

end

end

”hammer” variable is the input data

flen=300;

hammer=zeros(D1,flen);

for d1=1:D1

hammer(d1,1:100)=hammer(d1,1:100) + data(d1,2:101,2);

end

for d1=1:D1

Hammer_all(d1,:) = (fft(hammer(d1,:),Nfft)) **taking fft of input**

```
for index=1:Nfft
if(Hammer_all(d1,index)==0)
if (index==0) Hammer_all(d1,index)=Hammer_all(d1,index+1)/2
elseif (index==Nfft) Hammer_all(d1,index)=Hammer_all(d1,index-1)/2
else
Hammer_all(d1,index)=(Hammer_all(d1,index-1) + Hammer_all(d1,index+1)) /2
end
end
end
GFF_all(d1,:)=real(Hammer_all(d1,:) .* (conj(Hammer_all(d1,:))))
for d3=1:D3-2
v(d3,:,d1)=(a(d3,1:size(data,2)-1,d1)+a(d3,2:size(data,2),d1)) .* (Dt/2) \\numerical
integration
d(d3,:,d1)=(v(d3,1:size(data,2)-2,d1)+v(d3,2:size(data,2)-1,d1)) .* (Dt/2)
dlow(d3,1:Nfft,d1)=zeros(1,Nfft)
dlow(d3,1:Nfft-flen,d1)=d(d3,1:Nfft-flen,d1)
dlow(d3,:,d1)=filter(B,A,dlow(d3,:,d1))
end
end
\\taking average of input
for k=1:Nfft
Hammer_ave(k,1)=mean(Hammer_all(:,k))
end
hammer_ave=real(ifft(Hammer_ave))
for d3=D3-2:D3-2
for d1=1:D1
Output_freq1(d3,:,d1)=(fft(dlow(d3,:,d1),Nfft))
end
end
\\taking average of output
Output_freq=Output_freq1(3,:,:)
for k=1:Nfft
```

```
Output_freq_ave(k,1)=mean(Output_freq(1,k,:))
End
output_ave=real(ifft(Output_freq_ave))
\\frequency domain division
for d3=D3-2:D3-2
fd5dec=zeros(D3-2,Nfft)
for d1=1:D1
GXd5F(d3,:,d1)=(fft(dlow(d3,:,d1),Nfft)).* (conj(Hammer_all(d1,:)))
end
GXd5Fav=sum(GXd5F(d3,:,:),3)
GFFav=sum(GFF_all(:,:))
end
fd5dec(d3,:)=real(ifft(GXd5Fav ./ GFFav))
hf(:,1)=fd5dec(d3,:) \\estimated IRF from frequency domain division
u(:,1)=hammer_ave \\input time domain data
y(:,1)=output_ave \\ output time domain data
L=800 \\length of impulse response
M=200 \\length of input
for i=1:M
x1(i,1)=u(i,1) \\taken samples from input
end
for i=1:(M+L-1)
y1(i,1)=y(i,1) \\taken samples from output
end
z1=zeros(L-1,1) \\for toeplitz matrix
z2=zeros(1,L-1) \\for toeplitz matrix
C=[x1
z1] \\for toeplitz matrix
R=[x1(1,1) z2] \\for toeplitz matrix
X = toeplitz(C,R)
alfa=x1'*x1 \\scalar number
h(:,1)=(1/alfa)*X'*y1 \\Estimated IRF-1
```

```
h1(:,1)=inv(X'*X)*X'*y1 \\Estimated IRF-2
end
```

A.3 PFD Algorithm

```
clear all
N = input('model order : ');
M = input('mass matrix : ') ;
C = input('damping matrix : ') ;
K = input('stiffness matrix : ') ;
A1=inv(M)*C;
A0=inv(M)*K;
B0=inv(M);
I=eye(N,N);
Z=zeros(N,N);
A=[-A1 -A0;I Z];
B=[B0;Z];
C1=[-A1 -A0];
D=[B0];
sys=ss(A,B,C1,D);
w = 1:100;
H = freqresp(sys,w); \\Frequency response of the artificial system
for j=1:100
Asys(:,:,j)=[H(:,:,j) i*w(j)*H(:,:,j) (w(j)^2)*I];
Bsys(:,:,j)=(w(j)^2)*H(:,:,j);
end
sat=1;
p=1;
for k=1:100
for i=1:N
Asys1(p,:)=Asys(sat,:,k);
p=p+1;
sat=sat+1;
end
sat=sat-N;
k=k+1;
```

```
end
sat=1;
p=1;
for k=1:100
for i=1:N
Bsys1(p,:)=Bsys(sat,:,k);
p=p+1;
sat=sat+1;
end
sat=sat-N;
k=k+1;
end
[Q,R] = qr(Asys1);
x=pinv(R)*(Q')*Bsys1 \\ estimated A1, A0 and B0 matrices
for n=1:N
A12(n,:)=[x(n,:)]; \\estimated A0 matrix
end
f=1;
for n=(N+1):(N^2)
A11(f,:)=[x(n,:)]; \\estimated A1 matrix
f=f+1;
end
A111=[-A11 -A12;I Z]; \\ estimated A matrix
roots=eig(A111); \\estimated roots
for n1=1:N
estfreq(n1)=imag(roots(n1)); \\estimated natural frequency
estdamp(n1)=-real(roots(n1))/estfreq(n1); \\estimated damping ratio
end
```

A.4 LSCE Algorithm for Artificial Data

```
clear all
N = input('model order : ');
M = input('mass matrix : ') ;
C = input('damping matrix : ') ;
K = input('stiffness matrix : ') ;
A1=inv(M)*C;
A0=inv(M)*K;
B0=inv(M);
I=eye(N,N);
Z=zeros(N,N);
A=[-A1 -A0;I Z];
B=[B0;Z];
C1=[-A1 -A0];
D=[B0];
sys=ss(A,B,C1,D);
Fs =20; \\ Sampling Frequency
Dt=1/Fs;
T=250; \\ Duration of the signal
t=0:Dt:T; \\ Time interval
y=impulse(sys,t); \\ Impulse response functions of the system
num=Fs*T+1; \\ Number of samples
\\ Maps the impulse responses to variables h11,h12,h21,h22..
for k=1:N
for l=1:N
for j=1:num
eval(sprintf('h1%d%d(j,1)=y(j,k,l);',k,l));
end
eval(sprintf('h%d%d=h1%d%d";',k,l,k,l));
end
end
r0=50; \\ Reading offset
```

```
L=N*2; \\ Number of samples to be used for cosntructing the Hankel matrix
odet=4; \\ Overdetermine factor
for k=1:N
for l=1:N
eval(sprintf('Y%d%d = hankel(h%d%d(r0+1:r0+L*odet)',
h%d%d(r0+L*odet:r0+L*odet+N*2-1));',k,l,k,l,k,l)); \\ Hankel matrix
eval(sprintf('x%d%d=[h%d%d(r0+N*2+1:r0+L+L*odet)]';',k,l,k,l));
eval(sprintf('betavec%d%d = inv(Y %d %d'*Y %d %d)*Y %d %d'*x %d
%d;',k,l,k,l,k,l,k,l,k,l)); \\ Beta vector
eval(sprintf('rootspoly %d%d = roots(flipud([betavec % d % d ; -1]));',k,l,k,l));
eval(sprintf('w%d%d =
Fs*(sqrt(log(rootspoly%d%d).*log(conj(rootspoly%d%d))));',k,l,k,l,k,l)); \\ Identified
natural frequencies for h11,h12...
end
end
\\ Following routine calculates the real natural frequencies from M, C, K matrices
syms w
A=det(K-w^2*M);
Realfreq=eval(solve(A));
n=1;
for i=1:(N*2)
if Realfreq(i)>0
W(n,:)=Realfreq(i);
n=n+1;
end
end
W
```

A.5 LSCE Algorithm for Experimental Data

```
[N,nout]=size(data); \\ size of impulse response data
m=1; \\ number of output
len=min(ceil(N/5),205);
xlen=len;
nrow=floor((N-m)/len)*(len/xlen)-2;
\\ constructing Hankel matrix
datat=data';
for i=1:(nrow)
H0((i-1)*nout+1:i*nout,1:len)=datat(1:nout,(i-1)*xlen+1:(i-1)*xlen+len);
X0((i-1)*nout+1:i*nout,1)=datat(1:nout,i*len+1);
end
beta_vec = inv(H0'*H0)*H0'*X0; \\ beta vector for prony polynomial
roots_poly = roots(flipud([beta_vec ; -1])); \\ roots of polynomial
Fs=1/deltat; \\ sampling frequency
w = Fs*(sqrt(log(roots_poly).*log(conj(roots_poly)))); \\ Natural Frequencies
ksi = -log(roots_poly.*conj(roots_poly))*Fs/(2*w); \\ Damping ratio values
```

A.6 ERA Algorithm for Artificial Data

```
clear all
N = input('model order : ');
M = input('mass matrix : ') ;
C = input('damping matrix : ') ;
K = input('stiffness matrix : ') ;
A1=inv(M)*C;
A0=inv(M)*K;
B0=inv(M);
I=eye(N,N);
Z=zeros(N,N);
A=[-A1 -A0;I Z];
B=[B0;Z];
C1=[-A1 -A0];
D=[B0];
sys=ss(A,B,C1,D); \\defining the state-space model
Fs = 100; \\sampling frequency
Dt=1/Fs;
T=100; \\duration of the signal
t=0:Dt:T; \\time interval
y=impulse(sys,t); \\impulse response of the system
num=Fs*T+1; \\number of samples
for k=1:N
for l=1:N
for j=1:num
eval(sprintf('h1end
eval(sprintf('hend
end
r0=10; \\reading offset
L=100; \\number of samples to be used for cosntructing the Hankel matrix
H0 = hankel(h11(r0+1:r0+L)", h11(r0+L:r0+L+N*2-1)); \\hankel matrix H0
H1 = hankel(h11(r0+2:r0+L+1)", h11(r0+L+1:r0+L+N*2)); \\hankel matrix H1
```

order=rank(H0); **order of the system**

[Pmax,Dmax,Qmax] = svd(H0); **singular value decomposition of Hankel matrix**

P=Pmax(:,1:order); **reducing order of P**

D=Dmax(1:order,1:order); **reducing order of D**

Q=Qmax(:,1:order); **reducing order of Q**

A=[$D^(-0.5)$]*P'*H1*Q*[$D^(-0.5)$]; \\ **state matrix of minimum order realization**

z=eig(A); \\ **eigenvalues of the system**

a=1;

for i=1:2:order

omega(a)=imag(log(z(i))*Fs); **estimated natural frequencies**

dampratio(a)=-real(log(z(i))*Fs)/omega(a); \\ **estimated damping ratios**

a=a+1;

end

A.7 ERA Algorithm for Experimental Data

[N,nout]=size(data); \\ **size of impulse response data**

m=1; \\ **number of output**

len=min(ceil(N/10),205);

xlen=len;

nrow=floor((N-m)/len)*(len/xlen)-2;

\\ **constructing Hankel matrix**

datat=data';

for i=1:nrow

H0((i-1)*nout+1:i*nout,1:len)=datat(1:nout,(i-1)*xlen+1:(i-1)*xlen+len);

H1((i-1)*nout+1:i*nout,1:len)=datat(1:nout,(i-1)*xlen+2:(i-1)*xlen+len+1);

end

dimH=size(H0);

[Plong,Dlong,Qlong]=svd(H0,'econ'); \\ **singular value decomposition of Hankel matrix**

dimD=size(Dlong);

nfreqmax = min(50,dimD(2)/2);

nfreq=nfreqmax;

P=Plong(:,1:nfreqmax);

D=Dlong(1:nfreqmax,1:nfreqmax);

Q=Qlong(:,1:nfreqmax);

clear dimD;

dimD=size(D);

dimP=size(P);

dimQ=size(Q);

invrootD=zeros(dimD);

invrootD(1:nfreq,1:nfreq)=(D(1:nfreq,1:nfreq))^(-0.5);

A=invrootD * P' * H1 * Q * invrootD; \\ **estimate of state matrix**

(V, LAMBDA) = eig (A);

dimLAMBDA=size (LAMBDA);

LAMBDAv = LAMBDA * ones(dimLAMBDA(2),1);

for i=1:nfreq

if(abs(angle(LAMBDAv(i))) ¡ 1e-3) invrootD(i,i)=0; end

if(abs(abs(angle(LAMBDAv(i)))-pi) ¡ 1e-3) invrootD(i,i)=0; end

if(abs(LAMBDAv(i))¿1) invrootD(i,i)=0; end

end

Sv=(log(LAMBDAv))/deltat;

Lv=abs(log(LAMBDAv))/deltat;

zeta=-real(Sv(1:nfreq))./Lv(1:nfreq);

angular_resonant_freq=abs(imag(Sv(1:nfreq))./ sqrt(1 - zeta)); \\ **frequency estimates in rad/s**

freq=(angular_resonant_freq angular_resonant_freq(2*pi)); \\ **frequency estimates in Hz**

A.8 State Estimation with Kalman filter

```
clear
N = input('model order : ');
M = input('mass matrix : ') ;
C = input('damping matrix : ') ;
K = input('stiffness matrix : ') ;
A1=inv(M)*C;
A0=inv(M)*K;
B0=inv(M);
I=eye(N,N);
Z=zeros(N,N);
A=[0 I;-A0 -A1];
B=[Z;B0];
C=[I Z]; \\ continous time system
D=0;
dt=0.01; \\ sampling period  duration=20;
sys=ss(A,B,C,D); \\ continuous time system
sysd=c2d(sys,dt);
(a,b,c,d) = ssdata(sysd); \\ discrete time system
x(:,:,1)=[0;0]; \\ initial state
xest(:,:,1)=x(:,:,1); \\ initial state estimate
P(:,:,1)=b*b'; \\ initial error covariance
snr=10; \\ signal to noise ratio
u(1)=10000;
for j=2:5000
u(j)=0.0001; \\ dirac delta
end
i=1;
for t=0:dt:duration
x(:,:,i+1)=a*x(:,:,i)+b*u(i);
y1(:,:,i)=c*x(:,:,i+1);
i=i+1;
```

```
end
y2=squeeze(y1);
y=awgn(y2,snr); \\ adding noise to observation
k=1;
Sz=0.1;
\\ Kalman Filter Equations
for t=0:dt:duration
xest(:,:,k+1)=a*xest(:,:,k)+b*u(k);
P(:,:,k+1)=a*P(:,:,k)*a';
Inn=y(k)-c*xest(:,:,k);
S=c*P(:,:,k+1)*c'+Sz;
K=P(:,:,k+1)*c'*inv(S);
xest(:,:,k+1)=xest(:,:,k)+K*Inn;
P(:,:,k+1)=(I-K*c)*P(:,:,k+1);
k=k+1;
end
```