FAST DYNAMICS ANALYSIS AND ACTIVE CONTROL OF SPIKE-TYPE STALL INCEPTION IN A ONE STAGE AXIAL COMPRESSOR SYSTEM

by

Nirmeet Adhvaryu

A thesis submitted in partial fulfillment of the

requirements for the degree of

MASTER OF SCIENCE

IN

MEASUREMENT AND CONTROL ENGINEERING

IDAHO STATE UNIVERSITY

FALL 2017

Photocopy and Use Authorization

In presenting this thesis in partial fulfillment of the requirements for an advanced degree at Idaho State University, I agree that the Library shall make it freely available for inspection. I further state that permission to download and/or print my thesis for scholarly purposes may be granted by the Dean of the Graduate School, Dean of my academic division, or by the University Librarian. It is understood that any copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Signature_____

To the Graduate Faculty:

The members of the committee appointed to examine the thesis of NIRMEET R. ADHVARYU find it satisfactory and recommend that it be accepted.

_____

Dr. Marco P. Schoen, Major Advisor

_____

Dr. Kenneth Bosworth, Member

_____

Dr. Gene Stuffle, GFR

ACKNOWLEDGEMENTS

I would like to express my sincerest gratitude and special thanks to my major advisor Dr. Marco P. Schoen for his continued support, guidance, and patience throughout this research. The knowledge and expertise he has provided has been invaluable and highly appreciated. I would also like to thank the rest of my thesis committee: Dr. Ken Bosworth and Dr. Gene Stuffle for their support and insightful comments.

I would like to thank Idaho State University and the Department of Mechanical Engineering for providing me financial assistantship that helped me in funding to complete my degree.

Nobody has been more important to me in the pursuit of this degree than the members of my family. I would like to thank my parents for their immense love and support in whatever I pursue. It would not have been possible without their care and patience. I would also like to thank my sister and my brother-in-law for their unconditional love and my 2-year-old nephew for always making me smile. I would also like to extend my special thanks to my uncle Mr. Pancholi whose blessings are always with me.

Last but not the least, I would like to thank my close friends Ankit, Gaurav, Gunjan, Harman, Romal, and Ruchi for their unfailing moral and emotional support and continuous encouragement throughout my years of research.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Fast Dynamics Analysis and Active Control of Spike-type Stall Inception in a One Stage Axial Compressor System

Thesis Abstract--Idaho State University (2017)

The research presented in this thesis details the system identification, modelling and control of the fast dynamics of a one stage axial compressor system. A control strategy to delay the onset of rotating stall is proposed. The system identification is focused on the relationship between the compressor flow coefficient and pressure rise coefficient as well as the compressor dynamics in and around the rotor blade passages. Transfer function and state-space models are developed to model the compressor fast dynamics for different flow coefficients. An overall model incorporating a variable flow coefficient is proposed. The derivation of this proposed overall model employs a map of identified eigenvalues corresponding to different flow coefficients. An active control strategy is proposed to delay the onset of rotating stall using a Linear Quadratic Gaussian controller and air injection into the blade passage. Simulation results based on the proposed overall model and dynamic controller indicate that the compressor characteristics within a blade passage can be controlled to the extent that problematic regions can be stabilized. Future work includes the testing of the proposed system on a one stage axial compressor system.

CHAPTER 1 INTRODUCTION

1.1: Axial Flow Compressor

Most modern aircrafts are powered by jet engines which employ compressors to increase the pressure of the incoming air. There are mainly two types of compressors namely Axial and Centrifugal. The basic difference between an axial flow compressor and a centrifugal compressor is that in an axial flow compressor the working fluid flows parallel to the axis of rotation while in a centrifugal compressor the working fluid may enter axially, but exits the compressor radially. In this thesis the main interest is limited to axial flow compressors. Compressor performance has a large impact on the overall performance of the jet engine and thus it is of significant importance to have a compressor working at or near its maximum performance for engineering, economic, and environmental benefits.

In axial flow compressors, the pressure rise is obtained by first accelerating the working fluid and then having it diffused. The acceleration is done by a row of rotating airfoils (blades) called the rotor and the diffusion is attained by a row of stationary blades called the stator [1]. The schematic of a one stage axial flow compressor is shown in Figure 1.1. The fluid is fed through the inlet and served to the compressor where the working fluid is compressed. After the compressor stage, the flow is guided in to the plenum via the exit duct. A throttle is used at the end of the duct in order to vary the operating point of the compressor. This is done by changing the size of the opening of the outlet [2].

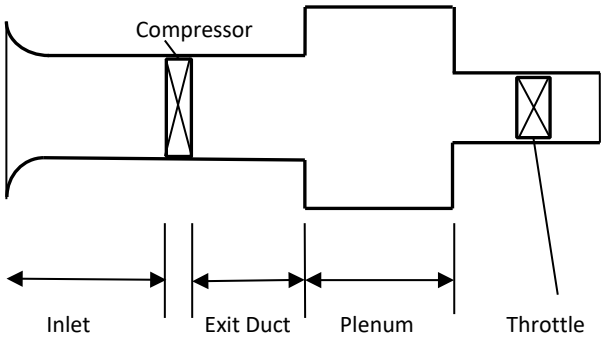

Figure 1.1: One Stage Axial Flow Compressor Schematics [2]

Boyce in [1] suggests that axial flow compressors— by producing low pressure increases of order 1.1:1 to 1.4:1— are able to attain very high efficiencies. This can be seen from Table 1.1

Table 1.1: Axial Flow Compressor Characteristics [1]

Type of Application	Type of Flow	Inlet Relative velocity Mach Number	Pressure Ratio per stage	Efficiency per stage
Industrial	Subsonic	0.4-0.8	1.05-1.2	88%-92%
Aerospace	Transonic	0.7-1.1	1.15-1.6	80%-85%
Research	Supersonic	1.05-2.5	1.8-2.2	75%-85%

The axial flow compressor in an advanced gas turbine these days usually consists of a multistage compressor having about 17-22 stages with exceedingly high-pressure ratio [1]. Figure 1.2 shows a multistage axial flow compressor rotor.

Figure 1.2: Multistage Axial Flow Compressor Rotor [1]

1.2: Compressor Instabilities: Rotating Stall and Surge

Compressor rotating stall and surge are the primary instabilities affecting the performance of a compressor and an engine altogether. The type and the degree of instability depends on the dynamics of the compressor [3]. Both of these phenomena are very difficult to predict accurately during design stage and can largely impact the performance of a compressor. Both rotating stall and surge can induce vibration, which can cause rapid airfoil failure and destruction of the compressor [4]. Rotating stall is a two-dimensional occurrence in which one or more regions of stalled flow keep rotating around the circumference of the compressor [5]. The regions circulate in the same course as the blades. The speed of the same is often between 1/5 and 1/2 of the wheel speed [6]. Rotating stall often occurs only in some parts of the machine. It is also regarded by many as an inception of a potentially more dangerous compressor instability called the surge. Surge occurs when the pressure in the plenum exceeds the compressor pressure rise [7]. Depending on the pressure fluctuations and amplitude of the flow many researchers have

distinguished surge in four categories [8]: mild surge, classic surge, modified surge, and deep surge. Mild surge is a phenomenon having small pressure fluctuations while a classic surge is the one with larger oscillations and a lower frequency than the mild surge. Modified surge is a mix of rotating stall and classic surge. A deep surge is a more severe form of the classic surge and the only one from all the other types where there is a possibility of a flow reversal. The transition from mild surge to other forms are characterized by flow fluctuations and increase in the amplitudes of the pressure. The transition from a normal compressor operation to stall is depicted in Figure 1.3 where, ϕ is a dimensionless parameter designated as the flow coefficient and Ψ is the dimensionless parameter pressure rise. The two parameters are defined by Equations 1.1 and 1.2 [2].

$$\phi = \frac{C_x}{U} \tag{1.1}$$

$$\Psi = \frac{\Delta P}{\frac{1}{2}\rho U^2} \tag{1.2}$$

Here, C_x is the axial flow velocity, U is the mean rotor velocity, ΔP is the change in pressure from inlet atmospheric pressure to plenum pressure, and ρ is the density of the fluid passing through the compressor. The pressure rise increases as the flow coefficient is decreased. This pattern continues until the system goes in to rotating stall or surge. For rotating stall, the compressor can operate at the peak of the characteristic at point A. A sudden transition in to rotating stall occurs at point B. During the transition from A to B (segment A-B) a significant drop in pressure rise and decrease in flow coefficient occurs. This trend will continue until the flow is again increased to point C [9]. At present, the only remedy to get out of rotating stall and surge is to shut down the engine and restart it [10]. For the reasons stated above, it is of utmost

importance to minimize the effect of these instabilities for the optimum performance of a compressor.

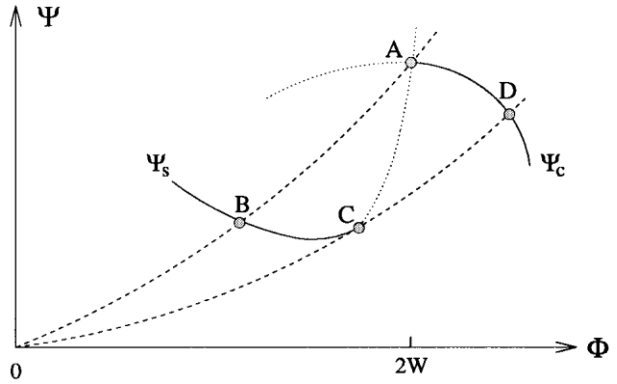

Figure 1.3: Compressor Characteristics of Rotating Stall [9]

1.3: Research History on Rotating Stall and Surge

Research on minimizing the effects of rotating stall and surge in compressors has been of high interest since the 1950's. For a volute type centrifugal pump, [11] found that the onset of surge is due to premature diffusion of the flow entering the tongue (part of the volute near the diffuser) region along with the destabilizing behavior of the diffuser. According to [12-14] the rotating stall is caused by the flow moving slower than the rotors around the compressor annulus. [13] found that the flow separation inside the blade passage was the reason for the onset of stall. In [15], it is suggested that for high speed multi stage axial compressors, the reason for rotating stall and surge is the blast wave originating from the back of the machine due to a sudden change triggered by the machine, whereas in [16], the author argues that the short rotating stall cells originating from the front of the machine is the reason for the onset of fully developed rotating stall cells. Rotating stall and surge causes rapid heating of the blades and the unsteady excitation

causes additional loads on the blades which consequently results in severe blade vibration and fatigue— which may also cause severe damage to the engine due to unacceptable levels of system vibration [8]. Hence, it appears that several factors are responsible for the onset of rotating stall and surge and all of them have not been identified yet.

Several measures are presently used to minimize the effects of surge [17] — which can be classified as follows:

1. **Surge control or avoiding surge**: Here the basic concept is to operate the machine in a region near and beyond the line of surge. This is often called as an open-loop strategy. The conventional anti-surge control is widely used with variable speed centrifugal compressors. The method is based on the proportionality between the signal from the flow element (differential pressure Δp_0) and the differential pressure across the compressor (Δp_c). The relationship is given by Equation 1.3, where Δp_0 is the flow element which is the differential pressure and Δp_c is the differential pressure across the compressor [17]

$$\Delta p_0 = k_1 . \Delta p_c + k_2 \tag{1.3}$$

   There have been many variations proposed in this basic concept of surge control. One such method is discussed in [18], where the researchers propose a drive torque assisted anti surge control scheme where along with the understanding of the traditional control scheme, drive torque is used to prevent the onset of surge.

2. **Surge detection and control/avoidance**: Here the surge avoidance/control system starts acting right at the inception of surge in the system. This is regarded as the closed-loop strategy. There have been numerous techniques and methods proposed for the detection of surge which are generally based on the variations in the parameters such as

temperature, pressure, flow etc. The strategies based on the monitoring of the vibrations in the compressor are not much reliable and usually fail because it gets difficult to distinguish between the vibrations caused by the onset of surge and the vibration from other sources in the system. One technique is proposed in [19], where the inlet pressure is sensed and used to detect surge. The temperature rate of change signal which is compared to a pre-calculated value maybe a function of other system parameters such as compressor discharge pressure or rotor speed. Several techniques have been proposed for surge control or detection. However, each of these techniques proposed have been rather specific to the machine in discussion and typically fails to be successfully applied to all the systems in general. A general model applicable to all compressor systems is yet to be invented.

3. **Active surge control:** A considerable amount of work has been done to identify and understand the different phenomenon causing rotating stall and surge. For active control, it is of utmost importance to have working knowledge of the compressor dynamics to be able to better predict the onset of stall and surge.

A breakthrough research in development of a model describing rotating stall and surge occurred in 1950's by E.M.Greitzer at the Massachusetts Institute of Technology. The research carried out is published in [20-21]. A collaborative research with F.K. Moore led to further refining of the Greitzer's model. The important aspect of this model was to integrate blade passage flow in Greitzer's original model. The results of the research are published in [10]. The collaborative model now widely known as the Moore-Greitzer model has been extensively used for further developments in this area of research. The resulting model has

been highly successful in capturing rotating stall and surge and has been widely used as the mathematical base model for further developments.

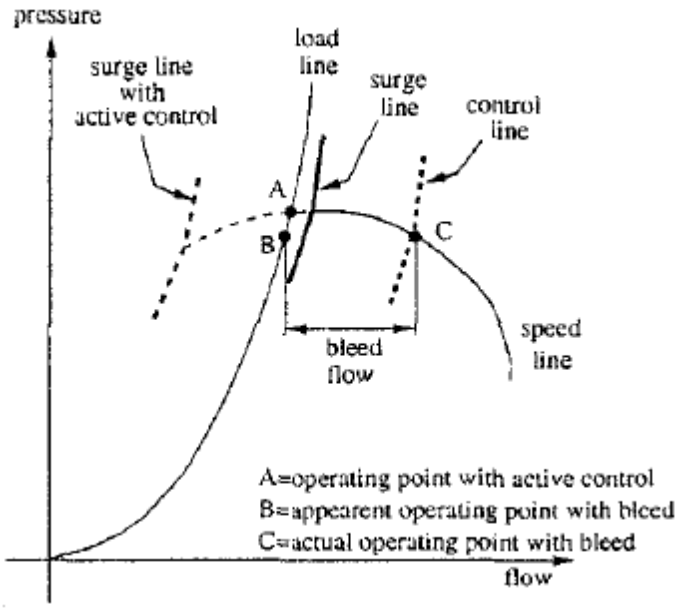

Figure 1.4: Scheme Showing Active Control Strategy [5]

Epstein et al. [22] proposed an active control theory where compressor dynamics were altered by feeding back disturbances (A small change in movement or behavior) in to the flow field. The performance of the compressor can be improved due to extension of the stable operating region as shown in Figure 1.4. A number of researchers used the bifurcation theory in order to validate the Moore-Greitzer model. An example of such is given in [23], where a generic nonlinear control scheme was proposed. A similar experiment was conducted by Mansoux et al. in [24], where three different experimental low speed compressors exhibiting different stall inception behavior were modelled and the results were compared with the nonlinear simulation. During late 1990's another control research area was developing which made use of air injectors as

actuators to increase the operating range of the compressor. In [25], Behnken et al. analyzed the effects of steady state compressor characteristic actuation scheme on a three state Moore-Greitzer model. A closed-loop controller was designed, and experiments were performed on a specific compressor. The experimental findings were compared with the simulation and both the results were found to be quite similar in nature. The designed injection controller was shown to have removed the hysteresis region associated with rotating stall. W kang et al. in [28] proposes a bifurcation stabilization method with a linear feedback control to eliminate surge. The stabilization results are demonstrated via numerical simulations.

In early 2000's Paduano et al in [29], presented methods to stabilize rotating stall and surge in a transonic single stage axial compressor. Their two different approaches involved one— where they find a way to actively damp the system while the other involved manipulating system dynamics to keep the operating range of the compressor near the instability.

1.4: Axial Compressor Fast Dynamics

The fast dynamics are referred to as the flow dynamics in and around the rotor blade. In axial compressors, stall inception is hypothesized to occur in two different forms—namely spike stall inception and modal stall inception. Spike type stall inception is a short length scale disturbance that occurs suddenly and develops in to rotating stall, while modal stall is a long length scale disturbance which gradually develops in to rotating stall [30]. Figure 1.5 shows spike type stall where one can see a sudden spike appearing on the velocity traces. As the spike begins to propagate, it increases in size and speed of rotation decreases. Figure 1.6 shows modal stall inception where one can see gentle undulation in the velocity traces before it develops in to fully developed rotating stall.

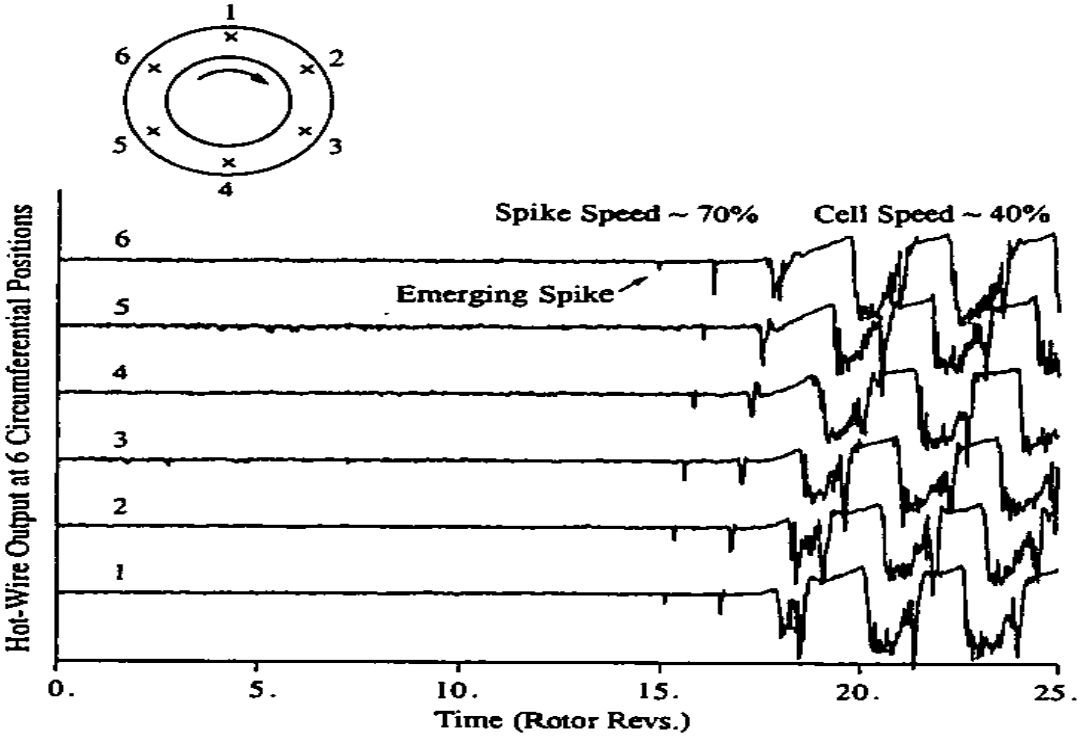

Figure 1.5: An Example Showing Spike Stall Inception [30]

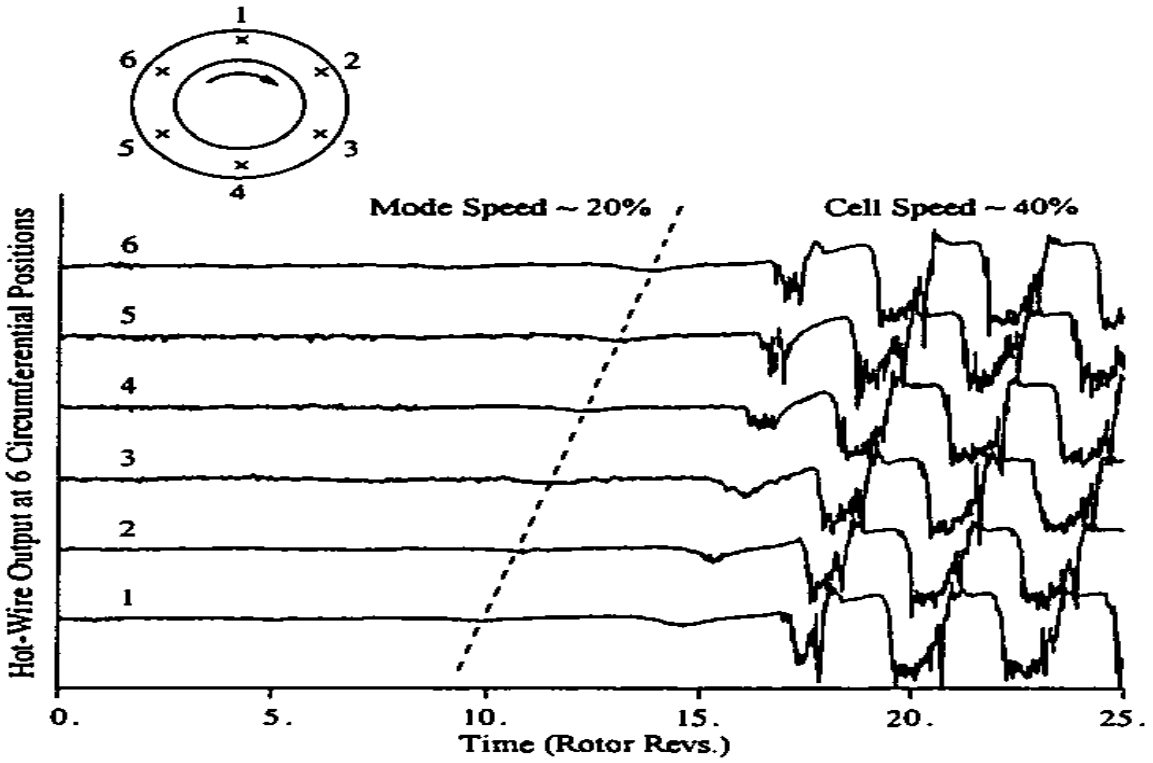

Figure 1.6: An Example Showing Modal Stall Inception [30]

It was suggested by I.J. Day in 1993 that the possibility of either of the phenomena to occur in a compressor depends on the change in the tip clearance. The spike type stall inception is physically defined by a sudden and sharp spike in the velocity traces and the high speed at which it rotates. However, spike inception lasts only for about a couple of blade rotations before it develops in to stall. Thus, this type of stall inception is difficult to use for control purposes as it does not give enough time for the controller to act. In contrast, modal oscillations appear for a lengthier time before it develops in to stall and intensifies as the flow rate reduces. As can be seen from Figure 1.7, for spike stall inception, stall tends to occur prior to the peak of the characteristic curve while for modal oscillations, it tends to occur near or at the peak on the characteristic curve.

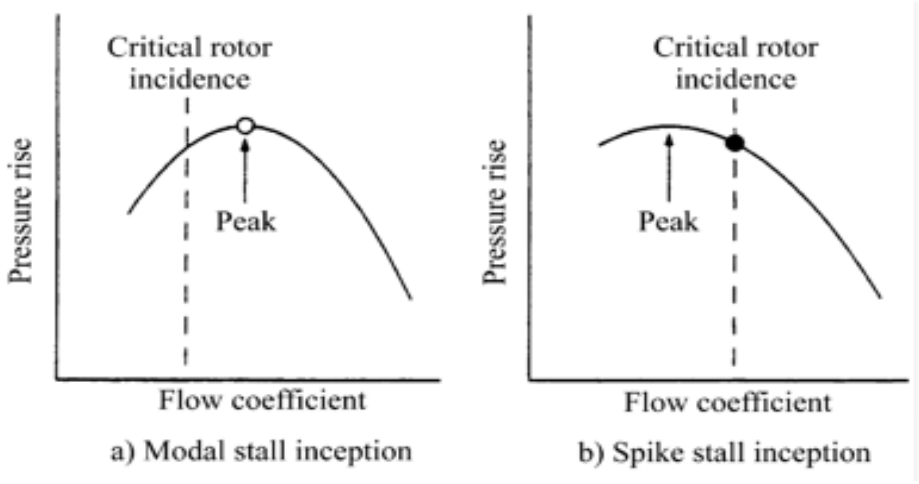

Figure 1.7: Location of Spike and Modal-type Stall Inception on a Compressor Characteristic Curve [30]

The experimental compressor at Chinese Academy of Sciences (CAS) used for model development for the research presented in this document—exhibits spike type stall inception. Thus, the research presented here is focusing on spike type stall inception.

1.5: Flow Features Associated with Spike-type Stall Inception

Camp et al. proved with their experimental results in [30] that the spike type stall inception occurs on a negatively sloped region of the compressor characteristic curve. Another feature associated with spike stall inception is the blade boundary layer near the blade tip. It was observed in [30]— that the spike stall inception typically occurred at a certain value of blade tip incidence. Spike stall inception exhibits localized flow separation on one particular blade row. If so, blade boundary layer separation can be related to spike stall inception [31]. The research by Vo [31] suggests that the flow within the blade passages can be characterized by the interface

line between the tip clearance flow and the main incoming flow. Figure 1.8 graphically represents how tip clearance flow interacts with the main incoming flow.

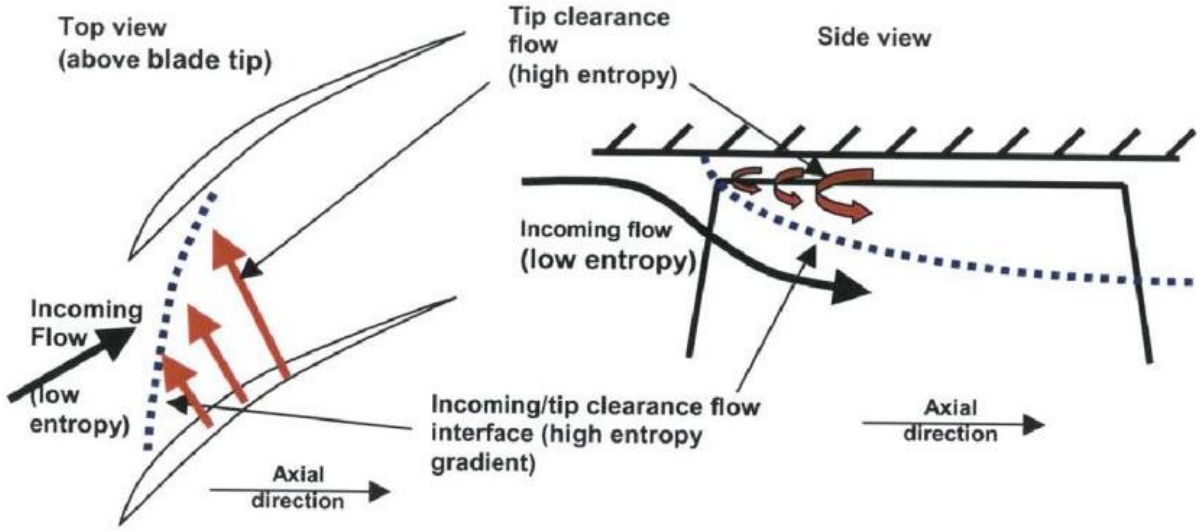

Figure 1.8: Representation of Interaction Between the Tip Clearance Flow and the Incoming Flow [31]

The interface line can be used to distinguish various flow conditions. The interface line begins from the low-pressure side of the first blade and extends near the leading edge. This creates a tip gap which allows the incoming flow to pass through the blade passage without any disturbance. This hypothesis has been adopted by a number of researchers to overcome stall and the same would be assumed to be true for the research presented in this document.

1.6: Research Objectives

The objective of this thesis is to identify and to some extend control the flow dynamics within a single blade passage. The assumed flow structure is shown in Figure 1.8. The flow is described by using an interface line that indicates the meeting line between the incoming flow, the tip leakage flow, and a potential flow from the injectors used as a controller.

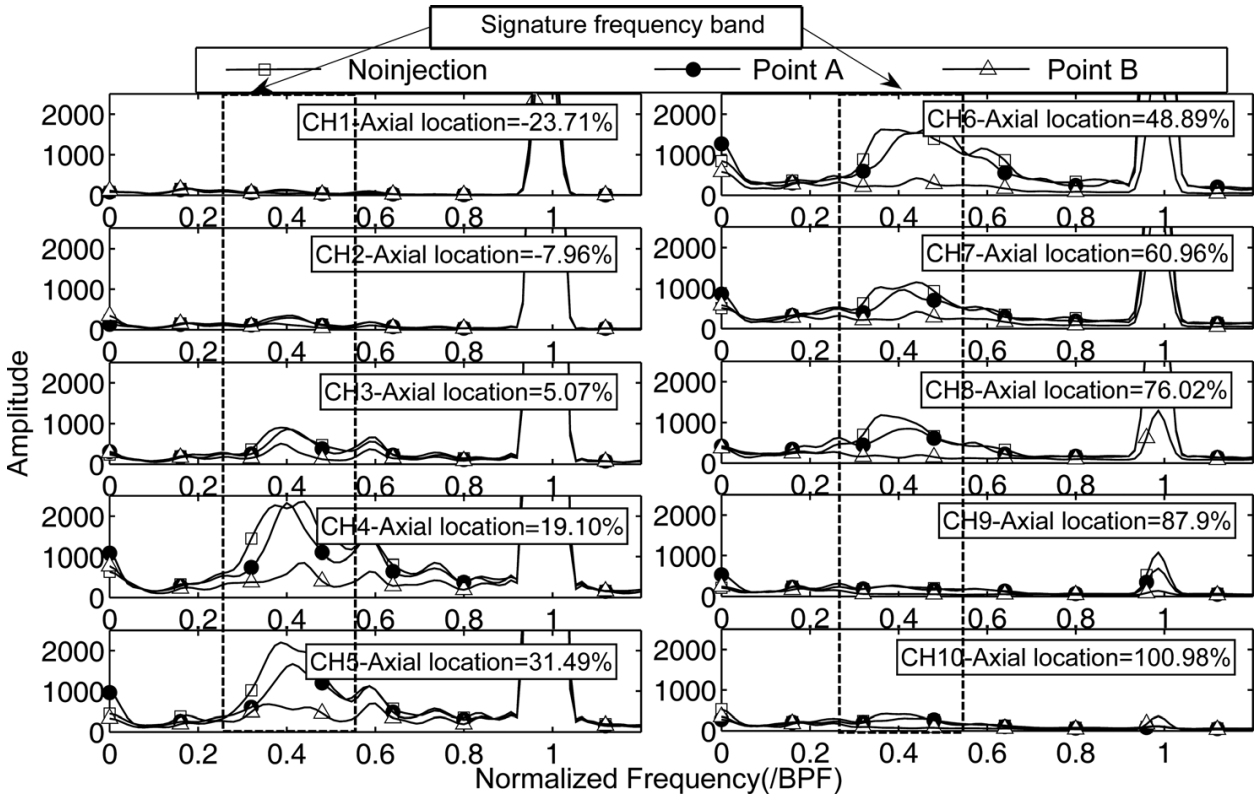

Figure 1.9: Signature Frequency band for Ten Sensors [32]

Injection has shown to be effective at an operating point close to stall in moving the stall margin further out [32]. Using a Power Spectral Density (PSD) analysis of the unsteady pressure measured in the casing above the blade passage, the signature frequency band can be determined. In Figure 1.9 [32], the PSD is plotted for ten sensors whose arrangement is given in Figure 1.10.

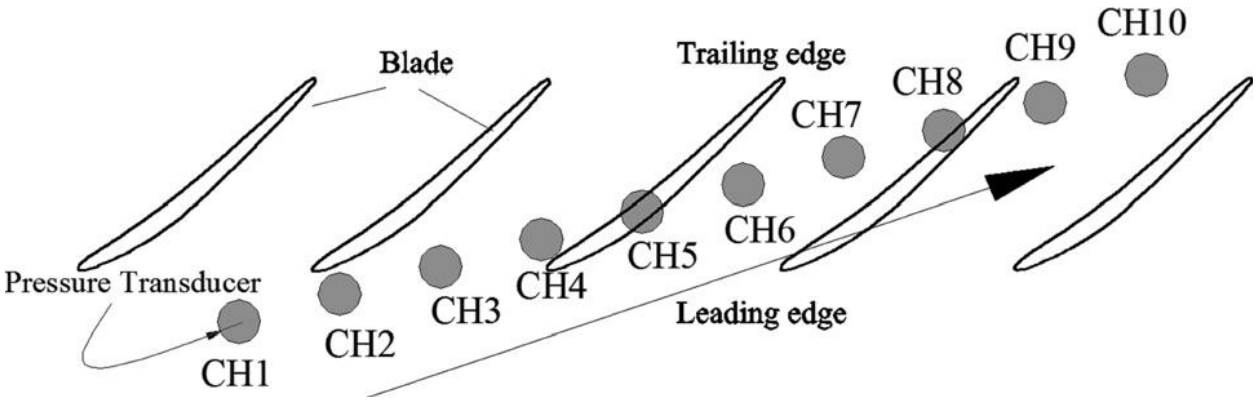

Figure 1.10: Sensor Location above Blade Passage [32].

From Figure 1.9, it seems that sensors 4 to 6 indicate the most sensitivity in terms of changes in PSD magnitude within the signature bandwidth. System identification for these sensors was used to extract the kind of dynamics occurring at this part of the blade passage for different flow coefficients. In particular, for the system identification, the input was selected to be a random injection at the leading edge of the compressor blades. The output was selected to be the correlation coefficient between the pressure data of a given sensor between one rotation of the rotor. As the sampling frequency of the injection (20,000 Hz to about 40 Hz), a spline curve approximation was used in order to resample the computed and estimated correlation data at 20,000 Hz. Figure 1.11 shows the extracted root locus of the system model for sensor 6. As the compressor approaches stall limit, the pole location of the extracted dynamical model moves towards the unstable region (pole 2 of a conjugate complex pole pair). As the spline curve approximation results in a loss of identifiable frequencies (it acts like a smoothing filter), these results are indicative in nature.

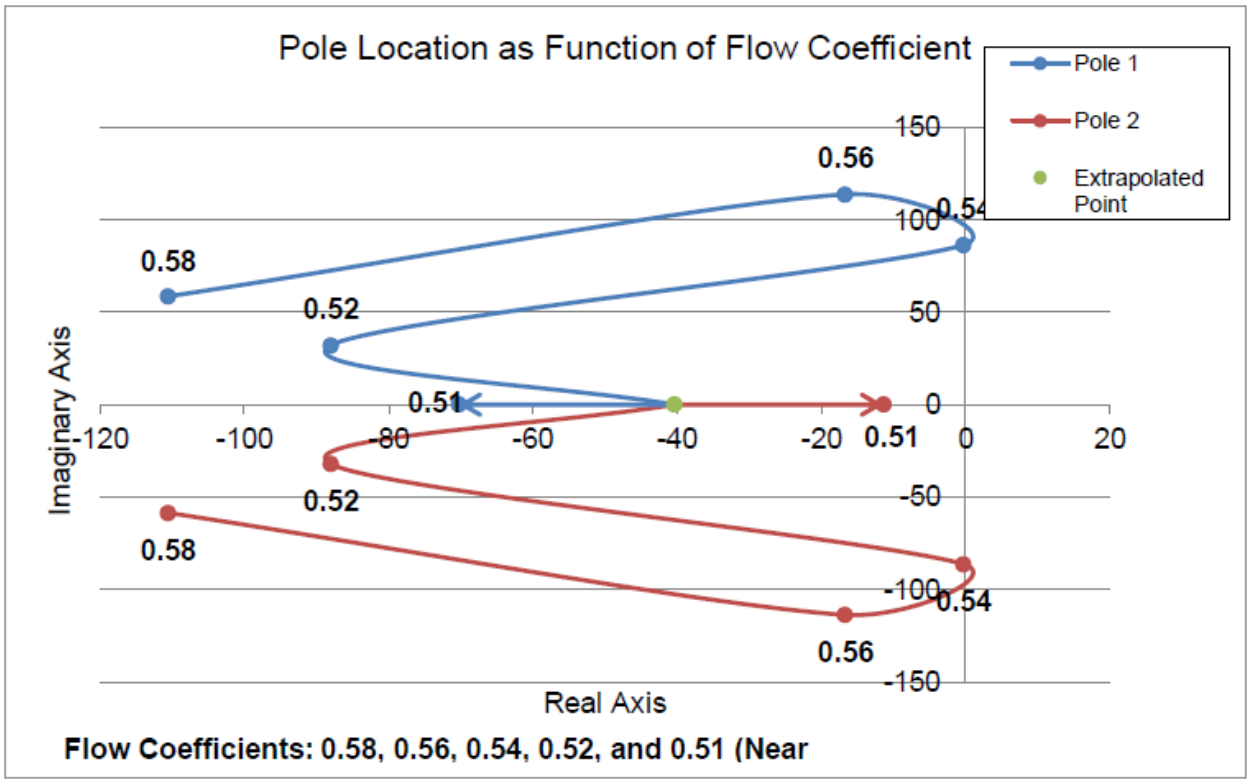

Figure 1.11: Root Locus of Extracted System Model. The Root Locus is a Function of the Flow Coefficient. [2]

Using the above given identification results, the plan is to design a controller based on injection to stabilize the controller just at or beyond instability.

Chapter 2: System Identification and Parametric Models

System identification is defined as “approximate modeling of a system for a specific application on the basis of observed data and prior system knowledge”, [35]. In other words, system identification is a method of building models for dynamic systems using system’s measured input and output in time or frequency domain. The two major approaches for using system identification for model development are:

1. **Grey box approach**: This approach is based on developing a model based on both insight of the system as well as the collected experimental data. Although all the parameters of the system are not known, a certain model is developed based on the preconceived knowledge of a system. System identification could be used to estimate the unknown parameters.
2. **Black box approach:** This approach just relies on the experimental input-output data to develop a model. No prior knowledge of the system is available. Bunge [27], describes the black box as “The constitution and structure of the box are altogether irrelevant to the approach under construction, which is purely external or phenomenological. In other words, only the behavior of the system would be accounted for”.

Figure 2.1: A Representation of the Black Box Approach [26]

System identification involves applying a model to the test input-output data sets and identifying model parameters from it. The parameters that best relate to the input-output is used for model development.

2.1: Mathematical Models

Mathematical models can be of different forms, based on the system being studied. Generally, mathematical models are needed for the following purposes: [36]

- **Diagnosis**- To obtain a larger insight in to a particular phenomenon occurring in the system
- **Control Design**- For designing a control system that improves the performance of a dynamic system
- **Simulation**- Analyzing system dynamics and behavior with the help of simulations
- **Estimation**- Estimating state variables which are usually difficult to measure in real time
- **Prediction**- Prediction of the behavioral pattern of particular variables

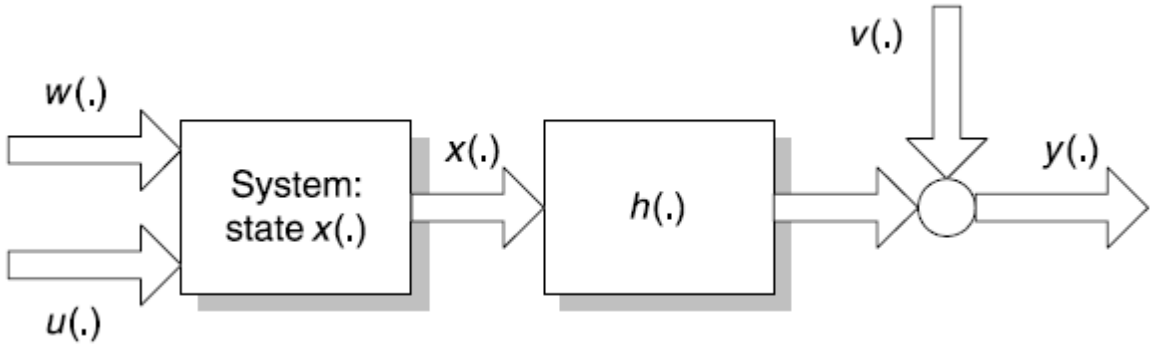

Figure 2.2: A Basic Representation of a Mathematical Model [35]

A basic representation of a mathematical model is shown in Figure 2.2. Here w(.) and v(.) are additive white noises. These noises are added to account for the errors in the system caused by measurement and modeling errors.

The input-output relationships for a single input and single output system can be represented by Equations 2.1 and 2.2 where, Equation 2.2 is the discrete version of Equation 2.1. [35]

$$y(t) = \int_{-\infty}^{t} g(t-\tau)u(\tau)d\tau \quad (2.1)$$

$$y(t) = \sum_{k=-\infty}^{t} g(t-k)u(k) \quad (2.2)$$

Here, y(t) represents the output, $g(t-\tau)$ is the impulse response of the system at time t, k is the time index, and τ is the variable for integration. These equations are also referred to as impulse response model representation.

2.2: System Identification Procedure

In practice, a perfect mathematical representation of a system is not possible because of the complexity of the system, the limited prior knowledge, and the errors associated with the observed data. Thus, system identification is considered as an appropriate modeling of the system on the basis of prior knowledge and available data. Figure 2.3 shows a block diagram of the basic identification process. The process of identification can be described by three main aspects: [36]

- **Data**: Data can be collected based on prior knowledge or objectives or maybe in the form of observed data to cater specific needs.

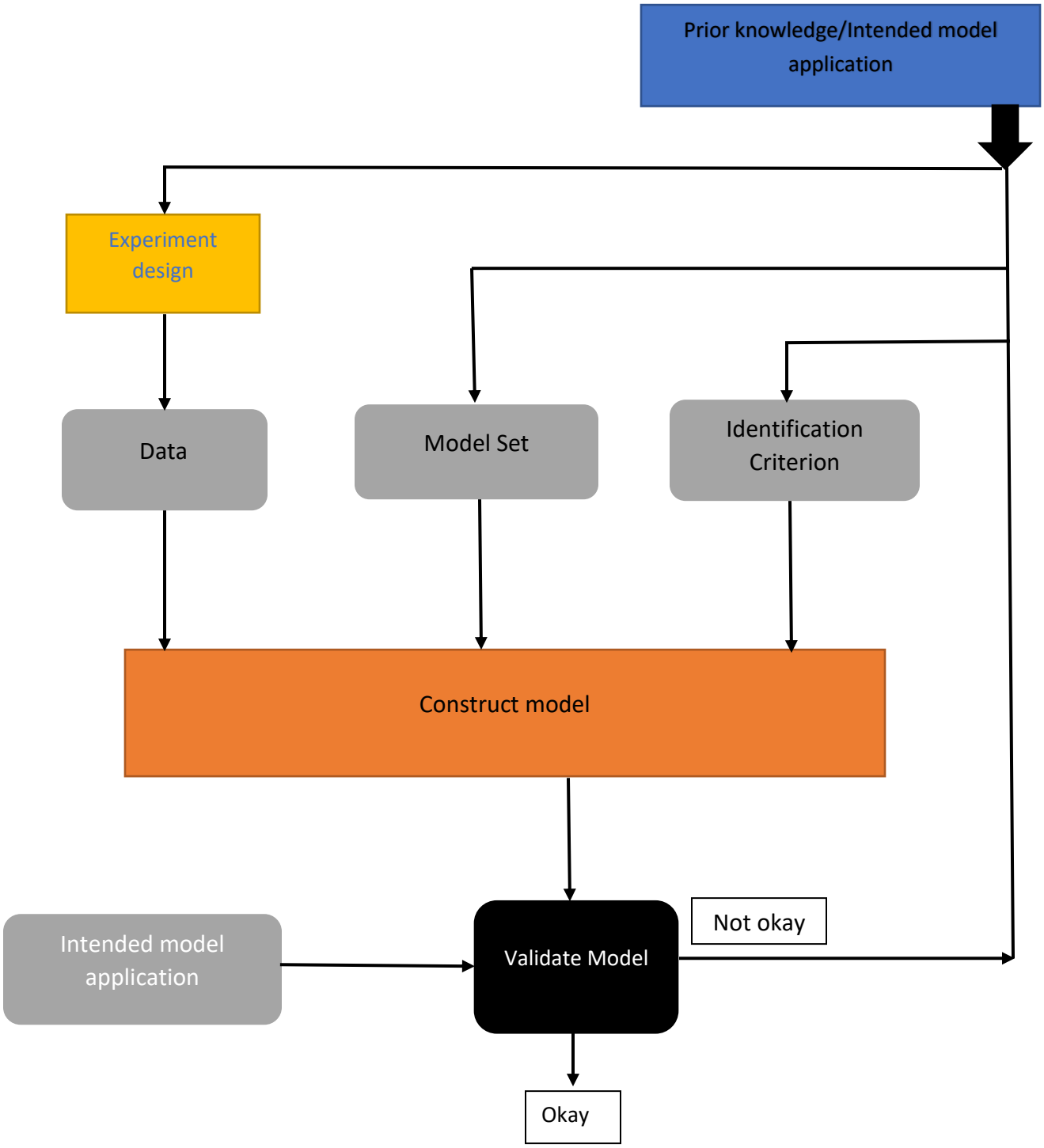

Figure 2.3: Block Diagram Representation of the System Identification Process [36]

- **Model set**: Often regarded as the most difficult step in the identification process, model set is where one choses a set of models from which the most accurate model is selected for the process. Here, one also specifies the properties of model like discrete/continuous, linear/non-linear etc.- which is generally done based on the prior knowledge of the system.
- **Identification Criterion**: In order to check the goodness of fit between the model parameters and the observed data, the identification criterion has to be specified.

2.3: Fast Dynamics System Identification

The Matlab^{TM} system identification toolbox was used for the system identification analysis. The approach used by Dane et al. [2] for preparing fast dynamics data for system identification is shown in Figure 2.4

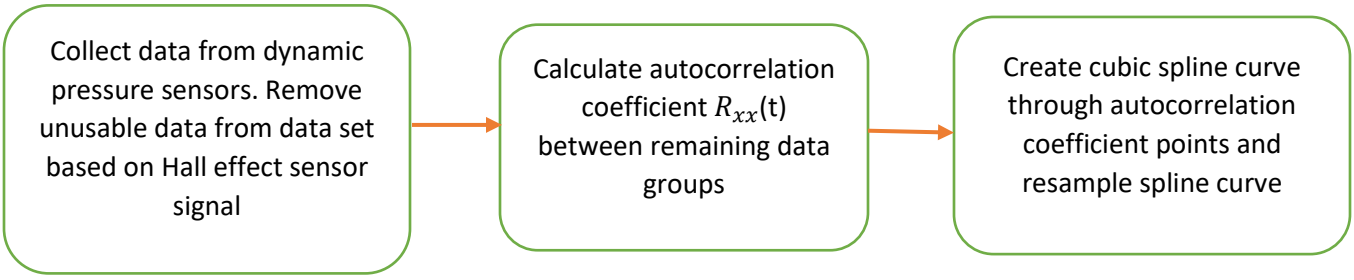

Figure 2.4: Block Diagram Representation of the Preparation of Fast Dynamics Data for System Identification [2]

Several models like the ARX, state space, and discrete and continuous time transfer function models were attempted. It was found that four poles and three zeroes transfer function model produced the best fit for the data [2]. A transfer function was developed at each of the nine

sensors. Equation 2.3 shows the form of the transfer function as a function of the complex frequency domain variable s. Y_n is designated as the output and U is the input.

$$\frac{\mathrm{Y_n(s)}}{\mathrm{U(s)}} = \mathrm{G_n(s)} \tag{2.3}$$

The transfer function can also be written as shown in Equation (2.4), where n is the sensor number.

$$G_n(s) = \frac{b_{0n}s^3 + b_{1n}s^2 + b_{2n}s + b_{3n}}{s^4 + a_{1n}s^3 + a_{2n}s^2 + a_{3n}s + a_{4n}} \tag{2.4}$$

Out of the nine sensor outputs, sensor number 4, 5, and 6 were used for system identification. It is believed that these three sensors were most suitable for capturing the system's fast dynamics mainly because of its location between the leading edge of the rotor blade to approximately the mid-chord length. For the system identification of fast dynamics— the input is the pressure sensor signal from the air injector actuator jet and the outputs are the resampled data sets from the spline curve fitted to the autocorrelation coefficient data points for the nine sensors used.

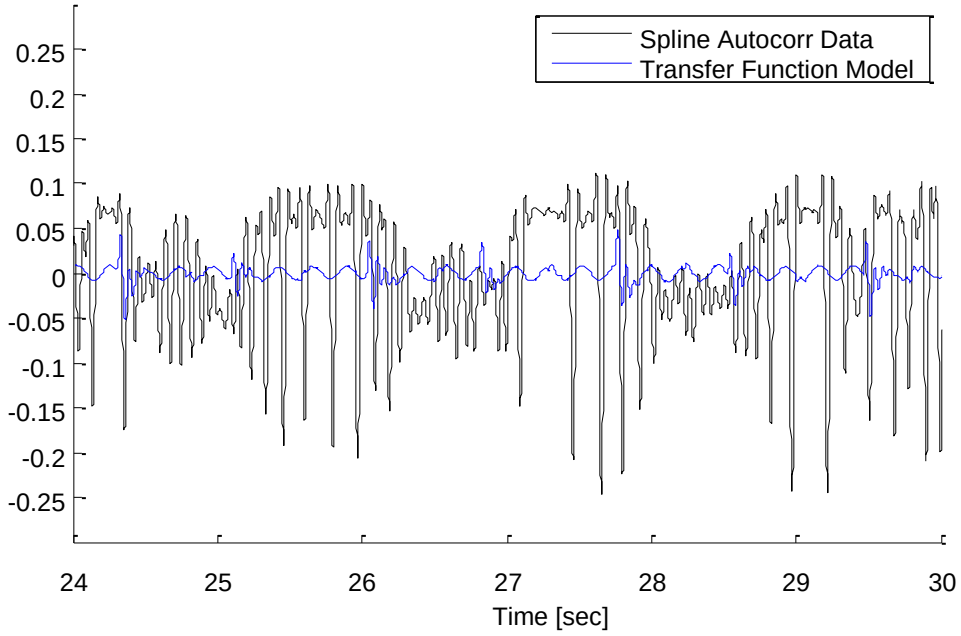

Figure 2.5: Transfer Function Model Comparison for Sensor No.5 for ϕ=0.56 [2]

Figure 2.5 shows the comparison of spline interpolated autocorrelation coefficient data (Black) with the transfer function model data (Blue) for flow coefficient ϕ=0.56 for sensor no.5 [2]. It can be seen from the figure that the goodness of fit percentage is low (around 10%). It is believed that these transfer function models are best able to capture the system's fast dynamics as they could capture the compressor natural frequency and also displayed certain pole trends as stall is approached. Thus, sensor no. 5 was chosen for development of a model for control purposes. The fast dynamics transfer function model data for sensor no.5 is shown in Table 2.1.

The initial approach was to get the polynomial equation of a smooth curve fitted to the combined pole and zero trends at different flow coefficients. However, a dynamic equation of a smooth curve fitting all the data points was not achievable. An attempt to add some data points on the path of the curve-to get the equation of the smooth curve was also made- but with undesirable results. Therefore, the second approach was to get the polynomial equations of how each pole/zero behaved at different flow coefficients and develop a model out of it from control purposes. Microsoft Excel™ "Trendline" function was used for getting the equation of the curves. The steps involved along with the screenshots are shown below:

1. Select the stored data and make a scatter plot.

Figure 2.6: Insert Chart Window

2. Right click on the data on the scatter plot and select “add trendline”

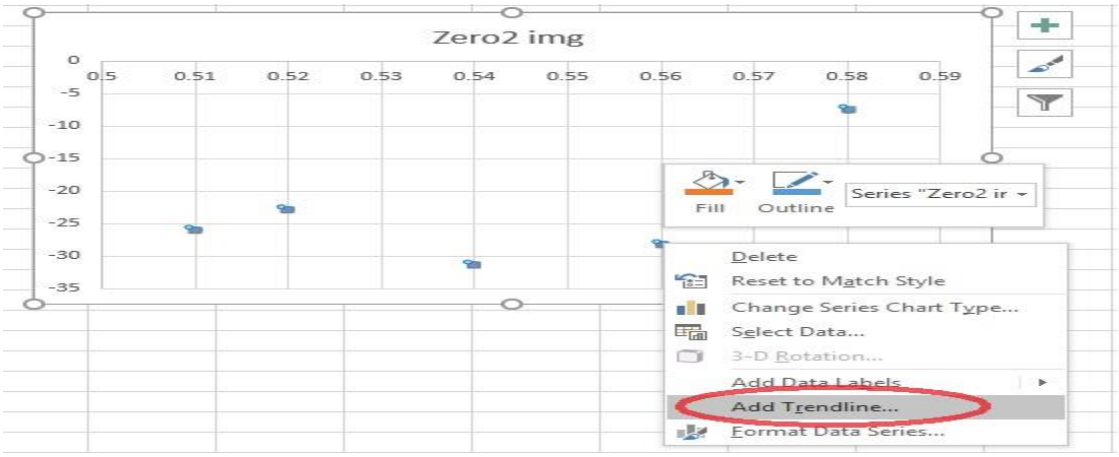

Figure 2.7: Adding Trendline Window

3. Select “polynomial” and pick the order that best fits the data. The options “display equation on chart” and “Display R-squared value on chart” can be selected.

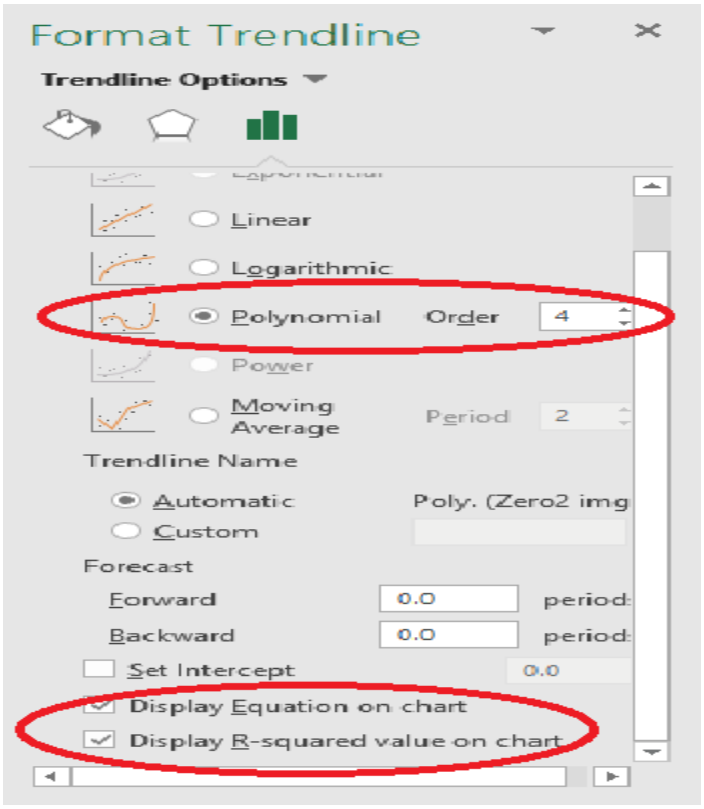

Figure 2.8: Format Trendline Window

4. Click on the legend and select “format trend line”. Select the option “number” and change the decimal number required as per the requirement.

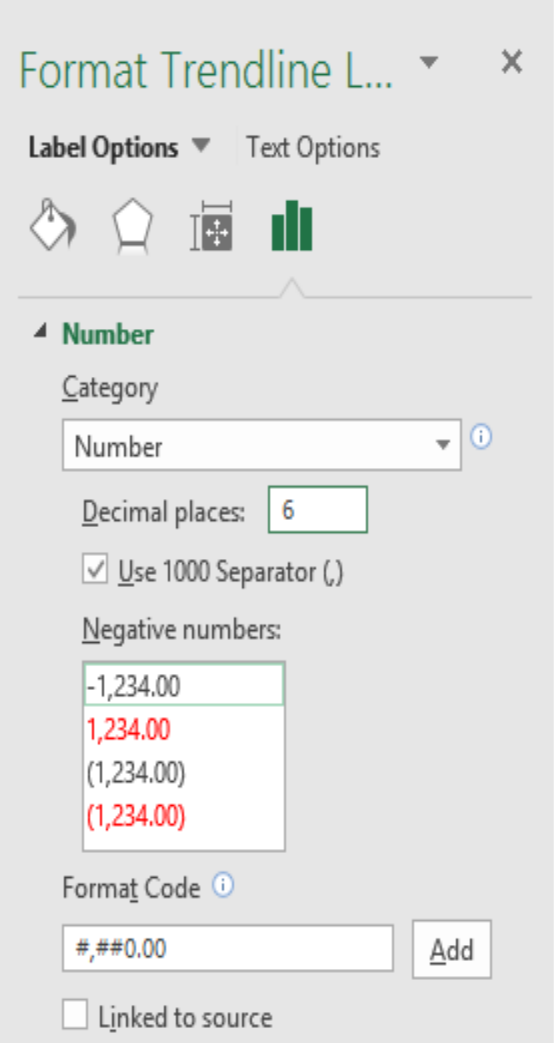

Figure 2.9: Choosing Number in the Format Trendline Window

Table 2.1: Fast Dynamics Transfer Function Model Data for Sensor no. 4 [2]

Flow Coefficient	Poles	Zeros	Natural Frequency (Hz)
0.58	-0.0000 +86.8311i	-133.71 + 0.0000i	13.820
	-0.0000 -86.8311i	-1.80 + 55.55i	13.820
	-7.8998 +20.4642i	-1.80 - 55.55i	3.491
	-7.8998 -20.4642i	-	3.491
0.56	-34.7345 +88.9319i	2.1037 +32.1154i	15.195
	-34.7345 -88.9319i	2.1037 -32.1154i	15.195
	-0.0000 +30.7684i	3.4567 + 0.0000i	4.897
	-0.0000 -30.7684i	-	4.897
0.54	-0.3740 +86.5791i	-5.1659+28.3489i	13.780
	-0.3740 -86.5791i	-5.1659-28.3489i	13.780
	-0.0361 +21.9828i	13.1705 +0.0000i	3.499
	-0.0361 -21.9828i	-	3.499
0.52	-76.0179+44.0606i	-0.0315+23.5119i	13.984
	-76.0179-44.0606i	-0.0315-23.5119i	13.984
	-0.0347 +23.5191i	6.0286 +0.0000i	3.743
	-0.0347 -23.5191i	-	3.743
0.51	-79.9413 +30.8245i	1.6096+26.0384i	13.636
	-79.9413 -30.8245i	1.6096-26.0384i	13.636
	-0.4161 +25.3275i	1.1757 + 0.0000i	4.032
	-0.4161 -25.3275i	-	4.032

Table 2.2: Fast Dynamics Transfer Function Model Data for Sensor no. 5 [2]

Flow Coefficient	Poles	Zeros	Natural Frequency (Hz)
0.58	-110.61+58.57i	19.8603+0.0000i	19.19
	-110.61-58.57i	-0.1802+7.5664i	19.19
	-0.18+7.50i	-0.1802-7.5664i	1.194
	-0.18-7.50i	-	1.194
0.56	-16.70+113.78i	0.191+28.1214i	18.303
	-16.70-113.78i	0.0191-28.1214i	18.303
	-0.0000+26.95i	9.7887+0.0000i	4.289
	-0.0000-26.95i	-	4.289
0.54	-0.3206+86.3066i	48.5023+0.0000i	13.736
	-0.3206-86.3066i	-0.8536+31.2561i	13.736
	-0.0000+31.6615i	-0.8536-31.2561i	5.039
	-0.0000-31.6615i	-	5.039
0.52	-88.0397+32.0824i	46.9253+0.0000i	14.913
	-88.0397-32.0824i	0.6126+22.8844i	14.913
	-0.0226+23.4365i	0.6126-22.8844i	3.730
	-0.0226-23.4365i	-	3.730
0.51	-70.2878+0.0000i	0.2443+25.9897i	11.187
	-11.3883+0.0000i	0.2443-25.9897i	1.813
	-0.1389+25.3343i	0.7730+0.0000i	4.032
	-0.1389-25.3343i	-	4.032

Table 2.3: Fast Dynamics Transfer Function Model Data for Sensor no. 6 [2]

Flow Coefficient	Poles	Zeros	Natural Frequency (Hz)
0.58	-88.7959 +60.8620i	-0.5506+18.1437i	17.133
	-88.7959 -60.8620i	-0.5506-18.1437i	17.133
	-0.0006 +17.4192i	10.7467+0.0000i	2.772
	-0.0006 -17.4192i	-	2.772
0.56	-92.4162 +60.8268i	-0.0011+23.3122i	17.609
	-92.4162 -60.8268i	-0.0011-23.3122i	17.609
	-0.0019 +23.3145i	10.7301+0.0000i	3.711
	-0.0019 -23.3145i	-	3.711
0.54	-65.3445+52.5376i	-0.2362+91.8688i	13.344
	-65.3445-52.5376i	-0.2362-91.8688i	13.344
	-0.0921 +88.6676i	73.4270+0.0000i	14.112
	-0.0921 -88.6676i	-	14.112
0.52	-12.9371+15.0633i	-7.7264+95.9229i	3.160
	-12.9371-15.0633i	-7.7264-95.9229i	3.160
	-0.0797+87.4101i	-11.6611+0.0000i	13.912
	-0.0797-87.4101i	-	13.912
0.51	-58.5959 +0.0000i	4.4825+76.3377i	9.326
	-0.8703 + 0.0000i	4.4825-76.3377i	0.139
	-0.3802 +81.7580i	-0.0969+0.0000i	13.012
	-0.3802 -81.7580i	-	13.012

The plots along with their polynomial equation (displayed inside the plot) for all the poles and zeroes for sensor no. 5 (Table 2.2) are shown below:

(x-axis)	(Y-axis)
0.58	-0.1802
0.56	0.0191
0.54	-0.8536
0.52	0.6126
0.51	0.2443

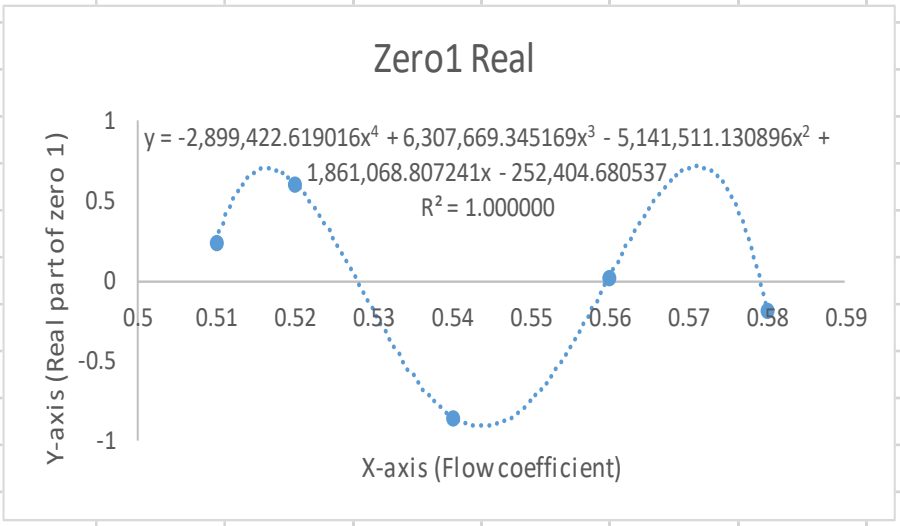

Figure 2.10: Plot and Equation of Real part of Zero 1 at Different Flow Coefficients.

X-axis	Y-axis
0.58	7.5664
0.56	28.1214
0.54	31.2561
0.52	22.8844
0.51	25.9897

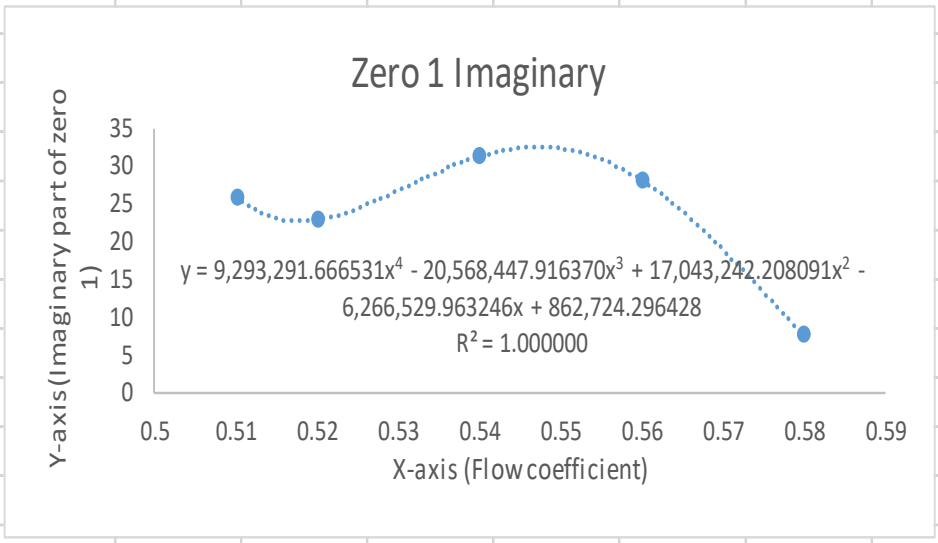

Figure 2.11: Plot and Equation of Imaginary part of Zero 1 at Different Flow Coefficients.

(x-axis)	(Y-axis)
0.58	-0.1802
0.56	0.0191
0.54	-0.8536
0.52	0.6126
0.51	0.2443

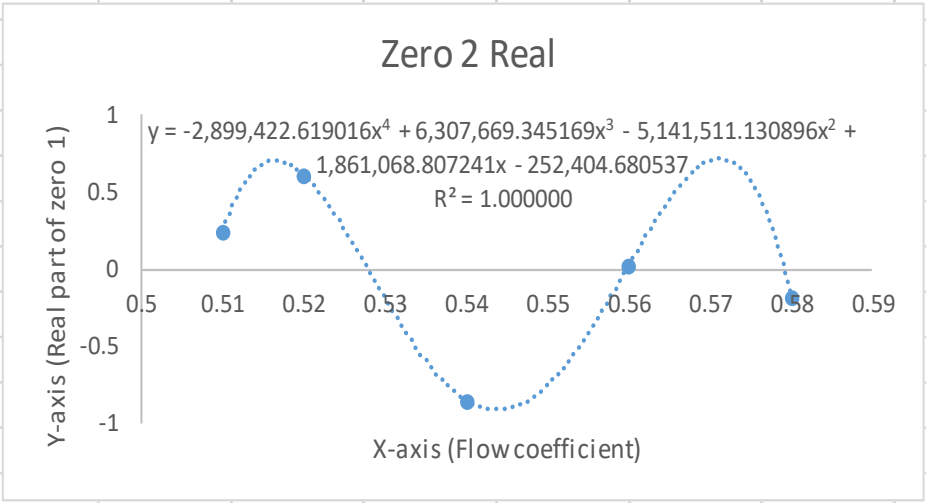

Figure 2.12: Plot and Equation of Real part of Zero 2 at Different Flow Coefficients

X-axis	Y-axis
0.58	-7.5664
0.56	-28.1214
0.54	-31.2561
0.52	-22.8844
0.51	-25.9897

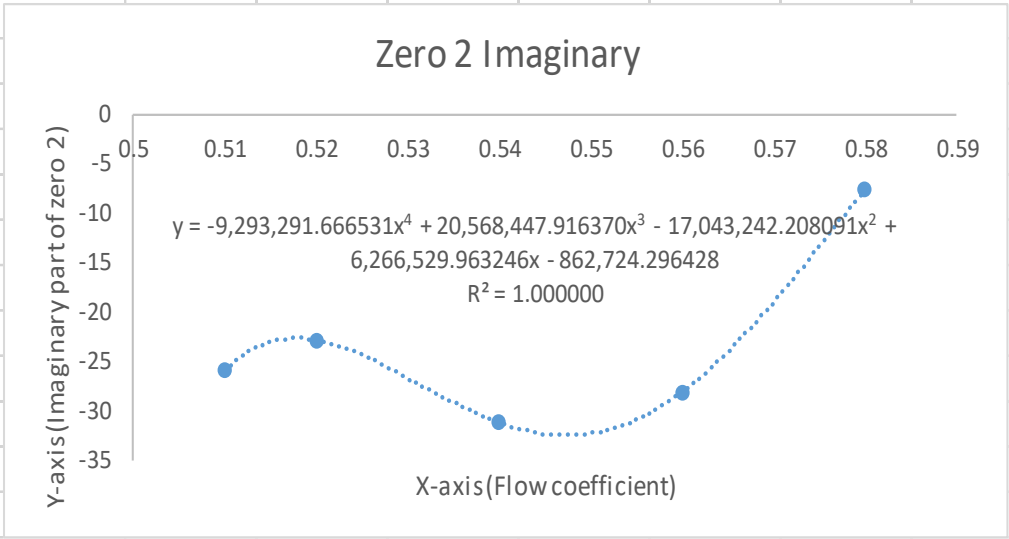

Figure 2.13: Plot and Equation of Imaginary part of Zero 2 at Different Flow Coefficients

X-axis	Y-axis
0.58	-110.61
0.56	-16.7
0.54	-0.3206
0.52	-88.0397
0.51	-70.2878

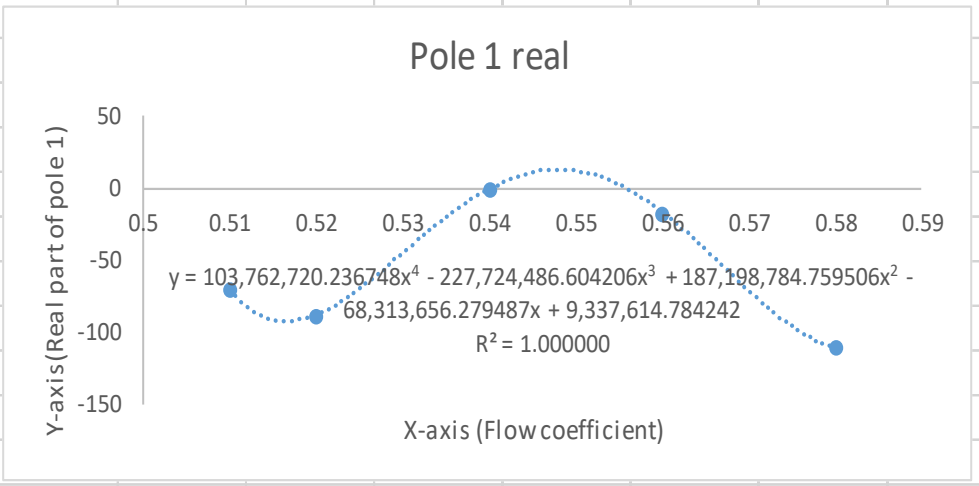

Figure 2.14: Plot and Equation of Real part of Pole 1 at Different Flow Coefficients

X-axis	Y-axis
0.58	58.57
0.56	113.78
0.54	86.3066
0.52	32.0824
0.51	0

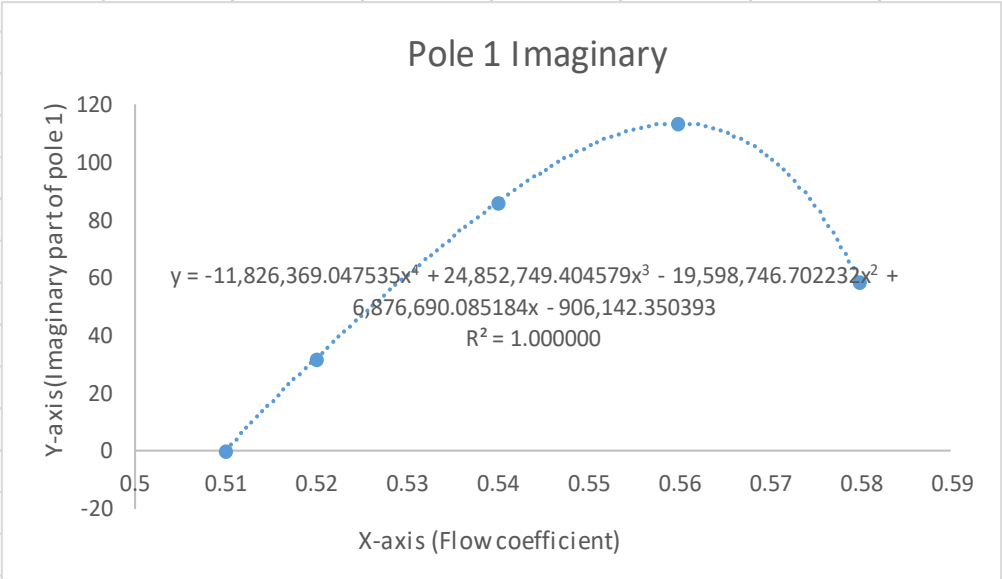

Figure 2.15: Plot and Equation of Imaginary part of Pole 1 at Different Flow Coefficients

X-axis	Y-axis
0.58	-110.61
0.56	-16.7
0.54	-0.3206
0.52	-88.0397
0.51	-11.3883

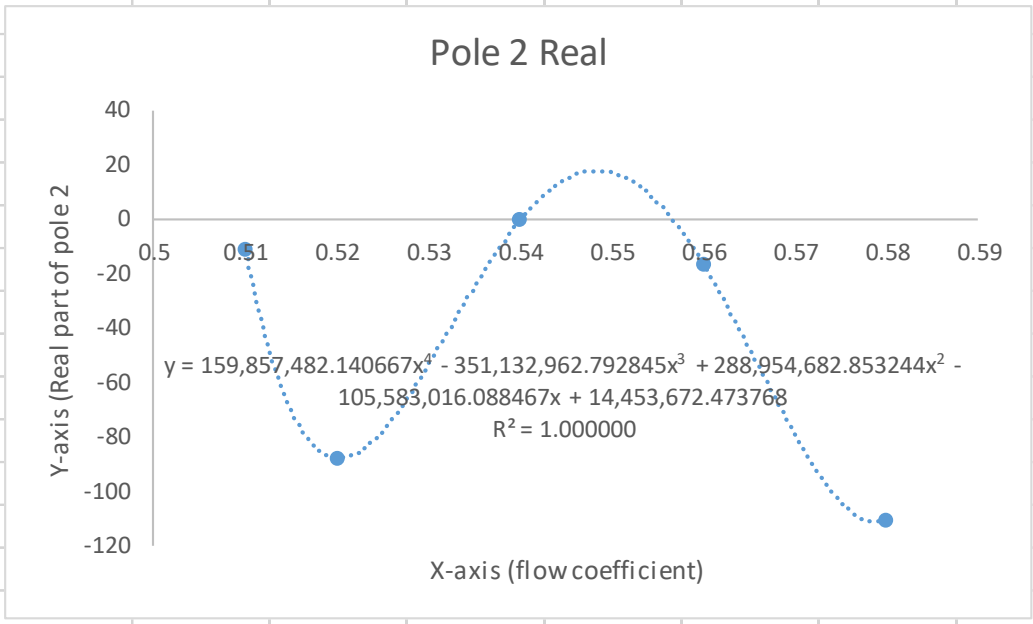

Figure 2.16: Plot and Equation of Real part of Pole 2 at Different Flow Coefficients

X-axis	Y-axis
0.58	-58.57
0.56	-113.78
0.54	-86.3066
0.52	-32.0824
0.51	0

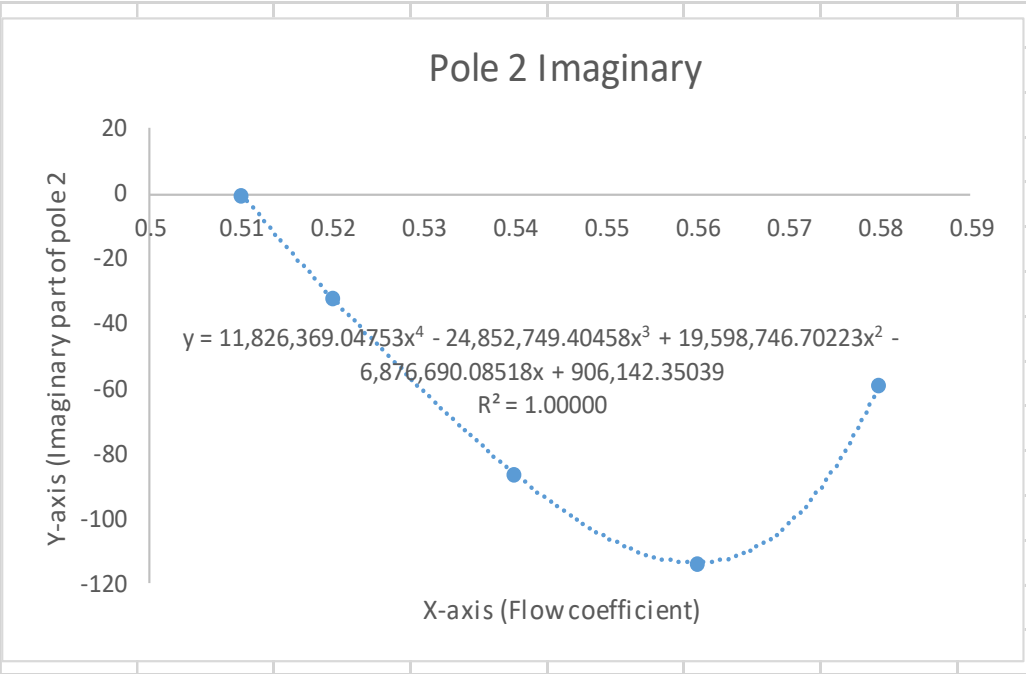

Figure 2.17: Plot and Equation of Imaginary part of Pole 2 at Different Flow Coefficients

X-axis	Y-axis
0.58	-0.18
0.56	0
0.54	0
0.52	-0.0226
0.51	-0.1389

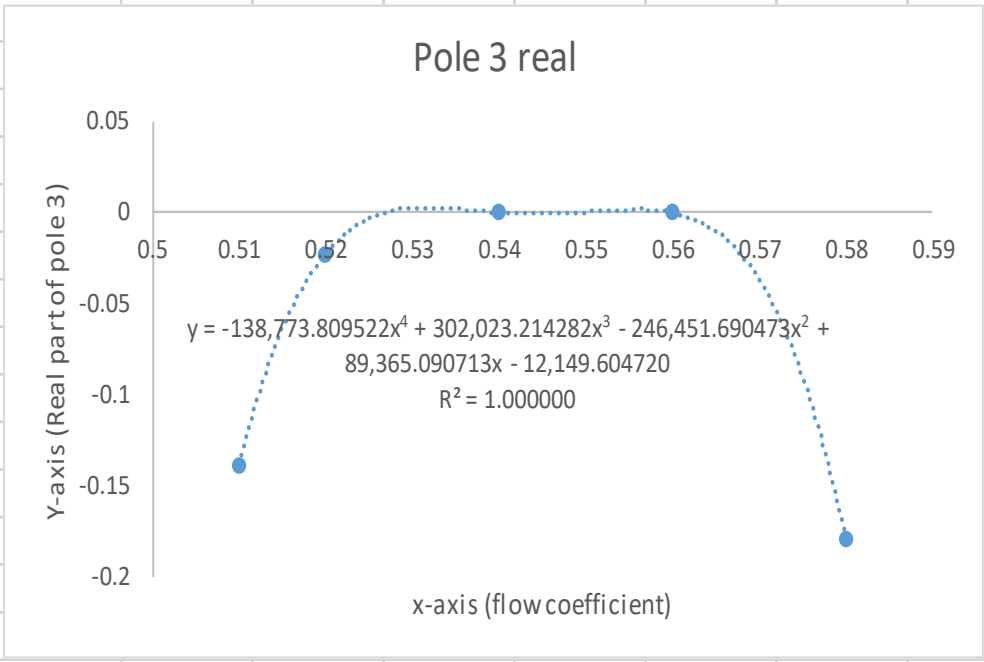

Figure 2.18: Plot and Equation of Real part of Pole 3 at Different Flow Coefficients

X-axis	Y-axis
0.58	7.5
0.56	26.95
0.54	31.6615
0.52	23.4365
0.51	25.3343

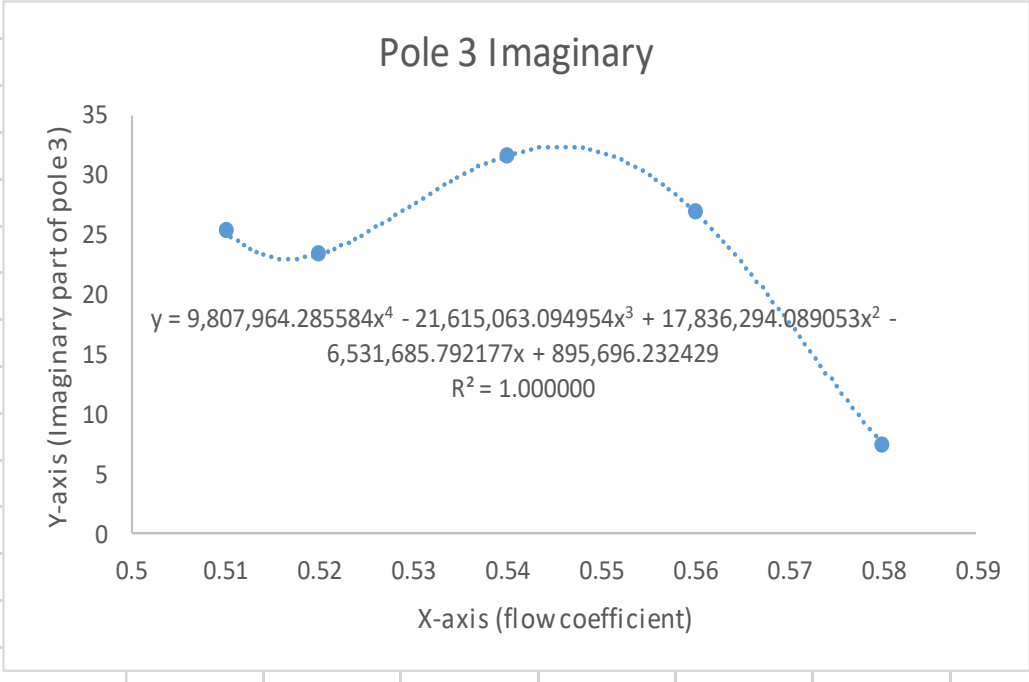

Figure 2.19: Plot and Equation of Imaginary part of Pole 3 at Different Flow Coefficients

X-axis	Y-axis
0.58	-0.18
0.56	0
0.54	0
0.52	-0.0226
0.51	-0.1389

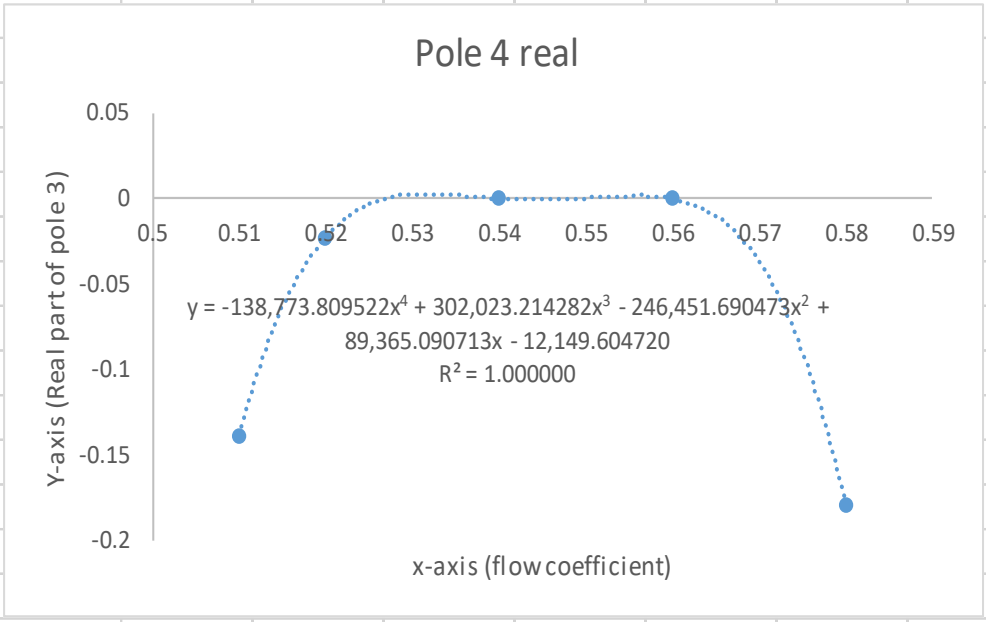

Figure 2.20: Plot and Equation of Real Part of Pole 4 at Different Flow Coefficients

X-axis	Y-axis
0.58	-7.5
0.56	-26.95
0.54	-31.6615
0.52	-23.4365
0.51	-25.3343

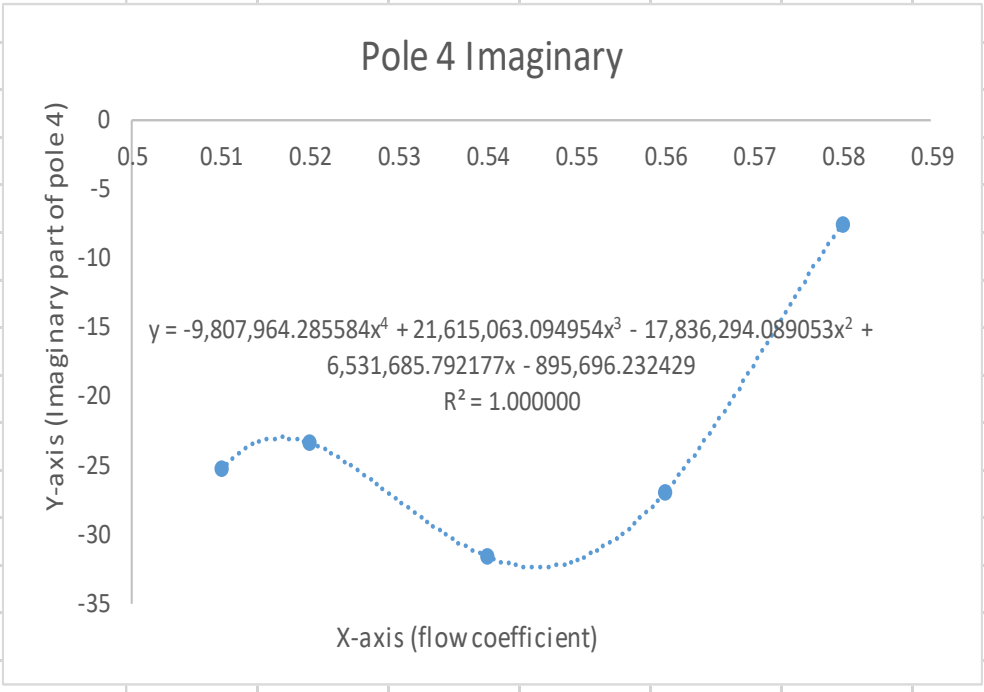

Figure 2.21: Plot and Equation of Imaginary part of Pole 4 at Different Flow Coefficients

2.4: Model Development

Transfer function and state space-models for different poles and zeroes combination were developed. The four models developed were:

- Four poles and two zeroes transfer function model
- Four poles and two zeroes state-space model
- Two poles and two zeroes transfer function model
- Two poles and two zeroes state-space model

Matlab™ was used to develop these models, the algorithm for model development are attached in the Appendix section. The basic syntax used for developing these models are:

sys=tf(num,den): The command creates a continuous time transfer function "sys" with numerator defined by "num" and denominator defined by "den". The output sys is:

- A transfer function model object, when the numerator and denominator are numeric arrays [33].
- A generalized state-space model when numerator or denominator are tunable parameters [33].
- An uncertain state space model when numerator and denominator are uncertain [33]

sys=ss(A,B,C,D): The command creates a state-space model object representing the continuous-time state-space model of the form:

$$\dot{x} = Ax + Bu \tag{2.5}$$

$$y = Cx + Du \tag{2.6}$$

For a model with Nx states, Ny outputs, and Nu inputs:

- A is a Nx-by-Nx real- or complex valued matrix [34].
- B is a Nx-by-Nu real- or complex valued matrix [34].
- C is a Ny-by-Nx real- or complex valued matrix [34].
- D is a Ny-by-Nu real- or complex valued matrix [34].

2.4.1: Controllable Canonical Form:

For converting the transfer function model in to a state-space model, controllable canonical form is used. To see how this method works, consider a third order transfer function:

$$H(s) = \frac{Y(s)}{U(s)} = \frac{b_0 s^2 + b_1 s + b_2}{s^3 + a_1 s^2 + a_2 s + a_3} \tag{2.7}$$

Let us start by multiplying by Z(s)/Z(s) and then solving for U(s) and Y(s) in terms of Z(s):

$$Y(s) = (b_0 s^2 + b_1 s + b_2) Z(s) \tag{2.8}$$

$$U(s) = (s^3 + a_1 s^2 + a_2 s + a_3) Z(s) \tag{2.9}$$

$$y = b_0 \ddot{z} + b_1 \dot{z} + b_2 z \tag{2.10}$$

$$u = \dddot{z} + a_1 \ddot{z} + a_2 \dot{z} + a_3 z \tag{2.11}$$

The next step is to choose the state variables:

$$q_1 = z \; ; \; q_2 = \dot{z} \; ; \; q_3 = \ddot{z} \tag{2.12}$$

$$\dot{q_1} = \dot{z} = q_2 \tag{2.13}$$

$$\dot{q_2} = \ddot{z} = q_3 \tag{2.14}$$

$$\dot{q_2} = \ddot{z} = q_3 \tag{2.15}$$

$$\dot{q_3} = \dddot{z} = u - a_1\ddot{z} - a_2\dot{z} - a_3 z$$

$$= u - a_1 q_3 - a_2 q_2 - a_3 q_1 \tag{2.16}$$

$$y = b_0\ddot{z} + b_1\dot{z} + b_2 z = b_0 q_3 + b_1 q_2 + b_2 q_1 \tag{2.17}$$

From these Equations, the state-space model can be represented as:

$$\dot{q} = Aq + Bu = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_3 & -a_2 & -a_1 \end{bmatrix} q + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u \tag{2.18}$$

$$y = Cq + Du = [b_2 \quad b_1 \quad b_0]\, q + 0.u \tag{2.19}$$

The above example presented is the basis for the one where the order of the numerator of the transfer function is less than the order of the denominator. The method does not work the same way if the order of the numerator and denominator of the transfer function are the same. For a general nth order transfer function given by:

$$H(s) = \frac{Y(s)}{U(s)} = \frac{b_0 s^n + b_1 s^{n-1} + \cdots + b_{n-1}s + b_n}{s^n + a_1 s^{n-1} + \cdots + a_{n-1}s + a_n} \tag{2.20}$$

The controllable canonical state-space model objects that works for all cases is given by:

$$A = \begin{bmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 1 \\ -a_n & -a_{n-1} & -a_{n-2} & \ldots & -a_1 \end{bmatrix} \tag{2.21}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad (2.22)$$

$$C = [(b_n - a_n b_0)\ (b_{n-1} - a_{n-1} b_0)\ \ldots (b_2 - a_2 b_0)\ (b_1 - a_1 b_0)] \quad (2.23)$$

$$D = b_0 \quad (2.24)$$

2.5: Chapter Summary

- Discussion on the approach used for the system identification of fast dynamics
- Fast dynamics transfer function model data for sensor no. 4,5, and 6 presented in Table 2.1, 2.2 and 2.3
- Steps involved in developing the polynomial equations of the behavioral curve of poles and zeroes at different flow coefficients of the transfer function model data presented in the form of screenshots
- Plots and equations of real and imaginary part of poles and zeroes at different flow coefficients are presented in the form of figures

Chapter 3: Controller

3.1: Transformation Matrix

For any dynamic systems, the realized state-space model is not unique. For comparison of any form between the analytical model and the realized state-space model, both the models have to be in the same coordinate frame. One such unique transformation matrix is derived in [37]. This unique transformation matrix can transform any realized state-space model to be in a from usually used for a structural dynamic system so that any identified system parameter can be compared with the corresponding analytical one [37]. This unique transformation matrix exists only when one half of the states can be measured directly. A general representation of a dynamic system is given by Equation 3.1, where p is the displacement, u is the control force, G is the control influence matrix, and M, D, and K are mass, damping, and stiffness matrices, respectively. After converting the realized discrete-time state-space system to continuous-time system, the parameters of which are (A_c, B_c, C), a unique transformation matrix T is given by Equation 3.2

$$M\ddot{p} + D\dot{p} + Kp = Gu \tag{3.1}$$

$$T_m = [t_1 \quad t_2] = \begin{bmatrix} C \\ CA_c \end{bmatrix}^{-1} \tag{3.2}$$

3.1.1: Derivation of the Transformation Matrix

The state-space equation and output equation of a dynamic system is given by Equation 3.3

$$\dot{x} = A_m x + B_m u \; and \; y = C_m x \tag{3.3}$$

where, $x = \begin{Bmatrix} p \\ \dot{p} \end{Bmatrix}$, $A_m = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}D \end{bmatrix}$, $B_m = \begin{bmatrix} 0 \\ M^{-1}G \end{bmatrix}$ and C_m is the output matrix. If the assumption is made that one-half states can be measured directly, one can have $C_m = [I \quad 0]$

After converting a realized discrete-time system $[A, B, C]$ to a continuous-time system $[A_c, B_c, C]$, If A is diagonalized by matrix Q, then

$$Q^{-1}AQ = A, A_c = Q\frac{\ln(A)}{T}Q^{-1}, B_c = (A - I)^{-1}A_cB \tag{3.4}$$

where, T is the sampling time. If the transformation matrix is given by Equation 3.2, then

$$T_m{}^{-1}T_m = \begin{bmatrix} C \\ CA_c \end{bmatrix}[t_1 \quad t_2] = \begin{bmatrix} Ct_1 & Ct_2 \\ CA_ct_1 & CA_ct_2 \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \tag{3.5}$$

$$T_m{}^{-1}A_cT_m = \begin{bmatrix} C \\ CA_c \end{bmatrix}A_c[t_1 \quad t_2] = \begin{bmatrix} CA_ct_1 & CA_ct_2 \\ CA_c{}^2t_1 & CA_c{}^2t_2 \end{bmatrix} = \begin{bmatrix} 0 & I \\ X & X \end{bmatrix} \tag{3.6}$$

$$T_m{}^{-1}B_c = \begin{bmatrix} CB_c \\ CA_cB_c \end{bmatrix} = \begin{bmatrix} C_mB_m \\ CA_cB_c \end{bmatrix} = \begin{bmatrix} 0 \\ X \end{bmatrix} \tag{3.7}$$

$$CT_m = [Ct_1 \quad Ct_2] = [I \quad 0] \tag{3.8}$$

We can see that $CT_m = C_m$. Thus, the transformation of a continuous time system $[A_c, B_c, C]$ to an analytical system is given by Equations 3.9, 3.10, and 3.11. After the transformation, both the identified model and the analytical model are in the same coordinate frame and can be compared.

$$A_c \xrightarrow{yields} T_m{}^{-1}A_cT_m \tag{3.9}$$

$$B_c \xrightarrow{yields} T_m{}^{-1}B_c \tag{3.10}$$

$$C \xrightarrow{yields} CT_m \tag{3.11}$$

3.2: Linear-Quadratic Gaussian(LQG) Controller

In control theory, linear-quadratic gaussian control problem is one of the most fundamental optimal control problem. It has many applications in the modern world ranging from aircraft and missile navigation control systems, nuclear power plants as well as medical processes controllers [38]. Linear-quadratic gaussian controller as the name suggests, is a controller that reduces the quadratic cost function of a linear system with additive white gaussian noises. The LQG controller is simply a combination of both concepts of linear- quadratic regulator(LQR) and Kalman filter—which is the linear-quadratic estimator(LQE).

3.2.1: Linear quadratic Regulator (LQR)

Consider a linear system modeled by Equation 3.12, where, $x(k) \in R^n$ and $u(k) \in R^m$

$$x(k+1) = Ax(k) + Bu(k), x(k_0) = x_0 \tag{3.12}$$

The pair (A, B) is controllable. The objective is to design a stabilizing linear state feedback controller $u(k) = -kx(k)$ which will minimize the quadratic cost function given by Equation 3.13, where Q and R are noise covariances and the gain matrix k is given by Equation 3.14. P is obtained by solving the Algebraic Riccati Equation (ARE) given by Equation 3.15

$$J = \sum_{k=0}^{\infty} (x^T(k)Qx(k) + u^T(k)Ru(k)) \tag{3.13}$$

$$k = (B^T PB + R)^{-1} B^T PA \tag{3.14}$$

$$P = A^T PA - P + Q - A^T PBS^{-1} B^T PA \tag{3.15}$$

3.2.2: Kalman State Estimator

Given the discrete plant:

$$x[n+1] = Ax[n] + Bu[n] + Gw[n] \tag{3.16}$$

$$y[n] = Cx[n] + Du[n] + v[n] \tag{3.17}$$

And the noise covariance data given by Equation 3.18

$$E(w[n]w[n]^T) = Q, \qquad E(v[n]v[n]^T) = R, \qquad E(w[n]v[n]^T) = 0 \tag{3.18}$$

The estimator has the following Equation:

$$x[n|n] = x[n|n-1] + M(y[n] - Cx[n|n-1] - Du[n]) \tag{3.19}$$

where, M is the gain matrix. The Ricaati solution is given by Equation 3.20 and the error covariances are given by Equation 3.21.

$$P = E\{(x[n|n-1] - x)(x[n|n-1] - x)^T\} \tag{3.20}$$

$$Z = E\{(x[n|n] - x)(x[n|n] - x)^T\} \tag{3.21}$$

3.3: Matlab™ codes used for development of the Controller

1. dlqry

The code designs a linear quadratic regulator with output weighting for discrete time systems.

The syntax:

[K,S,E]=dlqry(A,B,C,D,Q,R), calculates the optimal feedback gain matrix k such that the feedback law $u[n] = -Kx[n]$, minimizes the cost function given by Equation 3.22, Which are subject to the constraint Equations 3.23 and 3.24

$$J = Sum\{y'Qy + u'Ru\} \tag{3.22}$$

$$X[n+1] = Ax[n] + Bu[n] \tag{3.23}$$

$$Y[n] = Cx[n] + Du[n] \tag{3.24}$$

The syntax also returns S, the steady-state solution to the associated discrete matrix Riccati Equation and the closed loop eigenvalues given by Equation 3.25

$$E = EIG(A - B * K) \tag{3.25}$$

2. dlqe

This Matlab™ code designs the Kalman estimator for discrete time systems. The system state equation is given by Equation 3.26 and the measurements are given by Equation 3.27, Where, $w[n]$ is the process noise and $v[n]$ is the measurement noise with covariances given in Equation 3.28

$$x[n+1] = Ax[n] + Bu[n] + Gw[n] \tag{3.26}$$

$$y[n] = Cx[n] + Du[n] + v[n] \tag{3.27}$$

$$E\{ww'\} = Q, \qquad E\{vv'\} = R, \qquad E\{wv'\} = 0 \tag{3.28}$$

The syntax:

[M, P, Z, E] = dlqe(A,G,C,Q,R), returns the gain matrix M, such that the discrete, stationary Kalman filter with observation and time update equations shown in Equations 3.29 and 3.30 produces an optimal state estimate $x[n|n]$ of $x[n]$ given $y[n]$ and the past measurements. The syntax also returns the Riccati solution given by Equation 3.31, the error covariance given by Equation 3.32, and the estimator poles $E = EIG(A - A * M * C)$

$$x[n|n] = x[n|n-1] + M(y[n] - Cx[n|n-1] - Du[n]) \tag{3.29}$$

$$x[n+1|n] = Ax[n|n] + Bu[n] \tag{3.30}$$

$$P = E\{(x[n|n-1] - x)(x[n|n-1] - x)'\} \tag{3.31}$$

$$Z = E\{(x[n|n] - x)(x[n|n] - x)'\} \tag{3.32}$$

3. dreg

The syntax is:

[Ac,Bc,Cc,Dc] = dreg(A,B,C,D,K,L)

It produces the LQG controller based on the discrete system (A, B, C, D) with feedback gain matrix K, and Kalman gain matrix L, assuming all the inputs of the system are control inputs and all the outputs of the system are sensor outputs. The resulting state-space controller is given by Equations 3.33 and 3.34 where, $E = inv(I + KLD)$ and has control feedback commands uHat as

outputs and sensors y as inputs. The controller should be connected to the plant using negative feedback.

$$xBar[n+1] = [A - ALC - (B - ALD)E(K - KLC)]\, xBar[n] + [AL - (B - ALD)EKL]\, y[n] \quad (3.33)$$

$$uHat[n] = [K - KLC + KLDE(K - KLC)]xBar[n] + [KL + KLDEKL]\, y[n] \quad (3.34)$$

The syntax:

[Ac,Bc,Cc,Dc] = dreg(A,B,C,D,K,L,SENSORS,KNOWN,CONTROLS), forms the LQG controller using the sensors specified by "SENSORS", the additional known inputs specified by "KNOWN", and the control inputs specified by "CONTROLS". The resulting system has control feedback commands as outputs and the known inputs and sensors as inputs. The "KNOWN" inputs are non-stochastic inputs of the plant and are usually additional control inputs or command inputs.

3.4: Chapter Summary

- A unique transformation matrix to compare both identified model and the analytical model is presented
- A brief introduction on Linear Quadratic Gaussian controller
- Derivation of Linear Quadratic Regulator and Kalman State Estimator presented in the form of equations
- Description of the Matlab™ codes used for development of the controller

Chapter 4: Experimental Setup and Data Collection

4.1: Experimental Setup

The experimental compressor system that is used to obtain data for the analysis presented in this research is a one-stage axial compressor. The experimental compressor is set up at the Advanced Energy and Power Technology Laboratory at the Institute of Engineering Thermophysics (IET), Chinese Academy of Sciences, located in Beijing, China. The specifications of the experimental compressor are given in Table 4.1.

Table 4.1: Parameters of the Experimental Axial Compressor

Parameter	Numerical Value	Units
Design Speed	2400	rpm
Rotor Blade Number	58	-
Outer Casing Diameter	500	mm
Mass Flow Rate	2.9	Kg/s
Rotor Tip Chord	36.3	mm
Rotor Tip Stagger Angle	39.2	degree
Hub-tip ratio	0.75	-

The front and side view of the experimental compressor is shown in Figure 4.1 and 4.2. In Figure 4.2, the inlet is on the left and the outlet is located to the right. The schematic diagram of a one stage axial compressor is shown in Figure 1.1 (Chapter 1). As discussed in Chapter 1, the occurrence of stall has been linked to several factors like the tip clearance, inlet velocity of the air flow, tip incidence of the blades, flow separation etc. The research compressor at IET exhibits spike type stall inception. The experimental compressor rotor has 58 blades. The side view of the compressor rotor blades is shown in Figure 4.3.

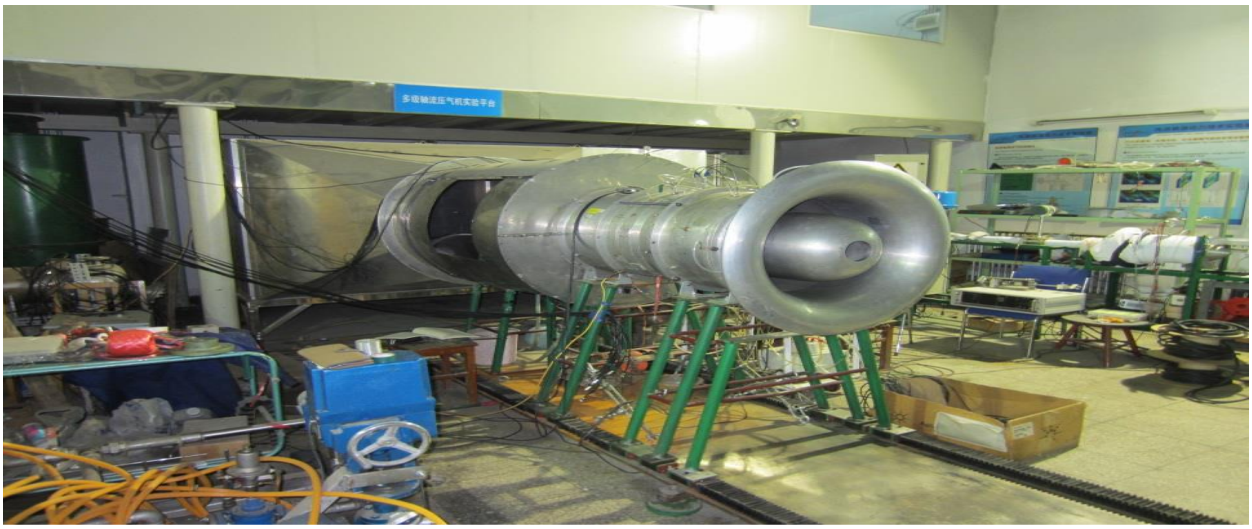

Figure 4.1: Front view of the Experimental Compressor [2]

Figure 4.2: Side view of the Experimental Compressor [2]

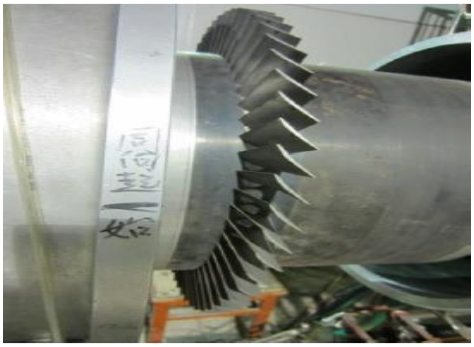

Figure 4.3 Side view of the Compressor Rotor Blades [2]

The outlet of the compressor consists of two parts. One is a stationary throttle cone and the other is a movable throttle outer cylinder. The outer cylinder passes through the wall and has a sliding mechanism for its back and forth movement. Figure 4.4 shows the outlet set-up of the experimental compressor.

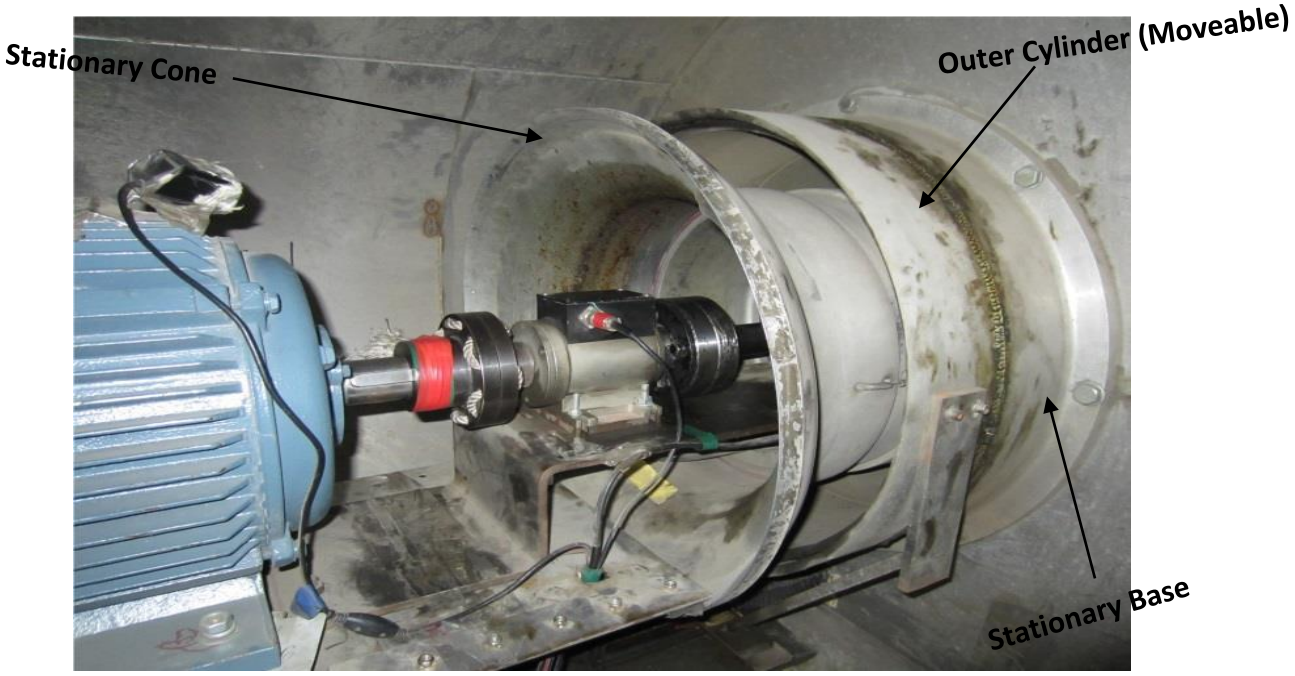

Figure 4.4: Outlet set-up of the Experimental Compressor at IET [2]

The cylinder is used to regulate the flow rate of the compressor. The movable outer cylinder is 10 mm thick and is made from aluminum. The outer diameter of the cylinder is 540 mm and is 118 mm in width. A stepper motor is used for the actuation of the outer cylinder. For the forward and the backward movement of the outer cylinder, a screw mechanism is used. The movable cylinder is bolted to a support structure that moves along the rail. The structure is connected to the stepper motor and the clockwise and anti-clockwise rotation of the stepper motor drives the structure connected to the outer cylinder in the forward and backward directions respectively. Figure 4.5 shows the mechanism of throttle actuation. The stepper motor used for

the actuation of the screw mechanism is mounted on the exterior of the compressor set up as shown in Figure 4.6. The stepper motor drives the outer cylinder at a constant speed of 2mm/sec [2].

Figure 4.5: Throttle Actuation Mechanism [2]

Figure 4.6: Stepper Motor used for Throttle Actuation Mounted on the Compressor Exterior [2]

4.2: Data Collection

To capture the fast dynamics of the experimental compressor, ten anemometer type pressure sensors are attached to the compressor casing. The sensors are positioned in a way that it spans the length of the rotor blade along the longitudinal axis of the compressor. The position of the sensors relative to the rotor blades is shown in Figure 4.7.

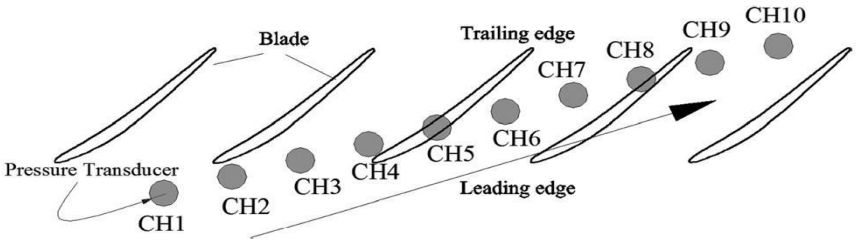

Figure 4.7: Sensor Location Relative to the Rotor Blades

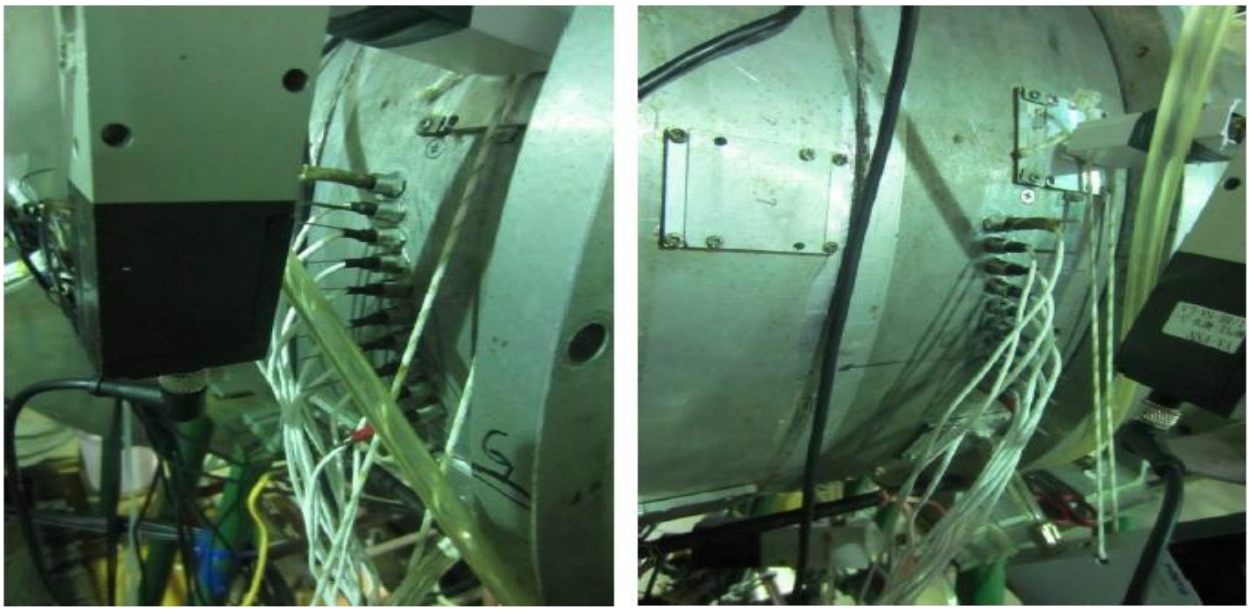

Figure 4.8: Arrangement of the Dynamic Sensors on the Compressor Casing [2]

The dynamic pressure sensor locations distances relative to the rotor blades are summarized in Table 4.2. The calibration factors between the sensor voltage signal and the pressure in pascals

are included in Table 4.3. Figure 4.8 shows the arrangement of the dynamic sensors on the casing of the experimental compressor.

Table 4.2: Pressure Sensor Location Distances Relative to the Compressor Rotor Blades [2]

Sensor Number	Axial Distance from leading edge of Rotor blade	Axial distance from leading edge of Rotor blade (% of Axial Rotor Length)
Sensor 1 (leading edge)	-5.33	-23.71
Sensor 2	-1.79	-7.96
Sensor 3	1.14	5.07
Sensor 4	4.295	19.10
Sensor 5	7.08	31.49
Sensor 6	10.99	48.89
Sensor 7	13.705	60.96
Sensor 8	17.09	76.02
Sensor 9 (Trailing)	19.76	87.90

Table 4.3 Dynamic Pressure Sensor Calibration Factors [2]

Sensor Number	Sensor Calibration Factor (Pa/V)
Sensor 1	968.3
Sensor 2	879.9
Sensor 3	891.9
Sensor 4	603.7
Sensor 5	599.2
Sensor 6	599.5
Sensor 7	596.8
Sensor 8	600.0
Sensor 9	917.7

To alter the tip leakage flow near the compressor blades, eight air injection actuators are used. These air injectors, capable of both macro-injection and micro-injection, are equally spaced around the compressor casing. The arrangement of these air injection actuators is shown in Figure 4.9. As seen in the picture, a tube is used to carry pressurized air into the actuator from a nearby pressurized storage vessel. In order to alter the angle at which the air is injected into the tip gap, the angle of the jets can be adjusted to angles such as 15° and 45°.

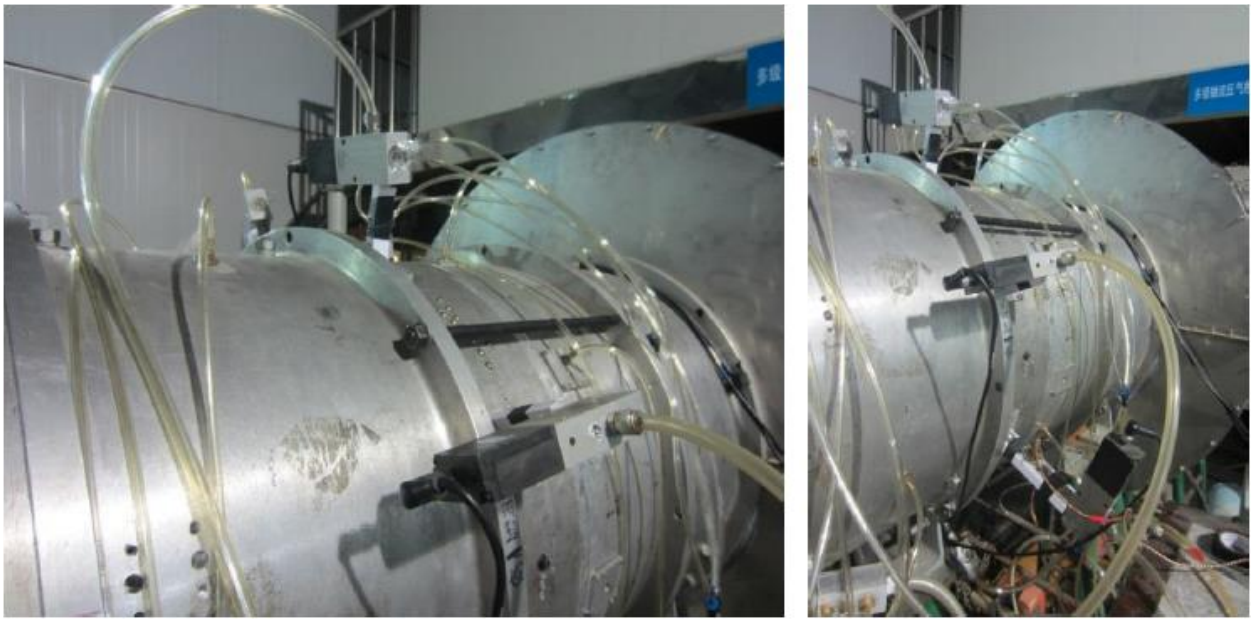

Figure 4.9: Arrangement of Air Injector Actuators [2]

CHAPTER 5: RESULTS AND DISCUSSION

5.1: Results

The developed model uses the air injection pressure signal as the input and the autocorrelation coefficient of the dynamic pressure data across the rotor blades as the output. The controller input is represented by Equation 5.1 where, $x(t = 0) = [0; 0]$, k is the feedback gain matrix and kr is the feed-forward gain matrix, and u_c is the command input. The complete Matlab™ code of the developed model along with the controller is attached in the Appendix section.

$$u(i) = -kx + kr * u_c \tag{5.1}$$

Simulations using actual data and generated data were performed to investigate the performance of the proposed controller. The different control inputs used to check the model output are shown in Equations 5.2-5.5. The collected pressure sensor data at different flow coefficients is designated as $VarName10$.

$$u_1 = u(i) = -kx + kr * VarName10 \tag{5.2}$$

$$u_2 = u(i) = -kx + kr * -1 \tag{5.3}$$

$$u_3 = u(i) = -kx + kr * 0 \tag{5.4}$$

$$u_4 = u(i) = -kx + kr * 1 \tag{5.5}$$

5.1.1: Simulation Results for Flow Coefficient 0.58

The flow coefficient of 0.58 corresponds to a stable, far away from stall inception, operating point (refer Figure 1.11). Figure 5.1 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.58. The input for the simulation given in Figure 5.1 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.1 b) is $\boldsymbol{u_1}$ (Equation 5.2), which corresponds to the collected data (input) from the compressor and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

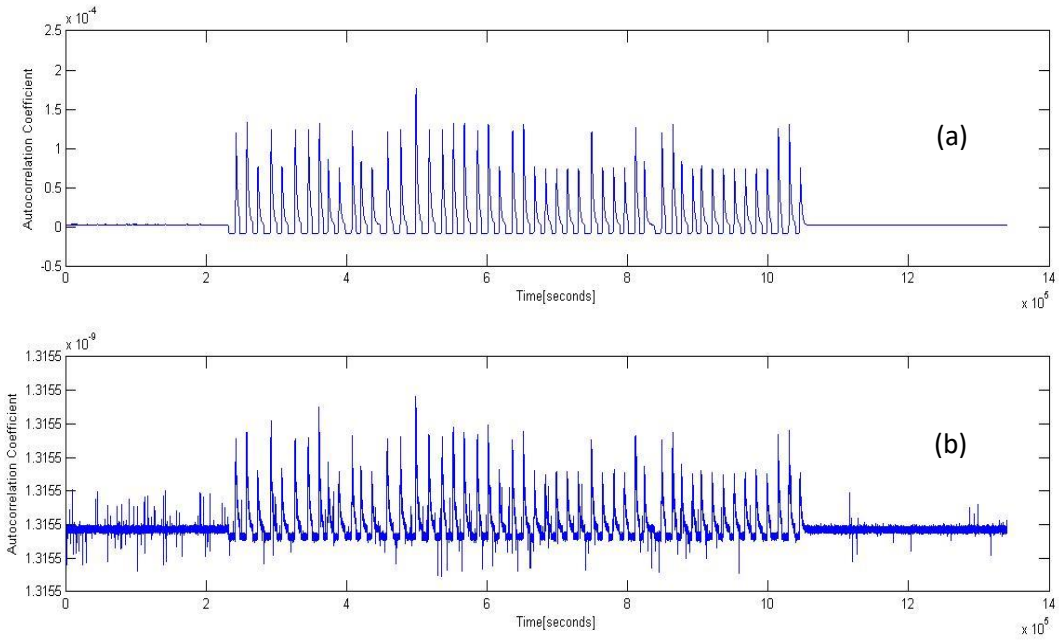

Figure 5.1: Plot of Uncontrolled System Output (a) vs. Controlled System Output using $\boldsymbol{u_1}$(b) for Flow Coefficient 0.58

Figure 5.2 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.58. The input for the simulation given in Figure 5.2 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.2 b) is $\boldsymbol{u_2}$ (Equation 5.3), which corresponds to a steady input and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

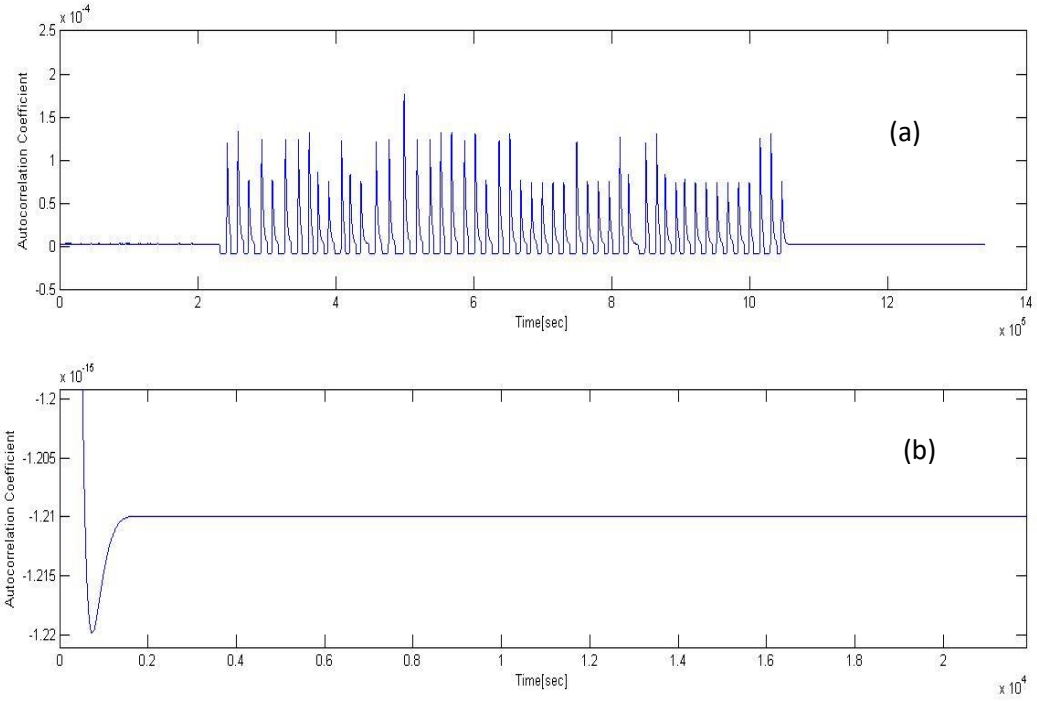

Figure 5.2: Plot of Uncontrolled System Output(a) vs. Controlled System Output using $\boldsymbol{u_2}$ (b) for ϕ=0.58

Figure 5.3 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.58. The input for the simulation given in Figure 5.3 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.3 b) is $\boldsymbol{u_3}$ (Equation 5.4), which corresponds to a steady input and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

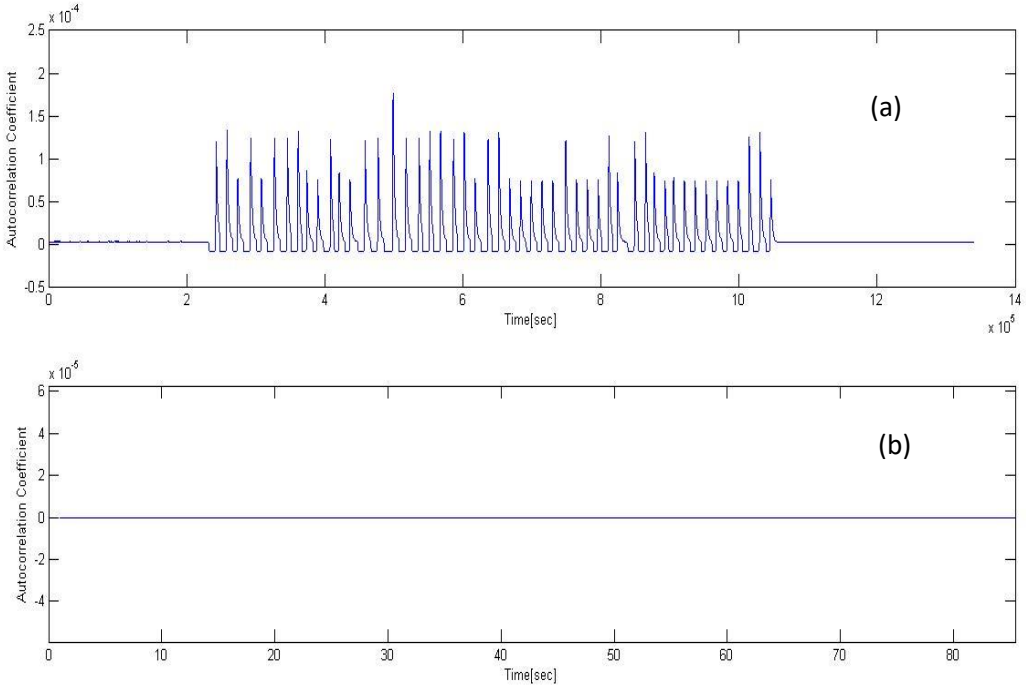

Figure 5.3: Plot of Uncontrolled System Output(a) vs. Controlled System Output using $\boldsymbol{u_3}$(b) for ϕ=0.58

Figure 5.4 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.58. The input for the simulation given in Figure 5.4 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.4 b) is $\boldsymbol{u_4}$ (Equation 5.5), which corresponds to a steady input and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

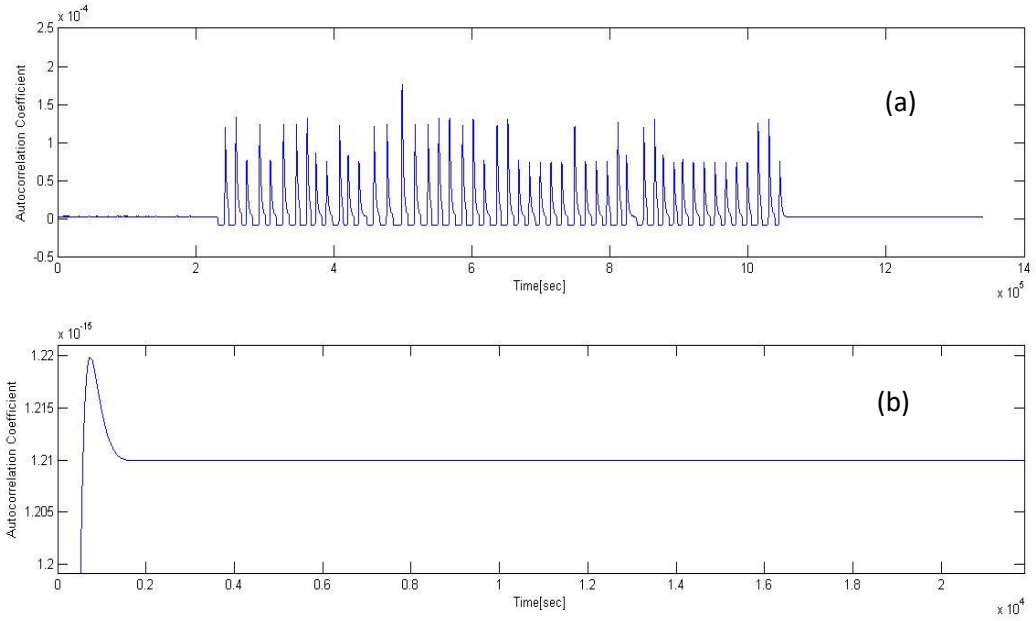

Figure 5.4: Plot of Uncontrolled System Output(a) vs. Controlled System Output using $\boldsymbol{u_4}$(b) for ϕ=0.58

Summary observation for flow coefficient 0.58

- In Figure 5.1, the command input u_c is given by measurement, which is a pulsing function as these pulses are used to excite the system for system identification. The response (as given as an autocorrelation) depicted in 5.1 b) matches the form of the input well. The magnitude is substantially smaller.
- In Figure 5.2, the command input u_c is a steady input of value -1. The response (as given as an autocorrelation) depicted in 5.2 b) stabilizes the compressor and matches the value of the command input. The magnitude is substantially smaller.
- In Figure 5.3, the command input u_c is a steady value of 0. The response (as given as an autocorrelation) depicted in 5.3 b) stabilizes the compressor at the given value of the command input.
- In Figure 5.4, the command input u_c is a steady value of 1. The response (as given as an autocorrelation) depicted in 5.4 b) stabilizes the compressor and matches the value of the command input.

5.1.2 Simulation Results for Flow Coefficient 0.56

The flow coefficient of 0.56 corresponds to a stable, away from stall inception, operating point (refer Figure 1.11). Figure 5.5 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.56. The input for the simulation given in Figure 5.5 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.5 b) is $\boldsymbol{u_1}$ (Equation 5.2), which corresponds to the collected data (input) from the compressor and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

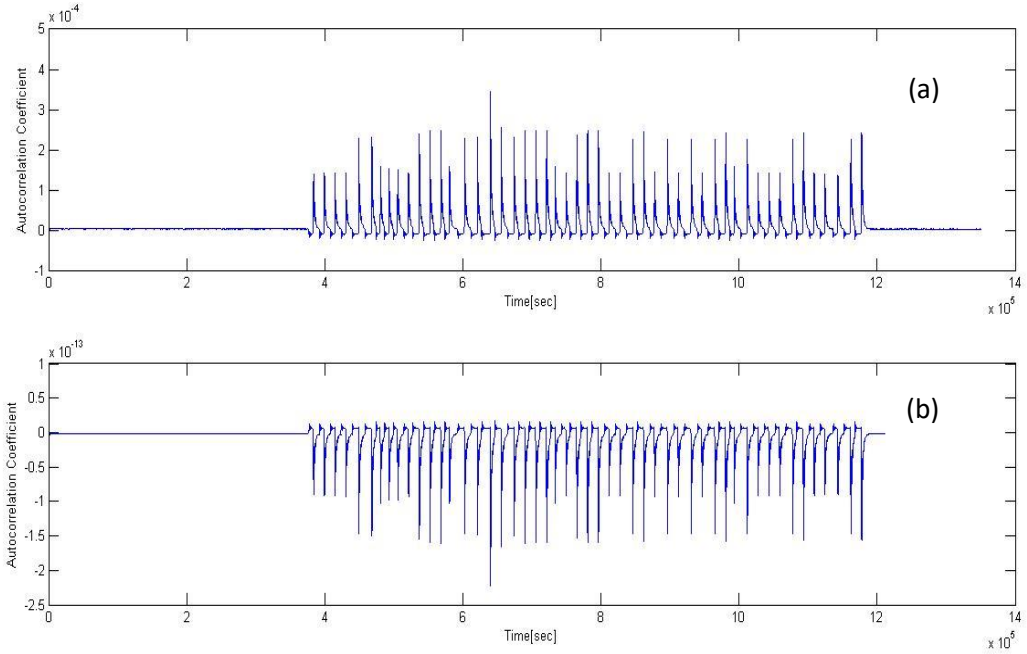

Figure 5.5: Plot of Uncontrolled System Output(a) vs. Controlled System Output using $\boldsymbol{u_1}$(b) for ϕ=0.56

Figure 5.6 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.56. The input for the simulation given in Figure 5.6 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.6 b) is $\boldsymbol{u_2}$ (Equation 5.3), which corresponds to a steady input and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

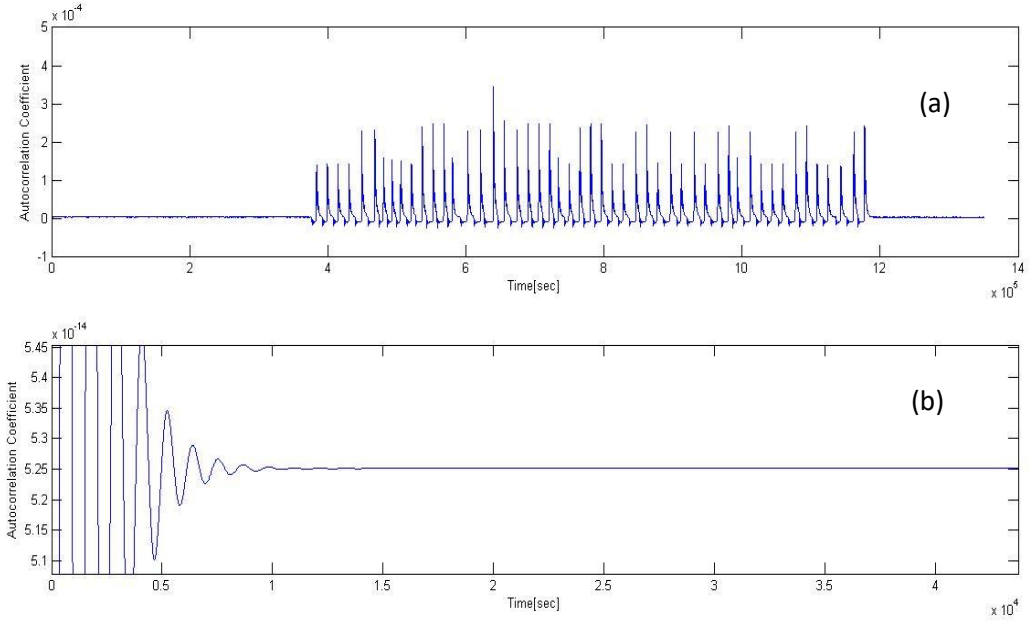

Figure 5.6: Plot of Uncontrolled System Output(a) vs. Controlled System Output using u_2(b) for ϕ=0.56

Figure 5.7 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.56. The input for the simulation given in Figure 5.7 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.7 b) is $\boldsymbol{u_3}$ (Equation 5.4), which corresponds to a steady input and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

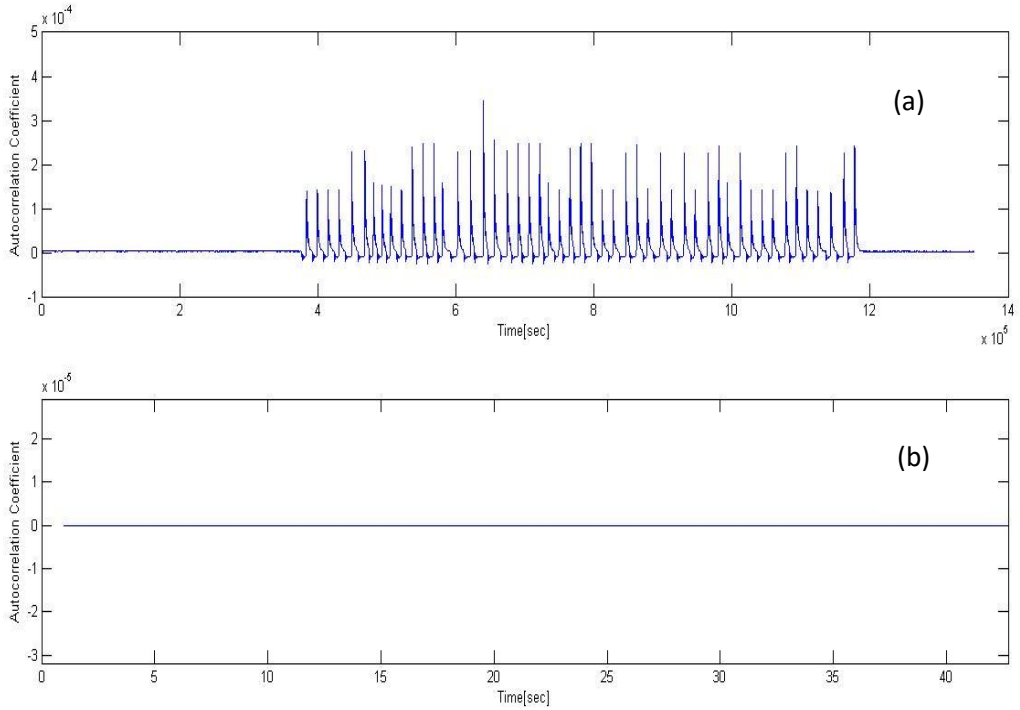

Figure 5.7: Plot of Uncontrolled System Output(a) vs. Controlled System Output using u_3(b) for ϕ=0.56

Figure 5.8 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.56. The input for the simulation given in Figure 5.8 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.8 b) is $\boldsymbol{u_4}$ (Equation 5.5), which corresponds to a steady input and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

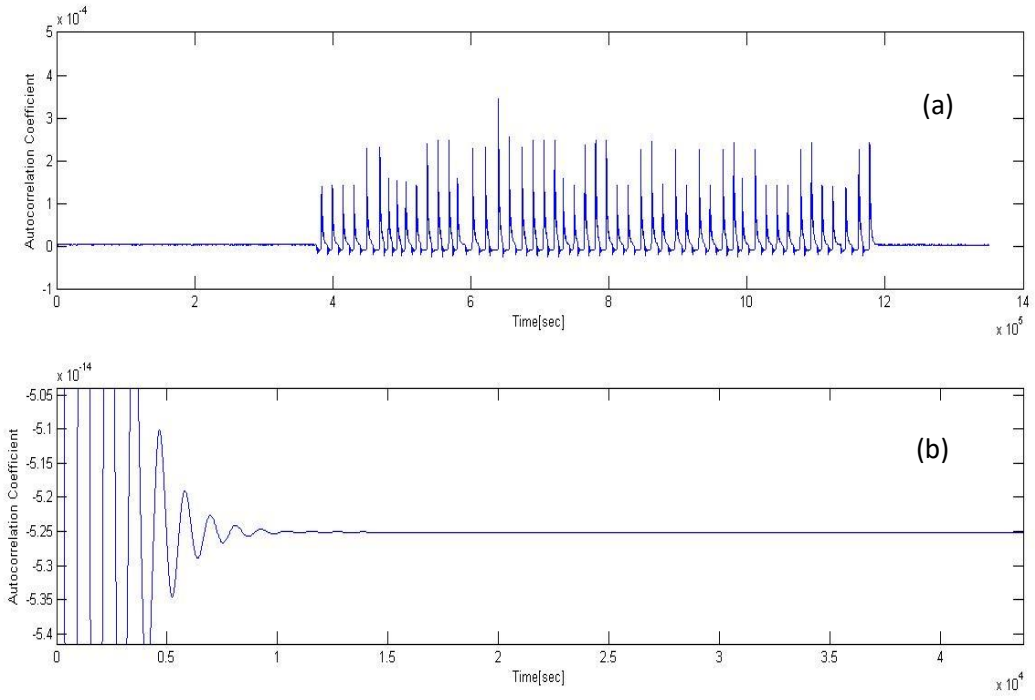

Figure 5.8: Plot of Uncontrolled System Output(a) vs. Controlled System Output using u_4(b) for ϕ=0.56

Summary observation for flow coefficient 0.56

- In Figure 5.5, the command input u_c is given by measurement, which is a pulsing function as these pulses are used to excite the system for system identification. The response (as given as an autocorrelation) depicted in 5.5 b) matches the form of the input but in an inverse pattern. The magnitude is substantially smaller.
- In Figure 5.6, the command input u_c is a steady input of value -1. The response (as given as an autocorrelation) depicted in 5.6 b) stabilizes the compressor but does not exactly match the value of the command input. The magnitude is substantially smaller.
- In Figure 5.7, the command input u_c is a steady value of 0. The response (as given as an autocorrelation) depicted in 5.7 b) stabilizes the compressor and matches the value of the command input.
- In Figure 5.8, the command input u_c is a steady value of 1. The response (as given as an autocorrelation) depicted in 5.8 b) stabilizes the compressor, but does not exactly match the value of the command input. The magnitude is substantially smaller.

5.1.3: Simulation Results for Flow Coefficient 0.54

The flow coefficient of 0.54 corresponds to a region of instability where the experimental compressor exhibits a sudden spike in the compressor output (refer Figure 1.11). Figure 5.9 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.54. The input for the simulation given in Figure 5.9 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.9 b) is $\boldsymbol{u_1}$ (Equation 5.2), which corresponds to the collected data (input) from the compressor and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

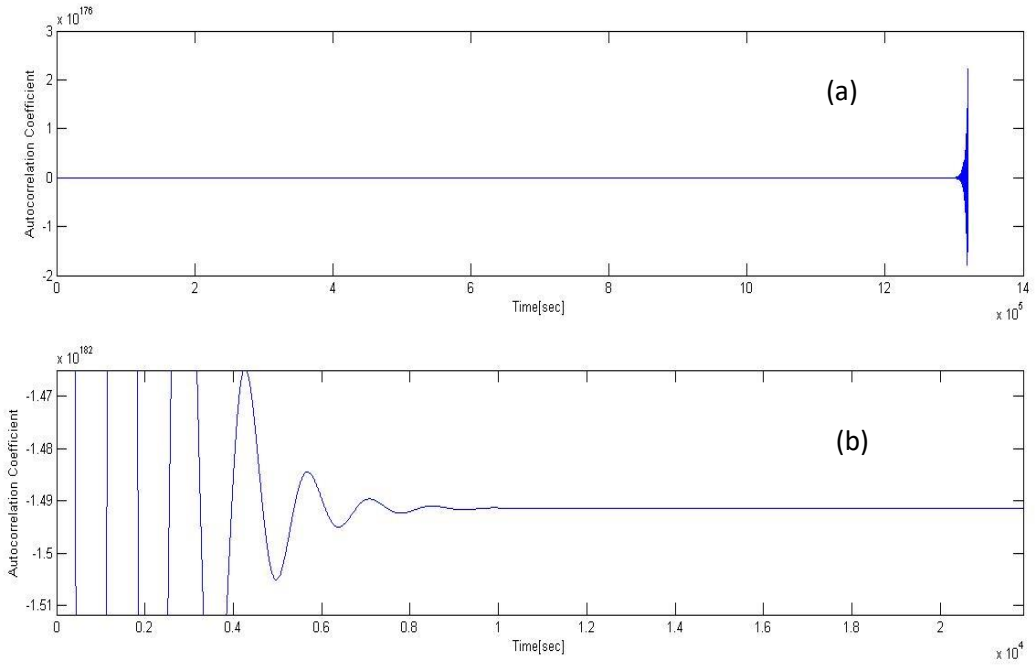

Figure 5.9: Plot of Uncontrolled System Output(a) vs. Controlled System Output using $\boldsymbol{u_1}$(b) for ϕ=0.54

Figure 5.10 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.54. The input for the simulation given in Figure 5.10 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.10 b) is $\boldsymbol{u_2}$ (Equation 5.3), which corresponds to a steady input and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

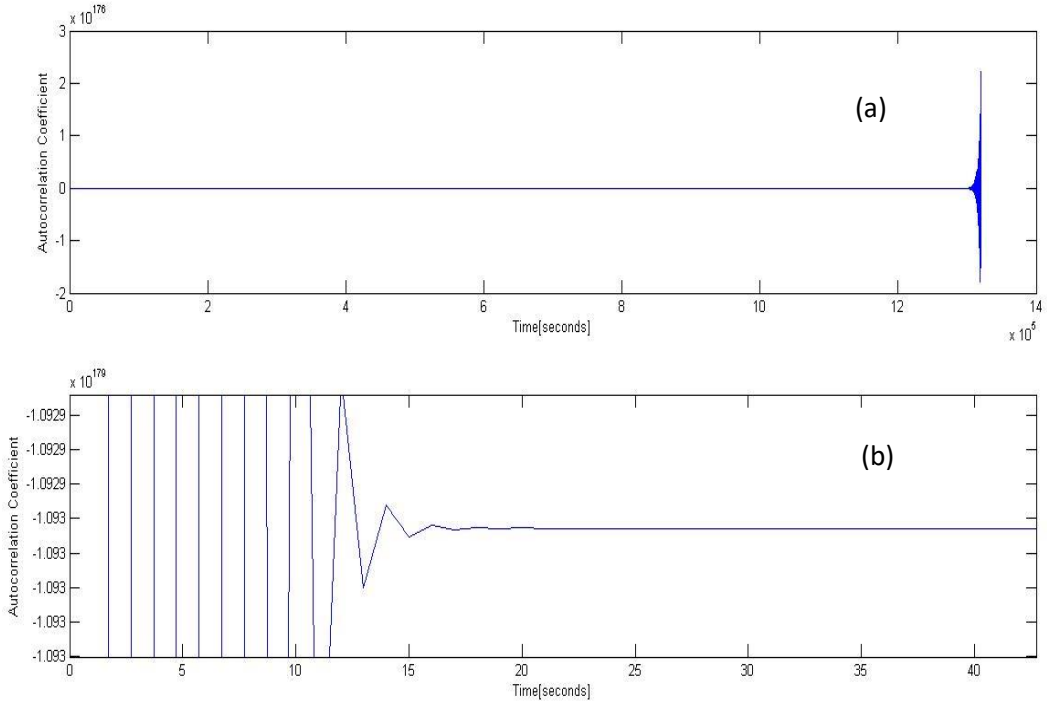

Figure 5.10: Plot of Uncontrolled System Output(a) vs. Controlled System Output using u_2(b) for ϕ=0.54

Figure 5.11 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.54. The input for the simulation given in Figure 5.11 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.11 b) is $\boldsymbol{u_3}$ (Equation 5.4), which corresponds to a steady input and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

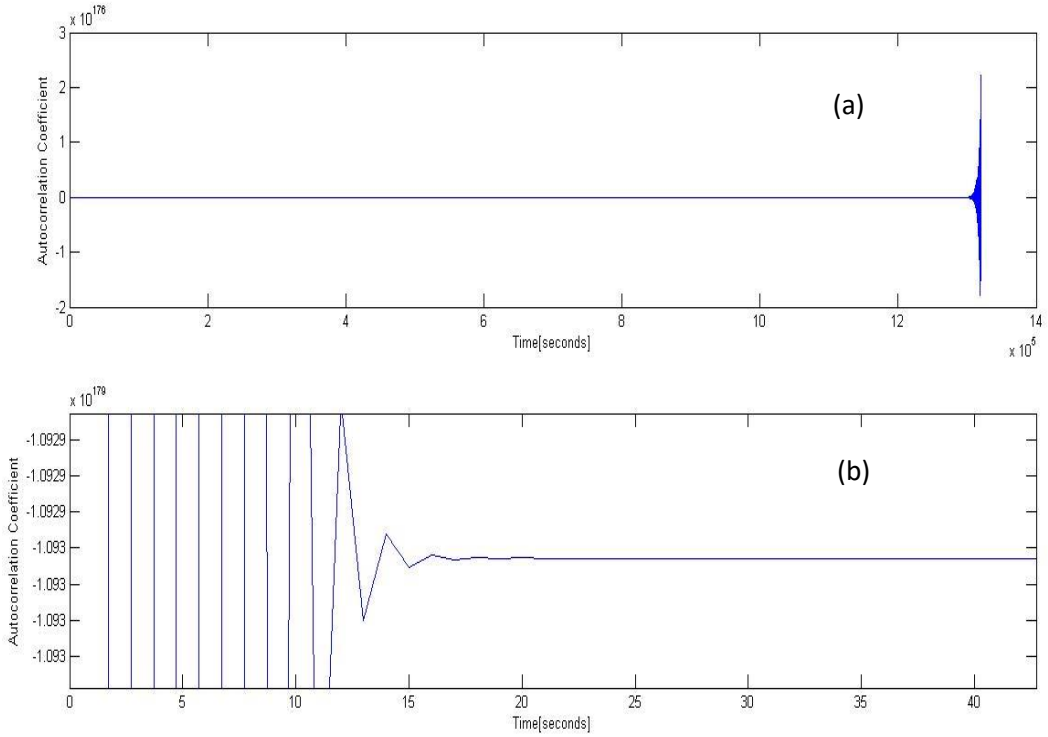

Figure 5.11: Plot of Uncontrolled System Output(a) vs. Controlled System Output using u_3(b) for ϕ=0.54

Figure 5.12 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.54. The input for the simulation given in Figure 5.12 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.12 b) is $\boldsymbol{u_4}$ (Equation 5.5), which corresponds to a steady input and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

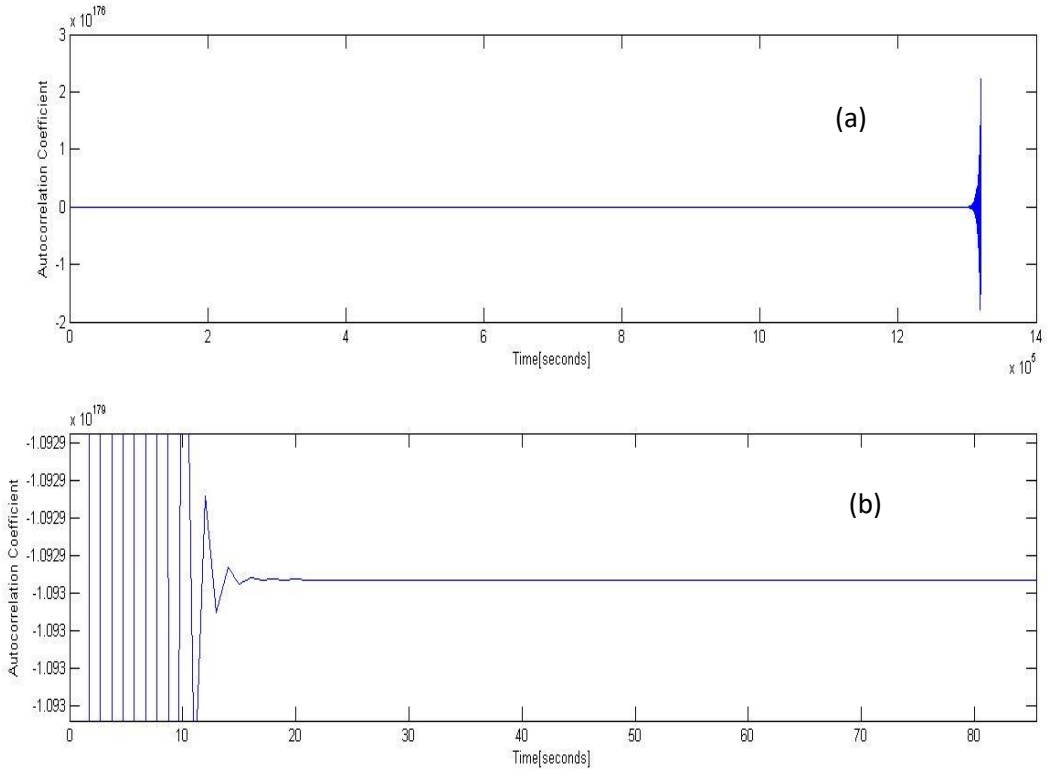

Figure 5.12: Plot of Uncontrolled System Output(a) vs. Controlled System Output using $\boldsymbol{u_4}$(b) for ϕ=0.54

Summary observation for flow coefficient 0.54

- In Figure 5.9, the command input u_c is given by measurement, which is a pulsing function as these pulses are used to excite the system for system identification. The response of the system without the controller is unstable. The controller is able to stabilize the compressor response (as given as an autocorrelation) depicted in 5.9 b) and has a very large magnitude. The response does not match the value of the command input.
- In Figure 5.10, the command input u_c is a steady input of value -1. The response (as given as an autocorrelation) depicted in 5.10 b) stabilizes the compressor. The response matches the value of the command input, but has a very large magnitude.
- In Figure 5.11, the command input u_c is a steady value of 0. The response (as given as an autocorrelation) depicted in 5.11 b) stabilizes the compressor. The response does not match the value of the command input and has a very large magnitude.
- In Figure 5.12, the command input u_c is a steady value of 1. The response (as given as an autocorrelation) depicted in 5.12 b) stabilizes the compressor. It matches the value of the command input inversely with a very large magnitude.

5.1.4: Simulation Results for Flow Coefficient 0.52

The flow coefficient of 0.52 corresponds to a stable, approaching stall inception, operating point (refer Figure 1.11). Figure 5.13 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.52. The input for the simulation given in Figure 5.13 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.13 b) is $\boldsymbol{u_1}$ (Equation 5.2), which corresponds to the collected data (input) from the compressor and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

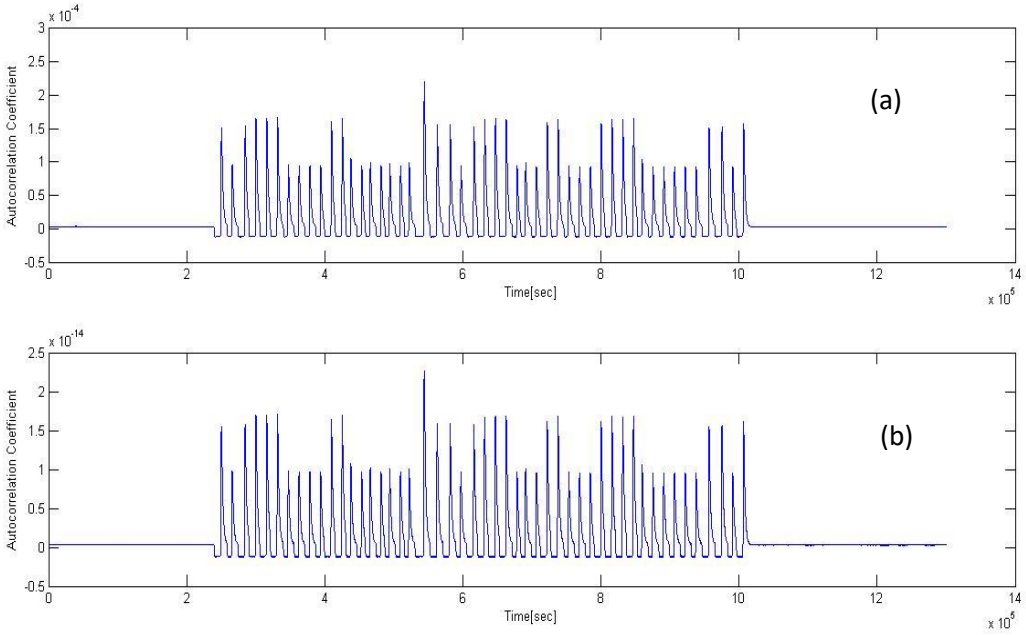

Figure 5.13: Plot of Uncontrolled System Output(a) vs. Controlled System Output using $\boldsymbol{u_1}$(b) for ϕ=0.52

Figure 5.14 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.52. The input for the simulation given in Figure 5.14 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.14 b) is $\boldsymbol{u_2}$ (Equation 5.3), which corresponds to a steady input and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

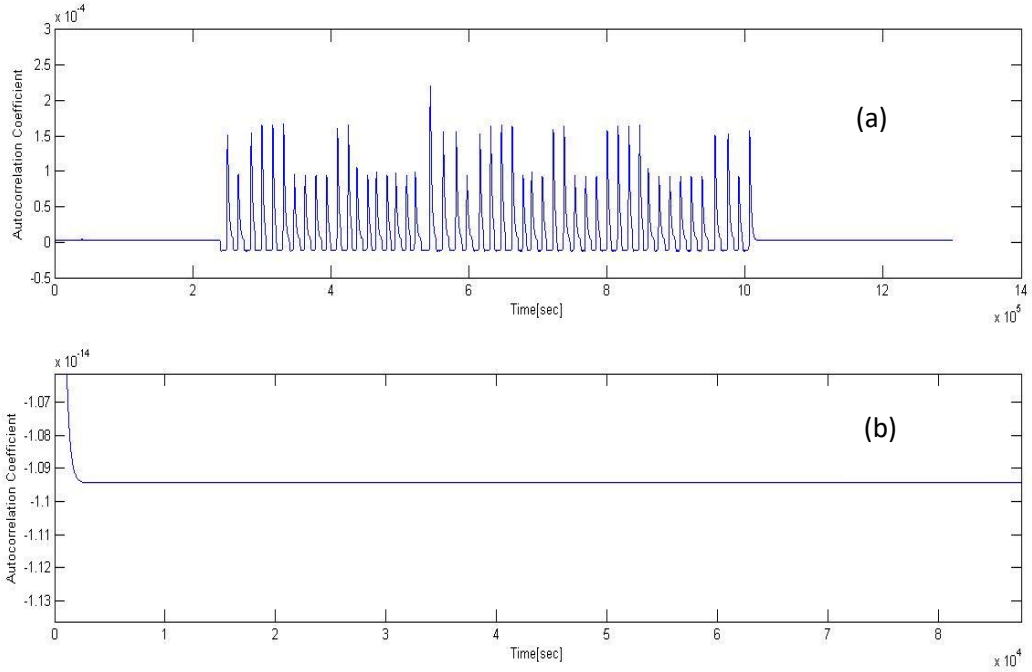

Figure 5.14: Plot of Uncontrolled System Output(a) vs. Controlled System Output using u_2(b) for ϕ=0.52

Figure 5.15 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.52. The input for the simulation given in Figure 5.15 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.15 b) is $\boldsymbol{u_3}$ (Equation 5.4), which corresponds to a steady input and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

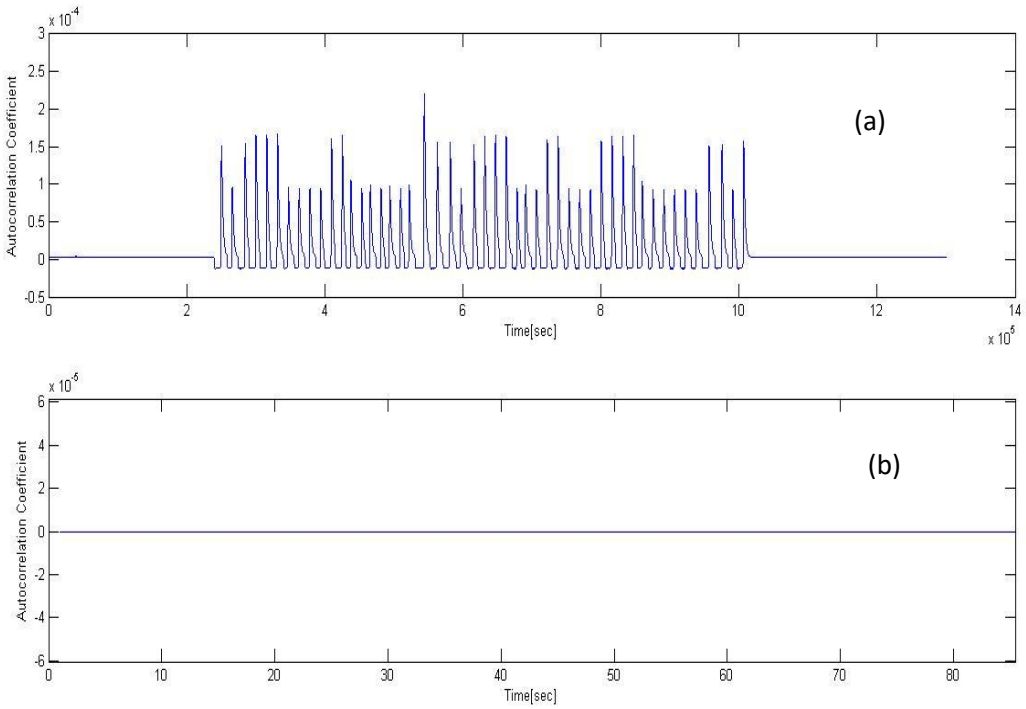

Figure 5.15: Plot of Uncontrolled System Output(a) vs. Controlled System Output using $\boldsymbol{u_3}$(b) for ϕ=0.52

Figure 5.16 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.52. The input for the simulation given in Figure 5.16 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.16 b) is $\boldsymbol{u_4}$ (Equation 5.5), which corresponds to a steady input and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

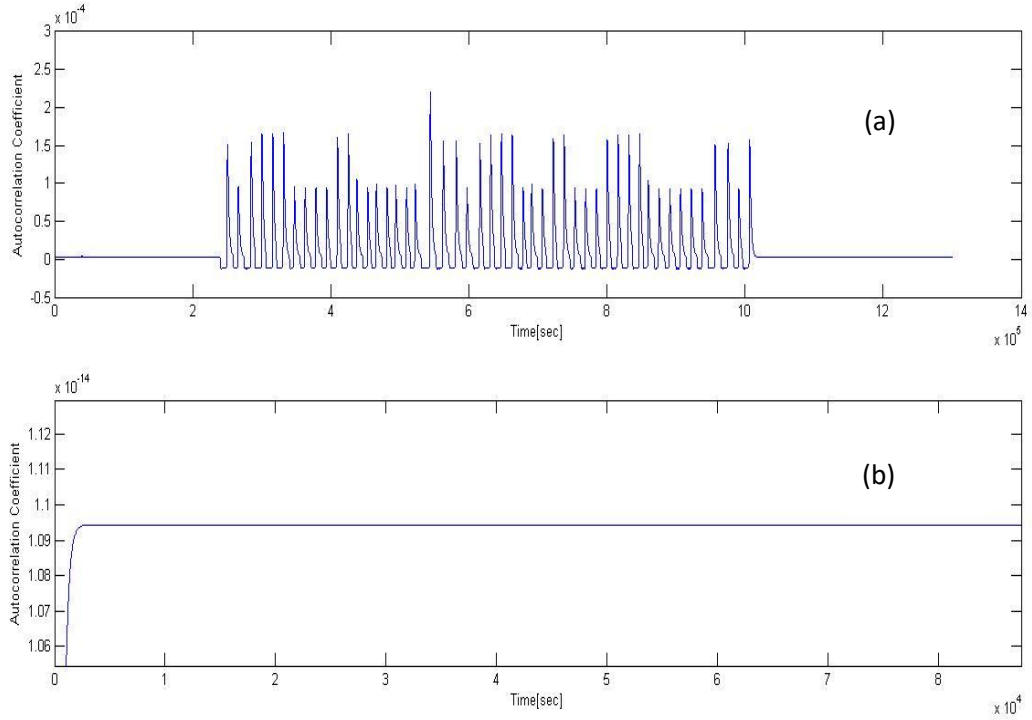

Figure 5.16: Plot of Uncontrolled System Output(a) vs. Controlled System Output using $\boldsymbol{u_4}$(b) for ϕ=0.52

Summary observation for flow coefficient 0.52

- In Figure 5.13, the command input u_c is given by measurement, which is a pulsing function as these pulses are used to excite the system for system identification. The response (as given as an autocorrelation) depicted in 5.13 b) matches the form of the input very well. The magnitude is substantially smaller.
- In Figure 5.14, the command input u_c is a steady input of value -1. The response (as given as an autocorrelation) depicted in 5.14 b) stabilizes the compressor and matches the value of the command input. The magnitude is substantially smaller.
- In Figure 5.15, the command input u_c is a steady value of 0. The response (as given as an autocorrelation) depicted in 5.15 b) stabilizes the compressor and matches the value of the command input.
- In Figure 5.16, the command input u_c is a steady value of 1. The response (as given as an autocorrelation) depicted in 5.16 b) stabilizes the compressor and matches the value of the command input. The magnitude is substantially smaller.

5.1.5 Simulation Results for Flow Coefficient 0.51

The flow coefficient of 0.51 corresponds to a stable, point of stall inception, operating point (refer Figure 1.11). Figure 5.17 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.51. The input for the simulation given in Figure 5.17 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.17 b) is $\boldsymbol{u_1}$ (Equation 5.2), which corresponds to the collected data (input) from the compressor and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

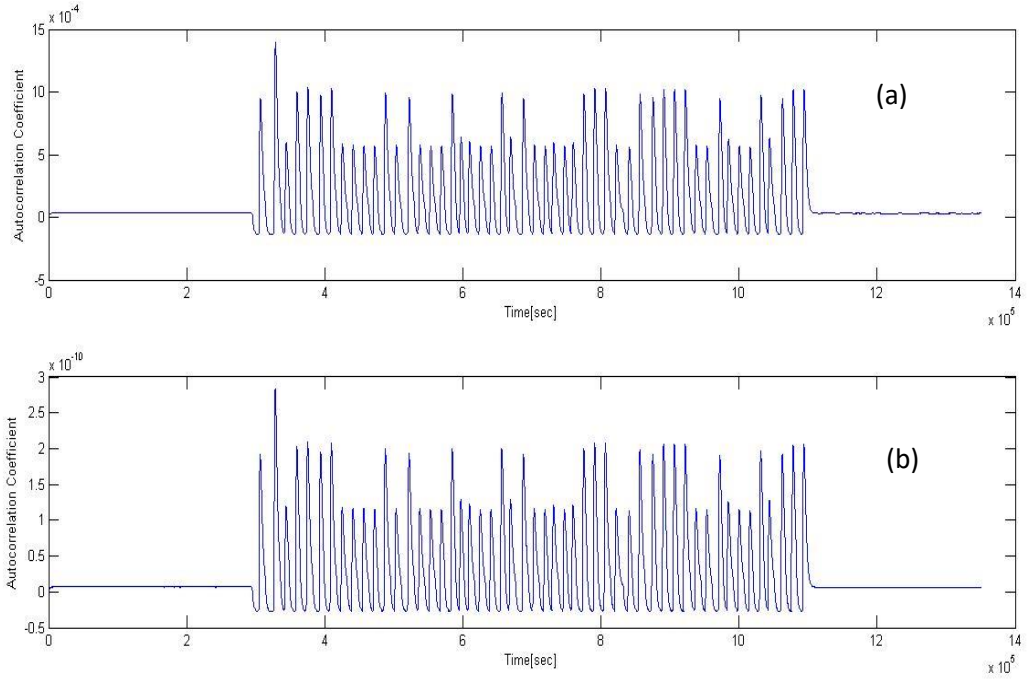

Figure 5.17: Plot of Uncontrolled System Output(a) vs. Controlled System Output using u_1(b) for ϕ=0.51

Figure 5.18 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.51. The input for the simulation given in Figure 5.18 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.18 b) is $\boldsymbol{u_2}$ (Equation 5.3), which corresponds to a steady input and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

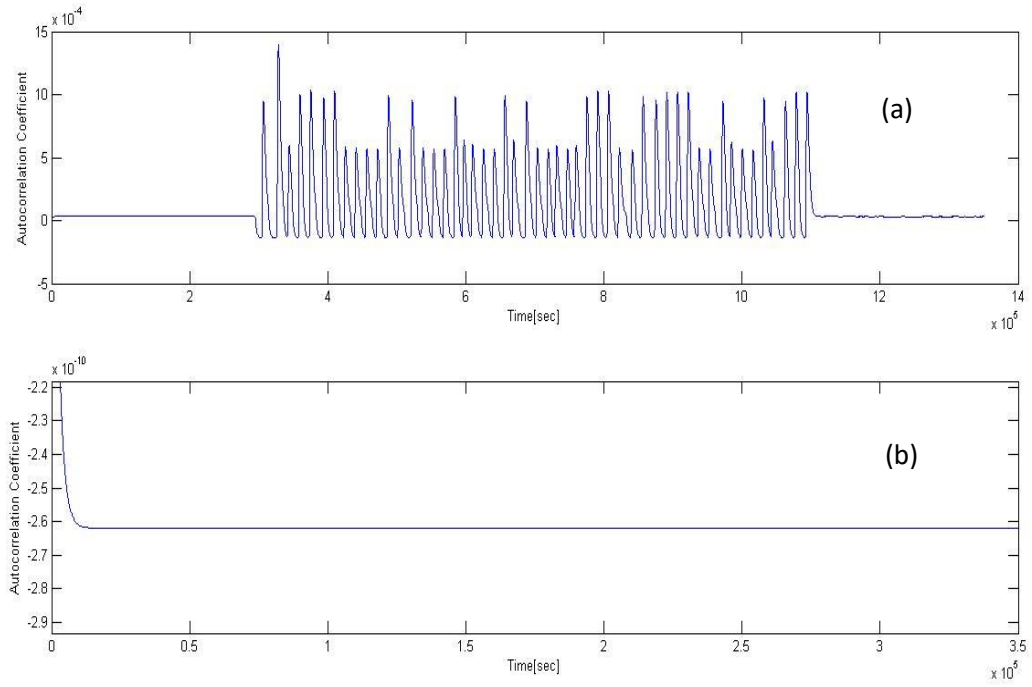

Figure 5.18: Plot of Uncontrolled System Output(a) vs. Controlled System Output using u_2(b) for ϕ=0.51

Figure 5.19 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.51. The input for the simulation given in Figure 5.19 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.19 b) is $\boldsymbol{u_3}$ (Equation 5.4), which corresponds to a steady input and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

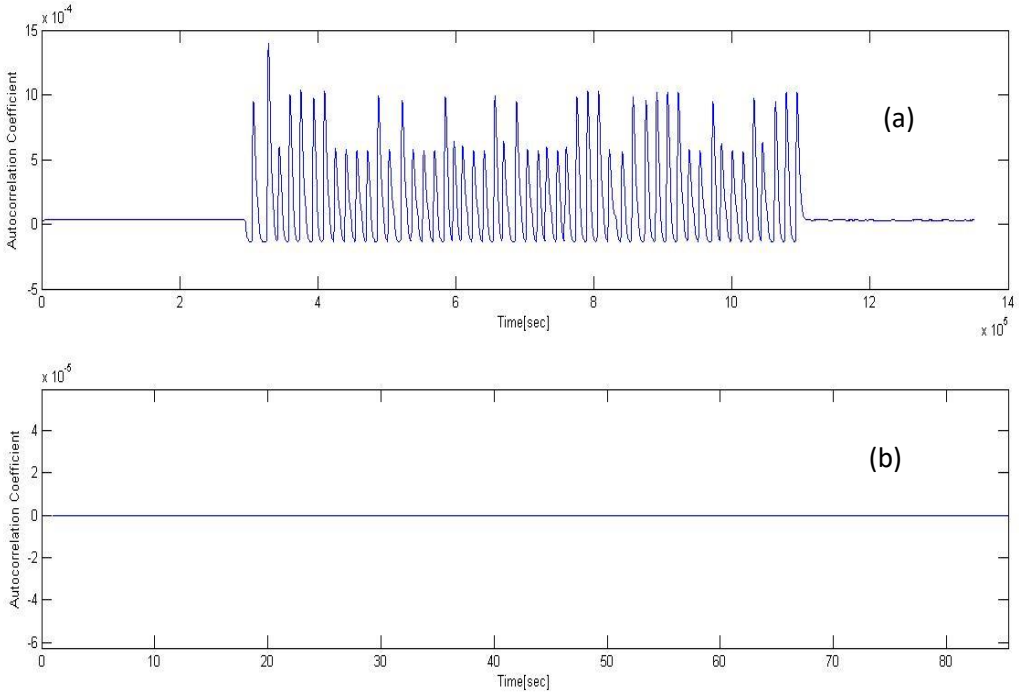

Figure 5.19: Plot of Uncontrolled System Output(a) vs. Controlled System Output using $\boldsymbol{u_3}$(b) for ϕ=0.51

Figure 5.20 shows the simulation results of the output of the developed model vs. the output of the model along with the controller – for a flow coefficient of 0.51. The input for the simulation given in Figure 5.20 a) is the collected pressure sensor data for the given flow coefficient. The input for the simulation results depicted in Figure 5.20 b) is $\boldsymbol{u_4}$ (Equation 5.5), which corresponds to a steady input and the proposed LQG controller. The output is the Autocorrelation coefficient designated as the y-axis. The x-axis is the time in seconds.

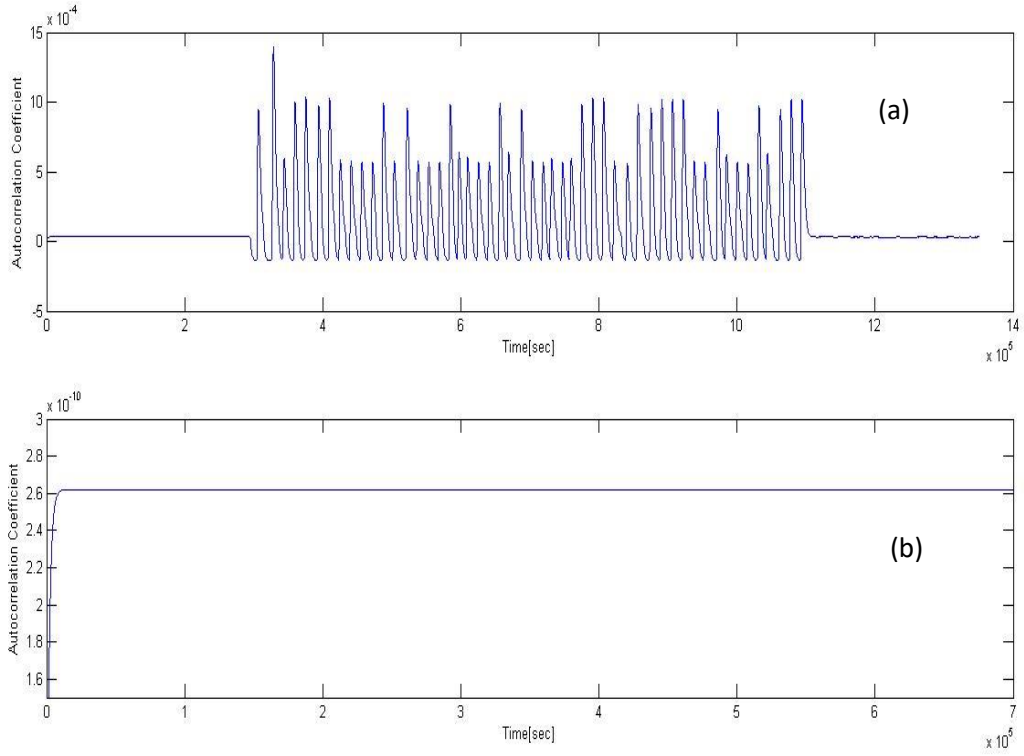

Figure 5.20: Plot of Uncontrolled System Output(a) vs. Controlled System Output using $\boldsymbol{u_4}$(b) for ϕ=0.51

Summary observation for flow coefficient 0.51

- In Figure 5.17, the command input u_c is given by measurement, which is a pulsing function as these pulses are used to excite the system for system identification. The response (as given as an autocorrelation) depicted in 5.17 b) matches the form of the input well. The magnitude is substantially smaller.
- In Figure 5.18, the command input u_c is a steady input of value -1. The response (as given as an autocorrelation) depicted in 5.18 b) stabilizes the compressor and roughly matches the value of the command input. The magnitude is substantially smaller.
- In Figure 5.19, the command input u_c is a steady value of 0. The response (as given as an autocorrelation) depicted in 5.19 b) stabilizes the compressor and matches value of the command input.
- In Figure 5.20, the command input u_c is a steady value of 1. The response (as given as an autocorrelation) depicted in 5.20 b) stabilizes the compressor and roughly matches the value of the command input. The magnitude is substantially smaller.

5.2: Discussion

The system identification results of the fast dynamics showed an interesting aspect of the compressor behavior. The identified transfer function and state space models which is a combination of the developed models at each flow coefficient agreed with the observations made at the actual system. The primary objective of the proposed LQG controller is to stabilize the compressor just at/beyond instability. General observation from the simulation results presented in this chapter is that the controller matches the command input in form and sign, but lacks the accuracy in the gain value. The change in gain of the responses for different command inputs is summarized in the Table 5.1.

Table 5.1 Gain Change of the Responses for Different Command Inputs

Flow Coefficient	Value of Gain Change with Input u_1	Value of Gain Change with Input u_2	Value of Gain Change with Input u_3
0.58	10^{-15}	10^{-5}	10^{-15}
0.56	10^{-14}	10^{-5}	10^{-14}
0.54	10^{179}	10^{179}	10^{179}
0.52	10^{-14}	10^{-5}	10^{-14}
0.51	10^{-10}	10^{-10}	10^{-10}

From Table 5.1, we can summarize that the proposed controller stabilizes the compressor, but lacks the accuracy in the gain value. To overcome this, a gain element can be added to the system to achieve the desired response. The gain element could also be computed by fitting a polynomial or spline curve through the data extracted and listed in Table 5.1.

CHAPTER 6: CONCLUSIONS AND FUTURE WORK

6.1: Conclusion

The fast dynamics analysis of the experimental one-stage axial compressor showed promising results. The system identification uses air injection pressure signal as the input and autocorrelation coefficient of the dynamic pressure data across the rotor blades as the output. This type of actuation has shown to be effective in extending the operating point far beyond the stall point. However, the energy consumed by such actuation is substantial and negates the efficiency improvement. This research resulted in a proposed LQG controller to address the control issues by providing a framework to design such controllers using air injection. The identified models for each flow coefficient are combined into an overall model—This model is a function of the flow coefficient. The pole behavior of the identified individual transfer function models as well as the state-space models appeared to have roughly capture the compressor natural frequency and exhibited observed behavior as the stall point is approached. For example, the sensor pressure at midspan of a blade passage at a flow coefficient of 0.54, indicates instability causing a spike in the compressor output. This observation is captured by the proposed overall model. In addition, the behavior far away and close to the stall point of the model agree with observations made at the actual system.

A Linear quadratic Gaussian controller is proposed to be designed in order to control the instability and to drive the compressor at or just beyond instability. Simulation results show that the controller is able to stabilize the compressor at a flow coefficient of 0.54—the region of instability. Testing of the developed model and proposed control strategy includes simulations for steady input, pulsed input and investigation of accuracy on the output with regard to

command following and gain change. Simulations results presented in Chapter 5 showed that the proposed controller is able to stabilize the compressor but lacks the accuracy in the gain value. A gain element can be added to the proposed system to achieve the desired response.

6.2 Future Work

Future work for this project may include driving the compressor at different flow coefficients and investigate the compressor behavior for stability utilizing the developed controller. Additionally, an implementation scheme to apply the controller to the real system may be developed. The new results of the compressor fast dynamics could be used for system identification to further validate and/or modify the model. In addition, using the proposed control scheme, the LQG controller can be used to minimize the energy loss due to control action, while allowing the system to operate beyond the stall point.

An alternate method of designing a controller could also be developed to compare the results of this research. Models developed from an alternate system identification method may be developed to compare the system identification of this research. Overall, the modelling and control of the fast dynamics of an axial compressor is an emerging area of study with many new opportunities of dynamic model development and applying different control strategies.

References

[1] Meherwan P. Boyce, “Axial flow compressors”

[2] Dane Sterbentz, “System Identification and modelling of the dynamics of a one stage axial flow compressor system”

[3] Epstein, A.H., Greitzer, E.M., Paduano,J.D., 2001, “Compression System stability and active control”, Annual review of fluid mechanics, 33(1), pp.491-517

[4] Rolls-royce

[5] F. Willems, B. D. Jager, “Modelling and control of rotating stall and surge: An overview”, International conference on control application, IEEE 1998.

[6] I.J. Day, “Stall inception in axial flow compressors.” J. Turbomachinery, vol,115, p1-9 Jan 1993

[7] H.O. Wang, R.A. Adomaitis, E.H. Abed, “Nonlinear Analysis and Control of Rotating Stall in Axial Flow Compressors”.

[8] Bram de Jager. Rotating stall and Surge control: A survey in proc proc of the 34th IEEE conference on decision and control Volume 2, New Orleans L A Dec 1995.

[9] Guoxiang Gu, Andrew Sparks, S.S. Banda, “An overview of Rotating Stall and Surge Control for Axial Flow Compressors.” IEEE transactions on control systems technology, Vol 7, No.6, Nov 1999.

[10] Moore, F.K., E.M. Greitzer., “A theory of Post-Stall Transients in Axial Compression Systems: Part I- Development of equations, “Journal of Engineering for Gas Turbines and Power, 108, pp. 68-76.

[11] A.M. Wo, J.P. Bons. “ Flow Physics Leading to System Instability in a Centrifugal Pump” International Gas Turbine and Aeroengine Congress and Exposition, Cincinnati, Ohio May 1993.

[12] Chi, J.N., Paduano, J.D., “New Concepts for Active Control of Rotating Stall and Surge,” American Control Conference Seattle, WA, pp. 2435-2442, 2008.

[13] Emmons. H.W., Pearson, C.E., H.P. Grant., 1955, “Compressor Surge and Stall Propagation,” Trans. ASME,79, pp. 455-469.

[14] Du, W., and Leonard, O., 2012, “Numerical Simulation of Surge in Axial Compressor,” International Journal of Rotating Machinery, pp. 1-10.

[15] A.M. Cargill and C. Freeman, “High Speed Compressor Surge with Application to Active Control” Gas Turbine and Aeroengine Congress and Exposition, Brussels, Belgium, June 1990.

[16] I.J. Day “Stall Inception in Axial Flow Compressors”, International Gas Turbine and Aeroengine Congress and Exposition, Orlando, FL June 1991.

[17] K.K Botros and J.F. Henderson, “Developments in Centrifugal Compressor Surge Control- A Technology Assessment” International Gas Turbine and Aeroengine Congress and Exposition, Cincinnati, Ohio, May 1993

[18] A. Cortinovis, D. Pareschi, M. Mercangoez, T. Besselmann, “Model Predictive Anti-Surge Control of Centrifugal Compressors with Variable-Speed Drives” Proc of the 2012 IFAC workshop on Automatic Control in Offshore Oil and Gas Production, Norway, May 2012

[19] Edmond Preti, Howard W. Ripy, “Surge Detectors for Gas Turbine Engines”, United Technologies Corporation, US Patent 4137710.

[20] Greitzer, E.M, 1976, “Surge and Rotating Stall in Axial Flow Compressors, Part I: Theoretical Compression System Model,” ASME J. Eng. For Power, 98(2), pp. 190-198

[21] Greitzer, E.M, 1976, “Surge and Rotating Stall in Axial Flow Compressors, Part II: Experimental Results and Comparison with Theory,” ASME J. Eng. For Power, 98(2), pp. 199-217

[22] A.H. Epstein,J.E Williams, E.M. Greitzer, “Active Supression of Aerodynamic Instabilities in Turbomachines”. J. Propulsion,5(2):204-221. March 1989

[23] J.S. Simon and L. Valavani, “A Lyapunov Based Nonlinear Control Scheme for Stabilizing a Basic Compression System Using a Close-Coupled Control Valve” Gas Turbine Laboratory, MIT, Cambridge.

[24] C.A. Mansoux, D.L Gysling, J.D. Setiawan, J.D. Paduano, “Distributed Nonlinear Modeling and Stability Analysis of Axial Compressor Stall and Surge” American Control Conference, Baltimore, June 1994

[25] R. L. Behnken, R. D. Andrea, R.M Murray, “Control of Rotating Stall in a Low-Speed Axial Flow Compressor Using Pulsed Air Injection: Modeling, Simulations and

Experimental Validation", Proc of the 34th Conference on Decision and Control, New Orleans, LA, Dec 1995.

[26] Krauss- Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=45608217

[27] Bunge, Mario; "A general black-box theory", Philosophy of science, Vol. 30, No.4, 1963, pp. 346-358.

[28] Kang, Wei and Gu, Guoxiang and Sparks, Andy and Banda, Siva, "Bifurcation test functions and surge control for axial flow compressors" Automatica, Vol-35, No.2, 1999, pp. 229-239

[29] Weigl, HJ and Paduano, JD and Frechette, LG and Epstein, AH and Greitzer, EM and Bright, MM and Strazisar, AJ, "Active Stabilization of Rotating Stall and Surge in a Transonic Single-Stage Compressor" Transactions of the ASME-T-Journal of Turbomachinery, Vol-120, No.4, 1998, pp.625-636

[30] Camp, TR and Day, IJ, "A study of spike and Modal Stall Phenomena in a Low-speed Axial Compressor" Journal of Turbomachinery, Vol-120, No.3, 1998, pp. 393-401

[31] Vo, H.D., 2001, "Role of Tip Clearance Flow on Axial Compressor Stability", "Doctor of Philosophy, Massachusetts Institute of Technology.

[32] J.Li, F. Lin, Z. Tong, C. Nie, J. Chen, "The Dual Mechanisms and Implementations of Stability Enhancement with Discrete Tip Injection in Axial Flow Compressors" Tran. Of ASME, Journal of Turbomachinery, Vol. 137, March 2015.

[33] https://www.mathworks.com/help/control/ref/tf.html

[34] https://www.mathworks.com/help/control/ref/ss.html?searchHighlight=ss&s_tid=doc_srchtitle

[35] Keesman, K. J., 2011, System Identification: An Introduction, Springer, New York.

[36] Paul, M. J., Van, den, Hof., 2012, System Identification-Data-Driven Modelling of Dynamic Systems, Lecture Notes, Department of Electrical Engineering, Eindhoven University of Technology

[37] Lee, Chang. H.,Hsiao, Hung. M., Huang, Kuang.J., Identification of Stochastic System And Controller Via Projection Filters, Proc. of the American Control Conference, Baltimore, Maryland, June 1994

[38] Kalbat, Abdulrahman., Linear Quadratic Gaussian(LQG) control of wind turbines, Electrical Engineering Department, Columbia University, Newyork

Appendix

1 Four Poles and Two Zeroes Transfer Function Model

```
% zero1 (real) is denoted as zr1
% zero1 (imaginary) is denoted as zimg1
% zero2 (real) is denoted as zr2
% zero2 (imaginary) is denoted as zimg2
% pole1 (real) is denoted as pr1
% pole1 (imaginary) is denoted as pimg1
% pole2 (real) is denoted as pr2
% pole2 (imaginary) is denoted as pimg2
% pole3 (real) is denoted by pr3
% pole3 (imaginary) is denoted by pimg3
% pole4 (real) is denoted by pr4
% pole4 (imaginary) is denoted by pimg4

zr1 = (-2899422.619016*(x/100).^4+6307669.345169*(x/100).^3-
5141511.130896*(x/100).^2+1861068.807241*(x/100)-252404.680537);
zimg1 = (9293291.666531*(x/100).^4-
20568447.916370*(x/100).^3+17043242.208091*(x/100).^2-
6266529.963246*(x/100)+862724.296428);

zr2 = (-2899422.619016*(x/100).^4+6307669.345169*(x/100).^3-
5141511.130896*(x/100).^2+1861068.807241*(x/100)-252404.680537);
zimg2 = (-9293291.666531*(x/100).^4+20568447.916370*(x/100).^3-
17043242.208091*(x/100).^2+6266529.963246*(x/100)-
862724.296428);

pr1 = (103762720.236748*(x/100).^4-
227724486.604206*(x/100).^3+187198784.759506*(x/100).^2-
68313656.279487*(x/100)+9337614.784242);

pimg1 = (-11826369.047535*(x/100).^4+24852749.404579*(x/100).^3-
19598746.702232*(x/100).^2+6876690.085184*(x/100)-
906142.350393);

pr2 = (159857482.140667*(x/100).^4-
351132962.792845*(x/100).^3+288954682.853244*(x/100).^2-
105583016.088467*(x/100)+14453672.473768);

pimg2 = (11826369.047535*(x/100).^4-
24852749.404579*(x/100).^3+19598746.702232*(x/100).^2-
6876690.085184*(x/100)+906142.350393);
```

```
pr3 = (-138773.809522*(x/100).^4+302023.214282*(x/100).^3+-
246451.690473*(x/100).^2+89365.090713*(x/100)-12149.604720);

pr4 = (-138773.809522*(x/100).^4+302023.214282*(x/100).^3+-
246451.690473*(x/100).^2+89365.090713*(x/100)-12149.604720);
pimg3 = (9807964.285584*(x/100).^4-
21615063.094954*(x/100).^3+17836294.089053*(x/100).^2-
6531685.792177*(x/100)+895696.232429);

pimg4 = (-9807964.285584*(x/100).^4+21615063.094954*(x/100).^3-
17836294.089053*(x/100).^2+6531685.792177*(x/100)-
895696.232429);

b0=1;

b1=((-2*zr1)-1i*(zimg2+zimg1));

b2=((zr1.^2)-(zimg1*zimg2)+1i*zr1*(zimg1+zimg2));

a1=((-pr4-pr2-pr1)-1i*(pimg4+pimg3+pimg2+pimg1));

a2=((-pr3+(pr3*pr4)-(pimg3*pimg4)+(pr4*pr2)+(pr2*pr3)-
(pimg2*pimg4)-(pimg3*pimg2)+(pr1*pr4)+(pr1*pr3)+(pr1*pr2)-
(pimg1*pimg4)-(pimg1*pimg3)-
(pimg1*pimg2))+1i*((pr3*pimg4)+(pimg3*pr4)+(pr2*pimg4)+(pr2*pimg
3)+(pr4*pimg2)+(pr3*pimg2)+(pr1*pimg4)+(pr1*pimg3)+(pr1*pimg2)+(
pimg1*pr4)+(pr3*pimg1)+(pr2*pimg1)));

a3=(((pimg3*pimg4*pr2)-
(pr2*pr3*pr4)+(pr3*pimg4*pimg2)+(pimg3*pimg2*pr4)-
(pr1*pr3*pr4)+(pr1*pimg3*pimg4)-(pr1*pr2*pr4)-
(pr3*pr1*pr2)+(pr1*pimg2*pimg4)+(pr1*pimg2*pimg3)+(pimg1*pr3*pim
g4)+(pimg3*pr4*pimg1)+(pr2*pimg1*pimg4)+(pr2*pimg1*pimg3)+(pimg1
*pimg2*pr4)+(pimg1*pimg2*pr3))+1i*((pimg3*pimg4*pimg2)-
(pr2*pr3*pimg4)-(pr2*pimg3*pr4)-(pr3*pimg2*pr4)-(pr1*pr3*pimg4)-
(pr1*pimg3*pr4)-(pr1*pr2*pimg4)-(pr1*pr2*pimg3)-(pr1*pimg2*pr4)-
(pr1*pimg2*pr3)-(pimg1*pr3*pr4)+(pimg3*pimg4*pimg1)-
(pr2*pimg1*pr4)-
(pr2*pimg1*pr3)+(pimg1*pimg2*pimg4)+(pimg1*pimg2*pimg3)));

a4=(((pr3*pr4*pr1*pr2)-(pr1*pr2*pimg3*pimg4)-
(pr1*pr3*pimg2*pimg4)-(pr1*pimg2*pimg3*pr4)-
(pr2*pimg1*pr3*pimg4)-(pr2*pimg1*pimg3*pr4)-
(pimg1*pimg2*pr3*pr4)+(pimg1*pimg2*pimg3*pimg4))+1i*((pr1*pr2*pr
3*pimg4)+(pr1*pr2*pimg3*pr4)+(pr1*pimg2*pr3*pr4)-
(pimg3*pimg4*pr1*pimg2)+(pr2*pimg1*pr3*pr4)-
```

```
(pimg3*pimg4*pr2*pimg1)-(pimg1*pimg2*pr3*pimg4)-
(pimg1*pimg2*pimg3*pr4)));

num = [b0 b1 b2];
den = [1 a1 a2 a3 a4];
sys = tf(num,den)
```

2. Four Poles and two Zeroes State-Space Model

```
% zero1 (real) is denoted as zr1
% zero1 (imaginary) is denoted as zimg1
% zero2 (real) is denoted as zr2
% zero2 (imaginary) is denoted as zimg2
% pole1 (real) is denoted as pr1
% pole1 (imaginary) is denoted as pimg1
% pole2 (real) is denoted as pr2
% pole2 (imaginary) is denoted as pimg2
zr1 = (-2899422.619016*(x/100).^4+6307669.345169*(x/100).^3-
5141511.130896*(x/100).^2+1861068.807241*(x/100)-252404.680537);

zimg1 = (9293291.666531*(x/100).^4-
20568447.916370*(x/100).^3+17043242.208091*(x/100).^2-
6266529.963246*(x/100)+862724.296428);

zr2 = (-2899422.619016*(x/100).^4+6307669.345169*(x/100).^3-
5141511.130896*(x/100).^2+1861068.807241*(x/100)-252404.680537);

zimg2 = (-9293291.666531*(x/100).^4+20568447.916370*(x/100).^3-
17043242.208091*(x/100).^2+6266529.963246*(x/100)-
862724.296428);

pr1 = (103762720.236748*(x/100).^4-
227724486.604206*(x/100).^3+187198784.759506*(x/100).^2-
68313656.279487*(x/100)+9337614.784242);

pimg1 = (-11826369.047535*(x/100).^4+24852749.404579*(x/100).^3-
19598746.702232*(x/100).^2+6876690.085184*(x/100)-
906142.350393);

pr2 = (159857482.140667*(x/100).^4-
351132962.792845*(x/100).^3+288954682.853244*(x/100).^2-
105583016.088467*(x/100)+14453672.473768);

pimg2 = (11826369.047535*(x/100).^4-
24852749.404579*(x/100).^3+19598746.702232*(x/100).^2-
6876690.085184*(x/100)+906142.350393);

pr3 = (-138773.809522*(x/100).^4+302023.214282*(x/100).^3+-
246451.690473*(x/100).^2+89365.090713*(x/100)-12149.604720);
pr4 = (-138773.809522*(x/100).^4+302023.214282*(x/100).^3+-
246451.690473*(x/100).^2+89365.090713*(x/100)-12149.604720);
```

```
pimg3 = (9807964.285584*(x/100).^4-
21615063.094954*(x/100).^3+17836294.089053*(x/100).^2-
6531685.792177*(x/100)+895696.232429);

pimg4 = (-9807964.285584*(x/100).^4+21615063.094954*(x/100).^3-
17836294.089053*(x/100).^2+6531685.792177*(x/100)-
895696.232429);

b0=1;

b1=((-2*zr1)-1i*(zimg2+zimg1));

b2=((zr1.^2)-(zimg1*zimg2)+1i*zr1*(zimg1+zimg2));

a1=((-pr2-pr1)-1i*(pimg1+pimg2));

a2=((pr1*pr2)-(pimg1*pimg2)+1i*((pr1*pimg2)+(pimg1*pr2)));

a3=(((pimg3*pimg4*pr2)-
(pr2*pr3*pr4)+(pr3*pimg4*pimg2)+(pimg3*pimg2*pr4)-
(pr1*pr3*pr4)+(pr1*pimg3*pimg4)-(pr1*pr2*pr4)-
(pr3*pr1*pr2)+(pr1*pimg2*pimg4)+(pr1*pimg2*pimg3)+(pimg1*pr3*pim
g4)+(pimg3*pr4*pimg1)+(pr2*pimg1*pimg4)+(pr2*pimg1*pimg3)+(pimg1
*pimg2*pr4)+(pimg1*pimg2*pr3))+1i*((pimg3*pimg4*pimg2)-
(pr2*pr3*pimg4)-(pr2*pimg3*pr4)-(pr3*pimg2*pr4)-(pr1*pr3*pimg4)-
(pr1*pimg3*pr4)-(pr1*pr2*pimg4)-(pr1*pr2*pimg3)-(pr1*pimg2*pr4)-
(pr1*pimg2*pr3)-(pimg1*pr3*pr4)+(pimg3*pimg4*pimg1)-
(pr2*pimg1*pr4)-
(pr2*pimg1*pr3)+(pimg1*pimg2*pimg4)+(pimg1*pimg2*pimg3)));

a4=(((pr3*pr4*pr1*pr2)-(pr1*pr2*pimg3*pimg4)-
(pr1*pr3*pimg2*pimg4)-(pr1*pimg2*pimg3*pr4)-
(pr2*pimg1*pr3*pimg4)-(pr2*pimg1*pimg3*pr4)-
(pimg1*pimg2*pr3*pr4)+(pimg1*pimg2*pimg3*pimg4))+1i*((pr1*pr2*pr
3*pimg4)+(pr1*pr2*pimg3*pr4)+(pr1*pimg2*pr3*pr4)-
(pimg3*pimg4*pr1*pimg2)+(pr2*pimg1*pr3*pr4)-
(pimg3*pimg4*pr2*pimg1)-(pimg1*pimg2*pr3*pimg4)-
(pimg1*pimg2*pimg3*pr4)));

A=[0 1 0 0;0 0 1 0;0 0 0 1;-a4 -a3 -a2 -a1];
B=[0;0;0;1];
C=[b2 b1 1 0];
D=0;
sys=ss(A,B,C,D)
```

3. Two Poles and Two Zeroes Transfer Function Model

```
% zero1 (real) is denoted as zr1
% zero1 (imaginary) is denoted as zimg1
% zero2 (real) is denoted as zr2
% zero2 (imaginary) is denoted as zimg2
% pole1 (real) is denoted as pr1
% pole1 (imaginary) is denoted as pimg1
% pole2 (real) is denoted as pr2
% pole2 (imaginary) is denoted as pimg2
zr1 = (-2899422.619016*(x/100).^4+6307669.345169*(x/100).^3-
5141511.130896*(x/100).^2+1861068.807241*(x/100)-252404.680537);

zimg1 = (9293291.666531*(x/100).^4-
20568447.916370*(x/100).^3+17043242.208091*(x/100).^2-
6266529.963246*(x/100)+862724.296428);

zr2 = (-2899422.619016*(x/100).^4+6307669.345169*(x/100).^3-
5141511.130896*(x/100).^2+1861068.807241*(x/100)-252404.680537);

zimg2 = (-9293291.666531*(x/100).^4+20568447.916370*(x/100).^3-
17043242.208091*(x/100).^2+6266529.963246*(x/100)-
862724.296428);

pr1 = (103762720.236748*(x/100).^4-
227724486.604206*(x/100).^3+187198784.759506*(x/100).^2-
68313656.279487*(x/100)+9337614.784242);

pimg1 = (-11826369.047535*(x/100).^4+24852749.404579*(x/100).^3-
19598746.702232*(x/100).^2+6876690.085184*(x/100)-
906142.350393);

pr2 = (159857482.140667*(x/100).^4-
351132962.792845*(x/100).^3+288954682.853244*(x/100).^2-
105583016.088467*(x/100)+14453672.473768);

pimg2 = (11826369.047535*(x/100).^4-
24852749.404579*(x/100).^3+19598746.702232*(x/100).^2-
6876690.085184*(x/100)+906142.350393);

b0=1;
b1=((-2*zr1)-1i*(zimg2+zimg1));
b2=((zr1.^2)-(zimg1*zimg2)+1i*zr1*(zimg1+zimg2));
a1=((-pr2-pr1)-1i*(pimg1+pimg2));
a2=((pr1*pr2)-(pimg1*pimg2)+1i*((pr1*pimg2)+(pimg1*pr2)));
num = [b0 b1 b2];
```

```
den = [1 a1 a2];
sys = tf(num,den)
```

4. Two Poles and Two Zeroes State Space Model

```
% zero1 (real) is denoted as zr1
% zero1 (imaginary) is denoted as zimg1
% zero2 (real) is denoted as zr2
% zero2 (imaginary) is denoted as zimg2
% pole1 (real) is denoted as pr1
% pole1 (imaginary) is denoted as pimg1
% pole2 (real) is denoted as pr2
% pole2 (imaginary) is denoted as pimg2

zr1 = (-2899422.619016*(x/100).^4+6307669.345169*(x/100).^3-
5141511.130896*(x/100).^2+1861068.807241*(x/100)-252404.680537);

zimg1 = (9293291.666531*(x/100).^4-
20568447.916370*(x/100).^3+17043242.208091*(x/100).^2-
6266529.963246*(x/100)+862724.296428);

zr2 = (-2899422.619016*(x/100).^4+6307669.345169*(x/100).^3-
5141511.130896*(x/100).^2+1861068.807241*(x/100)-252404.680537);

zimg2 = (-9293291.666531*(x/100).^4+20568447.916370*(x/100).^3-
17043242.208091*(x/100).^2+6266529.963246*(x/100)-
862724.296428);

pr1 = (103762720.236748*(x/100).^4-
227724486.604206*(x/100).^3+187198784.759506*(x/100).^2-
68313656.279487*(x/100)+9337614.784242);

pimg1 = (-11826369.047535*(x/100).^4+24852749.404579*(x/100).^3-
19598746.702232*(x/100).^2+6876690.085184*(x/100)-
906142.350393);

pr2 = (159857482.140667*(x/100).^4-
351132962.792845*(x/100).^3+288954682.853244*(x/100).^2-
105583016.088467*(x/100)+14453672.473768);

pimg2 = (11826369.047535*(x/100).^4-
24852749.404579*(x/100).^3+19598746.702232*(x/100).^2-
6876690.085184*(x/100)+906142.350393);

b0=1;

b1=((-2*zr1)-1i*(zimg2+zimg1));
b2=((zr1.^2)-(zimg1*zimg2)+1i*zr1*(zimg1+zimg2));

a1=((-pr2-pr1)-1i*(pimg1+pimg2));
```

```
a2=((pr1*pr2)-(pimg1*pimg2)+1i*((pr1*pimg2)+(pimg1*pr2)));

A=[0 1;;-a2 -a1];

B=[0;1];

C1=(b2-(a2*b0));

C2=(b1-(a1*b0));

C=[C1 C2];

D=1;

sys=ss(A,B,C,D)
```

5. Final Matlab Code with the Controller

```
% zero1 (real) is denoted as zr1
% zero1 (imaginary) is denoted as zimg1
% zero2 (real) is denoted as zr2
% zero2 (imaginary) is denoted as zimg2
% pole1 (real) is denoted as pr1
% pole1 (imaginary) is denoted as pimg1
% pole2 (real) is denoted as pr2
% pole2 (imaginary) is denoted as pimg2
% pole3 (real) is denoted as pr3
% pole3 (imaginary) is denoted as pimg3
% pole4 (real) is denoted as pr4
% pole4 (imaginary) is denoted as pimg4
% The equations are obtained from the "trendline function" in
excel the plots and equation of which are explained in detail in
Chapter 2
% First step before running this code is to specify the flow
coefficient*100 and name it "x" (example x=58,56,54,52,51)

zr1 = (-2899422.619016*(x/100).^4+6307669.345169*(x/100).^3-
5141511.130896*(x/100).^2+1861068.807241*(x/100)-252404.680537);
zimg1 = (9293291.666531*(x/100).^4-
20568447.916370*(x/100).^3+17043242.208091*(x/100).^2-
6266529.963246*(x/100)+862724.296428);

zr2 = (-2899422.619016*(x/100).^4+6307669.345169*(x/100).^3-
5141511.130896*(x/100).^2+1861068.807241*(x/100)-252404.680537);
zimg2 = (-9293291.666531*(x/100).^4+20568447.916370*(x/100).^3-
17043242.208091*(x/100).^2+6266529.963246*(x/100)-
862724.296428);

pr1 = (103762720.236748*(x/100).^4-
227724486.604206*(x/100).^3+187198784.759506*(x/100).^2-
68313656.279487*(x/100)+9337614.784242);

pimg1 = (-11826369.047535*(x/100).^4+24852749.404579*(x/100).^3-
19598746.702232*(x/100).^2+6876690.085184*(x/100)-
906142.350393);

pr2 = (159857482.140667*(x/100).^4-
351132962.792845*(x/100).^3+288954682.853244*(x/100).^2-
105583016.088467*(x/100)+14453672.473768);
```

```
pimg2 = (11826369.047535*(x/100).^4-
24852749.404579*(x/100).^3+19598746.702232*(x/100).^2-
6876690.085184*(x/100)+906142.350393);

pr3 = (-138773.809522*(x/100).^4+302023.214282*(x/100).^3+-
246451.690473*(x/100).^2+89365.090713*(x/100)-12149.604720);
pr4 = (-138773.809522*(x/100).^4+302023.214282*(x/100).^3+-
246451.690473*(x/100).^2+89365.090713*(x/100)-12149.604720);

pimg3 = (9807964.285584*(x/100).^4-
21615063.094954*(x/100).^3+17836294.089053*(x/100).^2-
6531685.792177*(x/100)+895696.232429);

pimg4 = (-9807964.285584*(x/100).^4+21615063.094954*(x/100).^3-
17836294.089053*(x/100).^2+6531685.792177*(x/100)-
895696.232429);

b0=1;

b1=((-2*zr1)-1i*(zimg2+zimg1));

b2=((zr1.^2)-(zimg1*zimg2)+1i*zr1*(zimg1+zimg2));

a1=((-pr2-pr1)-1i*(pimg1+pimg2));

a2=((pr1*pr2)-(pimg1*pimg2)+1i*((pr1*pimg2)+(pimg1*pr2)));

a3=(((pimg3*pimg4*pr2)-
(pr2*pr3*pr4)+(pr3*pimg4*pimg2)+(pimg3*pimg2*pr4)-
(pr1*pr3*pr4)+(pr1*pimg3*pimg4)-(pr1*pr2*pr4)-
(pr3*pr1*pr2)+(pr1*pimg2*pimg4)+(pr1*pimg2*pimg3)+(pimg1*pr3*pim
g4)+(pimg3*pr4*pimg1)+(pr2*pimg1*pimg4)+(pr2*pimg1*pimg3)+(pimg1
*pimg2*pr4)+(pimg1*pimg2*pr3))+1i*((pimg3*pimg4*pimg2)-
(pr2*pr3*pimg4)-(pr2*pimg3*pr4)-(pr3*pimg2*pr4)-(pr1*pr3*pimg4)-
(pr1*pimg3*pr4)-(pr1*pr2*pimg4)-(pr1*pr2*pimg3)-(pr1*pimg2*pr4)-
(pr1*pimg2*pr3)-(pimg1*pr3*pr4)+(pimg3*pimg4*pimg1)-
(pr2*pimg1*pr4)-
(pr2*pimg1*pr3)+(pimg1*pimg2*pimg4)+(pimg1*pimg2*pimg3)));

a4=(((pr3*pr4*pr1*pr2)-(pr1*pr2*pimg3*pimg4)-
(pr1*pr3*pimg2*pimg4)-(pr1*pimg2*pimg3*pr4)-
(pr2*pimg1*pr3*pimg4)-(pr2*pimg1*pimg3*pr4)-
(pimg1*pimg2*pr3*pr4)+(pimg1*pimg2*pimg3*pimg4))+1i*((pr1*pr2*pr
3*pimg4)+(pr1*pr2*pimg3*pr4)+(pr1*pimg2*pr3*pr4)-
(pimg3*pimg4*pr1*pimg2)+(pr2*pimg1*pr3*pr4)-
(pimg3*pimg4*pr2*pimg1)-(pimg1*pimg2*pr3*pimg4)-
(pimg1*pimg2*pimg3*pr4)));
```

```
A=[0 1 0 0;0 0 1 0;0 0 0 1;-a4 -a3 -a2 -a1];
B=[0;0;0;1];
C=[b2 b1 1 0];
D=0;
sys=ss(A,B,C,D);
% p is the transformation matrix (explained in Chapter 3)
% Ignore the complex part of "ad" and "Bd" and name them "Adn"
and "Bdn"
% "VarName10" is taken from data folder "20140708" for different
flow coefficients

p=pinv([C;C*A]);
At=pinv(p)*A*p;
Bt=pinv(p)*B;
 Ct=C*p;
 Ts=1/20000;
 [Ad,Bd]=c2d(At,Bt,Ts);
 Adn=real(Ad);
 Bdn=real(Bd);


%loop
%VarName10 is the data set from folder "20140708"
 x2=[0;0];
for i=1:1320000
    x2=Adn*x2+Bdn*VarName10(i);
    y2(i)=Ct*x2+D*VarName10(i);
end

%Controller
% kr is the feed forward gain matrix
% k is the feedback gain matrix
Q=100;
R=5;
Qw=eye(1);
Rv=eye(1);

[k,s]=dlqry(Adn,Bdn,Ct,D,Q,R);

l=dlqe(Adn,Bdn,Ct,Qw,Rv);

kr=inv(transpose(Bdn)*s*Bdn+R)*transpose(Bdn)*inv(eye(2)-
(transpose(Adn-Bdn*k)))*transpose(Ct)*Q;

[ac,bc,cc,dc]=dreg(Adn,Bdn,Ct,D,k,l);
[as1,bs1,cs1,ds1]=feedback(Adn,Bdn,Ct,D,ac,bc,cc,dc);
```

```
% controller loop

x3=[0;0];
for i=1:1320000
    u(i)=-k*x3+kr*VarName10(i);
    x3=ac*x3+bc*u(i);
    y3(i)=cc*x3+dc*u(i);
end
subplot(2,1,1)
plot(y2)
xlabel('Time[sec]')
ylabel('Autocorrelation Coefficient')
subplot(2,1,2)
plot(y3)
xlabel('Time[sec]')
ylabel('Autocorrelation Coefficient')
```

Experiment Description

The equations for poles and zeroes used for the modelling and control purposes are obtained by using the “trendline” function of excel. The steps for the same are explained in Chapter 2 of this thesis. The excel plots along with the polynomial equations are also explained in Chapter 2. The poles and zeroes are used from the transfer function model data for sensor number 5 (from Dane’s thesis), the same are presented in Table 2.2.

The data used for input (named as “VarName10” in the code) is taken from the data folder “20140708”. The explanation of the data is in the “Experiment Descriptions” section in the same folder.