NEAR REAL TIME SYSTEM IDENTIFICATION OF SERVO SYSTEMS USING FREQUENCY DOMAIN ANALYSIS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

EMRE GEYİK

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
MECHANICAL ENGINEERING

FEBRUARY 2022

Approval of the thesis:

NEAR REAL TIME SYSTEM IDENTIFICATION OF SERVO SYSTEMS USING FREQUENCY DOMAIN ANALYSIS

submitted by **EMRE GEYİK** in partial fulfillment of the requirements for the degree of **Master of Science** i**n Mechanical Engineering, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Sahir Arıkan
Head of the Department, **Mechanical Engineering** _____

Assist. Prof. Dr. Ali Emre Turgut
Supervisor, **Mechanical Engineering, METU** _____

Assoc. Prof. Dr. Mehmet Bülent Özer
Co-Supervisor, **Mechanical Engineering, METU** _____

Examining Committee Members:

Assoc. Prof. Dr. Ender Yıldırım
Mechanical Engineering, METU _____

Assist. Prof. Dr. Ali Emre Turgut
Mechanical Engineering, METU _____

Assoc. Prof. Dr. Mehmet Bülent Özer
Mechanical Engineering, METU _____

Assoc. Prof. Dr. Can Ulaş Doğruer
Mechanical Engineering, Hacettepe University _____

Assist. Prof. Dr. Selçuk Himmetoğlu
Mechanical Engineering, Hacettepe University _____

Date: 10.02.2022

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name Last name : Emre Geyik

Signature :

ABSTRACT

NEAR REAL TIME SYSTEM IDENTIFICATION OF SERVO SYSTEMS USING FREQUENCY DOMAIN ANALYSIS

Geyik, Emre
Master of Science, Mechanical Engineering
Supervisor : Assist. Prof. Dr. Ali Emre Turgut
Co-Supervisor: Assoc. Prof. Dr. Mehmet Bülent Özer

February 2022, 95 pages

Servo systems control the rotation angle or displacement of the machinery in order to follow reference input commands within required error limits. A typical servo system includes rotating or translating inertia, coupler elements, driver motor and feedback sensor. Mechanical characteristics of a servo system have considerable importance as to satisfy desired performance of the system.

In this thesis, a practical method to monitor mechanical characteristics of a servo drive is suggested. Invariance of system parameters is required for classical servo systems to operate within their respective design conditions. However, as operational time increases, systems are subjected to changes in their mechanical properties. By implementing the methods derived in this thesis, such changes can be tracked and addressed before severe system failures occur. In order to realize the aforementioned monitoring ability, a theoretical model is established and related equations are derived. Then the study is extended to experimental verification of the suggested method. It is shown that the theoretical model agrees with test results.

Keywords: Servo System Identification, Frequency Domain Analysis, Natural Frequency Analysis, Real Time System Monitoring, Predictive Maintenance

ÖZ

SERVO SİSTEMLERİN FREKANS TABANLI ANALİZ KULLANILARAK YAKLAŞIK GERÇEK ZAMANLI SİSTEM TANILAMASI

Geyik, Emre
Yüksek Lisans, Makina Mühendisliği
Tez Yöneticisi: Dr. Öğr. Üyesi Ali Emre Turgut
Ortak Tez Yöneticisi: Doç. Dr. Mehmet Bülent Özer

Şubat 2022, 95 sayfa

Servo sistemleri, belirlenen hata limitleri içerisinde girdi komutlarını takip edebilmek için makinelerin dönüş açılarını veya lineer deplasmanlarını kontrol eden sistemlerdir. Tipik bir servo sistemi; dönme veya ötelenme ataleti, bağlantı elemanları, tahrik motoru ve geri besleme sensöründen oluşur. Servo sisteminin mekanik karakteristiği, istenen sistem performansını sağlayabilmek için büyük önem taşır.

Bu tezde, servo tahrik sistemlerinin mekanik karakteristiklerini izleyebilmek adına pratik bir yöntem önerilmiştir. Sistem parametrelerinin değişmezliği, klasik servo sistemlerinin kendilerine özel dizayn kriterleri içerisinde çalışması için gereklidir. Ancak, çalışma süresi arttıkça sistemler mekanik özelliklerinde değişimlere maruz kalmaktadır. Tanıtılan teknikler kullanılarak, bahse konu değişimler takip edilebilir ve ciddi sistem arızaları yaşanmadan ele alınabilir. Bahsedilen izleme kabiliyetinin gerçekleştirilebilmesi için teorik bir model kurulmuş ve ilgili denklemler türetilmiştir. Ardından önerilen yöntemin deneysel doğrulaması yapılarak çalışma genişletilmiştir. Teorik modelin test sonuçlarıyla uyumlu olduğu gözlenmiştir.

Anahtar Kelimeler: Servo Sistem Tanılaması, Frekans Tabanlı Analiz, Doğal Frekans Analizi, Gerçek Zamanlı Sistem Gözleme, Önleyici Bakım

To my beloved parents…

ACKNOWLEDGMENTS

The author wishes to express his deepest gratitude to his supervisor Assist. Prof. Dr. Ali Emre Turgut and co-supervisor Assoc. Prof. Dr. Mehmet Bülent Özer for their guidance, advice, criticism, encouragements and insight throughout the research.

The technical assistance of Hazim Sefa Kızılay, Fetihhan Güran and Emin Deniz are gratefully acknowledged.

The author would also like to thank Cem Baykal for his suggestions and comments.

Experimental equipment for this study is funded by ASELSAN.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

SISO: Single Input Single Output

SDOF: Single Degree of Freedom

MDOF: Multi Degree of Freedom

PRBS: Pseudo Random Binary Signals

MIMO: Multi Input Multi Output

PMSM: Permanent Magnet Synchronous Motor

LG: Linear Graph

DTFT: Discrete Time Fourier Transform

FFT: Fast Fourier Transform

FRF: Frequency Response Function

FPGA: Field Programmable Gate Array

CNC: Computer Numerical Control

CAD: Computer Aided Design

FEA: Finite Element Analysis

TF: Transfer Function

CHAPTER 1

INTRODUCTION

1.1 Motivation of the Thesis

A servo system is a SISO system used for position or speed control of a given payload. The control scheme of servo systems generally requires a minimum bandwidth value for desired precision. As Ellis and Gao [1] stated, low frequency resonance characteristics of a servo drive results in narrow stability margins. Nominally, a servo drive may have been designed with larger natural frequencies. However, operational and environmental conditions may cause changes in mechanical design properties. Given the fact that natural frequencies of mechanical structure directly determine the bandwidth value and hence the stability margin, change in mechanical parameters over time may pose a significant risk for functionality of the system. In certain applications where continuous operability is crucial, constant monitoring and identification of the system may help to warn operators and technicians for an impending malfunction and take necessary maintenance actions without disruption of the operation.

Most frequent mechanical changes include loosening of coupler elements and increase in bearing and sealing friction. In addition to these phenomena, changes in material properties of transmission elements that can directly affect torsional stiffnesses of the drivetrain may occur. Consequently, such changes induce certain changes on the frequency response of the system.

As Pacas and Villwock [2] discussed, frequency response of the system allows for identification of its mechanical parameters under the condition that a certain model structure is assumed. General industrial applications have strict limits for the number of independent sensors due to constraints within the hardware. That leads to the fact that a practical solution must utilize a minimum amount of additional hardware, preferably only the existing sensing elements within the servo drive. Since a typical servo system has necessary sensing elements to execute closed loop control, it is convenient to use already existing sensors to perform frequency domain analysis to identify the system.

In this study, a theoretical approach for identification of servo drives will be presented. Later, the proposed approach will be tested on a servo drive setup as a proof-of-concept demonstration. Necessary near real time calculations and post processing techniques will also be discussed.

1.2 State of the Art

Currently, a number of methods exist to identify servo drives. Sensorless identification techniques that do not involve any speed sensors utilize torque generating stator currents [3][4], meanwhile a more general method is to use speed sensors to apply frequency domain analysis on the system. Main purpose of these identification techniques is to commission and monitor the servo drives. Detailed examination of parameter derivation algorithms of these studies is presented in the literature review chapter.

In terms of analytical modeling, a typical servo system is modeled as a two-mass system as in [2][3][8][12]. Three-mass system modeling is also prevalent in the literature [19][20]. In this study, a typical servo system is modeled as a two-mass system that includes payload and rotating shaft inertia. Single-mass representation of this system is also discussed and opted for parameter fitting.

In the literature, linear graph modeling of a servo system is not drawn in detail. Rather, the classical block diagram method for design of control systems is used. Linear graph modeling allows simple and straightforward construction of system equations. Thus, linear graph modeling of a typical servo drivetrain is discussed in this study.

Obtaining the transfer function of a system is widely employed in the literature. Analytical solutions for resonant frequencies allow back substitutions to be made in order to deduce mechanical parameters of the plant. In this study, a similar approach will be followed, though with a distinct method. Since relation between input and output shafts is utilized in this study, the transfer function deduction is different from those of similar studies.

Studies that involve servo system identification have significant emphasis on industrial practicality although direct application on a commercial end product is not extensively discussed in the literature. Although industrial application schemes are not described in these studies, this thesis also aims to discuss industrial practicality of proposed methodology.

In common approaches as described in [2][3][4], motor side sensors are used. These sensors include optical encoders or magnetic resolvers in direct coupling with the motor shafts. In these system identification techniques motor currents are taken as inputs and motor side encoder response is examined, rather than using the relation between input and output encoders. Since most industrial applications with ordinary machinery do not involve a secondary encoder attached to the load, studies that work on servo drives with load side encoders are not common in the literature.

However, certain applications that require precise position feedback typically take advantage of an additional encoder directly coupled with the load. This allows the system to operate with higher precision and ensures orientation critical payloads, such as radars or optics, to continue functioning with accurate position information even after a motor failure occurs.

In this study, a novel approach that employs only the motor side encoder and the payload side encoder and eliminates the need to monitor the motor currents is presented.

In addition to frequency domain analyses, there are also time domain analyses performed for servo drives [10][15]. These methods generally elaborate on nonlinear effects such as hysteresis or transmission backlashes. Time domain torque data is used in [10] to predict torsional damping of the servo system. Although many nonlinearities have sharper effects on time domain data, these studies do not exhibit a complete identification method for the entire drivetrain. Because of this reason, this study concentrates on analyses made on frequency domain rather than time domain.

1.3 Thesis Outline

Thesis starts with a literature review as chapter 2. This chapter includes a general literature overview regarding the frequency domain analysis of servo systems. Detailed investigation of literature for methods used in frequency domain analysis is presented. In addition to system identification techniques, studies with other possible techniques that can be utilized for system monitoring are also reviewed.

Following the literature survey, analytical formulations of the thesis are presented in chapter 3. First, linear graph modeling of a typical servo system is shown and explained as a means to mathematically represent the dynamics of servo drives. Then, transfer function formulation is presented to establish an analytical relation between input shaft speed and output shaft speed. Later, the obtained transfer function formulation is used in calculation of natural frequencies for analytically predicting the system behavior. Following the derivation of natural frequencies through forward substitution of system parameters, derivation of system parameters using natural frequencies is presented. The natural frequencies can be easily obtained

from frequency response of the system in practical applications. Detailed procedures of parameter fitting method are explained.

In chapter 4, the Simulink model is presented in order to verify analytical structure. First, the modeling of the system is introduced. Then, methods used for obtaining frequency response of the simulated model are explained. Lastly, the finite element analysis model of the experimental setup and obtained results are presented.

Experimental work for verification of the theoretical and numerical model derivations is presented in chapter 5. First, the test setup is introduced with a general overview of its core elements. Then, the developed software and hardware are discussed. A data processing algorithm is developed for processing the data in near real time. Finally, test results are presented together with outputs of the related data processing methods.

As the final chapter, the thesis is summarized in chapter 6. Advantages and limitations in the proposed method are discussed. The chapter is concluded with suggestions for future studies.

CHAPTER 2

LITERATURE REVIEW

This chapter consists of five headings:

- Applications of servo system identification

  Proposals for the applications of servo system identification found in the literature are discussed under this heading.

- Modeling of servo drives

  Under this heading, modeling approaches for servo drives are discussed.

- Frequency domain analysis techniques

  Different frequency domain analysis techniques are discussed under this heading.

- Other identification techniques

  Techniques that are not used in this study but relevant to servo identification are briefly discussed under this heading.

- Summary

  General overview of the literature is presented and motivation for a modified approach is explained.

2.1 Applications of Servo System Identification

As servo systems are in common use throughout the industry, their applications for identification are also wide. Bansal, Evans and Jones [5] state that high reliability supported by an effective maintenance system is a requirement for many machine applications. For such a system, an identification method is proposed in their study.

In [8][10] and [11], it is argued that identification of the shaft stiffness is crucial for prediction of an early failure of industrial robots, such as high-end medical equipments, CNC machine tools, energy systems and likewise machinery. Authors emphasize that the stiffness of the driveline decreases as fatigue of the mechanical parts increase, and hence the performance of the transmission mechanism reduces. Such changes in the system result in greater motion errors. Therefore, authors state that changes in the driveline stiffness are essential indicators for the health of system operation.

It can also be suggested that a servo identification method may also be used as a means to design specific controllers. [12] presents a dynamic model of a servo system with gear backlash in order to construct a basis for development of a controller. It is shown that such a basis can be achieved by identification of a real servo system with gear backlash using a dynamic model. As a similar application, [13] uses system identification as the quality control procedure in the production line of gear boxes.

[14] and [15] use frequency response of servo machines in order to identify failures within the rolling elements of driveline bearings. Although a general system identification is not aimed in these applications, such studies serve as examples of how broad and specific investigations can be made by using servo system identification methods.

In cases where control inputs cause torsional vibrations overlapping with natural frequencies of the servo drive, suppression of consequent resonances require appropriate filters. In an effort to design a successful control system, servo identification methods are also widely employed in filtering torsional shaft resonances ([1],[16] and [18]). Parameter identification is not needed but is possible in these applications, as detailed frequency response analysis is available.

2.2 Modeling of Servo Drives

Mechanical modeling of a servo system generally starts with determination of the number of masses, or rotational inertias, assumed to exist in the system. Common approaches include two-mass ([3], [8] and [19]) and three-mass ([2], [19] and [20]) system modeling. Both approaches model motor inertia as one of the inertias included in the system. In two-mass models, there is only one load inertia while in three-mass models there are two separate load inertias.

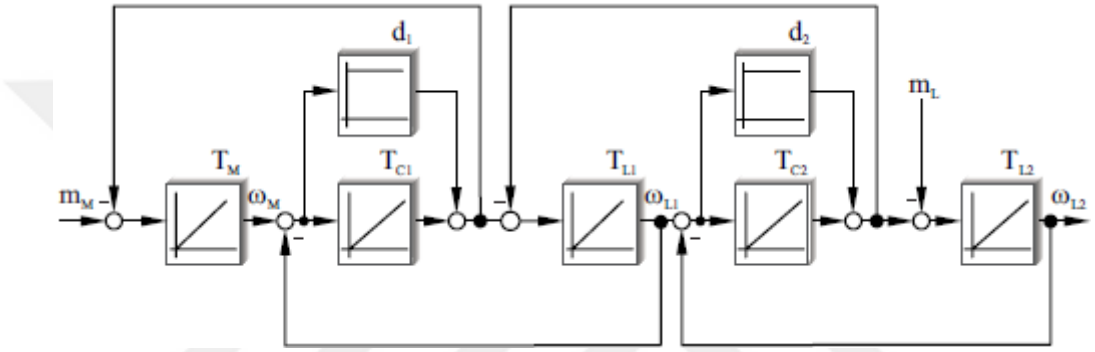

Figure 2.1. Block diagram of three-mass system (from [2])

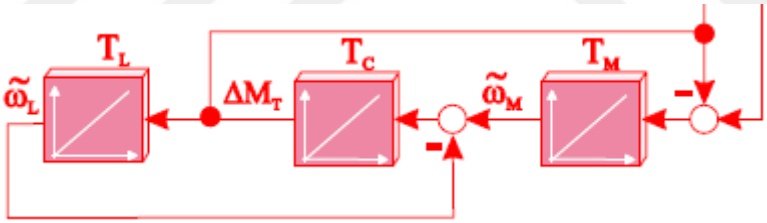

Figure 2.2. Block diagram of two-mass system (from [3])

As another topic, a servo motor is one of the critical components in a servo system. Permanent Magnet Synchronous Motors (PMSMs) are widely employed in many servo applications. Equations that mathematically model internal dynamics of the PMSM motors are used to construct an observer that can be used for speed control in [23]. In addition to realization of an observer, phase currents described in equations 2.2 and 2.3 can be used to measure generated torque by the PMSM motor. Equation 2.1 relates torque generating current to the torque generated by the motor.

$$T_e = K_T i_q \qquad \text{Eq. 2.1}$$

i_q: $Torque\ generating\ current$ K_T: $Torque\ constant\ of\ motor$

T_e: $Electromagnetic\ torque$

In all surveyed studies, it is observed that motors are modeled as torque generating components coupled with an inertia, rather than an across variable source (i.e., rotational velocity source) in terms of system dynamics [21]. This modeling approach allows for torque generating currents to be used as an input parameter but requires specific knowledge regarding internal dynamics of the driving motor. Torque generating current is not a directly measurable signal and requires to be deduced from phase currents. In order to calculate the torque generating current, Park-Clarke transformations must be carried out. These transformations are given below:

$$\begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} i_A \\ i_B \\ i_C \end{bmatrix} \qquad \text{Eq. 2.2}$$

Clarke transformation (from [8])

$$\begin{bmatrix} i_d \\ i_q \end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix} cos\theta & sin\theta \\ -sin\theta & cos\theta \end{bmatrix} \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} \qquad \text{Eq. 2.3}$$

Park transformation (from [8])

In these transformations, i_A, i_B and i_C denote the phase currents in a 3-phase PMSM motor. Resultant torque generating current is denoted by i_q while i_d represents the rotormagnetizing current which has no torque generation effect. In terms of output signals used in identification procedures, studies in the literature mostly portray motor encoders as only available rotational position sensors ([2], [8] and [22]).

Some studies even concentrate on encoderless identification techniques [3]. Thus, common practice in the literature is to obtain motor speed either by direct measurement via motor encoder or by indirect measurement with Luenberger observer structure ([2] and [19]). Luenberger observer takes phase currents and voltages as inputs and calculates relative angular speed between stator and rotor. Subsequently, motor speed signal is used as the output of the system while torque generating current is modeled as the input. Fig. 2.3 illustrates the relationship between motor speed and control loop of the torque generating current. Luenberger observer used in [2] and [19] utilizes current and voltage errors to compute angular speed of the motor.

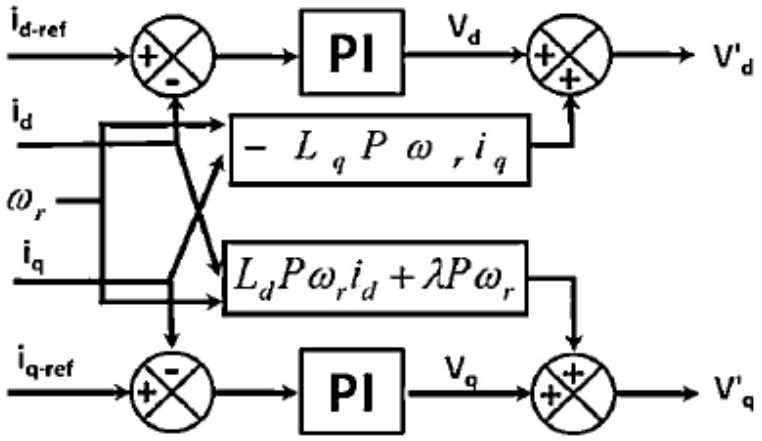

Figure 2.3. Current control loop of a PMSM motor (from [23])

Although motor speed can be calculated by implementing an observer structure, such a calculation method is susceptible to sensor noise and electromagnetic interference. It is important to point out that load side encoders are not used in aforementioned studies and hence formulating a transfer function relationship between a motor encoder and a load encoder stands out as an overlooked gap in the literature.

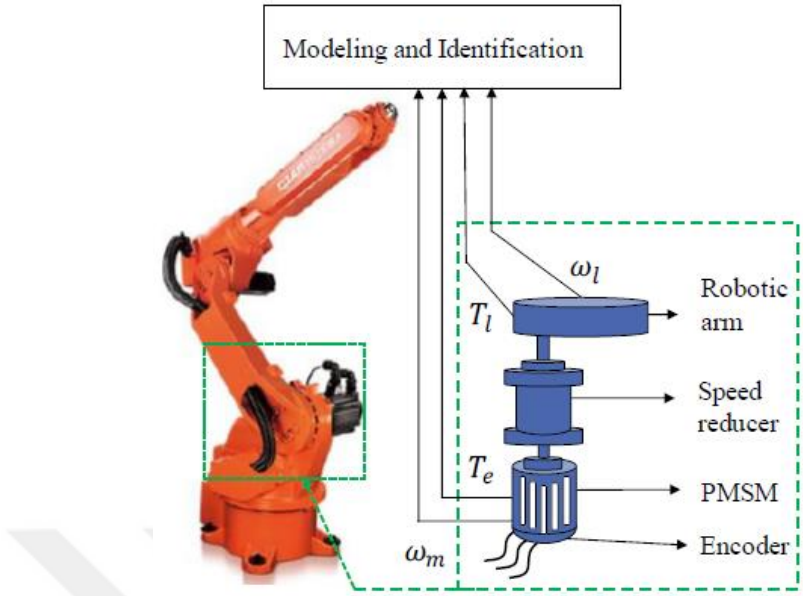

Figure 2.4. Common system outline of servo identification studies (from [8])

Since non-rigid coupler parts play a key role in characterization of frequency response behavior of servo systems, it is almost a universal approach to model connections between motor and loads as torsionally elastic elements. These elastic elements are generally modeled as linear, although some nonlinear models exist [10].

Depending on the choice of model outline, transfer functions of servo systems studied in the literature differ. It is stated in [19] that the transfer function of a three-mass system is much more complex than it is for a two-mass system. As mentioned before, studies existing in the literature use stator currents as inputs and motor speed as output. Hence, derived transfer functions establish a relationship between torque generating currents and motor speed. Derivation of system transfer function is either made by using block diagram approach [2] or state space representation of the system [22]. Studies use classical linear elementary equations for each constituent element of the system of interest. These elementary equations are also to be used in this study.

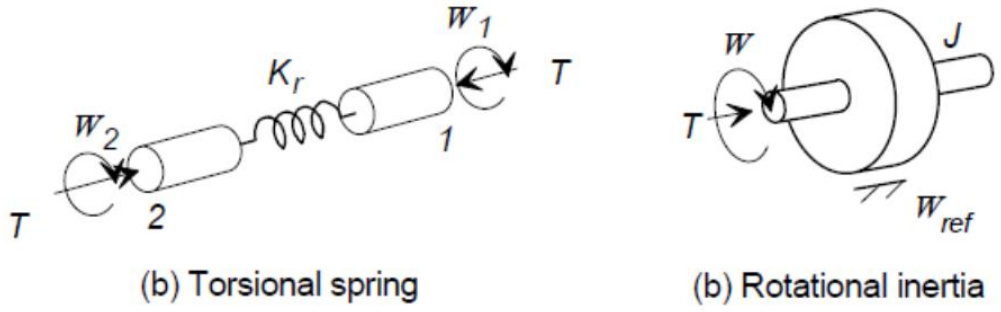

Figure 2.5. Fundamental elements for rotational mechanical systems (from [21])

Elemental equations for rotational elements shown in Fig. 2.5 are given below:

$$\Delta\Omega = \frac{1}{K_r}\frac{dT}{dt} \quad \text{Eq. 2.4}$$

$$T = J\frac{d\Omega}{dt} \quad \text{Eq. 2.5}$$

$$\Delta\Omega = \frac{1}{B_r}T \quad \text{Eq. 2.6}$$

$$\Omega: Rotational\ speed \quad T: Torque \quad J: Rotational\ inertia$$

$$K_r: Rotational\ stiffness \quad B_r: Rotational\ damper$$

Further details regarding elemental equations can be found in [21].

Linear graph (LG) modeling is considered to be a practical way to represent dynamic systems. In [25], it is argued that linear graph modeling is a systematic, unique, integrated, and unified method that employs a well-established set of steps leading to a single model. After forming a linear graph of the actual system, the standard form of the state-space model can be derived by methods explained in [26].

Any linear system model can be generalized as follows.

$$\dot{\boldsymbol{x}} = \boldsymbol{Ax} + \boldsymbol{Bu} \qquad \text{Eq. 2.7}$$

$$\dot{\boldsymbol{y}} = \boldsymbol{Cx} + \boldsymbol{Du} \qquad \text{Eq. 2.8}$$

The vector **x** represents the state variables of the system. Input variables are represented by the vector **u**. The matrices **A** and **B** are properties of the system and are determined by the system structure. The output equation matrices **C** and **D** are determined by the particular choice of output variables.

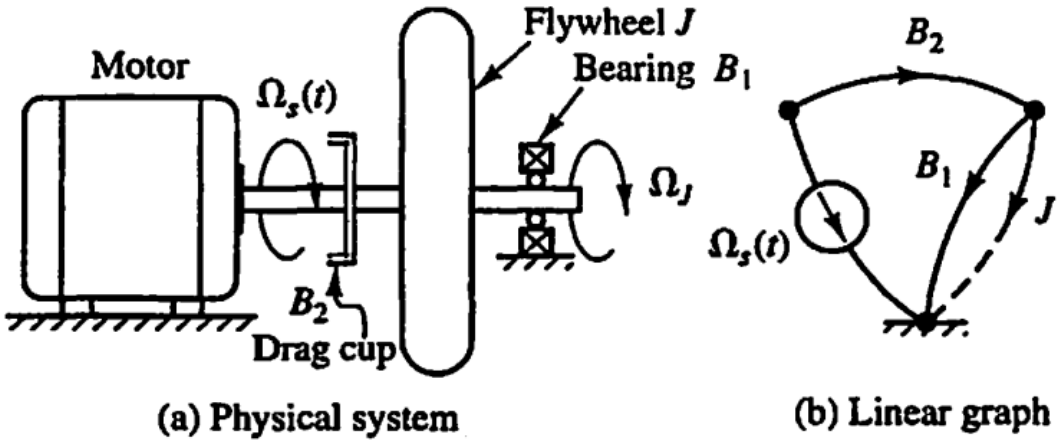

Figure 2.6. An example of linear graph modeling of a rotational drive (from [26])

In the literature, there are not many detailed studies that employ LG modeling for representing a servo system that is to be identified. In this study, flexibility of LG modeling method is used extensively. Details of the procedures followed for conversion from a linear graph model to state space model and subsequent derivation of transfer function between relevant variables are discussed in chapter 3.

2.3 Frequency Domain Analysis Techniques

In this section, important remarks found in the literature for frequency domain analysis of servo systems are discussed.

2.3.1 Excitation Input for the System

In terms of frequency domain analysis, one of the most important factors to consider is the excitation. As excitation signals, two signal types become prominent; these are swept-frequency signals (also known as the "chirp" signal) and pseudo random binary signals (PRBS)[8]. In [29], swept-frequency signal is used for excitation of the system. This signal can be expressed as a time-varying sinusoid function [29],

$$r_u(t) = A \cdot \cos(2\pi \cdot f(t) \cdot t + \varphi) \qquad \text{Eq. 2.9}$$

with time-varying function f(t) is written as,

$$f(t) = m \cdot t + f_0 \qquad \text{Eq. 2.10}$$

where f_0 is the starting frequency of the chirp, and m is the rate of frequency increase per unit time. This allows for linear sweeping of a certain frequency range. However, excitation signal with a certain frequency may cause certain analysis techniques to be affected and hence requires to be appropriately filtered. For example, it is discussed in [29] that since frequency of the input signal changes continuously, sinusoid to be tracked also changes. This requires fixed Kalman gains to be changed. As a result, the Kalman gain must be updated considering the tracked frequency.

In [8], a swinging robot arm is examined, and torque reference for "swinging" motion is taken as excitation signal. [8] argues that the trapezoidal curve is a commonly used speed profile for reciprocating motion of robot joints. In cases where the motion period is short and the maximum speed is not reached, the trapezoidal

curve becomes a triangular wave, for which torque corresponds to rectangular wave. It is argued that frequency spectrum of a rectangular wave is wideband, which can achieve a similar effect to the external excitation.

In [2],[3] and [19], the system is excited by pseudo random binary signals (PRBS). In these studies, PRBS is applied to the reference of torque generating stator current. A torque generation profile in the form of PRBS satisfies the need for full spectrum excitation in order to observe most of the natural frequencies of the system. It can also be argued that PRBS does not contaminate output signals with non-characteristic frequency responses as in the case of chirp signals. This makes PRBS a convenient input signal for system excitation.

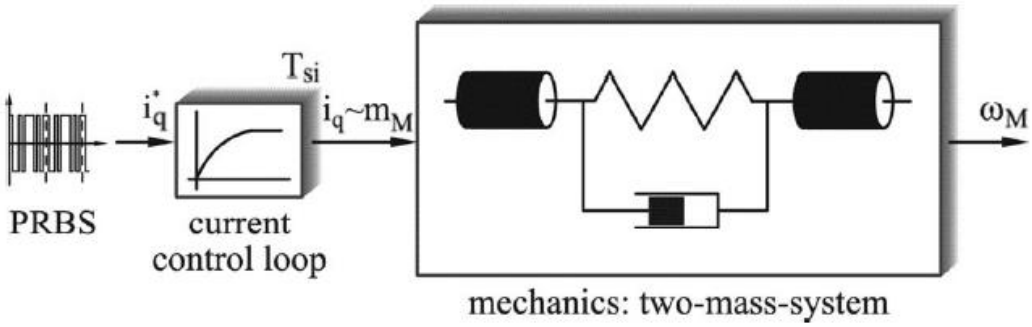

Figure 2.7. Representation of a system with PRBS as current reference (from [19])

In contrast to purely random signals, PRBS has periodicity since it is generated by a shift register consisting of “n” bits. In each cycle time, the most significant bit is shifted toward the output and the new least significant bit is generated via an antivalence function of 2 or more bits, depending on the length of the shift register [2], [3] and [19].

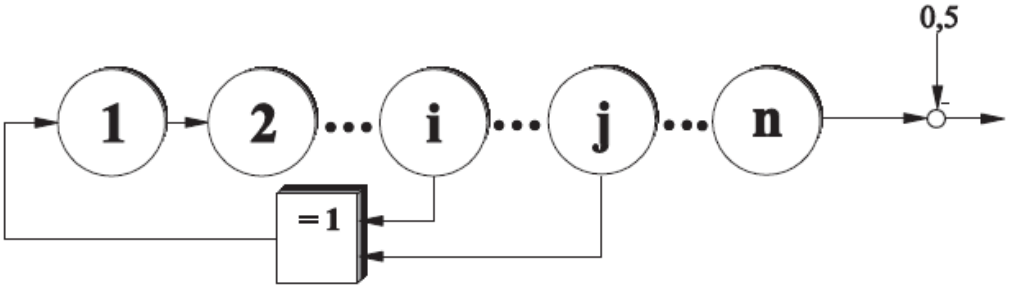

Figure 2.8. Structure principle of PRBS generator (from [3])

Especially in embedded algorithms running on device processors, generation of these pseudo random signals play a crucial role. In this study, generation of random signals is performed by using random number generating function in C, and then embedded into the hardware via flash programmer. Detailed generation scheme used in this study is discussed in Chapter 4 and Chapter 5.

2.3.2 Digital Signal Processing Techniques for Frequency Domain Analysis

In order to perform frequency domain analysis, discrete-time Fourier transform (DTFT) must be applied on the measured signals. Calculations performed in discrete values significantly affect the performance of the procedure, such as sample rate of the data. Choice of an appropriate rate is important to minimize distortions caused by discretization of a continuous signal, also called “aliasing”. In addition to importance of sample rate, selection of correct subsequence length for performing discrete Fourier transform (DFT) is also crucial. Because of the fact that digital computers can only work with information that is discrete and finite in length, converting a long sequence into smaller and manageable sized subsequences is a preferred approach in spectral analyses [32]. However, this technique entails loss of resolution in DTFT, which is directly affected by the choice of length of subsequences. Especially in cases where analyzed data contains noise and

disturbance, averaging the magnitude components of the multiple DFTs can be utilized to reduce the variance of the spectrum.

Two methods exist in the literature for spectral estimation of noisy signals using averaged periodograms, Bartlett's and Welch's methods. Welch's method is a further modification of Bartlett's method, which differs from the latter by modification of data segments with overlapping and windowing. Although introduction of overlapping spectral windows causes an error when estimating the spectral power density, it reduces the leakage effect. Together with reduction in leakage, spectral output also becomes less affected from noisy content in the data. Since these features turn Welch's method into a very strong method, it became a widely accepted method for spectral analysis. As an example, Discrete Transfer Function Estimator block in SIMULINK program uses Welch's method to estimate frequency domain transfer function of a system.

In [2] and [19], Welch method is used for calculating the frequency response of the system. In these studies, it is argued that the Welch method yields much better results than correlogram method, which is another popular method found in the literature. According to Villwock, Welch's method is a powerful digital signal processing method regarding the measured signals and yields better results.

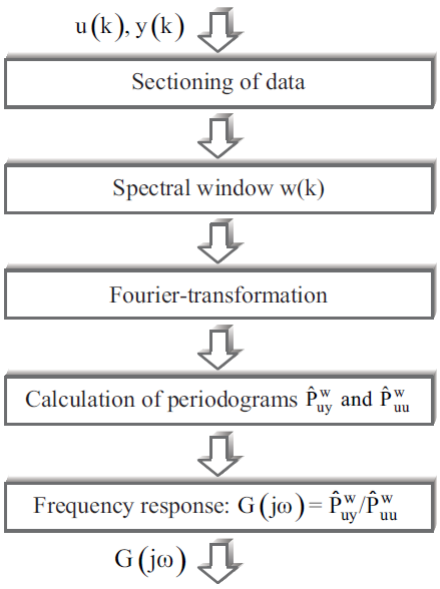

Figure 2.9. Schematic summary of the spectral analysis procedure (from [2])

2.3.3 Parameter Identification Algorithms

Although frequency response presents a good overview of the dynamic characteristics of a system, it does not yield parametric properties of the servo drive. In [2], it is argued that in order to perform a parameter fitting procedure, assumption of a certain model structure is required. Except for a "black box" identification procedure, system structure must be established in order to assign values to system parameters. As for parameter fitting techniques, least squares method and its derivations become prominent. In [2], [8] and [19], Levenberg-Marquardt method is used. This method is an improved form of Gauss-Newton method, which is also a type of least squares method. In the field of structural dynamics, Rational Fraction Polynomial (RFP) method is proposed [33]. It is argued that this method is convenient for use in small computers where computational power is limited. In this method, the analytical transfer function is written in its rational fraction form. Then, an error vector is defined as the difference between values of analytical frequency response and the measured frequency response. This error vector is then used in a squared error criterion for which there is only a single minimum value. Knowing this condition, parameter vectors can be found that satisfy the minimum squared error.

It is observed that RFP method is generally used in structural dynamics, but not encountered in servo system identification applications. RFP method has a simpler form than other methods used in servo system identification, such as Levenberg-Marquardt algorithm. Despite its simplicity, RFP method is not less effective in applications where noise and disturbance are present.

Since RFP method is more suitable and easier to program for embedded software, an adapted version of RFP method is employed in this study. Also, frequency sampling can be scaled in order to match hardware requirements. Flowchart of the method employed in this study is given in Figure 2.10.

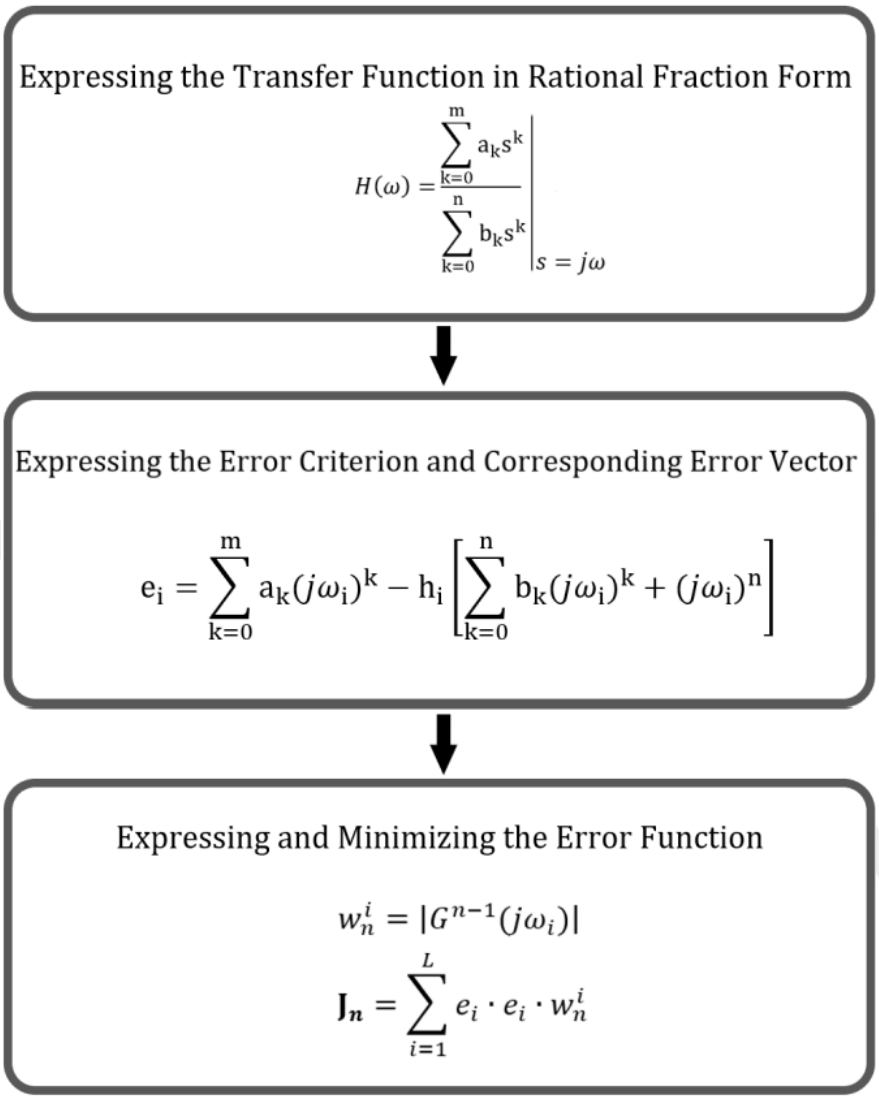

Figure 2.10. Abstract overview of RFP method for parameter fitting

As a first step, analytical transfer function must be expressed in rational fraction form. Secondly, an error term must be written that relates measured FRF with analytical FRF at i^{th} frequency. Then, this error term can be written as an error function in squared form. Equating partial derivatives of this error function to zero yields parameter vectors and hence system can be identified. Detailed explanation of parameter fitting procedure is discussed in Chapter 3. Application of this procedure is discussed in Chapter 5.

2.4 Other Identification Methods

Apart from system parameter identification methods, there are studies which use alternative methods such as neural network approach [5]. It is argued that neural network training approach avoids the need for costly measurement of system parameters. Although criterion for a measurement to be "costly" is not clearly described in this study, authors claim that their study introduces "an effective, real-time, predictive maintenance system based on the motion current signature". According to [5], conventional techniques require an accurate mathematical model in order to predict the dynamic characteristics of the system. Hence, it is required that system parameters are accurately known. Authors of [5] argue that tracking of the system parameter changes during the operation necessitates costly instrumentation. However, this statement cannot be accepted as a general outcome, since literature has considerable number of studies suggesting otherwise [2], [3], [8], [12] and [19].

In [22], an example of black-box identification is carried out on an industrial robot with flexibilities. In black-box modeling, there is no prior model available. Instead, system order is estimated by identification algorithm or inputted by the user, and then appropriate system structure is formed by fitting the parameter vector that defines the system. This parameter vector is determined through finding a set of vector values that minimizes the prediction error. Further details of black-box identification by using System Identification Toolbox can be found in [27].

Another system identification method that utilizes neural networks is studied in [28]. Studied self-commissioning system requires a structure and parameter identification of a nonlinear mechanic as basic building block. The identification is based on extraction of characteristic features from the system response and evaluation of these features by self-organizing neural networks, especially the Self-Organizing Feature Map.

2.5 Summary

As a summary for the literature survey, various studies exist for servo system identification. These studies generally include PMSM motors and two or three mass systems. It is observed that PRBS comes forward as an input excitation profile. Generally, torque generating current is taken as both excitation source and input parameter for the frequency domain analysis. There are no studies which utilize motor speed as input parameter. Since studies in this field do not employ a secondary encoder, there are also no studies which utilize "load side" encoder as output parameter.

In this study, input-output relationship will be formed between motor angular speed and load side angular speed, which can be described as novel in the literature. This approach offers more practical application than other methods which generally utilize torque generating currents, since both input and output signals are directly taken from angular measurements obtained from encoders. On the other hand, obtaining torque generating current of a PMSM motor requires Park-Clarke transformations to be carried out, which implies that utilized signal is susceptible to calculation errors and loss of accuracy.

In addition to such computational drawbacks, quality of phase current measurements is significantly lower than angular measurements which are based on optical sensing techniques. Hardware needed for a precise and accurate current measurement may not be available in servo machinery since frequency domain analysis for characterization of the system requires much sensitive signals than that is required for usual closed loop current control. Because of these reasons, utilization of transfer function relationship between angular speeds in two different machine locations comes out as a practical and but not yet studied aspect of servo system identification for the literature.

In terms of spectral analysis, it is observed that Welch's method is well known and widely accepted in the literature. It provides convenient handling of experimental data, which can have undesired noise and disturbance content. In this study, SIMULINK's Discrete Transfer Function Estimator is used to conduct primal spectral analyses which employs Welch's method.

Finally, prominent algorithms for parameter fitting of experimental results are examined. It is found out that, as expected, algorithms that employ nonlinear least squares regression method are extensively used in the literature. In this study, a practical application of RFP method is presented. As a simple vector fitting procedure, RFP method yields effective results with relatively less computational demand.

CHAPTER 3

ANALYTICAL MODEL

3.1 Linear Graph Modeling of a Classical Servo System

This study is conducted with a classical servo system taken as a basis. CAD image of examined servo system is shown in Figure 3.1. System consists of a PMSM motor with embedded encoder and reducer, a coupler joining motor shaft and the load, another coupler joining the load and encoder, and finally an encoder.

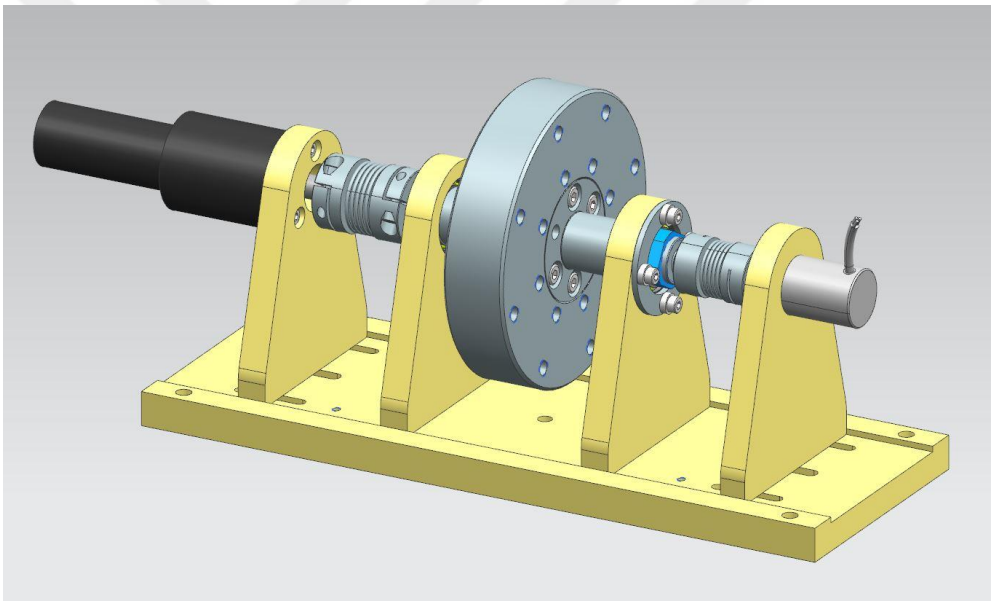

Figure 3.1. CAD model of the test setup

Linear graph approach is utilized for deriving a transfer function in order to create an analytical model. In the linear graph, motor is modeled as an across variable (angular velocity) source. Energy loss due to bearing and lubrication friction is modeled as linear damping. Load and shaft inertias are modeled as a single lumped inertia, and encoder is modeled as a secondary inertia. Elastic couplers are modeled as linear stiffness between the nodes. System parameters are given in Table 3.1.

Table 3.1 Parameters in LG Modeling of the System

Parameter	Explanation	Parameter Type
k_1	Stiffness of the coupler between motor and shaft	System Property
k_2	Stiffness of the coupler between shaft and encoder	System Property
J_{load}	Rotational inertia of the main shaft	System Property
$J_{residual}$	Rotational inertia of the encoder shaft	System Property
b	Linear viscous damping of bearings	System Property
ω_{motor}	Angular speed of the motor shaft	Motion Variable
ω_{load}	Angular speed of the main shaft	Motion Variable
$\omega_{encoder}$	Angular speed of the encoder shaft	Motion Variable

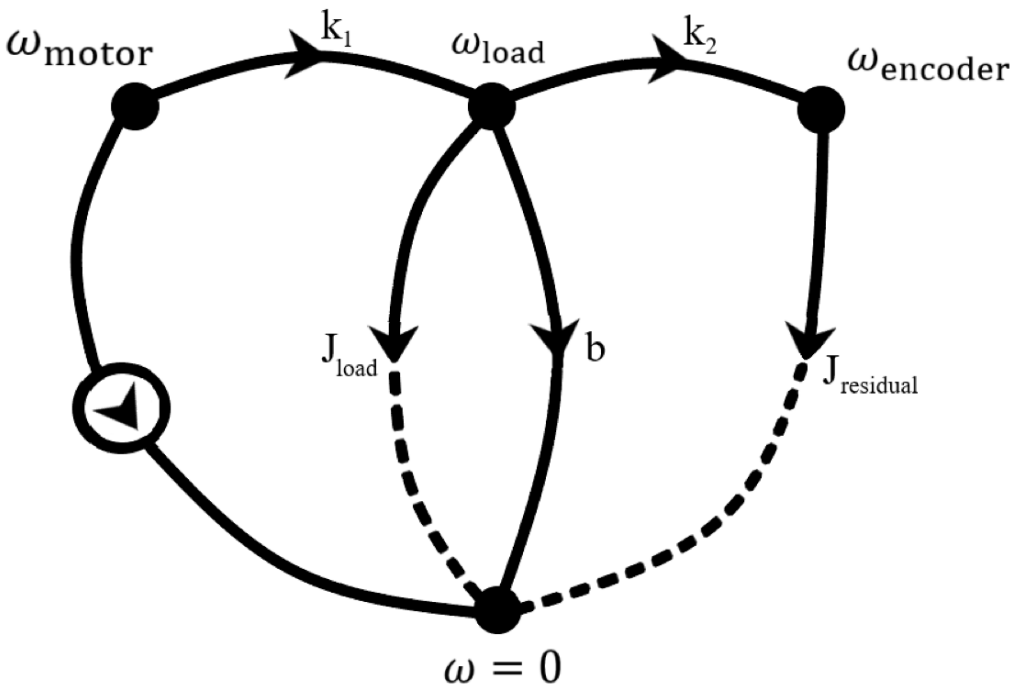

Figure 3.2. Linear graph representation of the servo system

Linear graph of the system is illustrated in Figure 3.2. State equations, which will later be used for derivation of the transfer function between motor speed and encoder speed, can be derived directly from linear graph model [30].

In the literature, using block diagram representations is accepted as a mainstream practice in order to construct transfer functions. However, mechanical connections and their respective transfer functions may not easily be deduced from block diagrams. On the other hand, linear graph (LG) modeling provides a very manageable and simple construction routine. Also being a powerful modeling tool, LG modeling allows easier adaptation to systems with different system configurations.

Initially, LG modeling is used for obtaining the elements of the matrices within the system model which is in the following form:

$$\dot{\boldsymbol{x}} = \boldsymbol{Ax} + \boldsymbol{Bu} \qquad \text{Eq. 3.1}$$

$$\boldsymbol{y} = \boldsymbol{Cx} + \boldsymbol{Du} \qquad \text{Eq. 3.2}$$

The matrices **A** and **B** are properties of the system and are deduced from the system configuration and elements. The output equation matrices **C** and **D** are determined by the specific choice of output variables. Modeling procedure consists of three steps:

1- Determination of the system order and selection of state variables
2- Generation of state equations
3- Derivation of system model by determining **A**, **B**, **C** and **D** matrices

In the first step, normal tree of linear graph of the system must be drawn. Normal tree is defined as a graph tree that includes all across variable sources and as many as possible A-type energy storage elements. An A-type energy storage element is defined as an element in which the stored energy is a function of across variable. In terms of a rotational mechanical system, across variable is the angular speed, and an A-type energy storage element is characterized as rotational inertia. Note that inertia of the load J_{load} is assumed to rotate in a rigid body motion with angular speed ω_{load}.

Likewise, inertia that represents the remainder rotating mass $J_{residual}$ is assumed to rotate in a rigid body motion with angular speed $\omega_{encoder}$.

Since across variable for rotational systems is defined as angular speed, an across variable source for a rotational system can be considered as a servo motor. In that case, angular speed of the motor shaft is modeled as an across variable source. This source is assumed to supply the system with angular speed ω_{motor}.

In the normal tree shown in Figure 3.3, all rotational inertias in the system, J_{load} and $J_{residual}$ are included in the branches. The servo motor which is modeled as a across variable source is also included in the normal tree.

According to this procedure, normal tree is derived as follows in Figure 3.3:

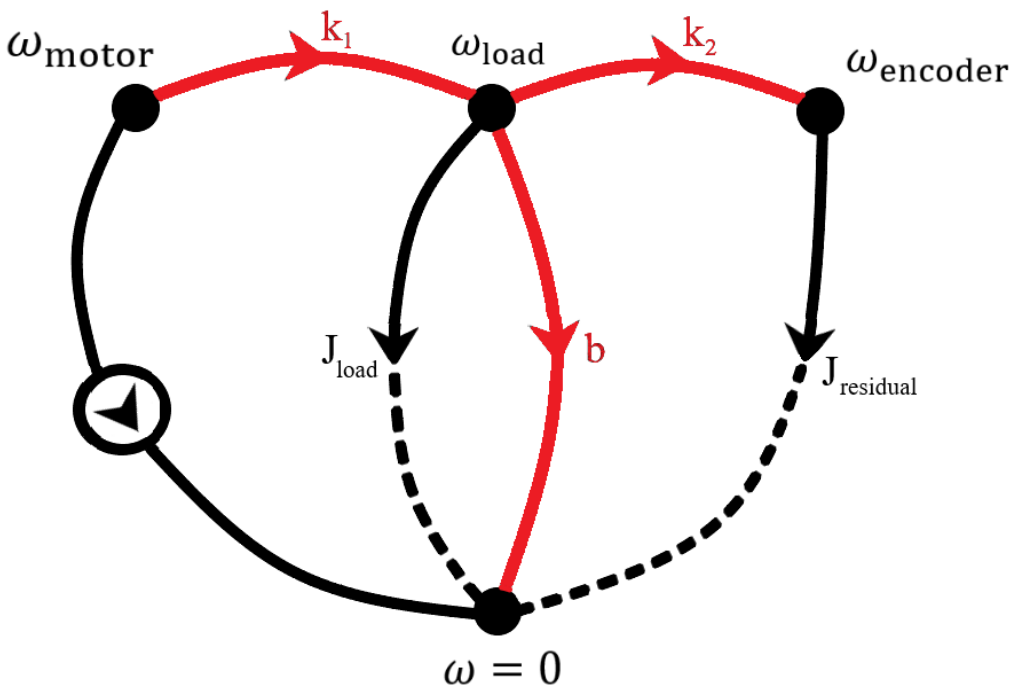

Figure 3.3. Normal tree of the system

Links of the normal tree are highlighted with red color. Branches of normal tree are highlighted with black color. The primary variables of the system are the across variables on the normal tree branches and the through variables on its links.

System order *n* is the number of independent energy storage elements in the system, that is the sum of the number of A-type elements in the normal tree branches and the number of T-type elements among the links. J_{load} and $J_{residual}$ constitute A-type elements in the normal tree. k_1 and k_2 constitute T-type elements among the links. System order is determined as 4. The *n* state variables are selected as the across variables of A-type elements and through variables of T-type elements. State vector **x** is determined as follows:

$$\mathbf{x} = \begin{bmatrix} \omega_{load} \\ \omega_{encoder} \\ T_{k_1} \\ T_{k_2} \end{bmatrix} \quad \text{Eq. 3.3}$$

After determining the state vector, input parameter can be written as follows:

$$u = \omega_{motor} \quad \text{Eq. 3.4}$$

The first step is completed with determination of system order and state variables. In the following step, state equations will be constructed with continuity, compatibility and elemental equations. Fundamental equations of five passive elements are given below:

$$\dot{\omega}_{load} = \frac{T_J}{J} \quad \text{Eq. 3.5}$$

$$\dot{\omega}_{encoder} = \frac{T_{J_r}}{J_r} = \frac{T_{k_2}}{J_r} \quad \text{Eq. 3.6}$$

$$\dot{T}_{k_1} = k_1(\omega_{motor} - \omega_{load}) \quad \text{Eq. 3.7}$$

$$\dot{T}_{k_2} = k_2(\omega_{load} - \omega_{encoder}) \quad \text{Eq. 3.8}$$

$$T_b = b\omega_{load} \quad \text{Eq. 3.9}$$

Note that state equations are formulated by expressing derivatives of state variables with system parameters and state variables themselves. Except Eq. 3.5, above equations are written in this form. In Eq. 3.5, T_J must be expressed with state variables. In the node of ω_{load}, compatibility equation is written as:

$$T_{k_1} = T_J + T_b + T_{k_2} \qquad \text{Eq. 3.10}$$

This equation can be reformulated with substitution of damping equation and expressing T_J with state variables as follows:

$$T_J = T_{k_1} - T_{k_2} - b\omega_{load} \qquad \text{Eq. 3.11}$$

After this step, four state equations can be written in system model form.

$$\begin{bmatrix} \dot{\omega}_{load} \\ \dot{\omega}_{encoder} \\ \dot{T}_{k_1} \\ \dot{T}_{k_2} \end{bmatrix} = \begin{bmatrix} -\frac{b}{J} & 0 & \frac{1}{J} & -\frac{1}{J} \\ 0 & 0 & 0 & \frac{1}{J_r} \\ -k_1 & 0 & 0 & 0 \\ k_2 & -k_2 & 0 & 0 \end{bmatrix} \begin{bmatrix} \omega_{load} \\ \omega_{encoder} \\ T_{k_1} \\ T_{k_2} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ k_1 \\ 0 \end{bmatrix} \omega_{motor} \qquad \text{Eq. 3.12}$$

By writing state equations in system model form, **A** and **B** matrices are determined. Finally, output ω_{load} can be expressed as:

$$y = [1 \quad 0 \quad 0 \quad 0] \begin{bmatrix} \omega_{load} \\ \omega_{encoder} \\ T_{k_1} \\ T_{k_2} \end{bmatrix} \qquad \text{Eq. 3.13}$$

In Eq. 3.2, matrix **D** which represents the direct effect of input on output is zero. As a conclusion, **A**, **B** and **C** matrices are found as follows:

$$\mathbf{A} = \begin{bmatrix} -\frac{b}{J} & 0 & \frac{1}{J} & -\frac{1}{J} \\ 0 & 0 & 0 & \frac{1}{J_r} \\ -k_1 & 0 & 0 & 0 \\ k_2 & -k_2 & 0 & 0 \end{bmatrix} \qquad \text{Eq. 3.14}$$

$$\mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ k_1 \\ 0 \end{bmatrix} \quad \text{Eq. 3.15}$$

$$\mathbf{C} = [1 \quad 0 \quad 0 \quad 0] \quad \text{Eq. 3.16}$$

3.2 Derivation of Transfer Function Between Input and Output Shafts

In order to construct transfer function between ω_{motor} and ω_{load}, recall the system formulation described in Section 3.1,

$$\dot{\boldsymbol{x}} = \boldsymbol{Ax} + \boldsymbol{Bu} \quad \text{Eq. 3.17}$$

$$\boldsymbol{y} = \boldsymbol{Cx} + \boldsymbol{Du} \quad \text{Eq. 3.18}$$

Meanwhile transfer function in Laplace domain is defined as,

$$G(s) = \frac{Y(s)}{U(s)} \quad \text{Eq. 3.19}$$

Where G(s) is the transfer function of the system, U(s) and Y(s) are the Laplace transformations of input and output respectively. It is possible to extract transfer function of the system from its state space representation if input and output vectors can be written in fractional form $\frac{\mathbf{y}}{\mathbf{u}}$**.** Recall that Laplace transformation of vector $\dot{\boldsymbol{x}}$ is equal to s**X**. If system equations are transformed, following form is obtained:

$$s\boldsymbol{X} = \boldsymbol{AX} + \boldsymbol{BU} \quad \text{Eq. 3.20}$$

$$\boldsymbol{Y} = \boldsymbol{CX} + \boldsymbol{DU} \quad \text{Eq. 3.21}$$

Vector **X** can be expressed as,

$$(\boldsymbol{sI} - \boldsymbol{A})\boldsymbol{X} = \boldsymbol{BU} \rightarrow \boldsymbol{X} = (\boldsymbol{sI} - \boldsymbol{A})^{-1}\boldsymbol{BU} \qquad \text{Eq. 3.22}$$

If vector **X** is substituted into equation above, following equation is obtained:

$$\boldsymbol{Y} = \boldsymbol{C}(\boldsymbol{sI} - \boldsymbol{A})^{-1}\boldsymbol{BU} + \boldsymbol{DU} \qquad \text{Eq. 3.23}$$

which can be written as:

$$\boldsymbol{Y} = (\boldsymbol{C}(\boldsymbol{sI} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D})\boldsymbol{U} \qquad \text{Eq. 3.24}$$

Since G(s) is defined as $\frac{Y(s)}{U(s)}$, relation between transfer function G(s) and system matrices is expressed as:

$$G(s) = \mathbf{C}(\mathbf{sI} - \mathbf{A})^{-\mathbf{1}}\mathbf{B} + \mathbf{D} \qquad \text{Eq. 3.25}$$

This transformation between transfer function and system matrices is also expressed in [31]. For SISO systems, resulting expression is a scalar fractional function of s. Note that MIMO systems will have transfer matrix as the result of this expression.

Since **D** is equal to zero for the studied system, equation can be written as:

$$G(s) = \mathbf{C}(\mathbf{sI} - \mathbf{A})^{-\mathbf{1}}\mathbf{B} \qquad \text{Eq. 3.26}$$

In order to derive the transfer function expression, system matrices are substituted explicitly as follows:

$$G(s) = \begin{bmatrix}1 & 0 & 0 & 0\end{bmatrix}\left[\begin{bmatrix}s & 0 & 0 & 0\\ 0 & s & 0 & 0\\ 0 & 0 & s & 0\\ 0 & 0 & 0 & s\end{bmatrix} - \begin{bmatrix}-\frac{b}{J} & 0 & \frac{1}{J} & -\frac{1}{J}\\ 0 & 0 & 0 & \frac{1}{J_r}\\ -k_1 & 0 & 0 & 0\\ k_2 & -k_2 & 0 & 0\end{bmatrix}\right]^{-1}\begin{bmatrix}0\\0\\k_1\\0\end{bmatrix} \qquad \text{Eq. 3.27}$$

Substituting (s**I-A)** expression into a single matrix further simplifies the equation to:

$$G(s) = [1 \quad 0 \quad 0 \quad 0]\begin{bmatrix} s+\frac{b}{J} & 0 & -\frac{1}{J} & \frac{1}{J} \\ 0 & s & 0 & -\frac{1}{J_r} \\ k_1 & 0 & s & 0 \\ -k_2 & k_2 & 0 & s \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ k_1 \\ 0 \end{bmatrix} \qquad \text{Eq. 3.28}$$

Note that the right-hand side of transfer function equation involves $(s\mathbf{I} - \mathbf{A})^{-1}$. This implies that G(s) can also be written as:

$$G(s) = \frac{Q(s)}{|s\mathbf{I} - \mathbf{A}|} \qquad \text{Eq. 3.29}$$

Where Q(s) is a polynomial in s. Notice that $|s\mathbf{I} - \mathbf{A}|$ is equal to the characteristic polynomial of G(s)[31]. Hence, eigenvalues of $|s\mathbf{I} - \mathbf{A}|$ yield poles of the system. Analytic expressions of natural frequencies can be derived as follows:

$$det(A) = \begin{vmatrix} s+\frac{b}{J} & 0 & -\frac{1}{J} & \frac{1}{J} \\ 0 & s & 0 & -\frac{1}{J_r} \\ k_1 & 0 & s & 0 \\ -k_2 & k_2 & 0 & s \end{vmatrix} \qquad \text{Eq. 3.30}$$

Expansion from second column yields:

$$det(A) = s\begin{vmatrix} s+\frac{b}{J} & -\frac{1}{J} & \frac{1}{J} \\ k_1 & s & 0 \\ -k_2 & 0 & s \end{vmatrix} + k_2\begin{vmatrix} s+\frac{b}{J} & -\frac{1}{J} & \frac{1}{J} \\ 0 & 0 & -\frac{1}{J_r} \\ k_1 & s & 0 \end{vmatrix} \qquad \text{Eq. 3.31}$$

If first determinant expression is expanded from third row, and second determinant expression is expanded from second row:

$$det(A) = s\left(-k_2\begin{vmatrix} -\frac{1}{J} & \frac{1}{J} \\ s & 0 \end{vmatrix} + s\begin{vmatrix} s+\frac{b}{J} & -\frac{1}{J} \\ k_1 & s \end{vmatrix}\right) + k_2\left(\frac{1}{J_r}\begin{vmatrix} s+\frac{b}{J} & -\frac{1}{J} \\ k_1 & s \end{vmatrix}\right) \qquad \text{Eq. 3.32}$$

Writing determinants explicitly gives:

$$det(A) = s\left(k_2\frac{s}{J} + s\left(s^2 + s\frac{b}{J} + \frac{k_1}{J}\right)\right) + k_2\left(\frac{1}{J_r}\left(s^2 + s\frac{b}{J} + \frac{k_1}{J}\right)\right) \quad \text{Eq. 3.33}$$

If parentheses are expanded, equation yields:

$$det(A) = s^4 + s^3\frac{b}{J} + s^2\frac{k_2}{J} + s^2\frac{k_1}{J} + s^2\frac{k_2}{J_r} + s\frac{bk_2}{JJ_r} + \frac{k_1k_2}{JJ_r} \quad \text{Eq. 3.34}$$

Then, roots of polynomial below yield the poles of the system.

$$poles = roots\left(s^4 + s^3\frac{b}{J} + s^2\left(\frac{k_1 + k_2}{J} + \frac{k_2}{J_r}\right) + s\frac{bk_2}{JJ_r} + \frac{k_1k_2}{JJ_r}\right) \quad \text{Eq. 3.35}$$

Assuming root(2) and root(4) are complex conjugates of root(1) and root(3) respectively, natural frequencies of the system can be expressed as follows:

$$f_1 = \frac{abs(Im(root(1)))}{2\pi}\ Hz \quad \text{Eq. 3.36}$$

$$f_2 = \frac{abs(Im(root(3)))}{2\pi}\ Hz \quad \text{Eq. 3.37}$$

Natural frequencies of a system with similar layout shown in Figure 3.2 can be calculated by a MATLAB script shown in Appendix A-1.

3.3 Analytical Calculations for Frequency Response of the System

Frequency response of a system with a given transfer function can be calculated analytically by substituting ***jω*** to Laplace domain variable **s**. This substitution yields G(jω), absolute value of which is defined as frequency response function of the system with transfer function G(s).

$$FRF(\omega) = abs(G(j\omega))\angle arg(G(j\omega)) \quad \text{Eq. 3.38}$$

As a result, FRF of the system can explicitly be written as:

$$|FRF(\omega)| = abs\left(\begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} j\omega + \frac{b}{J} & 0 & -\frac{1}{J} & \frac{1}{J} \\ 0 & j\omega & 0 & -\frac{1}{J_r} \\ k_1 & 0 & j\omega & 0 \\ -k_2 & k_2 & 0 & j\omega \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ k_1 \\ 0 \end{bmatrix} \right) \quad \text{Eq. 3.39}$$

MATLAB script given in Appendix A-2 is used for analytically calculating the response of the system for a certain range of frequencies, and then plotting the results on a log-log scale. Inertia values of the shafts are taken from CAD measurements (Fig. 3.5 and 3.6). Stiffness value is taken from manufacturer's datasheet (Table 3.2).

In Fig. 3.4, general overview of the physical setup is shown. In this setup, Maxon's EC 22 motor with PMG's PMG-8A reducer is used as angular velocity source (Right-hand side). FENAC's AS20 absolute encoder is used as load side encoder (Left-hand side). Two couplers that join the load shaft to the encoder and the motor are RINGFEDER's DKN15 metal bellow couplers. Bearings are GRW's 6803. All the other parts are of custom design.

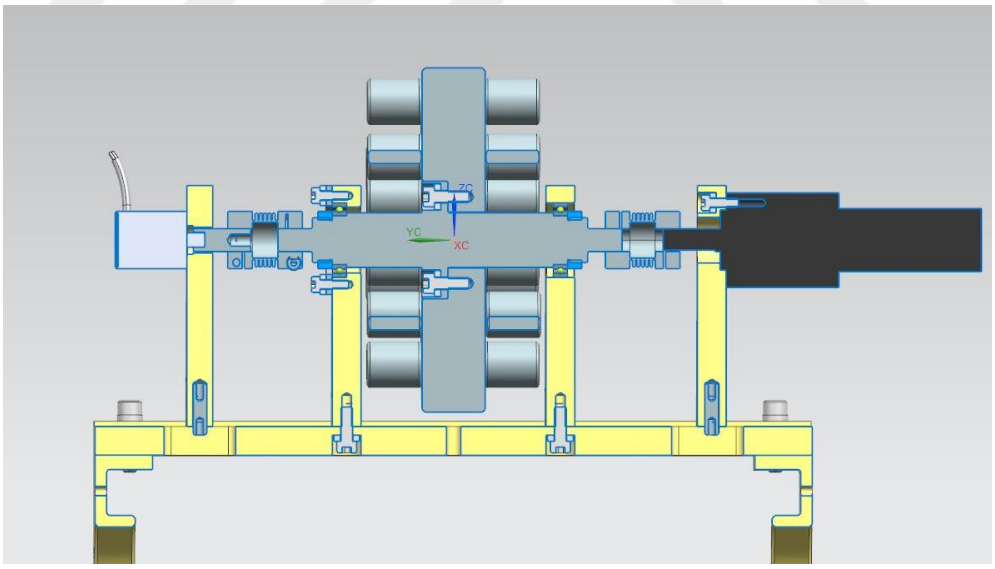

Figure 3.4. Layout of the system. *Y* axis in this view represents the axis of rotation.

As for the stiffness values of k_1 and k_2, datasheet provided by the coupler manufacturer is used. In Table 3.2, a section of RINGFEDER's datasheet for DKN series couplings is presented.

Table 3.2 Specifications of RINGFEDER DKN Series (taken from [24])

Size	T	n_{max}	C_{Tdyn}
	Nm	1/min	10^3 Nm/rad
4	0,5	15000	0,25
4	0,5	15000	0,19
4	0,5	15000	0,15
9	1,1	15000	0,50
9	1,1	15000	0,38
9	1,1	15000	0,30
15	1,75	15000	0,75

According to this specification, stiffness value of DKN15 couplers is given as:

$$k_1 = k_2 = 0.75 \times 10^3 \frac{Nm}{rad} = 750 \frac{Nm}{rad} \quad \text{Eq. 3.40}$$

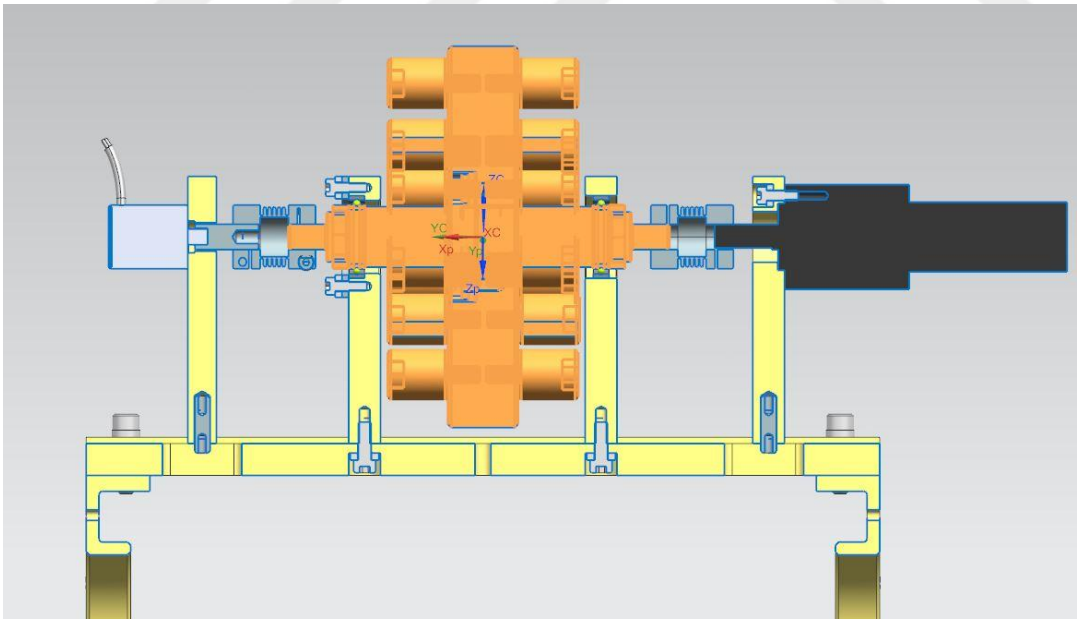

Figure 3.5. Parts of main rotating shaft that contribute to J_{load}

Parts highlighted in Fig. 3.5 are measured for their collective moment of inertia and following output is obtained:

```
Density               = 0.000008538 kg/mm³
Volume                = 384876.6553 mm³
Mass                  = 3.285969441 kg

Center of Mass
Xcbar, Ycbar, Zcbar   = 0.00000 mm,    0.00000 mm,   0.00000 mm

Moments of Inertia (WCS)
Ix, Iy, Iz            = 3464.891756292 kg·mm², 5573.436295805 kg·mm², 3464.885404558 kg·mm²
```

Figure 3.6. Measurement for J_{load} (except coupler inertia)

Since shaft is rotating on *Y* axis, J_{load} is determined as summation of I_y component of the measurement shown in Fig. 3.6, and the contribution of two couplers on both ends. It is assumed that collars of couplers that are attached to the load shaft will contribute to J_{load}. If contribution of a single coupler collar is assumed to be half of total coupler inertia, following expression can be written:

$$J_{load} = 5573.4\ kg \cdot mm^2\ +\ 2 \times \frac{J_{coupler}}{2} = 5573.4\ kg \cdot mm^2 + J_{coupler} \qquad \text{Eq. 3.41}$$

Moment of inertia for DKN15 couplers are given as:

$$J_{coupler} = 0.0011\ \times\ 10^{-3}\ kg \cdot m^2 \qquad \text{Eq. 3.42}$$

Hence, J_{load} is calculated as:

$$J_{load} = 0.0055734 +\ 0.0011\ \times\ 10^{-3}\ kg \cdot m^2 = 0.0055745\ kg \cdot m^2 \qquad \text{Eq. 3.43}$$

A similar procedure is carried out for $J_{residual}$. It is assumed that $J_{residual}$ consists of moments of inertia of encoder shaft and collar of the coupler connected to it.

Previously it was assumed that contribution of a single coupler collar is equal to half of total coupler inertia. These assumptions lead to following expression:

$$J_{residual} = J_{shaft,enc} + \frac{J_{coupler}}{2} \qquad \text{Eq. 3.44}$$

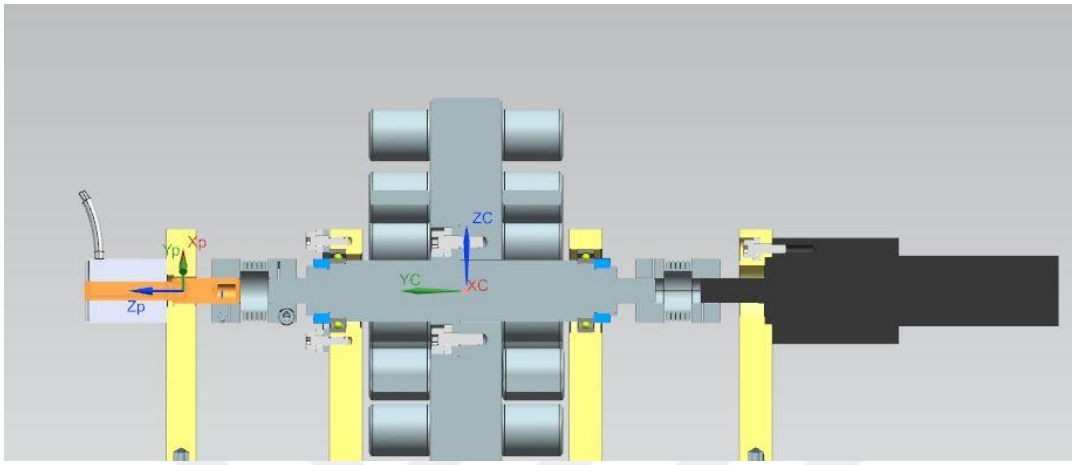

Figure 3.7. Parts of encoder shaft that contribute to $J_{residual}$

```
Density                 = 0.000005914 kg/mm³
Volume                  = 1630.216667895 mm³
Mass                    = 0.009640943 kg

Center of Mass
Xcbar, Ycbar, Zcbar     =    0.010239972 mm,    88.092115850 mm,    -0.009778652 mm

Moments of Inertia (WCS)
Ix, Iy, Iz              =   76.308247988 kg·mm²,    0.073246983 kg·mm²,    76.308316479 kg·mm²
```

Figure 3.8. Measurement for $J_{residual}$ (except coupler inertia)

Fig. 3.8 shows the measurement result for moment of inertia of encoder shaft. Finally, rotational inertia of $J_{residual}$ is calculated as:

$$J_{residual} = 0.07325 \times 10^{-6} + \frac{0.0011 \times 10^{-3}}{2} = 0.62325 \times 10^{-6}\ kg \cdot m^2 \qquad \text{Eq. 3.45}$$

After these calculations, only damping coefficient *b* remains unspecified. In this study, damping coefficient *b* is used to represent bearing losses caused by friction of ball elements during rotational motion. Bearing losses are difficult to determine, and various studies are conducted for this topic in detail. In [6], bearing company SKF presents a formula sheet for calculating friction losses in bearings.

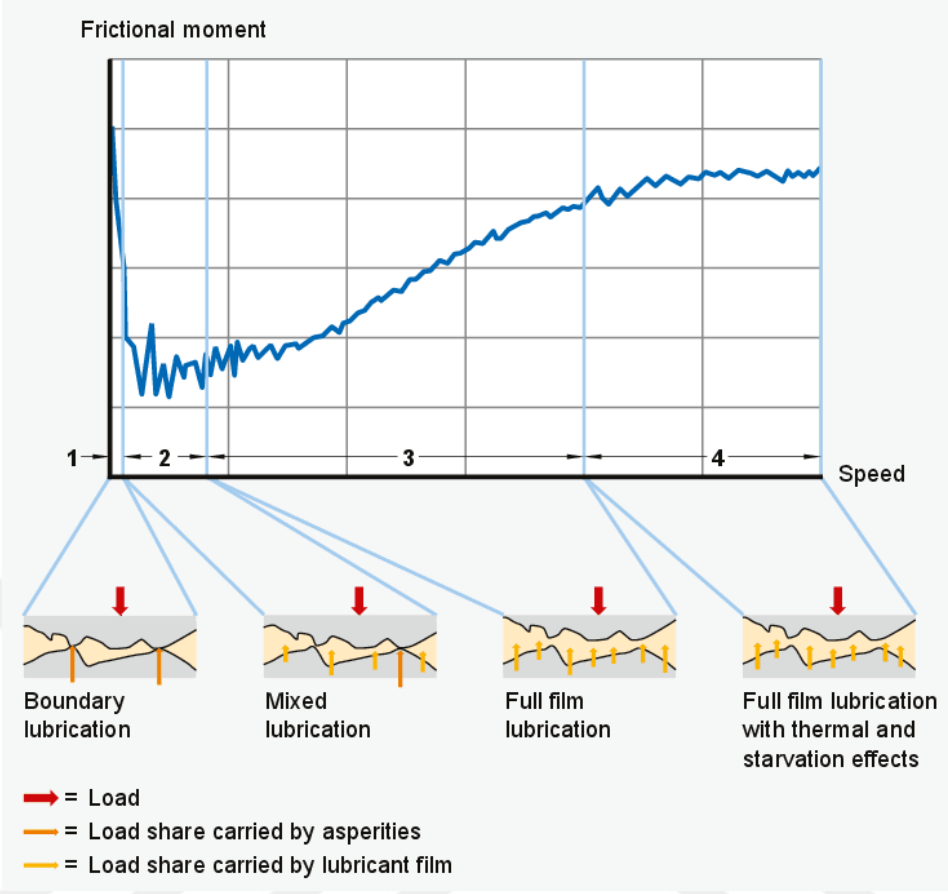

Figure 3.9. Frictional moment in relation to angular speed (from [6])

In Fig. 3.9, frictional moment behavior of bearings is shown with changing angular speed. Although not an exact linear behavior is observed, increasing frictional loss with increasing angular speed exhibits a convenient trend for linear approximation. As the first case of analytical calculations, undamped case is considered. Hence, b value is taken as zero. After determining initial inputs for analytical analysis, MATLAB script given in Appendix A-2 is run and frequency response of ω_{load} to ω_{motor} is obtained.

Also, MATLAB script given in Appendix A-1 is used to calculate expected natural frequencies by using analytical equations derived in section 3.2. Numerical values of system parameters used for the first case are summarized in Table 3.3.

Table 3.3 Values of system parameters for undamped case

Parameter	Value	Unit
k_1	750	Nm/rad
k_2	750	Nm/rad
J_{load}	0.0055745	kg . m^2
$J_{residual}$	0.00000062325	kg . m^2
b	0	Nms/rad

MATLAB code given in Appendix A-1 is run with values given in Table 3.3 and following natural frequencies are calculated:

$$f_1 = 58.3745\ Hz \qquad \text{Eq. 3.46}$$

$$f_2 = 5522.4\ Hz \qquad \text{Eq. 3.47}$$

In order to obtain frequency response of the system, MATLAB code given in Appendix A-2 is run and results are illustrated in Figure 3.10.

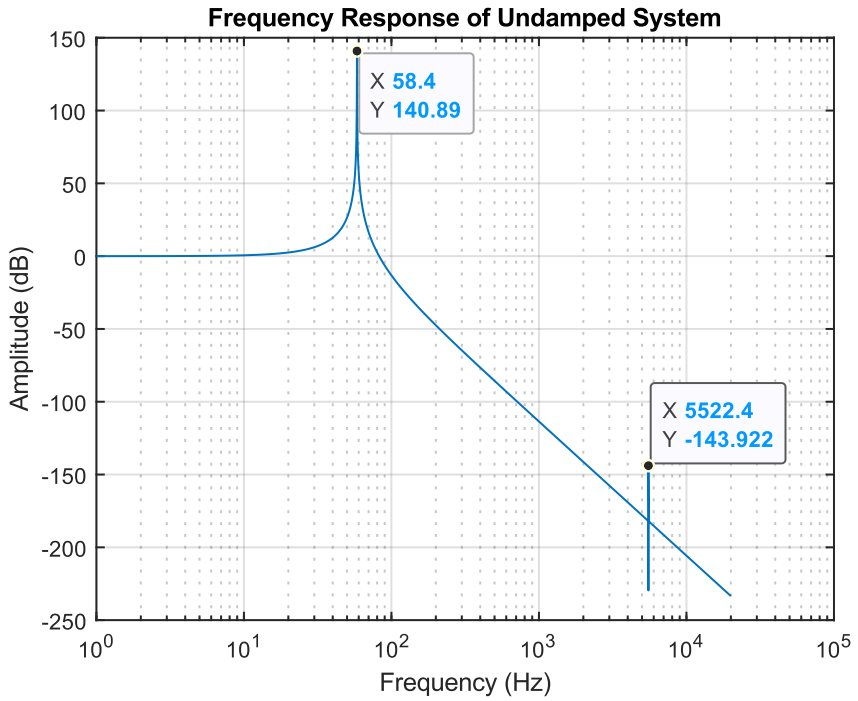

Figure 3.10. Frequency Response of Undamped System

For the test setup used in this study, it is apparent from Figure 3.10 that detecting the second rotational mode would practically be impossible due to extremely low amplitude and notably high frequency. Because of this reason, it is also possible to utilize a SDOF representation of the system for practical considerations. However, calculation methods for finding transfer function between input and output shafts do not differ by order of the system. Although derivation of individual parameters becomes algebraically complex as *n* increases, the methodology presented in this study covers all similar cases with any *nth* order system. If the same procedure is utilized for the SDOF representation of the test setup, transfer function given in Equation 3.48 is obtained.

$$G(s) = \frac{\frac{k_1}{J_{load}}}{s^2 + \frac{b}{J_{load}} + \frac{k_1}{J_{load}}} \qquad \text{Eq. 3.48}$$

Equivalently, this transfer function can be written in the following form:

$$G(s) = \frac{{\omega_n}^2}{s^2 + 2\zeta\omega_n + {\omega_n}^2} \qquad \text{Eq. 3.49}$$

where ζ denotes the damping ratio of the system. Similar to 2DOF representation of the system, first natural frequency is also calculated to be around 58Hz.

$$f_1 = \sqrt{\frac{k_1}{J_{load}}} = \sqrt{\frac{750}{0.0055745}} = 366.8\, rad/s \;=\; 58.38\, Hz \qquad \text{Eq. 3.50}$$

For cases where the rotational modes after first mode are not predominant, SDOF modeling of the system can be used conveniently. However, if modeled system has multiple loads with significant torque transmission, SDOF modeling may result in oversimplification of the system.

3.4 Analytical Application of Parameter Fitting using RFP Method

Rational Fraction Polynomial (RFP) method is employed in order to fit system parameters to the measured values of frequency response of the system. This method requires transfer function G(s) to be written in the rational fraction form. Transfer function G(s) of the system is expressed in Equation 3.28 in matrix form, and must be transformed into an explicit form. For this purpose, MATLAB script given in Appendix A-3 is used. This symbolic code parametrically yields transfer function as follows:

$$G(s) = \frac{J_r k_1 s^2 + k_1 k_2}{JJ_r s^4 + J_r b s^3 + (Jk_2 + J_r k_1 + J_r k_2)s^2 + bk_2 s + k_1 k_2} \quad \text{Eq. 3.51}$$

It is a common practice to make coefficient of highest order term in denominator of the transfer function unity in many applications. If both denominator and numerator is divided by JJ_r, G(s) takes following form:

$$G(s) = \frac{\frac{k_1}{J} s^2 + \frac{k_1 k_2}{JJ_r}}{s^4 + \frac{b}{J} s^3 + \left(\frac{k_2}{J_r} + \frac{k_1 + k_2}{J}\right) s^2 + \frac{bk_2}{JJ_r} s + \frac{k_1 k_2}{JJ_r}} \quad \text{Eq. 3.52}$$

In RFP method, error at i^{th} frequency is expressed as [33]:

$$e_i = \sum_{k=0}^{m} a_k (j\omega_i)^k - h_i \left[\sum_{k=0}^{n-1} b_k (j\omega_i)^k + (j\omega_i)^n \right] \quad \text{Eq. 3.53}$$

where m and n denote the orders of numerator and denominator polynomials respectively. h_i is the FRF measurement data at ω_i. In order to perform least squares approach, aggregate squared error for L number of frequencies can be written as:

$$\mathbf{J} = \sum_{i=1}^{L} e_i \cdot e_i = \{\mathbf{E}^*\}^t \{\mathbf{E}\} \quad \text{Eq. 3.54}$$

$*$: Denotes complex conjugate t : Denotes transpose

Error vector **E** can be written in terms of parameter vectors **A** and **B** in following form:

$$\{\mathbf{E}\} = [\mathbf{P}]\{\mathbf{A}\} - [\mathbf{T}]\{\mathbf{B}\} - \{\mathbf{W}\} \quad \text{Eq. 3.55}$$

where:

$$[\mathbf{P}] = \begin{bmatrix} 1 & j\omega_1 & \cdots & (j\omega_1)^m \\ 1 & j\omega_2 & \cdots & (j\omega_2)^m \\ \vdots & \vdots & & \vdots \\ 1 & j\omega_L & \cdots & (j\omega_L)^m \end{bmatrix} \quad \text{Eq. 3.56}$$

$$[\mathbf{T}] = \begin{bmatrix} h_1 & h_1(j\omega_1) & \cdots & h_1(j\omega_1)^{n-1} \\ h_2 & h_2(j\omega_2) & \cdots & h_2(j\omega_2)^{n-1} \\ \vdots & \vdots & & \vdots \\ h_L & h_L(j\omega_L) & \cdots & h_L(j\omega_L)^{n-1} \end{bmatrix} \quad \text{Eq. 3.57}$$

$$\{\mathbf{W}\} = \begin{Bmatrix} h_1(j\omega_1)^n \\ h_2(j\omega_2)^n \\ \vdots \\ h_L(j\omega_L)^n \end{Bmatrix} \quad \text{Eq. 3.58}$$

and **{A}**, **{B}** are parameter vectors written in form:

$$\{\mathbf{A}\} = \begin{Bmatrix} a_0 \\ a_1 \\ \vdots \\ a_m \end{Bmatrix}, \{\mathbf{B}\} = \begin{Bmatrix} b_0 \\ b_1 \\ \vdots \\ b_{n-1} \end{Bmatrix} \quad \text{Eq. 3.59}$$

For G(s) given in Eq. 3.52, $m = 2$ and $n = 4$. With these parameters set, matrices take following forms:

$$[\mathbf{P}] = \begin{bmatrix} 1 & j\omega_1 & (j\omega_1)^2 \\ 1 & j\omega_2 & (j\omega_2)^2 \\ \vdots & \vdots & \vdots \\ 1 & j\omega_L & (j\omega_L)^2 \end{bmatrix} \quad \{\mathbf{W}\} = \begin{Bmatrix} h_1(j\omega_1)^4 \\ h_2(j\omega_2)^4 \\ \vdots \\ h_L(j\omega_L)^4 \end{Bmatrix} \quad \text{Eq. 3.60}$$

$$[\mathbf{T}] = \begin{bmatrix} h_1 & h_1(j\omega_1) & h_1(j\omega_1)^2 & h_1(j\omega_1)^3 \\ h_2 & h_2(j\omega_2) & h_2(j\omega_2)^2 & h_2(j\omega_2)^3 \\ \vdots & \vdots & \vdots & \vdots \\ h_L & h_L(j\omega_L) & \ldots & h_L(j\omega_L)^3 \end{bmatrix} \quad \text{Eq. 3.61}$$

Finally **{A}** and **{B}** vectors are written as:

$$\{\mathbf{A}\} = \begin{Bmatrix} \frac{k_1 k_2}{J J_r} \\ 0 \\ \frac{k_1}{J} \end{Bmatrix}, \{\mathbf{B}\} = \begin{Bmatrix} \frac{k_1 k_2}{J J_r} \\ \frac{b k_2}{J J_r} \\ \frac{k_2}{J_r} + \frac{k_1 + k_2}{J} \\ \frac{b}{J} \end{Bmatrix} \qquad \text{Eq. 3.62}$$

Note that error function **J** is calculated by matrix multiplication of transpose of error vector $\{\mathbf{E}\}'\boldsymbol{s}$ complex conjugate with itself. Since error vector $\{\mathbf{E}\}$ depends on parameter vectors **{A}** and **{B}**, error function **J** can be written as a function of vectors **{A}** and **{B}**. Transpose of error vector $\{\mathbf{E}\}'\boldsymbol{s}$ complex conjugate can be written explicitly as:

$$\{\mathbf{E}*\}^{\boldsymbol{t}} = \{\{[\mathbf{P}]\{\mathbf{A}\} - [\mathbf{T}]\{\mathbf{B}\} - \{\mathbf{W}\}\}^*\}^{\boldsymbol{t}} \qquad \text{Eq. 3.63}$$

Transpose of complex conjugate of a matrix is also called *conjugate transpose* of a matrix [7]. Conjugate transpose has following properties:

$$(\mathbf{A} + \mathbf{B})^H = \mathbf{A}^H + \mathbf{B}^H \qquad \text{Eq. 3.64}$$

$$(\mathbf{AB})^H = \mathbf{B}^H \mathbf{A}^H \qquad \text{Eq. 3.65}$$

where *H* denotes conjugate transpose operation or Hermitian conjugate. These properties yield:

$$\{\mathbf{E}*\}^{\boldsymbol{t}} = \{[\mathbf{P}]\{\mathbf{A}\}\}^H - \{[\mathbf{T}]\{\mathbf{B}\}\}^H - \{\mathbf{W}\}^{\boldsymbol{H}} \qquad \text{Eq. 3.66}$$

$$\{\mathbf{E}*\}^{\boldsymbol{t}} = \{\mathbf{A}\}^H[\mathbf{P}]^{\boldsymbol{H}} - \{\mathbf{B}\}^H[\mathbf{T}]^{\boldsymbol{H}} - \{\mathbf{W}\}^{\boldsymbol{H}} \qquad \text{Eq. 3.67}$$

Since vectors **{A}** and **{B}** are real valued, their Hermitian conjugates are equal to their transpose.

$$\{\mathbf{E}*\}^{\boldsymbol{t}} = \{\mathbf{A}\}^{t}[\mathbf{P}]^{\boldsymbol{H}} - \{\mathbf{B}\}^{t}[\mathbf{T}]^{\boldsymbol{H}} - \{\mathbf{W}\}^{\boldsymbol{H}} \qquad \text{Eq. 3.68}$$

And error function **J** can be written as:

$$\mathbf{J} = \{\mathbf{E}^{*}\}^{t}\{\mathbf{E}\} = \{\{\mathbf{A}\}^{t}[\mathbf{P}]^{\boldsymbol{H}} - \{\mathbf{B}\}^{t}[\mathbf{T}]^{\boldsymbol{H}} - \{\mathbf{W}\}^{\boldsymbol{H}}\}\{[\mathbf{P}]\{\mathbf{A}\} - [\mathbf{T}]\{\mathbf{B}\} - \{\mathbf{W}\}\} \qquad \text{Eq. 3.69}$$

$$\begin{aligned}\mathbf{J} = {} & \{\mathbf{A}\}^{t}[\mathbf{P}]^{\boldsymbol{H}}[\mathbf{P}]\{\mathbf{A}\} - \{\mathbf{A}\}^{t}[\mathbf{P}]^{\boldsymbol{H}}[\mathbf{T}]\{\mathbf{B}\} - \{\mathbf{A}\}^{t}[\mathbf{P}]^{\boldsymbol{H}}\{\mathbf{W}\} - \{\mathbf{B}\}^{t}[\mathbf{T}]^{\boldsymbol{H}}[\mathbf{P}]\{\mathbf{A}\} \\ & + \{\mathbf{B}\}^{t}[\mathbf{T}]^{\boldsymbol{H}}[\mathbf{T}]\{\mathbf{B}\} + \{\mathbf{B}\}^{t}[\mathbf{T}]^{\boldsymbol{H}}\{\mathbf{W}\} - \{\mathbf{W}\}^{\boldsymbol{H}}[\mathbf{P}]\{\mathbf{A}\} \\ & + \{\mathbf{W}\}^{\boldsymbol{H}}[\mathbf{T}]\{\mathbf{B}\} + \{\mathbf{W}\}^{\boldsymbol{H}}\{\mathbf{W}\}\end{aligned} \qquad \text{Eq. 3.70}$$

Note that multiplication of a complex valued matrix with its Hermitian conjugate yields a real valued square matrix or a real scalar value. Also, summation of $[\mathbf{X}]^{\boldsymbol{H}}[\mathbf{Y}]$ with $[\mathbf{Y}]^{\boldsymbol{H}}[\mathbf{X}]$ yields $2\times\mathbf{Re}([\mathbf{X}]^{\boldsymbol{H}}[\mathbf{Y}])$ ***or*** $2\times\mathbf{Re}([\mathbf{Y}]^{\boldsymbol{H}}[\mathbf{X}])$ if matrix multiplication $[\mathbf{X}]^{\boldsymbol{H}}[\mathbf{Y}]$ results in a scalar value. According to these properties, error function **J** becomes:

$$\begin{aligned}\mathbf{J} = {} & \{\mathbf{A}\}^{t}[\mathbf{P}]^{\boldsymbol{H}}[\mathbf{P}]\{\mathbf{A}\} + \{\mathbf{B}\}^{t}[\mathbf{T}]^{\boldsymbol{H}}[\mathbf{T}]\{\mathbf{B}\} + \{\mathbf{W}\}^{\boldsymbol{H}}\{\mathbf{W}\} \\ & - 2\mathbf{Re}(\{\mathbf{A}\}^{t}[\mathbf{P}]^{\boldsymbol{H}}[\mathbf{T}]\{\mathbf{B}\}) \\ & - 2\mathbf{Re}(\{\mathbf{A}\}^{t}[\mathbf{P}]^{\boldsymbol{H}}\{\mathbf{W}\}) + \mathbf{2Re}(\{\mathbf{B}\}^{t}[\mathbf{T}]^{\boldsymbol{H}}\{\mathbf{W}\})\end{aligned} \qquad \text{Eq. 3.71}$$

Error function **J** takes its minimum value at a point where:

$$\frac{\partial \mathbf{J}}{\partial \mathbf{A}} = 0 \; and \; \frac{\partial \mathbf{J}}{\partial \mathbf{B}} = 0 \qquad \text{Eq. 3.72}$$

If partial derivatives $\frac{\partial \mathbf{J}}{\partial \mathbf{A}}$ and $\frac{\partial \mathbf{J}}{\partial \mathbf{B}}$ are explicitly written as follows:

$$\frac{\partial \mathbf{J}}{\partial \mathbf{A}} = 2[\mathbf{P}]^{\boldsymbol{H}}[\mathbf{P}]\{\mathbf{A}\} - 2\mathbf{Re}([\mathbf{P}]^{\boldsymbol{H}}[\mathbf{T}]\{\mathbf{B}\}) - 2\mathbf{Re}([\mathbf{P}]^{\boldsymbol{H}}\{\mathbf{W}\}) = \{\mathbf{0}\} \qquad \text{Eq. 3.73}$$

$$\frac{\partial \mathbf{J}}{\partial \mathbf{B}} = 2[\mathbf{T}]^{\boldsymbol{H}}[\mathbf{T}]\{\mathbf{B}\} - 2\mathbf{Re}([\mathbf{T}]^{\boldsymbol{H}}[\mathbf{P}]\{\mathbf{A}\}) + \mathbf{2Re}([\mathbf{T}]^{\boldsymbol{H}}\{\mathbf{W}\}) = \{\mathbf{0}\} \qquad \text{Eq. 3.74}$$

Further simplification by removing factor of 2 and placing constant terms to the right-hand side gives:

$$[\mathbf{P}]^{\boldsymbol{H}}[\mathbf{P}]\{\mathbf{A}\} - \mathbf{Re}([\mathbf{P}]^{\boldsymbol{H}}[\mathbf{T}]\{\mathbf{B}\}) = \mathbf{Re}([\mathbf{P}]^{\boldsymbol{H}}\{\mathbf{W}\}) \quad \text{Eq. 3.75}$$

$$[\mathbf{T}]^{\boldsymbol{H}}[\mathbf{T}]\{\mathbf{B}\} - \mathbf{Re}([\mathbf{T}]^{\boldsymbol{H}}[\mathbf{P}]\{\mathbf{A}\}) = -\mathbf{Re}([\mathbf{T}]^{\boldsymbol{H}}\{\mathbf{W}\}) \quad \text{Eq. 3.76}$$

Writing error minimizing equations in linear system of equations form yields:

$$\begin{bmatrix} \mathbf{Y} & \vdots & \mathbf{X} \\ \dots & \vdots & \dots \\ \mathbf{X}^{H} & \vdots & \mathbf{Z} \end{bmatrix} \begin{Bmatrix} \mathbf{A} \\ \dots \\ \mathbf{B} \end{Bmatrix} = \begin{Bmatrix} \mathbf{G} \\ \dots \\ \mathbf{F} \end{Bmatrix} \quad \text{Eq. 3.77}$$

where:

$$[\mathbf{Y}] = [\mathbf{P}]^{\boldsymbol{H}}[\mathbf{P}] \quad \text{Eq. 3.78}$$

$$[\mathbf{X}] = -\mathbf{Re}([\mathbf{P}]^{\boldsymbol{H}}[\mathbf{T}]) \quad \text{Eq. 3.79}$$

$$[\mathbf{Z}] = [\mathbf{T}]^{\boldsymbol{H}}[\mathbf{T}] \quad \text{Eq. 3.80}$$

$$\{\mathbf{G}\} = \mathbf{Re}([\mathbf{P}]^{\boldsymbol{H}}\{\mathbf{W}\}) \quad \text{Eq. 3.81}$$

$$\{\mathbf{F}\} = -\mathbf{Re}([\mathbf{T}]^{\boldsymbol{H}}\{\mathbf{W}\}) \quad \text{Eq. 3.82}$$

Solving these equations gives vectors **{A}** and **{B}**. Then, through known **{A}** and **{B}** vectors system parameters can be calculated. Generally, this approach yields an ill-conditioned solution matrix that does not give convenient results. However, by using orthogonal polynomial bases for independent variable **s** (or **j**ω), it is possible to avoid ill-conditioning of matrix equation. A simplified version of Forsythe method explained in [33] can be used to generate such polynomial bases. Using orthogonal basis simplifies the solution by turning **Y** and **Z** matrices into identity matrices.

Parameter vectors are replaced by **C** and **D**, which can be transformed back into normal parameter vectors. Related transformation script is given in Appendix A-4.

$$\begin{bmatrix} \mathbf{I}_1 & \vdots & \mathbf{X} \\ \dots & \vdots & \dots \\ \mathbf{X}^t & \vdots & \mathbf{I}_2 \end{bmatrix} \begin{Bmatrix} \mathbf{C} \\ \dots \\ \mathbf{D} \end{Bmatrix} = \begin{Bmatrix} \mathbf{H} \\ \dots \\ \mathbf{0} \end{Bmatrix} \quad \text{Eq. 3.83}$$

where:

$$\{\mathbf{H}\} = \mathbf{Re}([\mathbf{P}]^{\boldsymbol{H}}\{\mathbf{W}\}) \quad \text{Eq. 3.84}$$

$$\{\mathbf{C}\} = \{\mathbf{H}\} - [\mathbf{X}]\{\mathbf{D}\} \quad \text{Eq. 3.85}$$

$$[\mathbf{I} - [\mathbf{X}]^t[\mathbf{X}]]\{\mathbf{D}\} = -[\mathbf{X}]^t\{\mathbf{H}\} \quad \text{Eq. 3.86}$$

It is discussed in [9] that RFP method characteristically exhibits higher error values around the peak frequency despite its decent performance in locating the natural frequencies. This problem can be solved by implementing an error weighting vector for peak frequencies. By implementing this, error occurring around natural frequencies can have greater effect on overall calculations. Weighting vector can be determined by fitted response amplitude, and then progressively iterated to minimize the error around the peak region. Equations given below express the error weighting logic.

$$w_n^i = |G^{n-1}(j\omega_i)| \quad \text{Eq. 3.87}$$

$$\mathbf{J_n} = \sum_{i=1}^{L} e_i \cdot e_i \cdot w_n^i \quad \text{Eq. 3.88}$$

$$w_n^i : \ weighting\ factor\ of\ n^{th}\ iteration\ at\ i^{th}\ frequency$$

Since damping is the dominant factor around peak region, it is possible to pick damping ratio as an effective criterion to determine whether the curve fitting procedure successively approximated the system. Half-power rule can be used to extract damping ratio from measured FRFs. Equations given below describe the half-power rule. [17]

$$\zeta = \frac{f_{upper,3dB} - f_{lower,3dB}}{2 \times f_{peak}} \qquad \text{Eq. 3.89}$$

$$2\zeta\omega_n = 2 \times \frac{\omega_{upper,3dB} - \omega_{lower,3dB}}{2 \times \omega_n} \times \omega_n = \omega_{upper,3dB} - \omega_{lower,3dB} \qquad \text{Eq. 3.90}$$

For SDOF models, damping ratio obtained from FRF measurements can directly be compared with coefficient of the first order term in the denominator of SDOF transfer function given in section 3.3. For MDOF systems, it is possible to apply this criterion by expressing transfer functions in factorized form and comparing first order coefficient term for each mode.

3.5 Summary of the Chapter

As a summary for the analytical modeling chapter, a methodology is presented for derivation of analytical formulations for frequency response of the system. Different modeling approaches are discussed and compared. A complete procedure is explained for obtaining parametric transfer function equations that can be applied to SDOF and MDOF systems.

Subsequently, a parameter fitting algorithm is explained with improved peak region performance. Analytical background of the fitting procedure is explained and discussed.

All in all, necessary calculations used in development of a mathematical foundation for monitoring servo drives are presented.

CHAPTER 4

SIMULATION AND ANALYSIS

4.1 Basic Simulink Modeling of the Servo System

In addition to calculations presented in previous chapter, a SIMULINK model is created to cross check the results and examine the feasibility of available sampling frequency options on the hardware. The block diagram of the model is shown in Figure 4.1.

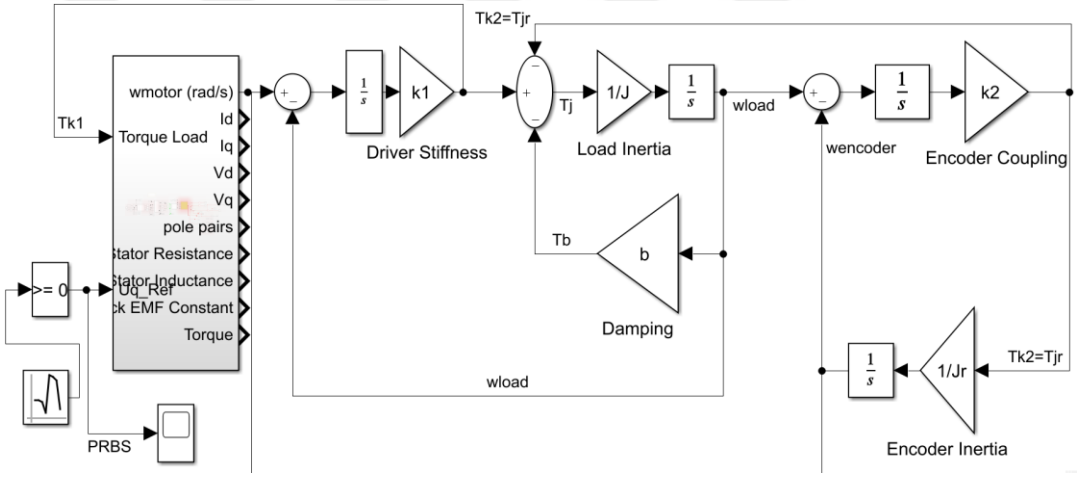

Figure 4.1. Block diagram of SIMULINK model

In addition to sampling frequency, it is also important to determine the random signal generation period for current signal. Generation period is described as the time interval between consecutive random binary signals. PRBS is an inherently discrete signal type, and its generation period has a direct effect on input and output velocities. Selecting very large periods may cause insufficient excitation of the system while selecting very narrow periods may lead to loss of information. Multiple simulation runs are conducted to pick the best possible period.

It is concluded that picking a generation period that is less than hundred times of the sampling period is not effective and must be avoided. A PRBS signal with 100ms of generation period is illustrated in Fig. 4.2.

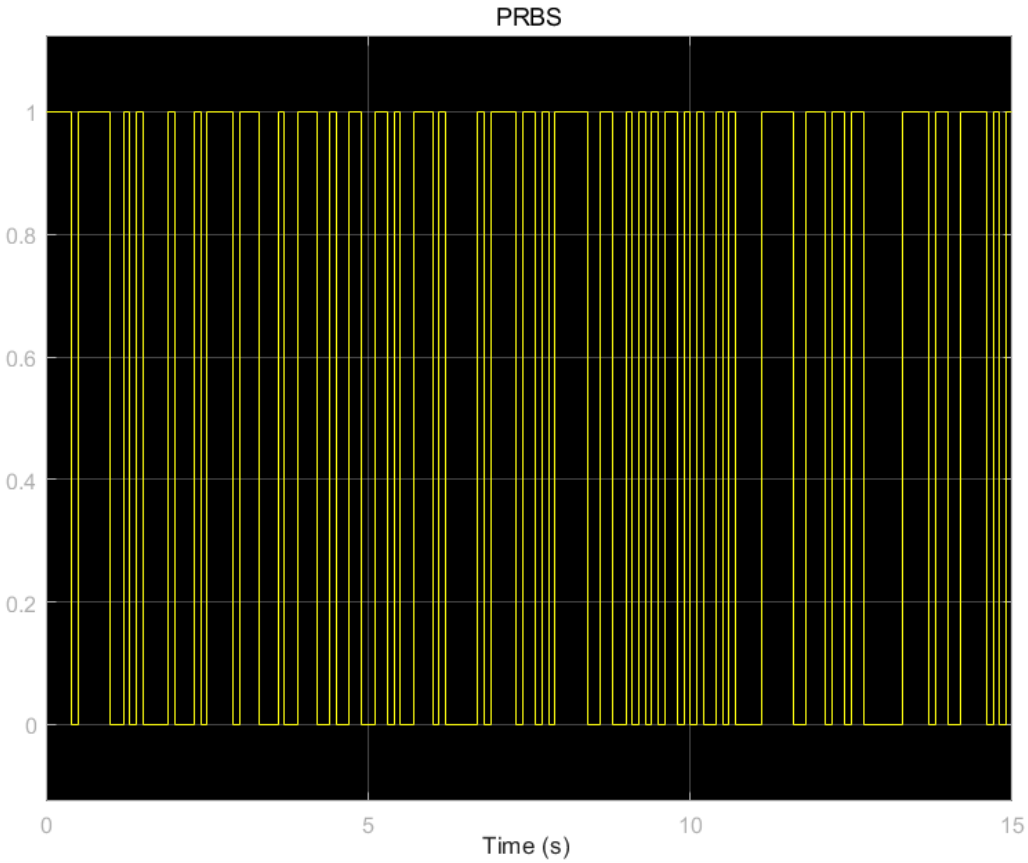

Figure 4.2. A pseudo random binary signal with 100ms generation period

Once the excitation signal is adjusted for sampling frequency, frequency response of modeled system is obtained through discrete transfer function estimator block. Parameters discussed in Chapter 3 are entered into simulation. Discrete transfer estimator block is fed with input and output values at 1 kHz. Simulation is run with a fixed step time of 1 microsecond. In addition to these predictive studies, SIMULINK model is extensively used as a basis for various data processing algorithms that are intended for experiments.

The expected natural frequency of 58 Hz is observed in resulting model response. Figure 4.3 shows the output of estimator block. Since serial communication infrastructure of the test setup can be adjusted to acquire clock frequencies as high as 5 kHz, it is expected that available hardware can satisfy necessary clock speeds during actual test. Also, PRBS generation period of 50ms is chosen for actual tests.

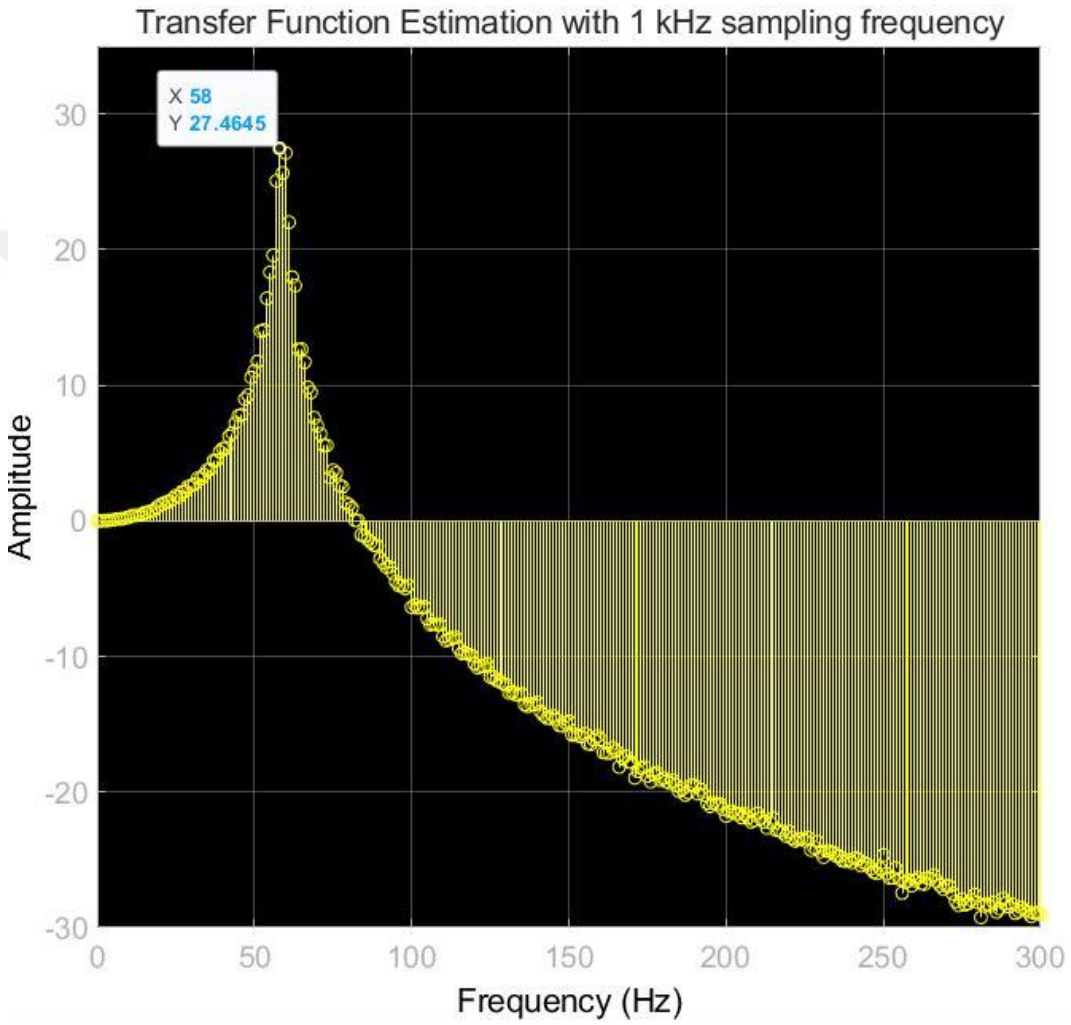

Figure 4.3. Frequency response of simulated system

4.2 Finite Element (FE) Simulation of the System

Up to this point, all the elements of studied servo drive are modeled as rigid bodies. Although predominant mode is not expected to originate from a flexible body motion except already modeled couplings, a supplementary analysis is conducted to perceive structural modes. Figure 4.4 illustrates the FE model of the experimental setup.

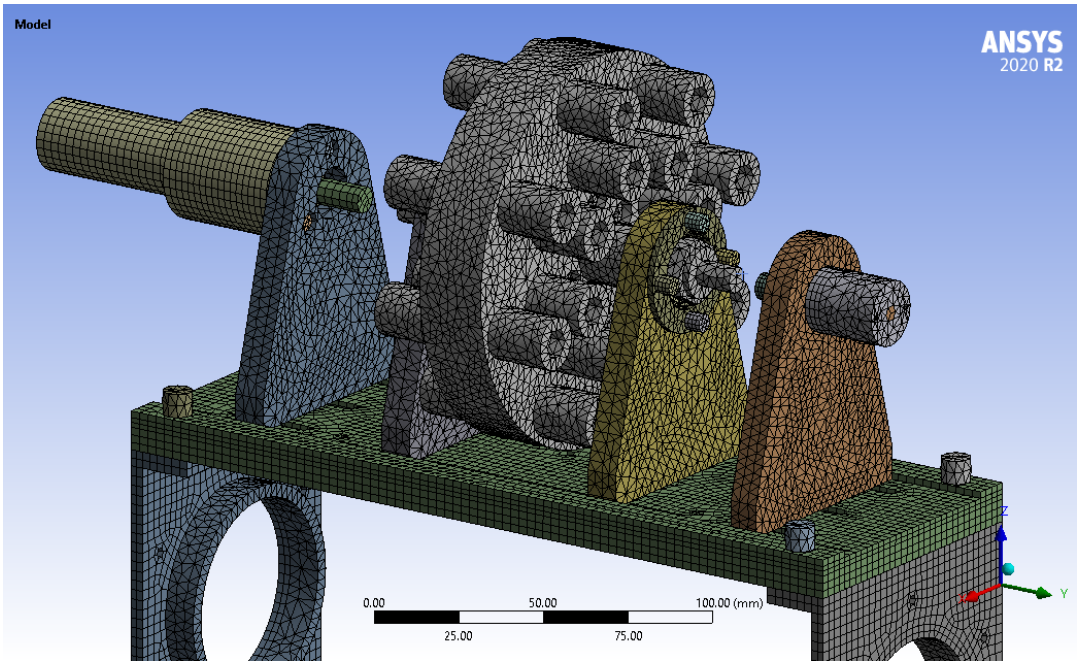

Figure 4.4. FE Model of the experimental setup

In this model, couplings are modeled as rotational springs. Stiffness value of springs are taken from datasheet specifications given in chapter 3. All other properties of the system are modeled with respect to CAD data.

As expected, first mode emerged in rotational motion of the load disc with a natural frequency of 57.5 Hz. Figure 4.5 shows the graphic representation of the first mode shape. Figure 4.6 shows the harmonic angular speed response of the load to the motor shaft. It can be concluded that other structural modes are observed to be unexcited for the rotational motion of the shafts. Thus, it can be stated that flexible effects during a rotational test do not constitute an error source for measurements.

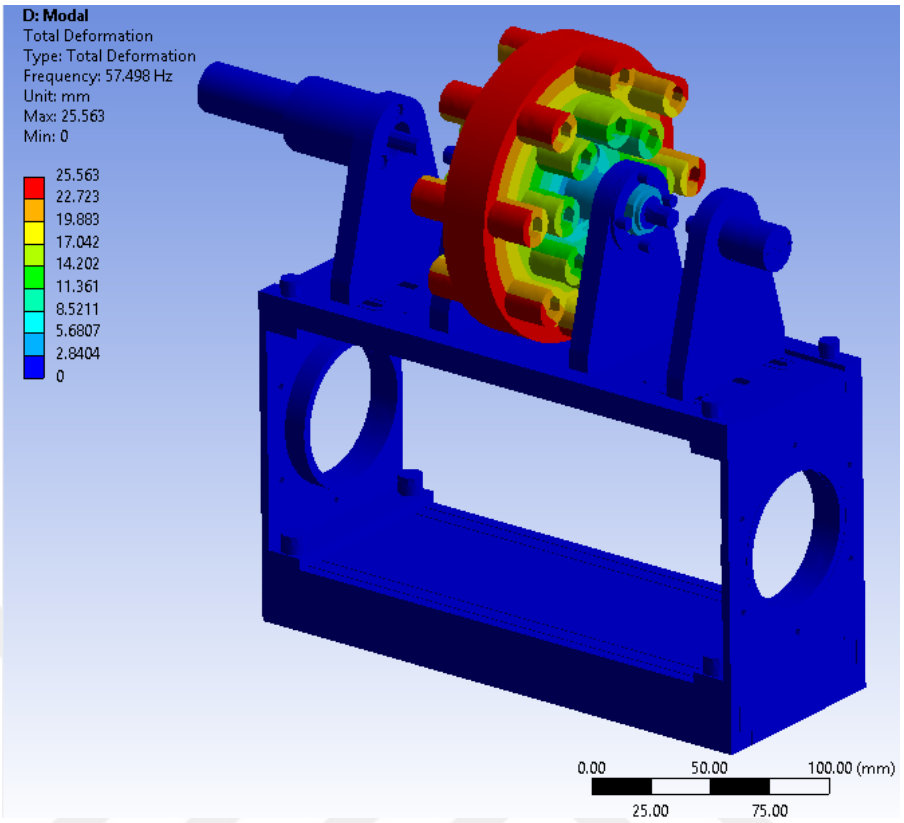

Figure 4.5. First mode shape of the experimental setup

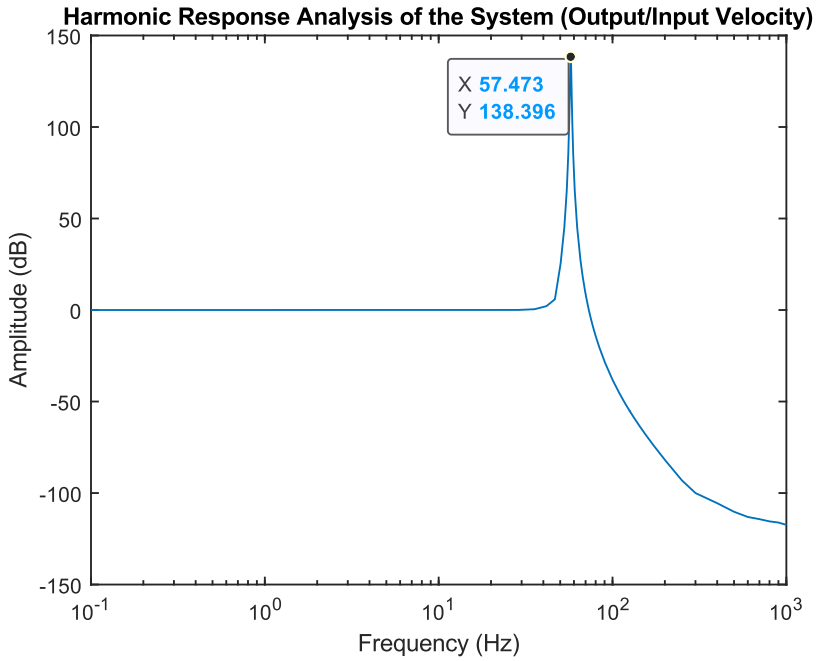

Figure 4.6. Harmonic response obtained from FE simulation

4.3 Summary of the Chapter

In this chapter, supplementary simulation and analysis procedures prior to testing is discussed. Selection of certain parameters such as generation period of excitation signal is conducted by taking advantage of simulated model. It is shown that better prediction of hardware requirements is possible with an accurate simulation model. Finally, a finite element simulation is performed in order to have a broader prospect on the system behavior.

CHAPTER 5

EXPERIMENTAL WORK

This chapter consists of four headings:

- General Outline of Experimental Setup

  Physical layout of the experimental setup is discussed under this heading.

- Test and Data Evaluation Procedures

  Detailed explanation of test results evaluation and data processing procedures are discussed under this heading.

- Impending Failure Test with Controlled Damaging

  Testing of an imitated gradual failure case is presented and its results are discussed under this heading.

- Summary of the Chapter

  Conclusions obtained from experiments are discussed under this heading.

5.1 General Outline of Experimental Setup

A physical setup is built for implementation of identification procedures discussed in previous chapters. The test setup includes a PMSM motor, an encoder, a custom design rotating load and other mechanical parts. In addition to these elements, a servo driver is used to drive the motor and read encoder measurements through SSI interface. Measurements taken by servo driver unit is sent to the computer via RS422 protocol with 5 kHz sampling frequency. Physical layout of the test setup is shown in Figure 5.1.

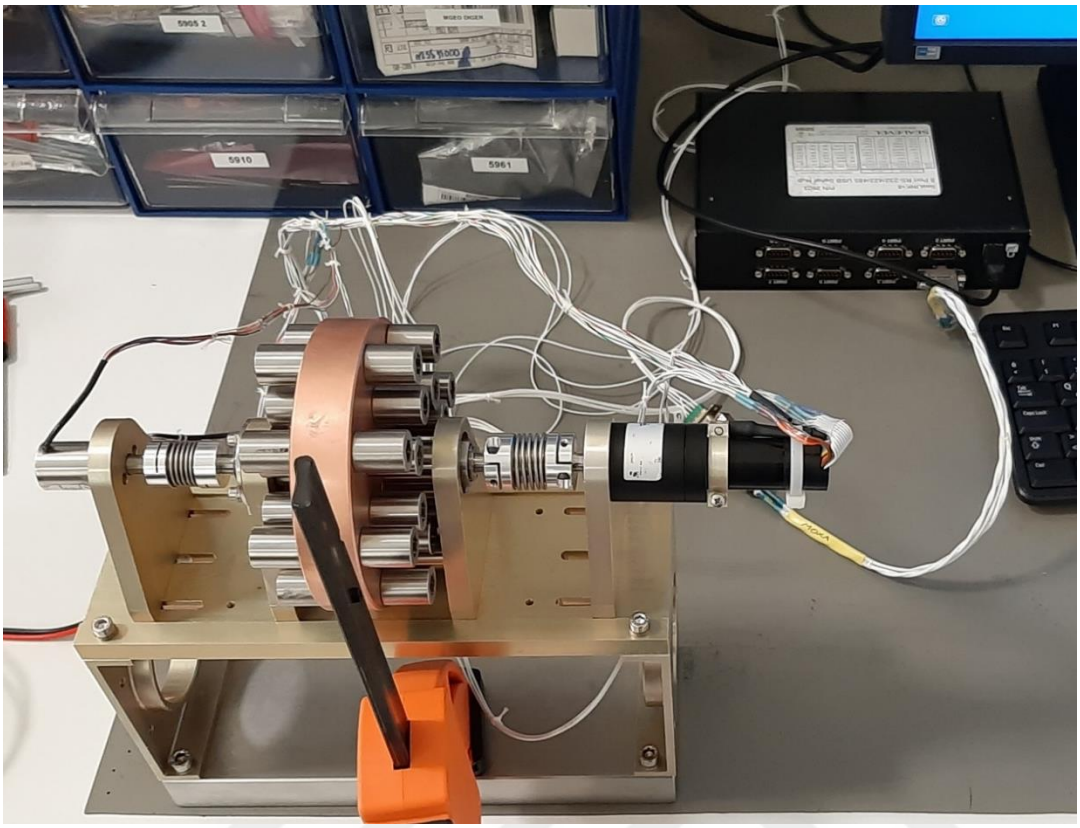

Figure 5.1. Test Setup for Servo System Identification Tests

As shown in Fig. 5.1, foundation of the test setup is made from thick steel block that is firmly fixed to the test bench with a clamp. Brackets that carry motor, encoder and the load are rigidly connected to the main plate with screws and locating pins. Main plate and the brackets are firmly connected to the foundation block and any clearance between parts that may cause unwanted motion during the tests is prevented. Bearings are mounted with fixed-free arrangement. Outer ring of the fixed bearing is constrained by a shoulder on the bracket and a bearing cap. Axial backlash of the shaft is taken away by axial tightening nuts on the shaft and hence only the rotational degree of freedom is left for motion. In addition, couplers are connected to the shafts by clamping hubs and radial marks are drawn upon connection to monitor any possible slippage. Throughout the tests, no observable slippage has occurred. Tightening nut and bearing cap together with clamping hubs of the coupling can be seen in Figure 5.2.

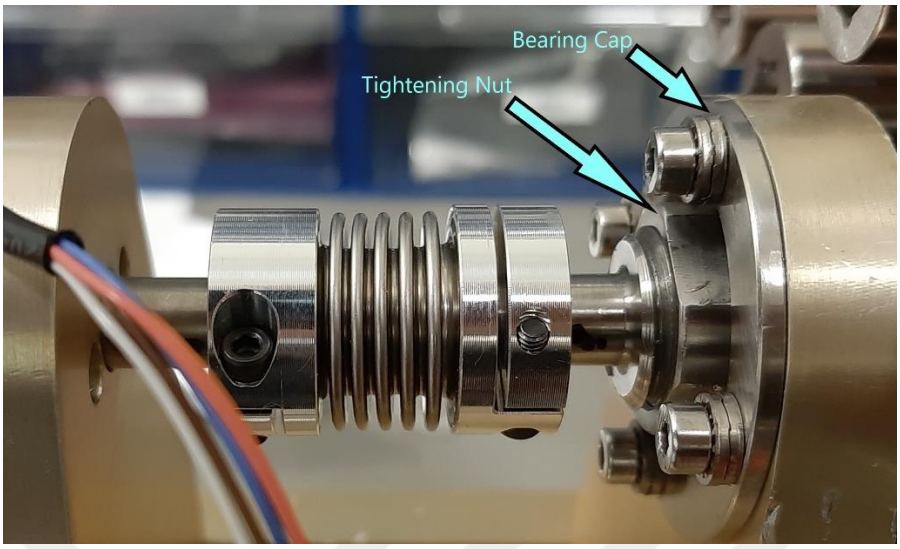

Figure 5.2. Clamping hubs of the coupler, tightening nut and the bearing cap

5.2 Test and Data Evaluation Procedures

In order to perform frequency response analysis on test results, obtained data must be processed prior to analysis. Raw data contains bitwise single-turn position information taken at moments separated by sampling period. Since angular speed of motor shaft will be related to angular speed of load in this study, raw data obtained from encoders must be processed to continuous angular speed in radians. Since both motor and load encoders are 16-bit absolute encoders, their raw output lies between 0 and 65535. 65535 refers to 1 angular bit before 360 degrees, and if the encoder completes a complete turn, its output comes back to zero. Hence, resolution of both encoders is 360/65536 degrees, which is approximately 0.0055 degrees. An example for a single-turn 16-bit absolute encoder output is given in Fig. 5.3. In Fig. 5.4, block diagram of transition procedure from raw data to angular speed is presented. First, raw bitwise data is checked for faulty signals. These data instances are corrected by simple linear regression in order to ensure local continuity. Then, single-turn position data is transformed to multi-turn position data by counting number of turns. After this step, bits are converted to radians. Subsequently, discrete position data is smoothened. As a result of these steps, processed position data is obtained. An example of processed position data is shown in Figure 5.5.

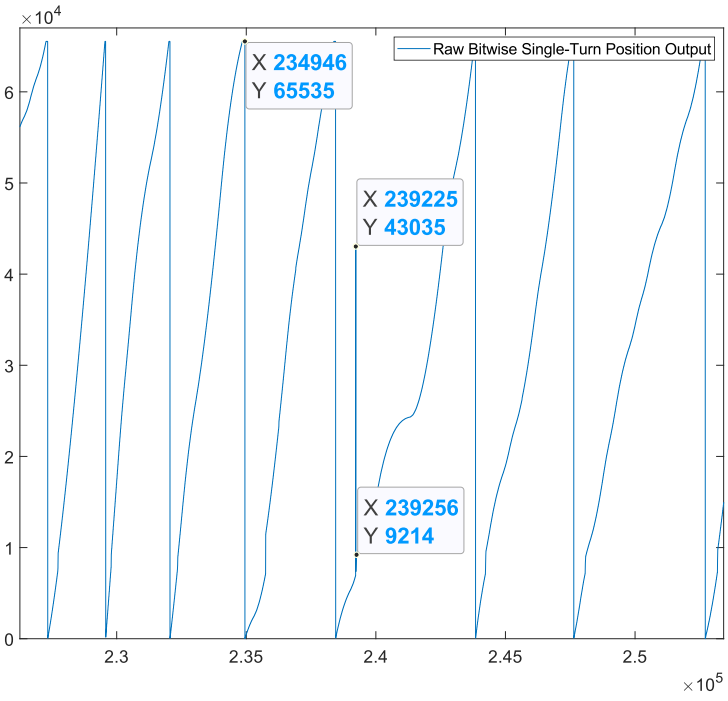

Figure 5.3. 16-bit absolute encoder output. Data no. 239225 is clearly erroneous.

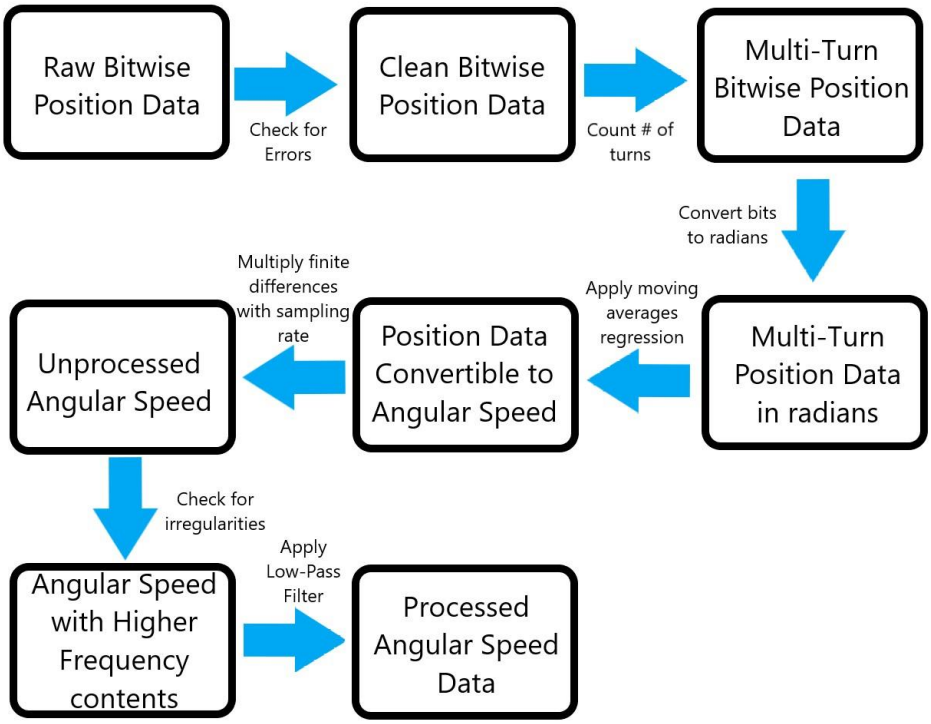

Figure 5.4. Flowchart of obtaining angular speed data from raw data

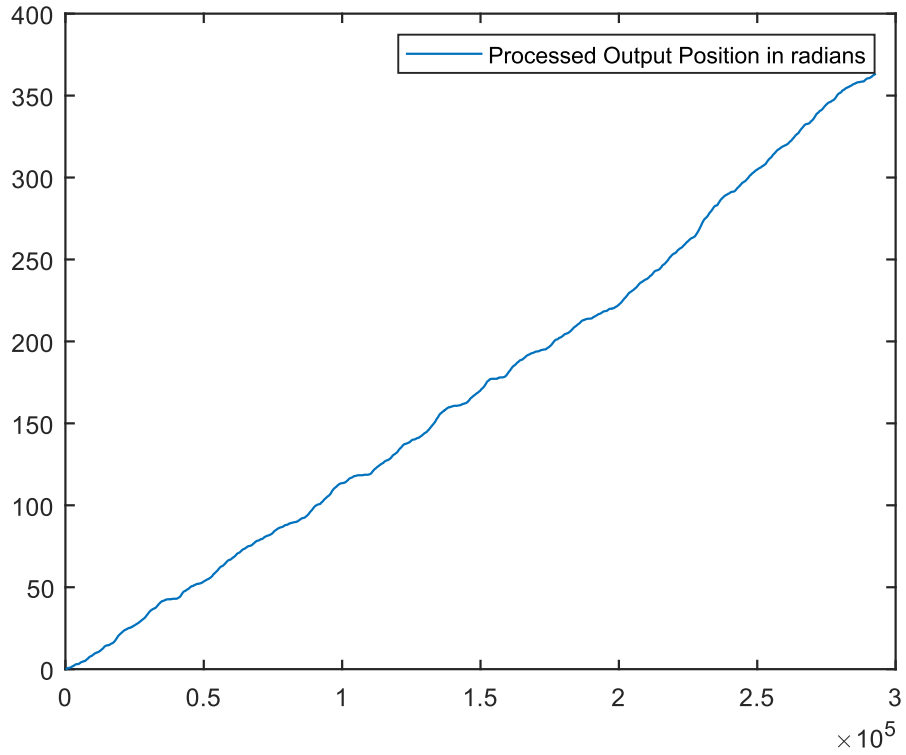

Figure 5.5. An example for processed position data

After obtaining motor and load side angular position data, angular speed can be calculated by multiplying finite differences of each data point with sampling rate. This is formulated in Eq. 5.1.

$$\omega = \frac{\Delta \boldsymbol{\theta}}{\Delta \boldsymbol{T}} = \Delta\theta \cdot f_s \qquad \text{Eq. 5.1}$$

Once angular speed data is obtained, it is important to be aware of the fact that high frequency contents or noisy signals may contaminate the data which are to be used in spectral analysis. Such contaminations may cause frequency analysis calculations to deviate from actual response characteristics of the system. In order to conduct spectral analyses with relatively clean data, an automatic irregularity detection algorithm is developed. Subsequently, low-pass filter is used to filter out the high frequency contents that are not expected to carry modal information. An example of processed angular speed data is shown in Figure 5.6.

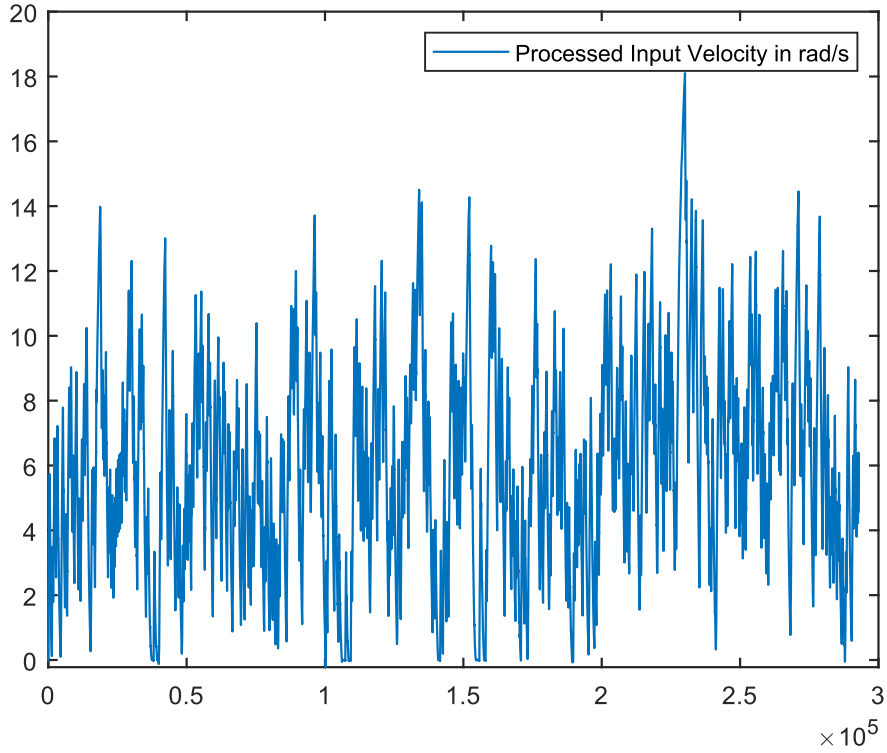

Figure 5.6. Example of a processed angular speed data in rad/s

Upon derivation of angular speed information of both input and output shafts, it is possible to perform spectral analysis to acquire frequency response characteristics of the system. However, a single momentary measurement may not reveal precise enough results due to instantaneous effects or noises. During in-situ identification applications, systems may only be available for measurements whenever triggered for action, such as sector sweeping gimbals or active seeking mechanisms. This requires getting multiple measurements and then averaging the frequency response of each measurement in order to obtain more accurate results. Once the measurements are obtained, it is possible to fit a transfer function to the averaged frequency response of the system for the timespan in which the measurements are taken. Afterwards, fitted transfer function can be used for approximation of system parameters and thus near real-time identification of the system can be performed. Flowchart of identification routine is presented in Figure 5.7.

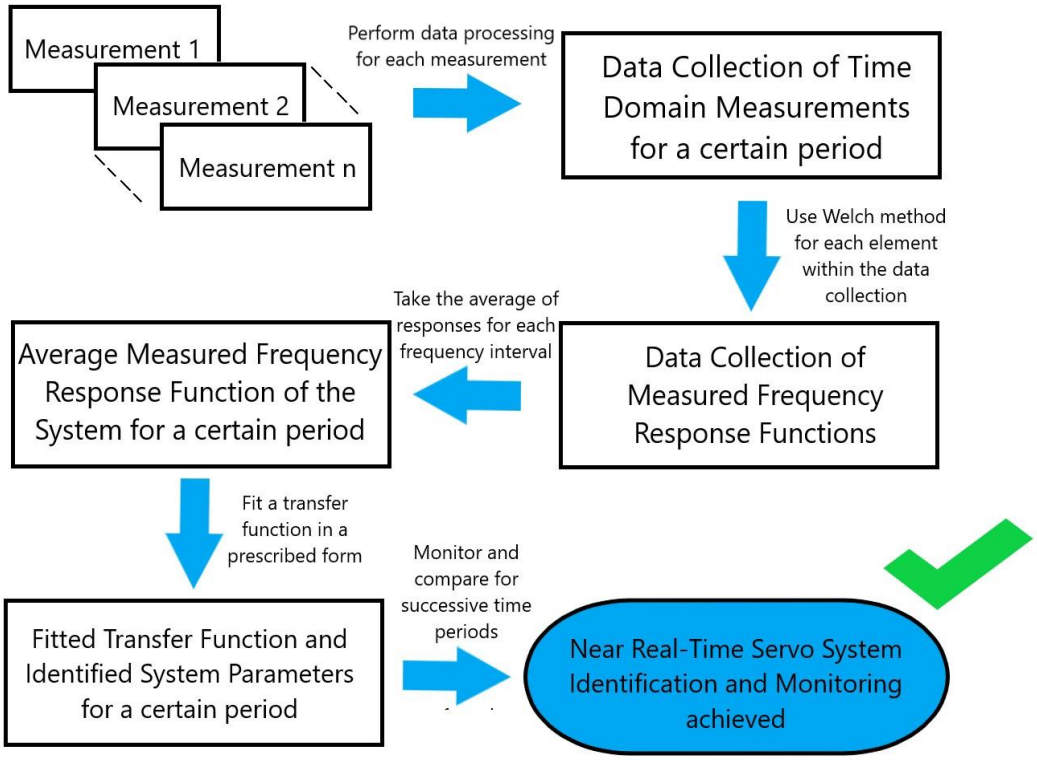

Figure 5.7. General outline of servo system identification procedure

The procedure explained in Fig. 5.7. can be embedded into an actual system with a dedicated processing unit, such as an FPGA based signal processing hardware, or a microcomputer connected to the servo driver. In this study, postprocessing is conducted on a different working station instead of an embedded processing unit that communicates with the system and performs near real-time analyses. However, this study forms a frame for such an application by presenting a proof-of-concept work that is obtained by monitoring results on case studies imitating real operating conditions. Thus, experiments conducted in this study aim to show that such an embedded system can offer an alternative to the other types of monitoring systems which require additional sensors.

Identification procedures explained in this chapter are performed automatically by MATLAB codes presented in Appendix B.

5.3 Impending Failure Test with Controlled Damaging

Following the completion of test setup, health monitoring experiments are conducted. Prior to tests with controlled damaging, a sacrificeable coupling is attached to motor shaft and series of aggressive signals were sent to the motor in order to observe a possible failure mode of the system. By this procedure, it is aimed to find out a realistic damage scenario on the driver coupling which can also occur during operation.

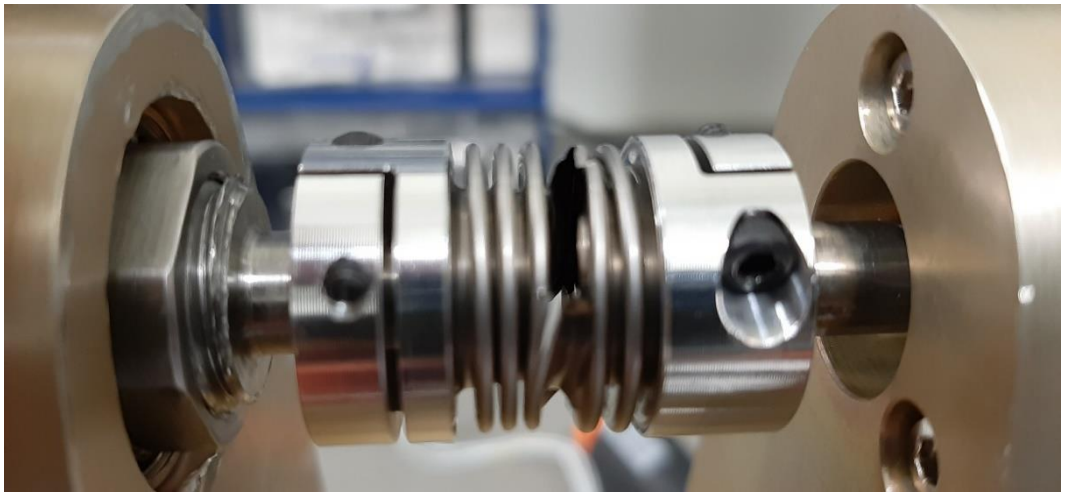

Figure 5.8. Bellow tearing is typical in bellow couplings due to fatigue loading.

These aggressive signals caused a propagating crack on a bellow of the driving coupling. Visual inspection of the coupling revealed that the crack propagation had stabilized after smoothening the signals, but an uncontrolled excitation could further damage and threaten the integrity of the element. Damage occurred on the coupling is shown in Figure 5.8. Revealing this failure mode provided an insight with regards to determining a method for gradual failure imitation. As a result, controlled and progressive tearing of metal bellows of the driving coupler is selected as the failure mode. Later, damaged coupling is removed and replaced with an unused coupling.

Before conducting the tests, all system elements are examined for inherent damage and verified for healthy functionality. Undamaged coupling is shown in Figure 5.9.

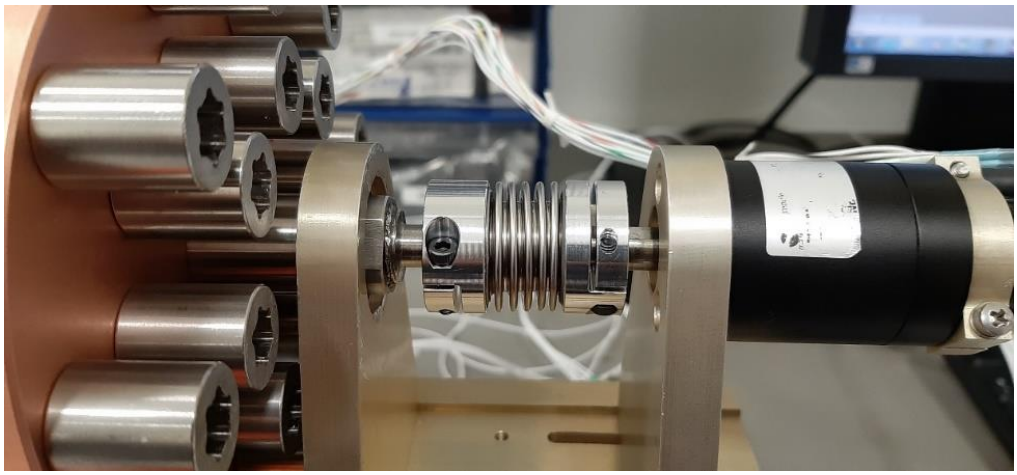

Figure 5.9. Undamaged driving coupler before testing

According to analytical calculations and simulation results, first natural frequency of the system was expected to be observed around 58 Hz. Initial result on frequency response of the system is illustrated in Figure 5.10.

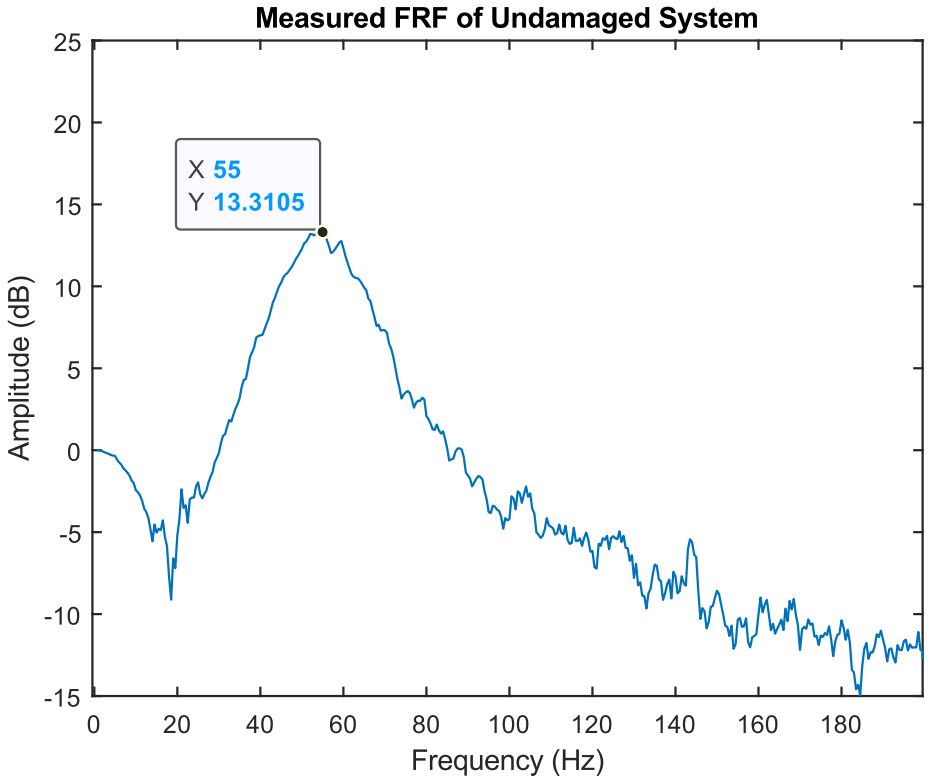

Figure 5.10. Amplitude of the frequency response of the system in decibels

As seen in Fig. 5.10, first natural frequency is observed around 55 Hz with considerable damping effect around the peak. It is possible that irregularities in driveline or asymmetry in rotating elements, though hardly noticeable, may have caused such a deviation from analytical expectation. After obtaining the experimental frequency response, parameter fitting procedure described in Chapter 3 is applied. Eq. 5.2 is the resulting transfer function.

$$G(s) = \frac{128302}{s^2 + 62.3s + 128302} \qquad \text{Eq. 5.2}$$

As explained in Chapter 3, single-mass representation of the system yields a transfer function that has zeroth order numerator and second order denominator, while having the same zeroth order coefficient for both. If the coefficient of the highest order term in the denominator is unity, this common coefficient gives the ratio of coupling stiffness to the rotational inertia of the load. This relationship can be formulated for calculating the coupling stiffness as described in Equations 5.3 and 5.4.

$$\frac{k}{J} = 128302 \qquad \text{Eq. 5.3}$$

$$k = J \times 128302 = 0.0055745 \times 128302 = 715.22 \ \frac{\text{Nm}}{\text{rad}} \qquad \text{Eq. 5.4}$$

Note that datasheet value for the stiffness of the coupling was 750 Nm/rad. This calculation deviates from specified value approximately by 5%. Such a deviation may not only be due to measurement error of the manufacturer and may indicate possible defects within test setup. However, determination of initial system parameters provides a good benchmark point as to identify future changes over time.

Frequency response of the estimated transfer function in Eq. 5.2 is illustrated in Figure 5.11. Notice the peak is occurring at 56.5 Hz, which can be interpreted as a close location to analytical and experimental findings for the peak location.

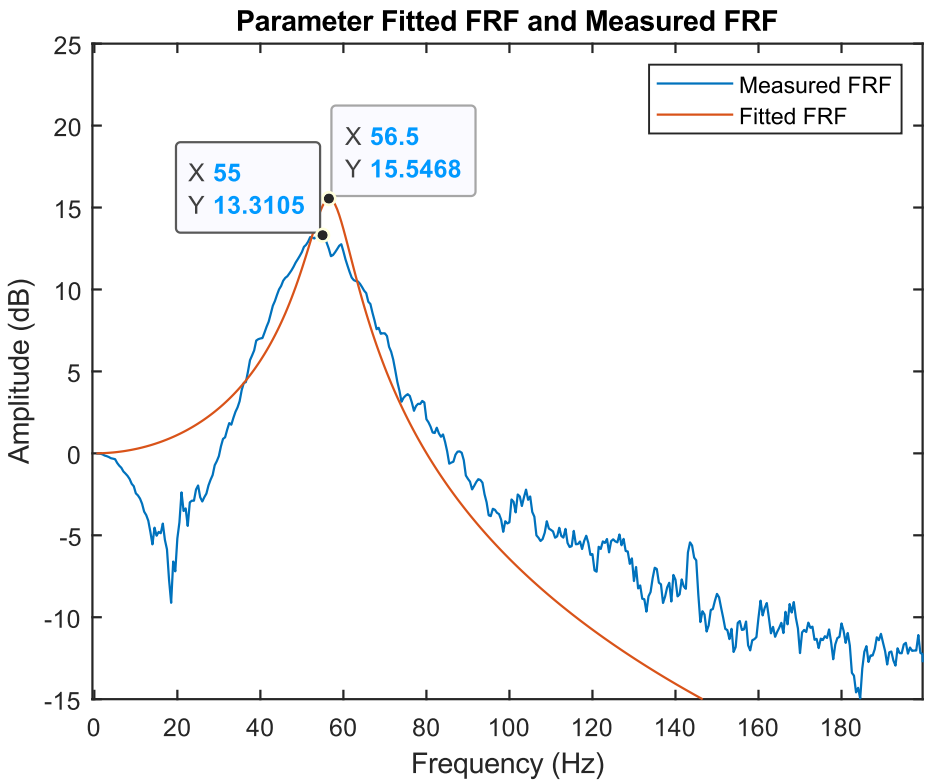

Figure 5.11. Frequency response of the fitted transfer function described in Eq. 5.2

After taking measurements for healthy system, controlled tearing of a single bellow on the coupling is performed with a handsaw. The figures 5.12 and 5.13 illustrate procedure and the resulting damage.

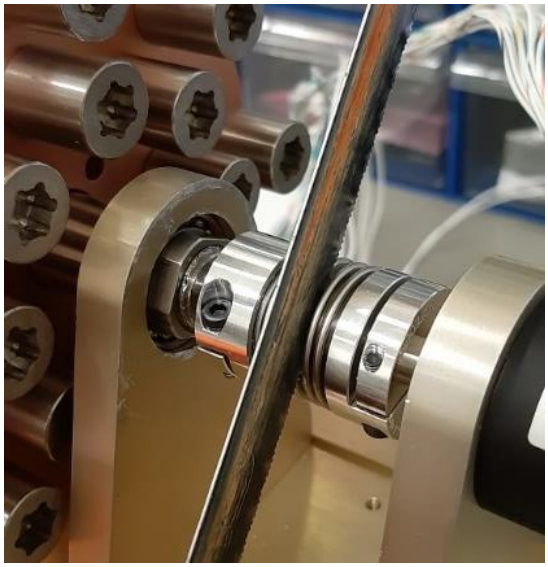

Figure 5.12. Cutting of a single bellow section on the metal bellow coupling

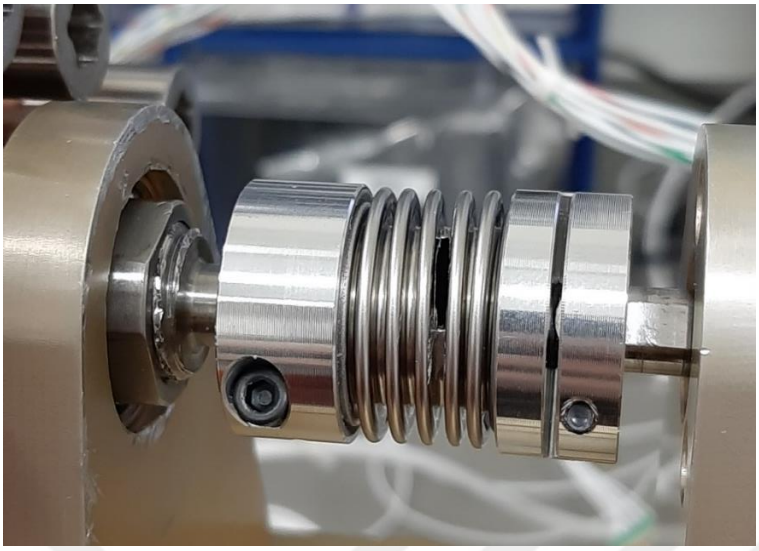

Figure 5.13. Initial damage on the coupling

Following first cut, tests conducted in order to observe expected changes. Fig. 5.14 shows the result of measurements on frequency response behavior of the damaged system.

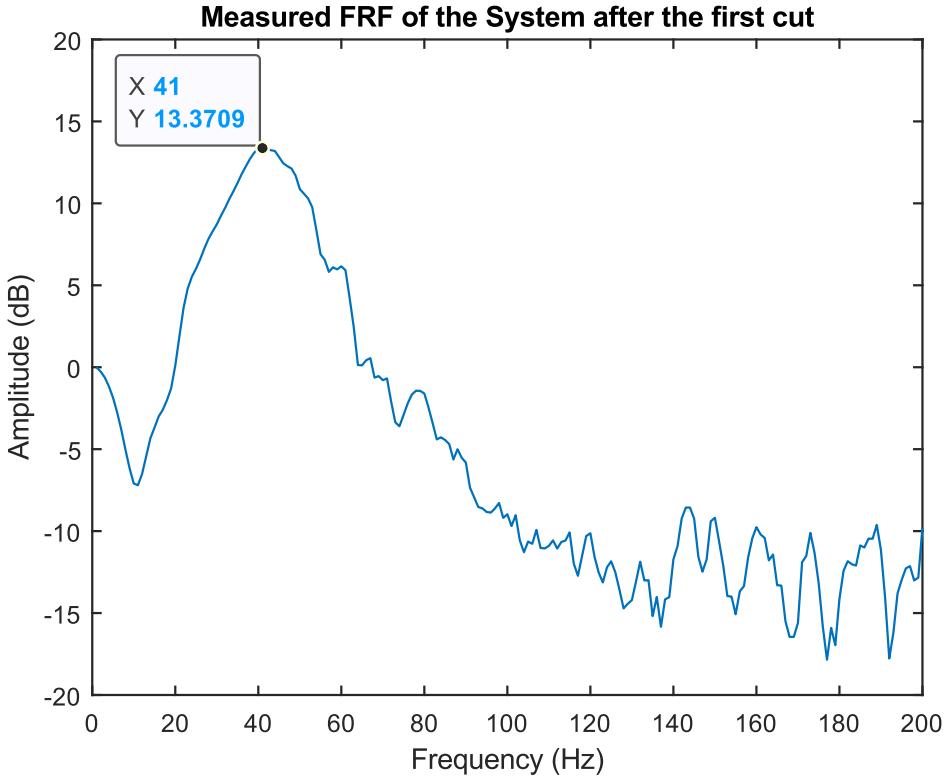

Figure 5.14. Frequency response of the system after first cut

Noticeable drop in the first natural frequency of the system can be observed in Figure 5.14. Once the results of the second test are obtained, parameter fitting is conducted to calculate the estimated transfer function of the system after partial damage. The resulting transfer function is given in Equation 5.5. Calculations for finding the change in the stiffness are presented in Eq. 5.6 and 5.7. FRF of the fitted transfer function is given in Figure 5.15.

$$G(s) = \frac{73027}{s^2 + 47.86s + 73027} \qquad \text{Eq. 5.5}$$

$$\frac{k}{J} = 73027 \qquad \text{Eq. 5.6}$$

$$k = J \times 73027 \;=\; 0.0055745 \times 73027 \;=\; 407.1\ \frac{Nm}{rad} \qquad \text{Eq. 5.7}$$

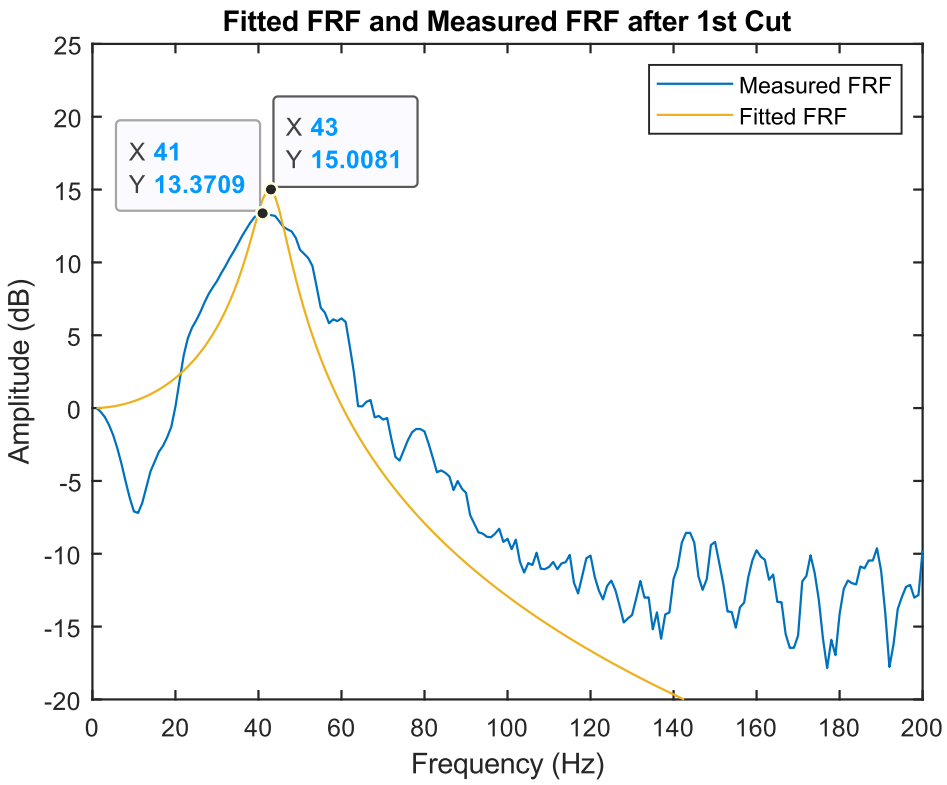

Figure 5.15. Frequency response of the fitted transfer function described in Eq. 5.5.

Notice that the peak occurs at 43 Hz, a point which stands slightly further by prominent peak of the measured FRF. It is important to realize that prominent peak of a measured FRF may not always indicate the existence of a modal frequency exactly on its spot, since resolution of averaged measurements may not be enough to catch exact location of the actual peak. Since the parameter fitting algorithm has a distributed error weighting along the frequency vector, the location of the natural frequency within the best fit may not exactly coincide with the measured FRF. However, overall error minimization of the fit satisfies the needed accuracy for estimation and yields convenient results. From Eq. 5.7, it is apparent that the driving coupler suffered considerable loss of stiffness due to tearing. After the initial damaging, another cut is applied to the coupling. In order to prevent premature failure, the secondary cut is located on a different bellow segment and directed at opposite side of the first cut. The situation of the coupling after the second cut is shown in Figure 5.16.

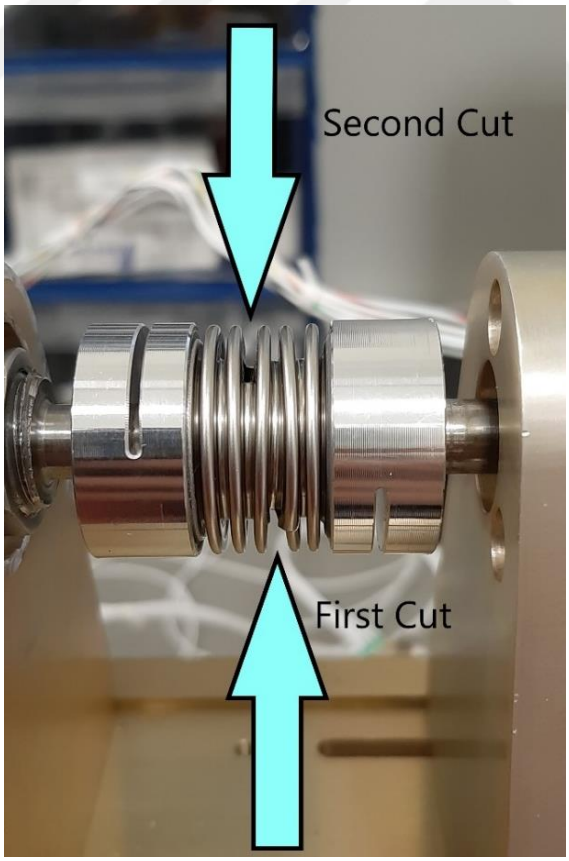

Figure 5.16. Secondary damage on the coupling

Fitted and measured FRF plots after secondary cut is given in Figure 5.17.

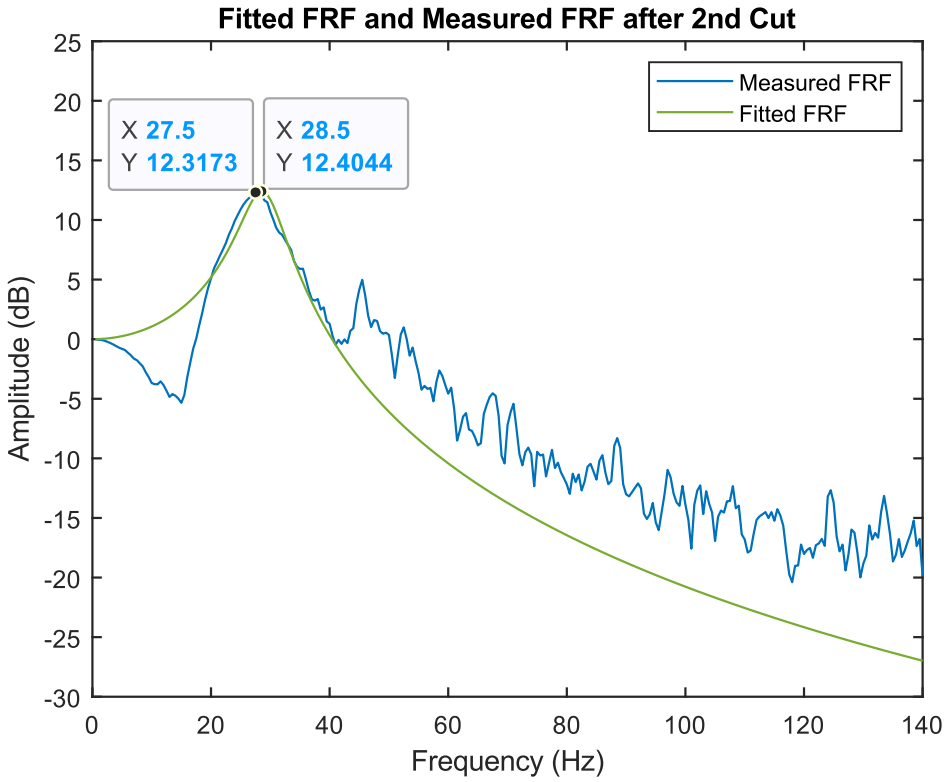

Figure 5.17. FRFs of fitted transfer function and the measurement after second cut

First natural frequency of the system after the second cut is observed to be at 27.5 Hz. The resulting transfer function is given in Equation 5.8. Calculations for finding the change in the stiffness is presented in Eq. 5.9 and 5.10. At this stage, system is being driven with a torque transmitting element that approximately lost 75% of its stiffness.

$$G(s) = \frac{33216}{s^2 + 43.12s + 33216} \qquad \text{Eq. 5.8}$$

$$\frac{k}{J} = 33216 \qquad \text{Eq. 5.9}$$

$$k = J \times 33216 \;=\; 0.0055745 \times 33216 \;=\; 185.16\ \frac{Nm}{rad} \qquad \text{Eq. 5.10}$$

After receiving such a damage, it is not possible for the coupler to transmit operating torques while retaining functionality. It is observed that after the second cut, nominal torque applied on the system is sufficient enough to cause quick propagation of cracks on the driving coupler. After the measurement illustrated in Figure 5.17, the stability of the coupler dropped significantly, and cracks propagated almost instantly. The last meaningful data received before total destruction of the coupling is shown in Figure 5.18.

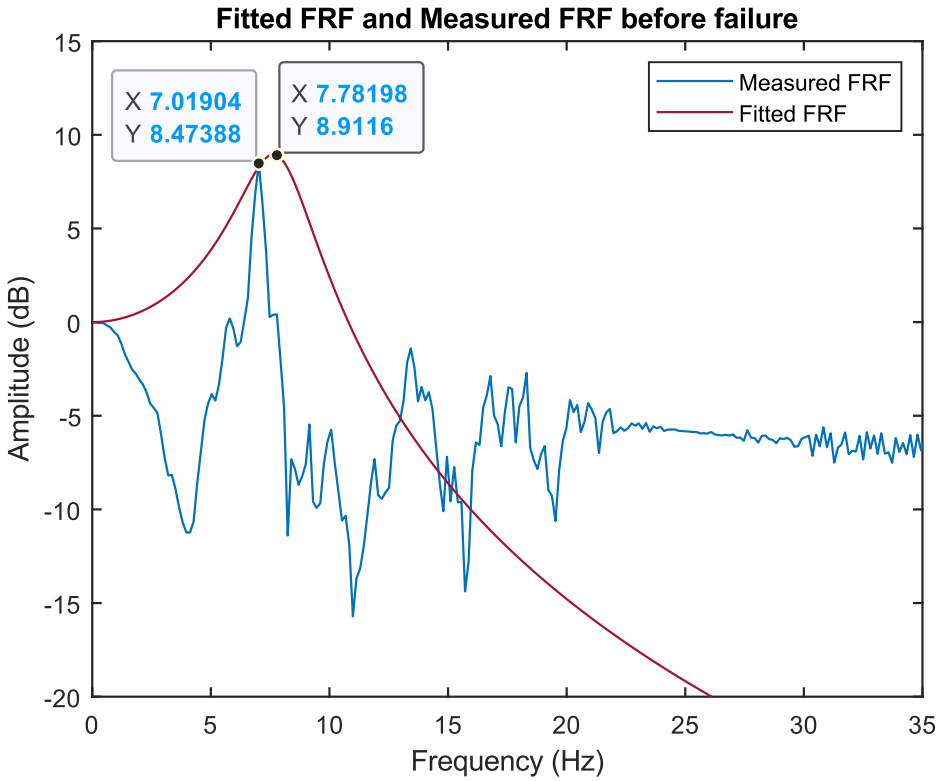

Figure 5.18. FRFs of fitted TF and the measurement prior to failure

Notice the other modes of vibration started to emerge as integrity of torque transmission parts degraded. This phenomenon may occur due to an increase in axial asymmetry, which develops further with crack propagation. After a certain point, parameter fitting may not be applicable due to nonlinear effects starting to dominate the response characteristics of the system. Nevertheless, it is observed that long paced track of an impending failure is possible with linear approximations.

Equation 5.11 is the resulting transfer function at an instant when the coupler is in its last stable form. Equations 5.12 and 5.13 express the derivation of stiffness estimate of the coupler before its destruction.

$$G(s) = \frac{2464}{s^2 + 17.46s + 2464} \qquad \text{Eq. 5.11}$$

$$\frac{k}{J} = 2464 \qquad \text{Eq. 5.12}$$

$$k = J \times 2464 \;=\; 0.0055745 \times 2464 \;=\; 13.73\ \frac{\text{Nm}}{\text{rad}} \qquad \text{Eq. 5.13}$$

Comparative plot for successive FRF measurements is presented in Figure 5.19 where the impending failure of the system can easily be seen. The plot for fitted FRFs of these measurements is shown in Figure 5.20.

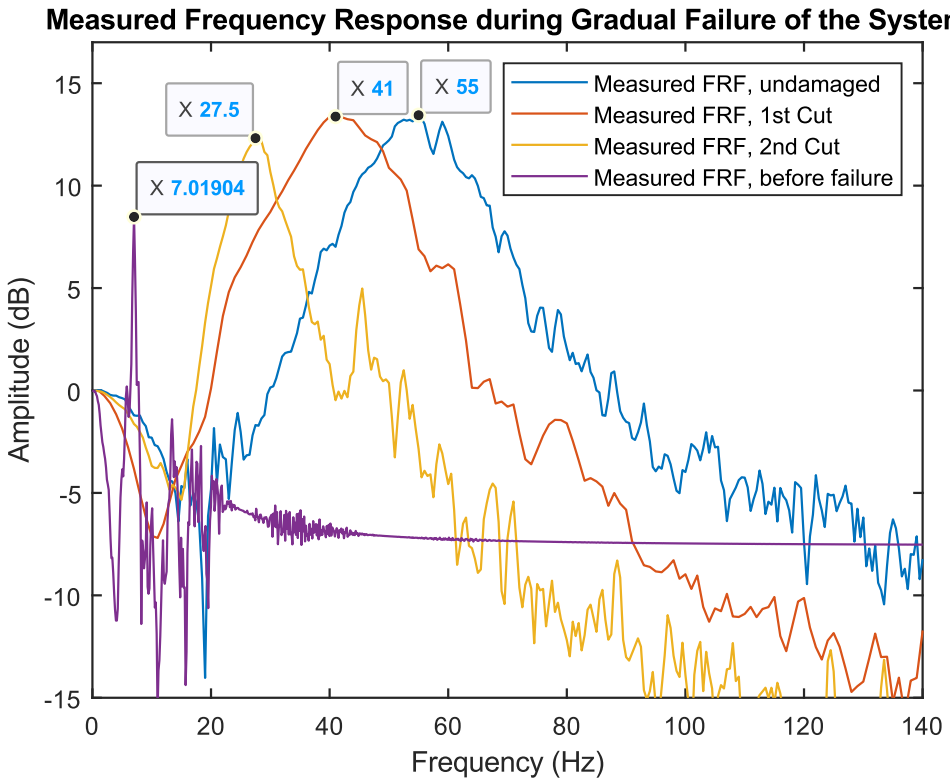

Figure 5.19. Comparative plot of Measured FRFs

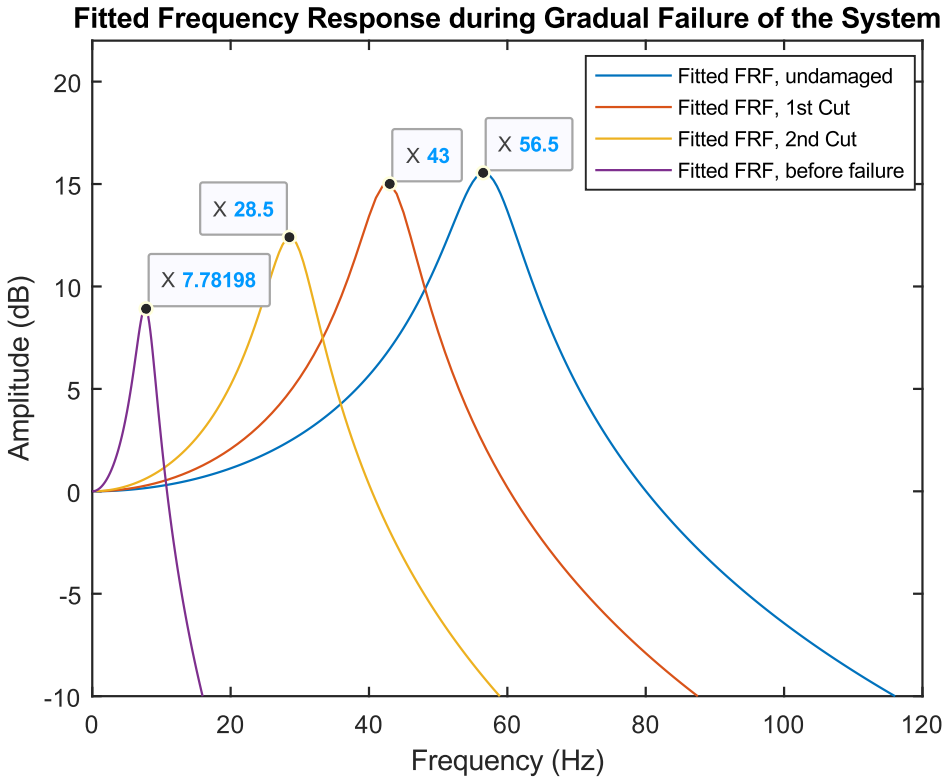

Figure 5.20. Comparative plot of Fitted FRFs

Following the identification of the system through input-output relationship of the servo drive, a modal impact hammer test is conducted in order to verify the obtained results. Since a torque impact induced by modal hammer is likely to excite first rotational mode, the consequent frequency response is expected to include a prominent peak around 7 Hz. The system during modal testing is shown in Fig. 5.21. Plot of the impact response of the system is given in Figure 5.22. Finally, the comparison of results obtained by different techniques for healthy and damaged states of the coupling are illustrated in Table 5.1 and 5.2.

Following the measurement of the last processable data, further continuation of the operation resulted in the complete destruction of the coupling shown in Figure 5.23. It can be concluded that suggested monitoring method is reliable and can provide up-to-date information for operational safety and predictive maintenance queries.

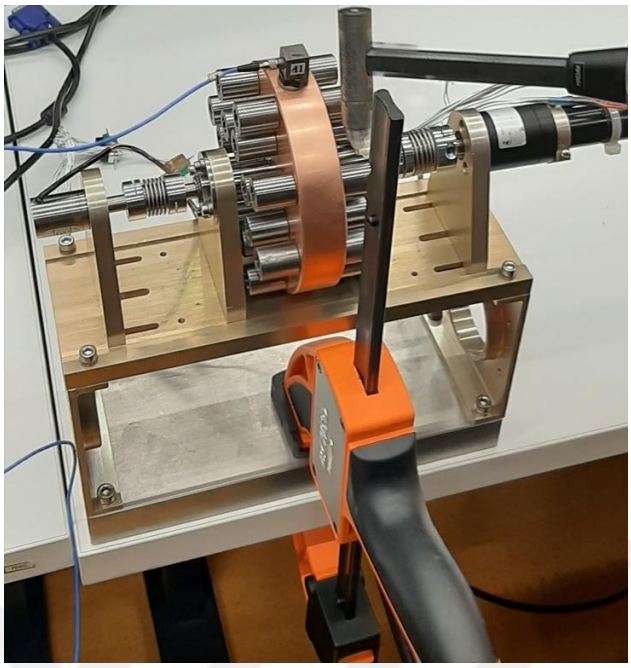

Figure 5.21. Impact hammer modal testing on experimental setup

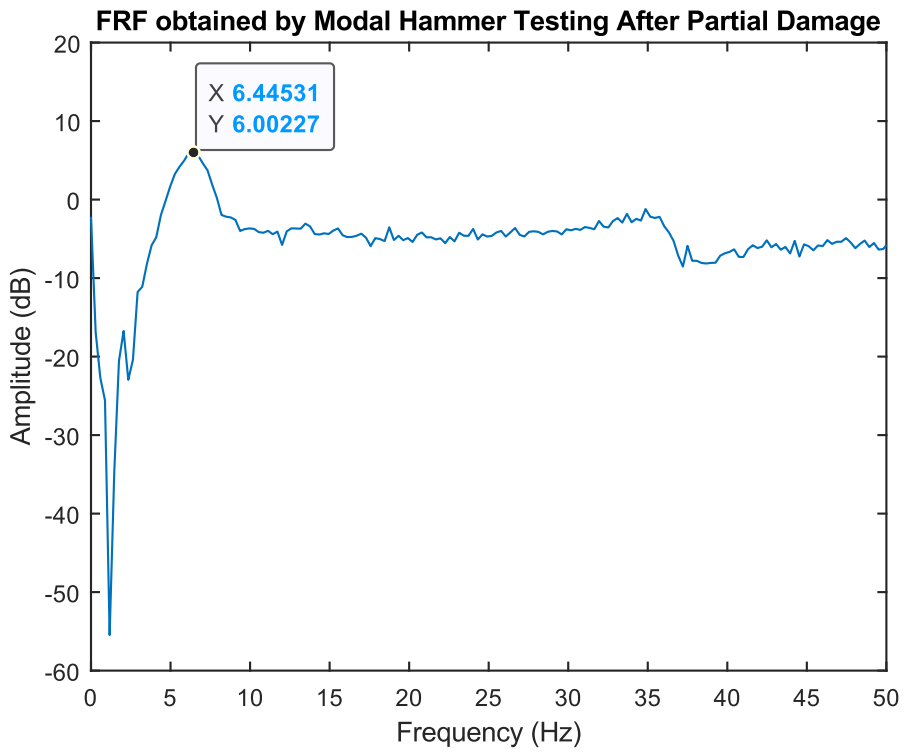

Figure 5.22. Frequency response obtained from impact hammer test

Prominent peaks occurring at 6.44 Hz in the hammer test, and 7 Hz in the second experiment indicate that two tests are consistent and yield similar results.

Table 5.1 Natural frequency of healthy system obtained by different methods

Method	Frequency (Hz)	Amplitude (dB)
Analytical Solution (undamped)	58.4	140.89
Discrete SIMULINK Solution	58	27.5
FE Analysis Solution (undamped)	57.5	138
Experimental Measurement	55	13.3
Curve Fitted Transfer Function Solution	56.5	15.5

Table 5.2 Natural frequency of damaged system obtained by different methods

Method	Frequency (Hz)	Amplitude (dB)
Modal Hammer Test	6.44	6.00
Experimental Measurement	7.02	8.47
Curve Fitted Transfer Function Solution	7.78	8.91

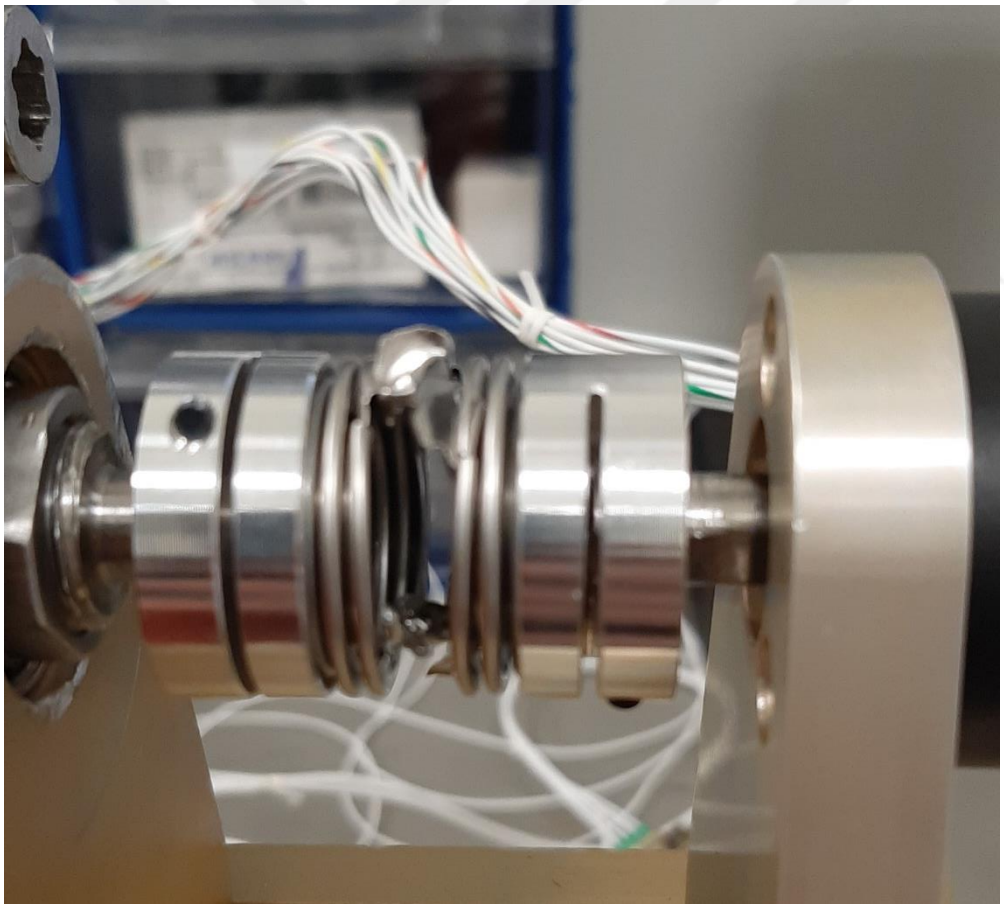

Figure 5.23. Completely damaged coupling

5.4 Summary of the Chapter

Experiments are conducted in order to demonstrate the effectiveness of an identification procedure for servo drives. The use of Welch method along with a least squares parameter algorithm yields a practical approach. Experiments show that establishing the input-output formulation as a speed-to-speed relationship is applicable and easily adaptable to many drivetrain types.

In addition to regular test results obtained from optical encoders, a modal test with impact hammer is performed with the intention of verifying obtained measurements. It is observed that both methods yield similar results.

Finally, a test on gradual degradation of the driver coupling is conducted for evaluating the system monitoring performance of the implemented procedure. It is concluded that the proposed methodology in this study can be used for monitoring systems that may be subject to progressive failure types.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

Primary goal of this thesis is to demonstrate that operational servo system monitoring can be achieved by utilizing existing sensors in the system. It is shown that in applications where continuous operation is deemed crucial, common industrial encoders can be used to track the health of systems efficiently. This approach provides high flexibility in design considerations due to absence of need for additional sensors or operator intervention.

Input-output relationship for a servo system is established only with rotational parameters. Formulation of such a relationship introduces greater simplification in data processing and error minimizing since the need for actuator dynamics is removed from consideration. This also enables systems with different actuator types to be identified independent of their driving element's electrical layout. It is shown that system behavior can successfully be deduced by implementing speed-to-speed relationship even though system operates at relatively low speeds.

After explanation of model structure, a generalizable approach in analytical modeling is presented. Significance of choosing the correct degree of freedom for system of interest is discussed. Following the explanation of background calculations, an experimental work is presented. It is shown that usage of pseudo random binary signals as excitation profile proved to be effective and convenient. This result also infers that usual operational control references utilized in seeker mechanisms and sector sweeping drives can easily be used for time domain measurements in order to perform periodic system identification.

It is important to note that certain amount of time must be allocated for identification procedure to be completed. In experiments explained in chapter 5, each data package is collected for 10 seconds of operation. Since averaging or concatenating multiple packets reduces noise effects and produces more accurate results, the time required to conveniently identify a system may increase further. Because of this reason, suggested method in this thesis does not provide a solution to identify sudden failures instantly. Although the suggested method can be used with smaller data packages sampled in shorter time periods, accuracy and repeatability of acquired results may not be convenient for system monitoring.

In addition to aforementioned limitations, it must be noted that any unmodeled system property may cause incorrect identification of system parameters. For instance, the effect of crack propagation on shafts cannot be identified since shafts in this thesis are assumed to be rigid and unchanging during the operation.

Also, the detectable range of natural frequencies directly depends on the equipment used for data acquisition. For a sampling rate in the order of couple MHz, it is possible to detect most characteristic natural frequencies of a servo drive. However, higher sampling rates may affect undesired delays in servo control itself due to frequent transmission of data. An embedded identification technique can bring considerable contributions to a system as long as available infrastructure is confirmed to be feasible.

Advantages of the proposed method include the flexibility to extend the application for any higher order system with different elements, such as gears, belts, chains or energy transformation units like hydraulic valves. Suggested procedure also provides a chance to compare the system with analytical expectations, instead of blind estimations. Thus, utilizing a predefined model structure as a foundation for system identification enables to interpret obtained results conveniently.

6.2 Future Work

Detection of gradual changes in a servo system is shown in this study, however, measured signals are not processed in an embedded physical unit. The performance of developed system monitoring procedure can further be understood with an attached programmable device which can communicate with the servo driver. Although practicality of the approach is made clear in overall terms, exact requirements for a proper hardware can only be determined by studying real operational conditions. Because of that reason, this approach can be studied on an actual system and investigated for possible improvements.

It should be noted that excitation profile applied on the system in this study does not resemble any functional reference. However, in an actual system, small random perturbations can be added to the velocity reference generated by controller. Performance of identification approach presented in this study can be examined for such a scenario, which would provide a continuous and uninterrupted health monitoring capability while retaining operational functionality.

REFERENCES

[1] G. Ellis, Z. Gao, "Cures for Low-Frequency Mechanical Resonance in Industrial Servo Systems", IEEE Industry Applications Society Annual Meeting, 2001.

[2] M. Pacas, S. Villwock, "Development of an Expert System for Identification, Commissioning and Monitoring of Drives", 13th International Power Electronics and Motion Control Conference, 2008.

[3] H. Zoubek, M. Pacas, "Encoderless Identification of Two-Mass-Systems Utilizing an Extended Speed Adaptive Observer Structure", IEEE Transactions on Industrial Electronics, Vol. 64, No.1, 2017.

[4] "Damping in a Rolling Bearing Arrangement". [Online]. Available: https://evolution.skf.com/damping-in-a-rolling-bearing-arrangement/. [Accessed: 02-Jan-2022].

[5] D. Bansal, D. J. Evans, B. Jones "A real-time predictive maintenance system for machine systems", International Journal of Machine Tools & Manufacture, Vol. 44, pp. 759-766, 2004.

[6] "The SKF model for calculating the frictional moment" [Online]. Available: https://www.skf.com/binaries/pub12/Images/0901d1968065e9e7-The-SKF-model-for-calculating-the-frictional-moment_tcm_12-299767.pdf [Accessed: 01-Feb-2022].

[7] "Adjoint matrix", Encyclopedia of Mathematics, ISBN 1402006098.

[8] K. Xu, X. Wu, X. Liu, D. Wang "Identification of Robot Joint Torsional Stiffness Based on the Amplitude of the Frequency Response of Asynchronous Data", Machines 2021, Vol. 9, p. 204, 2021.

[9] A. Carcaterra, W. D'ambrogio "An Iterative Rational Fraction Polynomial Technique for Modal Identification", Meccanica, Vol. 30, pp. 63-75, 1995.

[10] E. Madsen, O. S. Rosenlund, D. Brandt “Comprehensive modeling and identification of nonlinear joint dynamics for collaborative industrial robot manipulators”, Control Engineering Practice. , Vol. 101, 104462, 2020.

[11] L. Morales-Velazquez, R. Romero-Troncoso, R. A. Osornio-Rios, G. Herrera-Ruiz, J. J. de Santiago-Perez “Special purpose processor for parameter identification of CNC second order servo systems on a low-cost FPGA platform”, Mechatronics, Vol. 20, pp. 265-272, 2010.

[12] Z. Yumrukçal, E. Söylemez “Dynamic modeling of High Precision Servo Systems with Gear Backlash”, Middle East Technical University, 2013.

[13] A. Panda, M. Jancik, M. Behun “Transmission Error In The Systems of Mechatronics”, Transfer inovácií, Vol. 24, 2012.

[14] J. R. Stack, T. G. Habetler, R. G. Harley “Fault Classification and Fault Signature Production for Rolling Element Bearings in Electric Machines”, IEEE Transactions on Industry Applications, Vol. 40, No. 3, 2004.

[15] J. R. Stack, T. G. Habetler, R. G. Harley “An Amplitude Modulation Detector for Fault Diagnosis in Rolling Bearings Element Bearings”, IEEE Transactions on Industrial Electronics, Vol. 51, No.5, 2004.

[16] S. Vukosavic, M. Stojic “Suppression of torsional oscillations in a high-performance speed servo drive”, IEEE Transactions on Industrial Electronics, Vol. 45, pp. 108-117, 1998.

[17] M. J. Casiano “Extracting Damping Ratio From Dynamic Data and Numerical Solutions”, NASA, No. TM-2016-218227, 2016.

[18] P. Schmidt, T. Rehm “Notch filter tuning for resonant frequency reduction in dual inertia systems”, Proc. of IEEE IAS, pp. 1730-1734, 1999.

[19] S. Villwock, M. Pacas “Application of the Welch-Method for the Identification of Two- and Three-Mass-Systems”, IEEE Transactions on Industrial Electronics, Vol. 55, No. 1, 2008.

[20] I. Müller, P. Mutschler “Two Reliable Methods for Estimating the Mechanical Parameters of a Rotating Three-Inertia System”, Institut für Stromrichtertechnik und Antriebsregelung, 2002.

[21] D. Rowell “Linear Graph Modeling: One-Port Elements”, Massachusetts Institute of Technology, Department of Mechanical Engineering, 2003.

[22] M. Östring, S. Gunnarson, M. Norrlöf “Closed Loop Identification of an Industrial Robot Containing Flexibilities”, Control Engineering Practice, Vol. 11, pp. 291-300, 2003.

[23] O. M. Arafa, G. A. Abdelaziz, M. I. Abu El-Sebah, A. Aly Mansour “Observer-based sensorless speed control of PMSM: A focus on drive’s startup”, Journal of Electrical Systems and Information Technology, Vol. 3, pp. 181-209, 2016.

[24] “Metal Bellows Couplings RINGFEDER GWB DKN”. [Online]. Available: https:// https://www.ringfeder.com/products/metal-bellows-couplings/gwb-dkn/. [Accessed: 20-Dec-2021].

[25] E. McCormick, H. Lang, C. W. de Silva “Dynamic Modeling and Simulation of a Four-wheel Skid-Steer Mobile Robot using Linear Graphs”, International Conference on Robotics: Current Trends and Future Challenges (RCTFC), 2016.

[26] D. Rowell, D. N. Wormley “System Dynamics: An Introduction”, Prentice Hall, ISBN 0132108089, 1997.

[27] L. Ljung “System Identification Toolbox for use with MATLAB”, The MathWorks Inc., 2011.

[28] F. Schütte, S. Beineke, H. Grotstollen, N. Fröhleke, U. Witkowski, U. Rückert, S. Rüping “Structure and Parameter Identification for a Two-Mass System with Backlash and Friction using a Self-Organizing Map”, University of Paderborn, 1997.

[29] N. Nevaranta, S. Derammelaere, J. Parkkinen, B. Vervisch, T. Lindh, M. Niemela, O. Pyrhönen “Online Identification of a Two-Mass System in Frequency Domain using a Kalman Filter”, Modeling, Identification and Control, Vol. 37, No. 2, pp. 133-147, 2016.

[30] D. Rowell “Linear Graph Modeling: State Equation Formulation”, Massachusetts Institute of Technology, Department of Mechanical Engineering, 2004.

[31] K. Ogata “Chapter 2: Mathematical Modeling of Control Systems”, Modern Control Engineering, 5th ed., Prentice Hall, 2013, p. 44, eq. 2-29.

[32] S. W. Smith "Chapter 8: The Discrete Fourier Transform", The Scientist and Engineer's Guide to Digital Signal Processing (2nd ed.), California Technical Publishing, ISBN 978-0-9660176-3-2, 1999.

[33] M. H. Richardson, D. L. Formenti “Parameter Estimation from Frequency Response Measurements using Rational Fraction Polynomials”, 1st IMAC Conference, 1982.

APPENDICES

A. MATLAB SCRIPTS FOR ANALYTICAL CALCULATIONS

A-1: Natural Frequency Calculation for 2DOF Systems modeled as in Figure 3.2.

```
function [f1,f2,ph1,ph2] = analytical_nat_freq(k1,k2,b,J,Jr)

p1 = J*Jr ;
p2 = Jr*b ;
p3 = J*k2 + Jr*k1 + Jr*k2 ;
p4 = b*k2 ;
p5 = k1*k2 ;

p = [p1 p2 p3 p4 p5] ;

r = roots(p);

f1 = abs(imag(r(1)))/(2*pi) ; %nat. freq. of 1st mode
f2 = abs(imag(r(3)))/(2*pi) ; %nat. freq. of 2nd mode
ksi1 = sqrt(1/((imag(r(1))/real(r(1)))^2+1)) ;
ksi2 = sqrt(1/((imag(r(3))/real(r(3)))^2+1)) ;

ph1 = atan(sqrt(1-ksi1^2)/ksi1)*180/pi ; %phase of 1st mode
ph2 = atan(sqrt(1-ksi2^2)/ksi2)*180/pi ; %phase of 2nd mode

end
```

A-2: Analytical FRF Calculation for 2DOF Systems modeled as in Fig. 3.2.

```
k1=750;
k2=750;
Jcoupler = 0.0011*10^(-3) ;
J=0.0055734+Jcoupler;
Jr=0.000000073+Jcoupler/2;
b=0;
f = [1:0.1:10000];
w = 2*pi()*f;
C = [1 0 0 0];
B = transpose([0 0 k1 0]);
A=[-b/J 0 1/J -1/J ; 0 0 0 1/Jr ; -k1 0 0 0 ; k2 -k2 0 0];
G = [1:length(w)];

for k=1:length(w)

    G(k)= C*inv(1i*w(k)*eye(4)-A)*B;

end

figure
semilogx(f,mag2db(abs(G)))
title('Analytical Frequency Response of the System')
xlabel('Frequency (Hz)')
ylabel('Amplitude (dB)')
grid on
```

A-3: Explicit Transfer Function Derivation from State Space Representation

```
syms s k1 k2 b Jl Jm
C = [0 1 0 0];
B = transpose([0 0 k1 0]);
A=[0 0 1/Jm -1/Jm ; 0 -b/Jl 0 1/Jl ; -k1 0 0 0 ; k2 -k2 0 0];
G(s)= C*inv(s*eye(4)-A)*B;
```

A-4: Rational Fraction Polynomial Method

```
function parameters=rfp_method(m_frf,omega,m,n)

[r,c]=size(omega);
if r<c
    omega=omega.';
end
[r,c]=size(m_frf);
if r<c
    m_frf=m_frf.';
end
scaled_omega=max(omega);
omega=omega./scaled_omega;
[PHI,coeff_A]=ortho_basis(m_frf,omega,1,m);
[THETA,coeff_B]=ortho_basis(m_frf,omega,2,n);
[~,c]=size(PHI);
Phi=PHI(:,1:c);
[~,c]=size(THETA);
T=sparse(diag(m_frf))*THETA(:,1:c-1);
W=m_frf.*THETA(:,c);
X=-2*real(Phi'*T);
G=2*real(Phi'*W);
d=-inv(eye(size(X))-X.'*X)*X.'*G;
C=G-X*d;
D=[d;1];

A=coeff_A*C;
[r,~]=size(A);
A=A(r:-1:1).';
B=coeff_B*D;
[r,~]=size(B);
B=B(r:-1:1).';

[R,P,~]=residue(A,B);
[r,~]=size(R);
for n=1:(r/2)
   RES(n,1)=R(2*n-1);
   POLS(n,1)=P(2*n-1);
end
[r,~]=size(RES);
RES=RES(r:-1:1)*scaled_omega;
POLS=POLS(r:-1:1)*scaled_omega;
    freq=abs(POLS);
    damp=-real(POLS)./abs(POLS);
Ai=-2*(real(RES).*real(POLS)+imag(RES).*imag(POLS));
Bi=2*real(RES);
const_modal=complex(Ai,abs(POLS).*Bi);
    Ci=abs(const_modal);
    Oi=angle(const_modal).*(180/pi);

parameters=[freq, damp, Ci, Oi];
```

```
function [POLYS,COES]=ortho_basis(m_frf,omega,numorden,max_order)

if numorden==1
    q=ones(size(omega));
elseif numorden==2
    q=(abs(m_frf)).^2;
else
    error('Enter 1 for numerator or 2 for denominator.')
end
R_minus1=zeros(size(omega));
R_0=1/sqrt(2*sum(q)).*ones(size(omega));
R=[R_minus1,R_0];
COES=zeros(max_order+1,max_order+2);
COES(1,2)=1/sqrt(2*sum(q));

if max_order>0
    for k=1:max_order
    Vkm1=2*sum(omega.*R(:,k+1).*R(:,k).*q);
    Sk=omega.*R(:,k+1)-Vkm1*R(:,k);
    Dk=sqrt(2*sum((Sk.^2).*q));
    R=[R,(Sk/Dk)];
    COES(:,k+2)=-Vkm1*COES(:,k);
    COES(2:k+1,k+2)=COES(2:k+1,k+2)+COES(1:k,k+1);
    COES(:,k+2)=COES(:,k+2)/Dk;
    end
end
R=R(:,2:max_order+2);
COES=COES(:,2:max_order+2);

i=sqrt(-1);
for k=0:max_order
   POLYS(:,k+1)=R(:,k+1)*i^k;
   jk(1,k+1)=i^k;
end
COES=(jk'*jk).*COES;
```

B. MATLAB CODES FOR EXPERIMENTAL SETUP IDENTIFICATION

```
function
[tf,f]=AutoFRF(folder,fs,pre,post,figures,windowratio,roverlap,upto
,res)

Files=dir(folder);
data = {};
for k=1:length(Files)
   FileName=Files(k).name;
   if contains(FileName,'.csv')
        data = AddData(data,FileName,fs,pre,post,figures);
   end
end


[tf,f]=AvgTF(data,windowratio,roverlap,upto,res);




end

function [avgtf,f]=AvgTF(data,windowratio,roverlap,upto,res)

[tfs,fs] = testdata(data,windowratio,roverlap);
[avgtf,f]=AvgPSD(tfs,fs,upto,res);

semilogx(f,mag2db(abs(avgtf)))
title('Averaged FRF')
xlabel('Frequency (Hz)')
ylabel('Magnitude (dB)')


end
```

```
function [TF,f] = AvgPSD(TFc,fc,upto,res)

whereami = ones(length(TFc),1);

f = zeros(upto/res,1);
ITF = zeros(length(TFc),length(f));
TF = zeros(length(f),1);

for i = 1:upto/res

   f(i)= i*res;
   disp(['Evaluating frequency of ',int2str(f(i)),'Hz...'])


   for j=1:length(TFc)
       flag=false;
       counter=0;
       for k=whereami(j):length(TFc{j})

           if fc{j}(k)<f(i)+(res/2) && fc{j}(k)>f(i)-(res/2)
             ITF(j,i)=ITF(j,i)+TFc{j}(k);
             flag=true;
             counter = counter + 1;
           elseif flag==true
               whereami(j)=k;
               break
           end
       end
       if counter==0
           ITF(j,i)=0;
       else
           ITF(j,i)=ITF(j,i)/counter;
       end

   end

   TF(i)=sum(ITF(:,i))/length(TFc);
end
end

function [TFcells,fcells]=testdata(data,windowratio,roverlap)

TFcells = cell(length(data),1);
fcells = cell(length(data),1);

for i=1:length(data)

[TFcells{i},fcells{i}]=WelchPSD(data{i}{1},data{i}{2},data{i}{3},wi
ndowratio,roverlap);
end
end
```

```
function [TF,f] = WelchPSD(input,output,fs,windowratio,roverlap)

if length(input)~=length(output)

   error('Input and Output arrays must be of same size!')

end

window = 2^(nextpow2(floor(length(input)*windowratio)));
truncation = mod(length(input),window);

disp(['Window length is ',int2str(window),'. (Windowing ratio
',int2str(window/length(input)*100),'%)'])
disp(['Truncation is ',int2str(truncation),'. (Out of
',int2str(length(input)),' data points.)'])

input = input(1:end-truncation);
output = output(1:end-truncation);

[pxx,f1] =
pwelch(input,window,floor(window*roverlap),window,fs,'onesided');
[pxy,f2] =
cpsd(input,output,window,floor(window*roverlap),window,fs,'onesided
');


TF = pxy./pxx;
f=f1;

end

function
array=AddData(array,data,fs,preprocess,postprocess,figures)

if figures; disp(['Data is reading from ',data,'...']); end
DATA=xlsread(data);
if figures; disp('Data reading complete. Parsing..'); end

LOAD=DATA(:,2);
idx=find(LOAD~=0,1,'first');
LOAD=LOAD(idx:end);
if figures
    figure(1); plot(LOAD);legend('Raw Bitwise Single-Turn Position
Output')
    disp('Parsing of output data completed.')
end

if figures; disp('Checking output data for irregularities'); end
LOAD=cleardata(LOAD,figures,100);
if figures; figure(2); plot(LOAD); legend('Clean Bitwise Single-
Turn Position Output'); end
MOTOR=DATA(:,1);
```

```
MOTOR=MOTOR(idx:end);
if figures
    figure(3); plot(MOTOR);legend('Raw Bitwise Single-Turn Position
Input')
    disp('Parsing of input data completed.')
end
if figures; disp('Checking input data for irregularities'); end
MOTOR=cleardata(MOTOR,figures,100);
if figures; figure(4) ; plot(MOTOR); legend('Clean Bitwise Single-
Turn Position Input'); end
pause(1)
if length(MOTOR)~=length(LOAD)
   if figures; disp('Warning! Your data sizes are not
compatible.'); end
end
if figures
    disp('Converting Bitwise Single-Turn Position Data to Bitwise
Position Data...')
    pause(1)
end
bittorad = 2*pi/65535;
turn = 0;
MOTOR_POS_ABS=MOTOR;
LOAD_POS_ABS=LOAD;
for i=2:1:length(MOTOR)
    difference = MOTOR(i)-MOTOR(i-1);
    if difference>60000
        turn=turn-1;
    elseif difference<-60000
        turn=turn+1;
    end
    MOTOR_POS_ABS(i)=MOTOR(i)+turn*65535;
end
if figures; disp('Input conversion complete. Switching to output
conversion...'); end
turn=0;
for i=2:1:length(LOAD)
    difference = LOAD(i)-LOAD(i-1);
    if difference>60000
        turn=turn-1;
    elseif difference<-60000
        turn=turn+1;
    end
    LOAD_POS_ABS(i)=LOAD(i)+turn*65535;
end
if figures
    disp('Output conversion complete. Switching to unit
conversion...')
    figure(5); plot(MOTOR_POS_ABS);legend('Bitwise Input Position')
    figure(6); plot(LOAD_POS_ABS);legend('Bitwise Output Position')
end
MOTOR_POS_ABS = MOTOR_POS_ABS*bittorad;
LOAD_POS_ABS = LOAD_POS_ABS*bittorad;
if figures
```

```
    figure(7); plot(MOTOR_POS_ABS);legend('Input Position in
radians')
    figure(8); plot(LOAD_POS_ABS);legend('Output Position in
radians')
    disp('Unit conversion complete. Checking preprocess
request...')
end
MOTOR_POS_ABS = dataprocessor(preprocess,MOTOR_POS_ABS,fs);
LOAD_POS_ABS = dataprocessor(preprocess,LOAD_POS_ABS,fs);
if figures
    pause(1)
    disp([preprocess,' is applied on position data. Generating
velocity data in rad/s...'])
end
MOTOR_VEL = diff(MOTOR_POS_ABS)*fs;
LOAD_VEL = diff(LOAD_POS_ABS)*fs;
if figures
   disp('Velocity data generation completed. Checking postprocess
request...')
   figure(9); plot(MOTOR_POS_ABS);legend('Processed Input Position
in radians')
   figure(10); plot(LOAD_POS_ABS);legend('Processed Output Position
in radians')
   figure(11); plot(MOTOR_VEL);legend('Unprocessed Input Velocity
in rad/s')
   figure(12); plot(LOAD_VEL);legend('Unprocessed Output Velocity
in rad/s')
end
[MOTOR_VEL,motormustcut]=clearjumps(MOTOR_VEL,figures,std(MOTOR_VEL
)*1.5,floor(0.1*fs));
[LOAD_VEL,loadmustcut]=clearjumps(LOAD_VEL,figures,std(LOAD_VEL)*1.
5,floor(0.1*fs));
if motormustcut || loadmustcut
    MOTOR_VEL = MOTOR_VEL(floor(0.1*fs):end-floor(0.1*fs));
    LOAD_VEL = LOAD_VEL(floor(0.1*fs):end-floor(0.1*fs));
end
MOTOR_VEL = dataprocessor(postprocess,MOTOR_VEL,fs);
LOAD_VEL = dataprocessor(postprocess,LOAD_VEL,fs);
if figures
   pause(1)
   disp([postprocess,' is applied on velocity data.'])
   figure(13); plot(MOTOR_VEL);legend('Processed Input Velocity in
rad/s')
   figure(14); plot(LOAD_VEL);legend('Processed Output Velocity in
rad/s')
end
addition = {MOTOR_VEL LOAD_VEL fs};
array{end + 1} = addition;
if figures
    pause(1)
    disp(['Data added as ',int2str(length(array)),'. entry of data
array.'])
end
```

```
function output = dataprocessor(process,input,fsample)
        switch process
            case 'smooth100'
                output = smooth(input,100);
            case 'smooth50'
                output = smooth(input,50);
            case 'smooth20'
                output = smooth(input,20);
            case 'smooth10'
                output = smooth(input,10);
            case 'smooth5'
                output = smooth(input,5);
            case 'loess10'
                output = smooth(input,10,'loess');
            case 'loess20'
                output = smooth(input,20,'loess');
            case 'loess100'
                output = smooth(input,100,'loess');
            case 'lowpass5'
output=lowpass(input,fsample/5,fsample,'ImpulseResponse','iir','Ste
epness',0.95);
            case 'lowpass10'
output=lowpass(input,fsample/10,fsample,'ImpulseResponse','iir','St
eepness',0.95);
            case 'lowpass20'
output=lowpass(input,fsample/20,fsample,'ImpulseResponse','iir','St
eepness',0.95);
            case 'lowpass50'
output=lowpass(input,fsample/50,fsample,'ImpulseResponse','iir','St
eepness',0.95);
            case 'lowpass100'
output=lowpass(input,fsample/100,fsample,'ImpulseResponse','iir','S
teepness',0.95);
            case 'none'
                output = input;
            otherwise
                error('Process entry is invalid!')
         end
    end
function output = cleardata(input,figureflag,peakprominence)
        output=input;
        [MN,IN]=findpeaks(input*(-
1),'MinPeakProminence',peakprominence);

[MP,IP]=findpeaks(input,'MinPeakProminence',peakprominence);
        for q=1:length(MN)
            if (MN(q)>-65100 && MN(q)<-400)
                output(IN(q))=(input(IN(q)-10)+input(IN(q)-
5)+input(IN(q)+10)+input(IN(q)+5))/4;
                if figureflag; disp(['Irregularity found in
',int2str(IN(q)),'. element of data array. Irregularity is
normalized.']); end
            end
        end
        for q=1:length(MP)
```

```
            if (MP(q)<65100 && MP(q)>400)
                output(IP(q))=(input(IP(q)-10)+input(IP(q)-
5)+input(IP(q)+5)+input(IP(q)+10))/4;
                if figureflag; disp(['Irregularity found in
',int2str(IP(q)),'. element of data array. Irregularity is
normalized.']); end
            end
        end
end
function [output,youmustcut] =
clearjumps(input,figureflag,peakprominence,trimrange)
        output=input;
        youmustcut = false;
        [MN,IN]=findpeaks(input*(-
1),'MinPeakProminence',peakprominence);
 [MP,IP]=findpeaks(input,'MinPeakProminence',peakprominence);
        IA = sort([IN; IP]);
        IA = rmvrepeaters(IA,trimrange);

        for q=1:length(IA)
            if IA(q)+floor(trimrange/2)<length(input) && IA(q)-
floor(trimrange/2)>0
              output(IA(q)-
floor(trimrange/2):IA(q)+floor(trimrange/2))=smooth(input(IA(q)-
floor(trimrange/2):IA(q)+floor(trimrange/2)),floor(trimrange/2),'rl
oess');
                if figureflag; disp(['Irregularity found in
',int2str(IA(q)),'. element of data array. Irregularity is
normalized.']); end
            else
                youmustcut = true;
            end
        end
end
function idxs = rmvrepeaters(I,range)
        idxs = [];
        idxmeminit = I(1);
        idxmemfinal = I(1);
        for y=1:(length(I)-1)
            if I(y+1)>idxmeminit+range || (idxmemfinal-
idxmeminit>range)
                idxs(end+1)=floor((idxmeminit+idxmemfinal)/2);
                idxmeminit = I(y+1);
            end
            idxmemfinal = I(y+1);
        end
end
end
```