ANALYSIS OF COMPUTED TORQUE CONTROL APPLIED WITH COMMAND SHAPING TO MINIMIZE RESIDUAL VIBRATION IN A FLEXIBLE-JOINT ROBOT

A Thesis

Submitted to the Faculty

of

Purdue University

by

Ruiwen Wei

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Mechanical Engineering

May 2020

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF THESIS APPROVAL

Dr. Peter H. Meckl, Chair

School of Mechanical Engineering

Dr. George T. C. Chiu

School of Mechanical Engineering

Dr. Raymond J. Cipra

School of Mechanical Engineering

Approved by:

Dr. Nicole Key

Head of the School Graduate Program

Dedicated to my parents

Jinling Luo and Tianlin Wei

and my brother

Haowen Wei

for their unsolicited support and encouragement

ACKNOWLEDGMENTS

I would like to extend my sincere gratitude to my major advisor, Prof. Peter Meckl, for all his invaluable suggestions, help and encouragement throughout my graduate study and research. I would also like to thank the other members of my advisory committee, Prof. George Chiu and Prof. Raymond Cipra for their willingness to serve in this capacity. I would also like to extend my appreciation to Zijian He for setting up and conducting the experiments for the system identification research project. I would like to thank my research partners Pratheek Patil and Aparajit Venkatesh for asking me inquisitive questions during our research meetings which helped me to gain deeper understanding of the research project. Last but certainly not least, I would like to thank Lingxiao Xie for her love and support throughout my graduate study.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

Figure Page

ABSTRACT

Wei, Ruiwen MSME, Purdue University, May 2020. Analysis of Computed Torque Control Applied with Command Shaping to Minimize Residual Vibration in a Flexible-Joint Robot. Major Professor: Peter H. Meckl, School of Mechanical Engineering.

During fast point-to-point motion, the inherent joint flexibility could be detrimental in terms of residual vibration. Aiming to minimize the vibration, the command shaping method has been developed so as to remove critical energy from the input profile at resonant frequencies. Since this method requires information of a physical model in order to find the target frequencies, the quality of the shaped command profile relies on the accuracy of the model parameter estimation. Therefore, in this work, a system identification method using Instrumental Variables is applied from the literature. Compared with the classic Ordinary Least Square method, the IV approach has successfully improved the estimation of parameters, based on simulation results. The accuracy of parameter estimation influences the command profile, as does the feedback controller. In this work, starting from a mathematical derivation with a mismatch model due to a feedback controller called Computed Torque Control, insight for the closed-loop system is given with regard to the interaction between control gains and the actual resonant frequencies. It is found that the control gain is able to modify the actual resonant frequency curve, and push it into or out of the shaping bounds which are generated from the command shaping method. Further analysis based on the simulation results shows that the overlap area between the shaping bounds and the actual frequencies affects the level of residual vibration. In light of this fact, an optimal control gain exists and is found when the estimation error is in a certain range. At the end, recommendations for choosing the control gains are provided.

1. INTRODUCTION

1.1 Motivation

As average cost of robots has declined over the years and more models appear on the market, a robotic arm has become a popular choice not only in research labs but also in traditional industry. As a type of mechanical system, a robotic arm is designed to be programmable and universal so that it could work on different kinds of jobs such as painting for automobiles, welding, assembling, and so on. The reason for being versatile is mainly because a robotic arm can increase its number of degrees of freedom by successively adding extra links onto its body. However, bringing more joints onto the robot does not necessarily improve its efficiency. On the contrary, severe end-point vibration may be observed, degrading overall performance. Among possible reasons causing this issue, joint flexibility is of interest in this study.

Joint flexibility is an inherent property of mechanical systems, and most of the time comes from gear set, belt drive, and rotational spring. Influence due to joint flexibility could be beneficial in some cases so as to protect humans from collisions with the robot. This characteristic, in many other examples, may not be desired. After operating fast point-to-point motion, a robot with joint flexibility would suffer from long-lasting vibration. As many applications require precise control of the end effector that is attached on the robotic arm, it is necessary to wait for a long time until most of the energy dissipates. Only after that could the next operation command be executed.

To preserve the efficiency, a solution called "Command shaping" has been applied to minimize residual vibration by taking energy out of resonant frequencies. This technique allows the reference signal to contain less energy at selected frequencies when it is sent to the system. Thus, the robot is able to settle down faster as

less residual vibration is induced. In the case of an open-loop system, the input command shaping method is sufficient to avoid long-lasting vibration. In reality, to ensure robustness and to reject disturbances, a closed-loop controller is desired.

Therefore, the general objective of this thesis is to analyze interaction between closed-loop controller and input command shaping method. A classic control law named "Computed torque control" is used throughout this research work. After analyzing the influence from the closed-loop controller on the input command shaping, a design procedure is proposed to synthesize the controller design and trajectory shaping process.

1.2 Literature Review

A considerable amount of effort has been put into robotics research, aiming to minimize vibration and to shorten time spent between motions. In general, three major categories has received most of the attention. Researchers in [1–3] give a good review and background for each of them.

The first category is known as command shaping. It aims to carefully construct the reference input to the robot in order to minimize end-point vibration while ensuring accomplishment of motion within a reasonable time frame. The mathematical proof given in [4] shows that taking energy out of a system at the resonant frequencies sufficiently reduces residual vibration at the end of motion. Bhat and Miu [5] showed that, for a system without damping, residual vibration could be eliminated by forming an input signal whose spectrum has no energy at system natural frequencies. For a damped system, however, residual vibration would still exist. Another approach in command shaping called Posicast control [6] is able to work with underdamped systems. The general idea behind it is to apply wave cancellation through a delayed input signal at the end of motion. However, its performance highly relies on an accurate system model; it is sensitive to system parameter errors and precise calculation of the resonant frequencies. Singer and Seering [7] explored the application of Finite

Impulse Response (FIR) Filter in a technique now called Input Shaping. Efforts in this work had been restricted to particular frequencies of interest, and later to design the FIR filter such that it removes as much energy around desired frequencies as possible to address system uncertainty. Other types of filters such as low-pass filter and band-stop filter have also been investigated by Mohamed and Tokhi [8]. Requiring minimum knowledge of the system, these filters are able to minimize residual vibration but at the cost of longer move time. Meckl [4, 9] introduced a new approach that uses harmonics of selected functions such as Ramped Sinusoid and Segmented Versine. This method makes the command shaping process more flexible to meet design requirements and constraints of application. Tokhi and Azad [10] compared different command shaping methods and discussed the experimental results obtained. Beazel [11] discusses the robustness of command shaping methods when modeling errors exist. It also confirms that this technique is effective for systems with multi modes. In [12, 13], researchers showed that for time-varying systems and systems with configuration-dependent natural frequencies, the command shaping technique can effectively reduce residual vibration. On the basis of this technique, Agrawal [14] extended this approach to numerical optimization and suggested Least Absolute Deviation (LAD) as a new cost function in order to avoid the Gibbs phenomenon, which is a common issue experienced in numerical optimization. Furthermore, Wu [15] presented a way to constrain the peak acceleration instead of actuation time because, in practice, actuators have limited power that has to be considered during the shaping process. Wu [16] combined the constraint on peak acceleration with the numerical optimization strategy to attenuate energy at system natural frequencies for Ramped Sinusoid inputs. Patil [17] later applied the peak acceleration constraint and numerical optimization to the Versine inputs.

In addition to command shaping, system identification has attracted many researchers over the years. Since most control methods and techniques are model-based, their qualities in application depend both on an accurate system model and parameter estimation. Many studies used a rigid body to model the robotic arm when they

either tried to develop shaped inputs or advanced controllers. Assuming a rigid-body model greatly simplifies derivations but could potentially end up with worse response. Futami and other researchers [18,19] showed that neglecting flexibility in each joint of the robot could degrade final results. Without including the effect of joint flexibility, estimating parameters based on such model would yield bias during the system identification stage. To avoid this issue, Sweet and Good [19] derived an analytical model for a compliant drive system, and [20, 21] used a sequential identification procedure for a robotic arm. These two methods had been adopted by Scheel [22] for a two-link flexible-joint robot arm. Besides the error due to unaccounted joint flexibility, in reality, measurement noise also greatly affects estimates. To improve accuracy, work in [23–26] had been focused on modifying the estimation method based on classic Ordinary Least Square (OLS) to minimize bias but it assumes that measurement noise is normally distributed. However, as mentioned in [27–29], when error is not normally distributed, which is the case for most apparatus, final estimates would have bias.

The last topic falls into feedback control strategy. It is helpful to ensure global asymptotic stability, robustness, and ability to reject disturbances. In general, controller design could be divided into feedforward and feedback approaches. With knowledge of the system model, inverse compensation, one method in feedforward control, would determine the reference input first and directly calculate input torques or forces by using the inverse of system dynamics, as shown in [30]. Nevertheless, for a non-minimum phase system, its inverse would result in system instability. Among those feedback controllers, Tomei [31] showed that a simple Proportional-Derivative (PD) control can achieve stability for a three-joint elastic robot by assuming no energy is added from these flexible joints. As extension, researchers have explored controller design with feedback linearization [32, 33]. De Luca [34] successfully applied dynamic feedback linearization to linearize and decouple the nonlinear dynamics. Recently, computed torque control has obtained substantial interest not only for typical robot models but also for robots with flexibility. As a type of feedback controller, its main structure combines the idea of feedback linearization and a PD controller [35]. Lin-

earization technique serves as a model-based portion that tries to decouple the system while the PD controller provides servo-based control to accomplish tasks, such as following a trajectory.

1.3 Overview of Thesis

This research studies the interaction between command shaping method and computed torque controller when they are applied to a two-link flexible joint robot at the same time. Although previous work in the shaping process has taken out energy at a selected frequency range for the closed-loop system, it is not clear how to choose controller parameters to further minimize energy being injected back into the system without affecting the overall quality of command shaping. To address this question, the closed-loop system due to the computed torque controller is derived first. Instead of using the compact model format of a motor-drive robotic arm, the link and motor subsystems are decoupled. Further analysis is provided when estimation error is present during the decoupling process. Simulations of different controller settings are run to see the influence on residual vibration. More simulation results for mismatched-model cases are shown. At the end, recommendations for choosing control gain are provided.

Introducing the background and theory of command shaping, Chapter 2 also describes the physical set up of the robotic arm and its mathematical model derivation. Application of command shaping on the robotic arm is provided. In Chapter 3, a system identification method known as Instrumental Variable (IV) is applied from literature. Comparison between IV estimation and the Recursive Least Square method is made. Chapter 4 shows application of the computed torque controller on the robotic arm. Stability is discussed first, followed by eigenvalue analysis. Comparison and discussion on simulation results are shown in order to outline controller design procedures. Finally, Chapter 5 summarizes all results from this research work, and gives suggestions on potential future work.

2. COMMAND SHAPING AND ROBOT MODEL

2.1 Background and Theory of Command Shaping

Typically, to achieve fast point-to-point motion, a Bang-Bang acceleration profile is desired. Under the command of this profile, actuators would switch instantaneously between the peak acceleration and peak deceleration, generating a jump in its time function. However, for a mechanical assembly that has several flexible modes, such type of command profile tends to generate excessive amount of vibration on the mechanical system even when the command profile has ended. As a result, longer time is spent before executing the next motion command.

Such vibration could be linked to the amount of energy being injected at system natural frequencies. To mathematically find the correlation between residual vibration amplitude and the system natural frequency, a classic linear two-mass single-mode system shown in Figure 2.1 is used. While keeping important dynamic properties, this linear model serves as a good starting point to analyze a two-link robot arm which will be discussed later. In this system, M_1 denotes the inertia of the motor, M_2 is the endpoint inertia, and k represents the transmission structural stiffness. Through input force f, energy is injected to the two-mass system, causing displacement of $\mathrm{Y_m}$ and $\mathrm{Y_e}$, respectively. Based on the two-mass model, Meckl [4] derived an analytical relationship between the amplitude of residual vibration and the magnitude of the spectrum of the forcing function evaluated at the flexible mode. The mathematical relationship is given by

$$A^* = \omega_n T_f |F^*(\omega_n T_f)|, \tag{2.1}$$

where A^* denotes the dimensionless amplitude of residual vibration, ω_n is the natural frequency of the system, and T_f stands for the end of actuation time. In this equa-

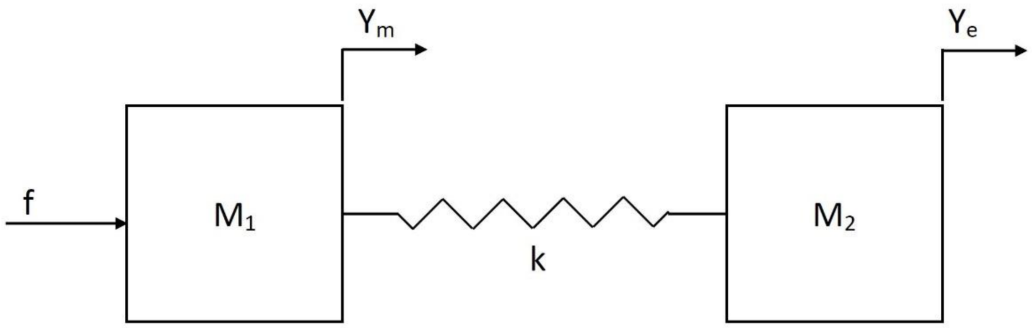


Figure 2.1. Linear Two-Mass System.

tion, the dimensionless magnitude of the Fourier Transform of the forcing function, $|F^*(\omega_n T_f)|$ is defined as

$$|F^*(\omega_n T_f)| = \frac{|F(\omega_n)|}{F_{max} T_f} \tag{2.2}$$

where $|F(\omega_n)|$ is evaluated at $\omega = \omega_n$ and obtained from the following equations,

$$F(\omega) = \int_0^{T_f} f(t) e^{-j\omega t} \delta t = [F_R(\omega) + jF_I(\omega)] e^{-j\frac{\omega T_f}{2}} \tag{2.3}$$

$$|F(\omega)| = \sqrt{F_R^2(\omega) + F_I^2(\omega)}\,. \tag{2.4}$$

In general, Equation 2.3 is the Fourier Transform of forcing function $f(t)$ evaluated from time zero to time T_f while Equation 2.4 calculates its spectral magnitude at ω based on the real part F_R and the imaginary part F_I. Moreover, F_{max} in Equation 2.2 is the peak value of the forcing function $f(t)$. More specifically, the forcing function is defined as

$$f(t) = \sum_{l=1}^{L} B_l \Phi_l^*(t)\,, \tag{2.5}$$

where B_l is the coefficient of l^{th} harmonic, Φ_l^* is the l^{th} harmonic of a basis function, and L is the total number of terms that form the forcing function. Two distinct basis functions are covered in Section 2.1.1 and 2.1.2. Accordingly, the dimensionless version of the forcing function is defined as

$$f(t)^* = \sum_{l=1}^{L} B_l^* \Phi_l^*(t)\,, \tag{2.6}$$

where B_l^* is the normalized dimensionless coefficient.

To reduce residual vibration and shorten total time to finish one motion command, the command shaping method comes into play. In general, there are two aspects influencing the shaping process. From a time-domain perspective, the Bang-Bang profile is the time-optimal solution for a rigid body. Because with constrained peak acceleration and a given actuation time, it injects the most energy to the system, generating a fast point-to-point motion. In light of this fact, to minimize move time, an optimal profile needs to mimic the Bang-Bang profile in order to maximize energy input. Meanwhile, Equation 2.1 indicates that, for any forcing function, its spectral magnitude at the system natural frequency greatly affects residual vibration; thus, the optimal profile should also avoid injecting energy around the system natural frequencies. As such, an objective function J capturing these two essential concepts could be formulated by the following equation

$$J = \frac{1}{T_f}\int_0^{T_f} [f^*_{BangBang}(t) - f^*(t)]^2 \delta t + \rho \sum_{i=1}^{11N_{mode}} (\omega_i T_f)^2 |F^*(\omega_i T_f)|^2 \,, \tag{2.7}$$

where ρ is the weighting factor, N_{mode} is the total number of mode frequencies to be shaped. As for $f^*_{BangBang}(t)$ profile, it is defined as

$$f^*_{BangBang}(t) = \begin{cases} 0 & t < 0 \\ 1 & 0 \le t < \frac{T_f}{2} \\ -1 & \frac{T_f}{2} \le t < T_f \\ 0 & T_f \le t \end{cases} \tag{2.8}$$

In the objective function, the first part, from a time-domain perspective, mainly enforces the penalty due to the difference between the forcing function and the Bang-Bang profile. The second part, from a frequency-domain point of view, explicitly imposes a penalty due to the squared magnitude of the spectrum of the corresponding forcing function evaluated at the system natural frequencies. In this part, 11 frequencies surrounding each natural frequency are chosen. From the standpoint of robustness, this ensures small magnitude throughout the selected frequency band

while avoiding having to integrate all frequencies over the spectrum. For a $\pm 10\%$ frequency range, bounds of ω_i are $0.9\omega_n \leq \omega_n \leq 1.1\omega_n$. Note that for a two-link flexible-joint robot, $N_{mode} = 2$. The coefficient B_l^*, as a design parameter of the forcing function, could be found by differentiating the objective function J with respect to B_l^* and setting the result to zero:

$$\frac{\partial J}{\partial B_l^*} = 0, \qquad (l = 1, 2, ..., L) \tag{2.9}$$

To proceed, two basis functions named Ramped Sinusoid and Versine are selected as the forcing functions. Each of them is covered in Sections 2.1.1 and 2.1.2, respectively.

2.1.1 Ramped Sinusoid

By definition, the Ramped Sinusoid is described by the following mathematical representation

$$\Phi_l^*(t) = \frac{1}{\alpha_l}(\frac{1}{2} - \tau) + \frac{1}{\alpha_l^2}\sin(\alpha_l\tau) - \frac{1}{2\alpha_l}\cos(\alpha_l\tau) \tag{2.10}$$

where τ is the dimensionless time defined by

$$\tau = \frac{t}{T_f} \tag{2.11}$$

and T_f, same as before, denotes the total actuation time of the input profile. α_l is the characteristic number associated with the l^{th} harmonic. To achieve smoothness and avoid discontinuity, zero magnitude and slope at the beginning and the end of motion are imposed on the Ramped Sinusoid function by the following constraint

$$\alpha_l \sin(\alpha_l) + 2\cos(\alpha_l) - 2 = 0, \qquad (\alpha_l \neq n\pi)\,, \tag{2.12}$$

where n is an even integer. The first ten characteristic numbers for the harmonics of the Ramped Sinusoid and its first five harmonics are given in Table 2.1 and Figure 2.2, respectively. Note that from the given Ramped Sinusoid basis function, it is possible to approximate a full cycle of the Bang-Bang profile. Plugging the forcing function Ramped Sinusoid into the objective function defined in Equation 2.7, the

Table 2.1. Characteristic Numbers for First Ten Harmonics of Ramped Sinusoid

l^{th} Harmonic	Value
α_1	8.9688
α_2	15.4505
α_3	21.8082
α_4	28.1324
α_5	34.4415
α_6	40.7426
α_7	47.0389
α_8	53.3321
α_9	59.6232
α_{10}	65.9128

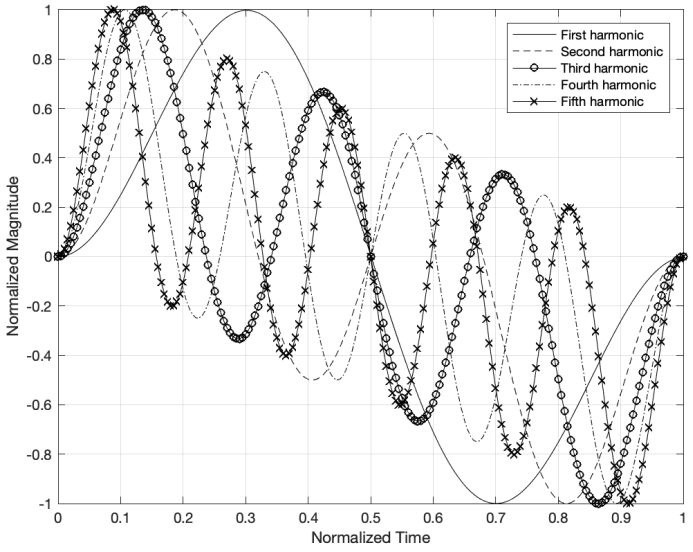


Figure 2.2. First Five Harmonics of Normalized Ramped Sinusoid Function

analytical solution is given in [4]. Since T_f and B_l^* are both unknown before one of them is determined, an iteration process has been incorporated into the solution. As

an alternative to the analytical solution, a numerical optimization approach, shown in Figure 2.3, was proposed by Agrawal [14] to obtain the forcing function.

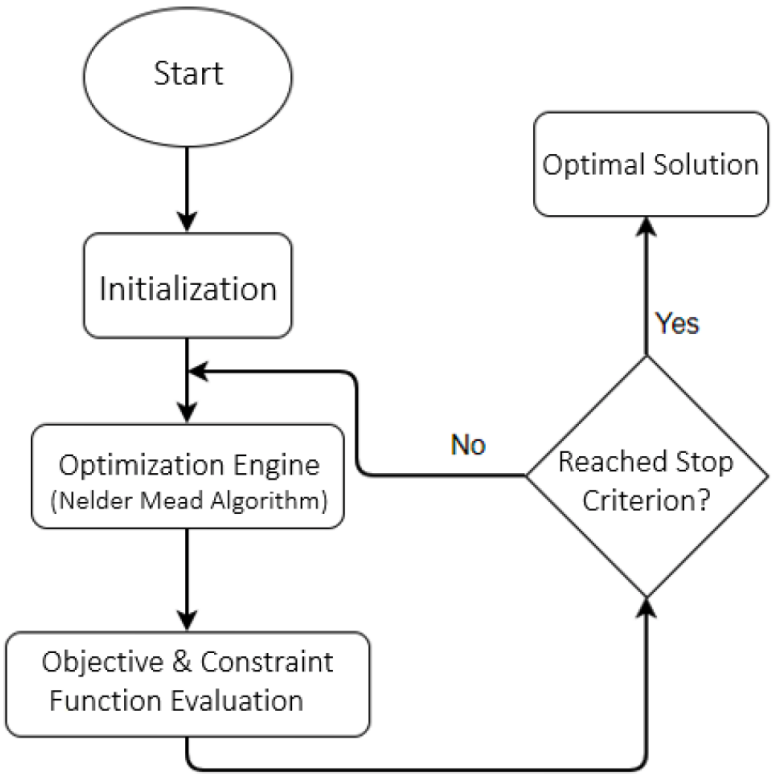


Figure 2.3. Flowchart of the Numerical Optimization Approach [14]

2.1.2 Segmented Versine

In addition to the Ramped Sinusoid, the Versine is another candidate for approximating the Bang-Bang profile. By definition, it is given by the following equation

$$\Phi_l^*(t) = 1 - \cos(2\pi l\tau), \qquad (l = 1, 2, ..., L)\,, \tag{2.13}$$

where l denotes the l^{th} harmonic, and τ is the dimensionless time defined by

$$\tau = \frac{t}{T_{p,i}} \tag{2.14}$$

where $T_{p,i}$ is the actuation time of the i^{th} segment. Since a full command profile may consist of acceleration, constant-velocity motion, and deceleration, the total actuation time would be the sum of actuation times from each of them, and could be written as

$$T_f = \sum_{i=1}^{N_{segment}+1} T_{p,i} \tag{2.15}$$

where $N_{segment} + 1$ is the number of segments making up one full input profile plus the constant velocity segment. Figure 2.4 shows the first three harmonics of the Versine profile. Unlike the Ramped Sinusoid, which approximates a full cycle of the

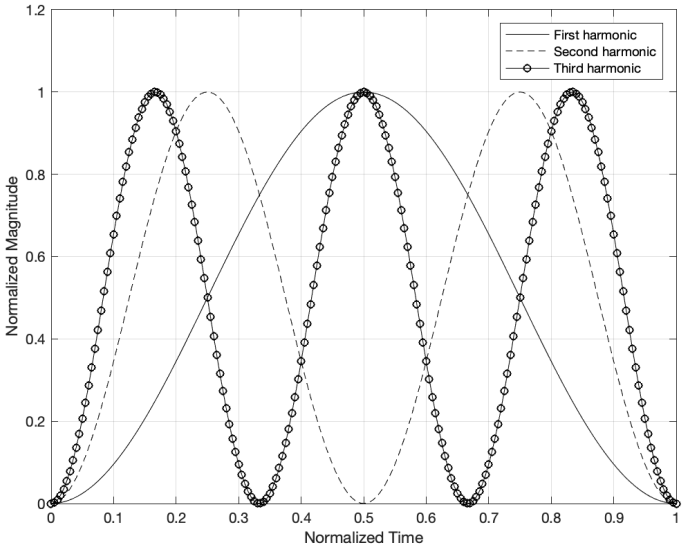


Figure 2.4. First Three Harmonics of Normalized Versine Function

Bang-Bang profile, the Versine mimics either the positive or negative square pulse of the Bang-Bang profile. Beazel in [11] proposed dividing the command shaping method into multiple segments. This is useful because for a mechanical system with configuration-dependent natural frequency, each segment has the flexibility to choose a different frequency band based on position. With precisely tailored and adjusted segments, more critical energy associated with the residual vibration could be removed. As a result, the ability of the Versine to approximate the Bang-Bang profile

with multiple segments allows the targeted frequency to change over time, and to minimize the spectrum energy only relevant to that segment. After carefully designing each segment, the command shaping method combines all segments together with respect to time, and yields the full command profile for the system. What's more, for a mechanical system that has a velocity limit, the segmented Versine profile is able to incorporate this constraint during the shaping process. With known peak velocity, one segment drives the system to achieve the limit velocity, maintains that constant velocity for some time, and then the second segment decelerates the system to zero velocity at the end. To ensure reaching the final position, command shaping can also enforce a constraint on the integration of velocity with respect to time. Such an advantage makes the segmented Versine relatively more desirable over the Ramped Sinusoid which cannot enforce peak velocity as both acceleration and deceleration are shaped simultaneously. Again, the analytical solution of incorporating the Versine function into the objective function defined in Equation 2.7 is given in [4]. A similar numerical optimization solution can obtained by using the method proposed in [14]

2.2 Two-Link Flexible-Joint Robot

For research purposes, a two-link robot with joint flexibility in the Ruth and Joel Spira Laboratory for Electromechanical Systems in the School of Mechanical Engineering at Purdue University is used in this work. With carefully built-in flexibility, it represents the testbed for the command shaping method. To minimize the effect due to gravity, both joints are designed and assembled in a way that the robot can only move in the horizontal plane. Figure 2.5 [16] shows details of the hardware that is set up on an experiment platform. As seen, the robotic arm consists of two links, referred to as link one and link two, or shoulder link and elbow link. Through the belt drive, each link is driven by a DC motor, namely motor one or motor two. More specifically, motor one is mounted on a fixed stand that is attached on the platform so that minimum energy could be transfered from the platform to the robot, and vice

versa. Motor 2, as shown, is mounted on the first link. To complete, there are two sets of torsional springs and sprockets on each link. As a result, each link is indirectly driven by its motor with a gear ratio of five. With this hardware setup, the robot is able to achieve a horizontal sweeping motion with typical flexibility. In addition, from the standpoint of feedback control, there are a total of four encoders and two accelerometers taking measurements from the robot. Two of the four encoders are directly attached to the motors so that actual displacement from each motor can be recorded. The other two encoders are placed on top of the end of each belt drive, recording the relative displacement between the motor angle and link angle. Two accelerometers are attached at the end of each link. Note that the built-in flexibility is realized by the torsional spring between each link and each sprocket. The spring coefficient ranges from 10^3 to 10^5 Nm/rad [16], which is much smaller than the typical joint stiffness of a robotic arm.

The controller was implemented firstly by Chatlatanagulchai [36] in LabVIEW (Version 8.5.1). Other than the robotic arm, the experimental apparatus includes a desktop PC, a National Instruments (NI) PXI-7831R field programmable gate array (FPGA), and two motor drives. The FPGA comes with 16 analog I/O ports and 96 digital channels with Virtex-II M gate so that it is capable of handling large amounts of data transmission. The FPGA is able to operate up to 200kHz to sample measurements. But for control purposes, both feedback signals and control outputs are received and transmitted at 2kHz. Encoder measurement and other essential input signals are connected to the FPGA through two National Instruments SCB-68 Shielded I/O Connector Blocks, respectively. The DC motor used by the shoulder link is a version of Electro Craft DPP242, providing torque constant of 0.118 Nm/A and maximum torque of 2.47 Nm. The second motor, Inland T-3108-A, drives the elbow link with a constant torque of 0.61 Nm/A and maximum torque of 1.35 Nm. Two motor drives called Advanced Motion Control pulse-width-modulated (PWM) transconductance servo amplifiers are able to convert the voltage command from the FPGA to a current source for the motor, respectively. Encoders from Renco Encoders

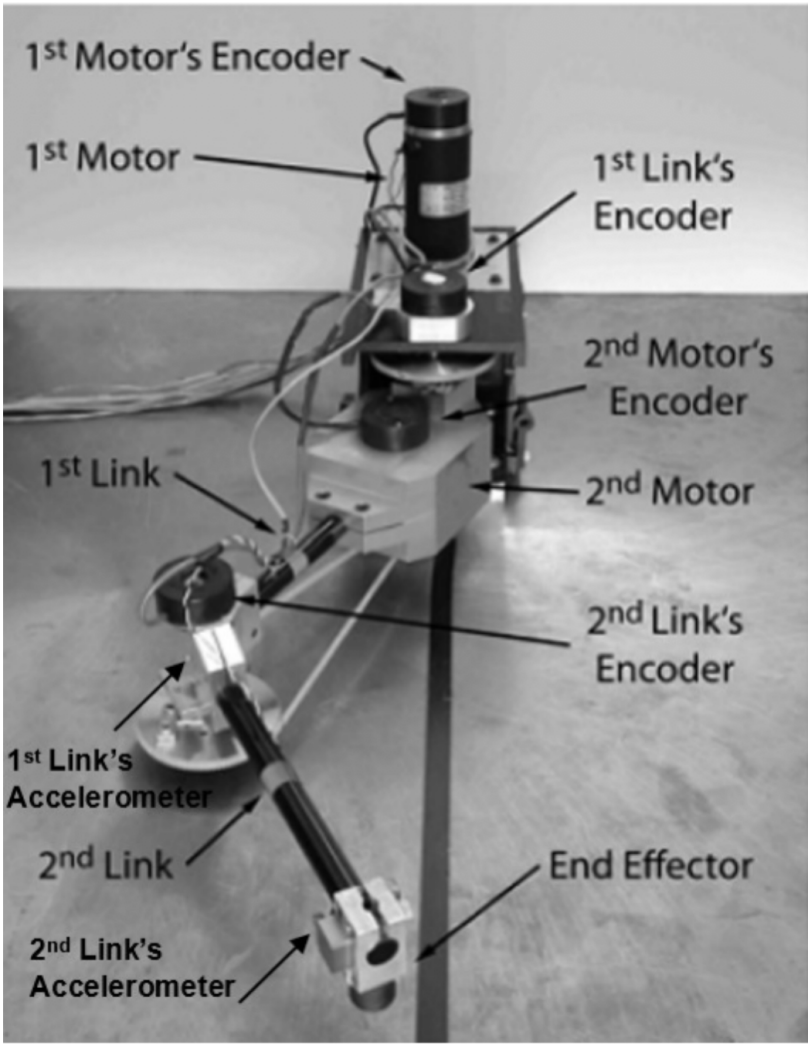


Figure 2.5. Physical Set up of the Robotic Arm [16]

are used. All of them have the same resolution of 4000 counts per revolution. The only difference is the model used for the shoulder and elbow links. R80 encoder is mounted on the first link while RM21 for the second link. Lastly, the accelerometers that are installed at the end of each link are both model 8315A from Kistler. They measure single-axis linear acceleration with a range of $\pm$ 2g, sensitivity of 2 V/g at frequency band of 0 to 250Hz, and a resolution of 0.35mg.

2.3 Robot Mathematical Model

To derive the mathematical model for the two-link flexible-joint robotic arm, the schematic is shown in Figure 2.6, given by Nho [37]. In this schematic, physical parameters are labeled and will be used in Section 2.3.1, which provides the complete Lagrangian Model developed by Nho [37]. Section 2.3.2 provides the reduced model introduced by Spong [38].

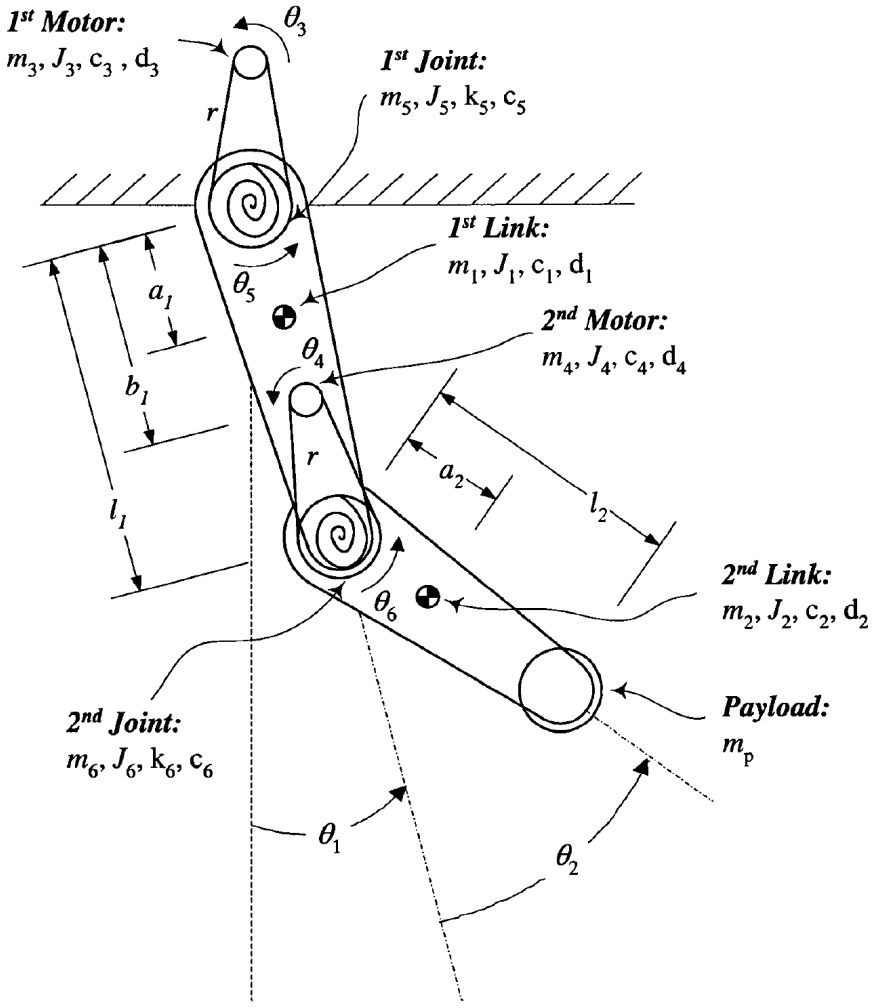


Figure 2.6. Schematic of the Two-Link Robotic Arm with Joint Flexibility [37]

As seen from the schematic of the robotic arm, θ_1 and θ_2 measure the angular position of link one and link two, respectively, while θ_3 and θ_4 measure motor one and motor two angular position, respectively. Note that both θ_1 and θ_3 take measurements with respect to the inertia frame but θ_2 and θ_4 take measurements with respect to the link one coordinate frame. The numerical values of the robot dimensions are listed in Table 2.2 from [16]. The expression of p_1 is defined in Equation 3.21.

Table 2.2. Physical Parameter Values of the Robotic Arm [16]

Parameter	Value	Parameter	Value
p_1	0.140 $(\frac{\text{kgm}^2}{\text{rad}})$	c_5	0.005 $(\frac{\text{Nms}}{\text{rad}})$
$m_2a_2^2 + J_2$	0.0196 $(\frac{\text{kgm}^2}{\text{rad}})$	c_6	8.128×10^{-5} $(\frac{\text{Nms}}{\text{rad}})$
$l_1m_2a_2$	0.0234 $(\frac{\text{kgm}^2}{\text{rad}})$	k_5	2.848 $(\frac{\text{Nm}}{\text{rad}})$
J_3	4.157×10^{-5} $(\frac{\text{kgm}^2}{\text{rad}})$	k_6	2.848 $(\frac{\text{Nm}}{\text{rad}})$
J_4	7.543×10^{-4} $(\frac{\text{kgm}^2}{\text{rad}})$	d_1	0.0199 (Nm)
J_5	0.025 $(\frac{\text{kgm}^2}{\text{rad}})$	d_2	0.0323 (Nm)
J_6	0.025 $(\frac{\text{kgm}^2}{\text{rad}})$	d_3	0.005 (Nm)
c_1	0.04 $(\frac{\text{Nms}}{\text{rad}})$	d_4	0.0271 (Nm)
c_2	0.0214 $(\frac{\text{Nms}}{\text{rad}})$	b_1*	0.064 (m)
c_3	1.894×10^{-4} $(\frac{\text{Nms}}{\text{rad}})$	l_1*	0.318 (m)
c_4	1.497×10^{-3} $(\frac{\text{Nms}}{\text{rad}})$	l_2*	0.291 (m)

* These parameters are directly measured from the robot.

2.3.1 Lagrangian Model

The complete Lagrangian Model was derived by Nho [37]. In this work, payload is set to zero; therefore, the end effector has minimum effect on the robot dynamics.

Starting from the Lagrangian Model, a general equation of motion for the robotic arm can be derived and written as

$$\mathbf{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{V}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{C}\dot{\boldsymbol{\theta}} + \mathbf{D}(\boldsymbol{\theta}) + \mathbf{K}\boldsymbol{\theta} = \mathbf{T}\,, \tag{2.16}$$

where $\boldsymbol{\theta}$ is a 4 by 1 column vector consisting of link one, link two, motor one, and motor two angles. Each element of $\boldsymbol{\theta}$ is denoted by θ_i with $i = 1, 2, 3, 4$ to represent a specific generalized angle, as shown in Figure 2.6. In this work, the derivative of $\boldsymbol{\theta}$ with respective to time is denoted as $\dot{\boldsymbol{\theta}}$, namely the angular velocity. Similarly, angular acceleration uses the symbol of $\ddot{\boldsymbol{\theta}}$. In Equation 2.16, $\mathbf{M}(\boldsymbol{\theta})$, as a function of $\boldsymbol{\theta}$, represents the inertia matrix for the robot and is given as

$$\mathbf{M}(\boldsymbol{\theta}) = \begin{bmatrix} \mathbf{M_1} & \mathbf{M_2} \\ \mathbf{M_2}^T & \mathbf{M_3} \end{bmatrix} \tag{2.17}$$

where

$$\mathbf{M_1} = \begin{bmatrix} \begin{pmatrix} m_1 a_1^2 + m_2(l_1^2 + a_2^2) \\ +m_4 b_1^2 + m_6 l_1^2 + J_1 + J_2 \\ +J_4 + J_6 + m_p(l_1^2 + l_2^2) \\ +2l_1(m_2 a_2 + m_p l_2)\cos(\theta_2) \end{pmatrix} & \begin{pmatrix} m_2 a_2^2 + J_2 \\ +m_p l_2^2 + l_1(m_2 a_2 \\ +m_p l_2)\cos(\theta_2) \end{pmatrix} \\ \\ \begin{pmatrix} m_2 a_2^2 + J_2 + m_p l_2^2 \\ +l_1(m_2 a_2 + m_p l_2)\cos(\theta_2) \end{pmatrix} & \left(m_2 a_2^2 + J_2 + m_p l_2^2\right) \end{bmatrix} \tag{2.18}$$

$$\mathbf{M_2} = \begin{bmatrix} 0 & (J_4 + \frac{J_6}{r}) \\ 0 & 0 \end{bmatrix} \tag{2.19}$$

and

$$\mathbf{M_3} = \begin{bmatrix} (J_3 + \frac{J_5}{r^2}) & 0 \\ 0 & (J_4 + \frac{J_6}{r^2}) \end{bmatrix}\,. \tag{2.20}$$

Note that $\mathbf{M_2}^T$ is the transpose of matrix $\mathbf{M_2}$. In the $\mathbf{M_1}$ matrix, m_i denotes the lumped mass, l_i represents the length of the link, J_i is the moment of inertia, and a_i is the distance between the center of gravity of link i and joint i. b_i is the distance

between joint i and motor $i+1$. Gear ratio for the sprocket/chain transmission is denoted by r, and equal to five in this work. As mentioned before, no payload would be put onto the robot, so that $m_p = 0$ applies to all equations.

As for $\mathbf{V}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$, this matrix consists of elements due to Coriolis and Centrifugal forces, and is written as

$$\mathbf{V}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \begin{bmatrix} -l_1(m_2 a_2 + m_p l_2)(2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2)\sin(\theta_2) \\ l_1(m_2 a_2 + m_p k_2)\dot{\theta}_1^2 \sin(\theta_2) \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \mathbf{V}_{link}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) \\ \mathbf{V}_{motor}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) \end{bmatrix} . \tag{2.21}$$

The system viscous damping matrix $\mathbf{C}$ consisting of damping coefficients is given by

$$\mathbf{C} = \begin{bmatrix} c_1 + c_5 & 0 & -\frac{c_5}{r} & 0 \\ 0 & c_2 + c_6 & 0 & -\frac{c_6}{r} \\ -\frac{c_5}{r} & 0 & c_3 + \frac{c_5}{r^2} & 0 \\ 0 & -\frac{c_6}{r} & 0 & c_4 + \frac{c_5}{r^2} \end{bmatrix} = \begin{bmatrix} \mathbf{C_1} & \mathbf{C_2} \\ \mathbf{C_2} & \mathbf{C_3} \end{bmatrix} = \begin{bmatrix} \mathbf{C}_{link} \\ \mathbf{C}_{motor} \end{bmatrix} . \tag{2.22}$$

where c_i denotes the damping coefficient shown in Figure 2.6, and $\mathbf{C}_{link}$ and $\mathbf{C}_{motor}$ are 2×4 matrices.

Coulumb friction is shown in the column vector $\mathbf{D}$ and is given by

$$\mathbf{D}(\dot{\boldsymbol{\theta}}) = \begin{bmatrix} d_1 \text{sign}(\dot{\theta}_1) \\ d_2 \text{sign}(\dot{\theta}_2) \\ d_3 \text{sign}(\dot{\theta}_3) \\ d_4 \text{sign}(\dot{\theta}_4) \end{bmatrix} = \begin{bmatrix} \mathbf{D}_{link}(\dot{\boldsymbol{\theta}}) \\ \mathbf{D}_{motor}(\dot{\boldsymbol{\theta}}) \end{bmatrix} , \tag{2.23}$$

where sign is an operation defined as

$$\text{sign}(x) = \begin{cases} 1 & \text{if} \quad x > 0 \\ 0 & \text{if} \quad x = 0 \\ -1 & \text{if} \quad x < 0 \end{cases} . \tag{2.24}$$

The system stiffness matrix $\mathbf{K}$ is given by

$$\mathbf{K} = \begin{bmatrix} k_5 & 0 & -\frac{k_5}{r} & 0 \\ 0 & k_6 & 0 & -\frac{k_6}{r} \\ -\frac{k_5}{r} & 0 & \frac{k_5}{r^2} & 0 \\ 0 & -\frac{k_6}{r} & 0 & \frac{k_6}{r^2} \end{bmatrix} = \begin{bmatrix} \mathbf{K_1} & \mathbf{K_2} \\ \mathbf{K_2} & \mathbf{K_3} \end{bmatrix} = \begin{bmatrix} \mathbf{K}_{link} \\ \mathbf{K}_{motor} \end{bmatrix}, \tag{2.25}$$

where k_i denotes specific stiffness due to the torsional spring, and $\mathbf{K}_{link}$ and $\mathbf{K}_{motor}$ are 2×4 matrices.

Lastly, the input Torque is denoted by the column vector $\mathbf{T}$, and is defined as

$$\mathbf{T} = \begin{bmatrix} 0 \\ 0 \\ T_1 \\ T_2 \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{T}_{motor} \end{bmatrix}, \tag{2.26}$$

where T_1 is the driving torque generated by motor one while T_2 by motor two.

Note that the matrices $\mathbf{M}(\boldsymbol{\theta})$, $\mathbf{C}$, and $\mathbf{K}$ are all symmetric. Immediately, it can be seen that coupling exists between links and motors; thus, none of these matrices is diagonal. In general, the dimension of Equation 2.16 is different from that of a typical two-link robotic arm. This is because, in this work, actuators are included in the model, so that the flexibility between links and motors can be captured. Including flexibility is important as it serves as the foundation for the command shaping method, which requires a flexible model representation of the system.

2.3.2 Reduced Model

In [38] Spong provided a simplified model based on the full Lagrangian Model. The reduced model, in general, removes coupling elements between links and motors. There are two assumptions to achieve the simplified model. The first assumption states that the majority of the kinetic energy of each motor is due to its own rotation. This assumption requires the gear ratio between each link and its motor to be greater than 1 so that the motor has greater angular velocity than that of the link. Thus,

the inertia matrix $\mathbf{M}(\boldsymbol{\theta})$ drops coupling terms due to each set of link and motor. The second assumption is to assume viscous damping coefficients c_5 and c_6 of the torsional springs are negligible because they are small enough when compared to other viscous damping coefficients. This assumption drops coupling terms between links and motors in matrix $\mathbf{C}$ and results in $\mathbf{C}_{reduced}$. These simplifications greatly reduce the complexity of the system, and yet preserve essential dynamics of the robot. By definition, $\mathbf{M}_{reduced}(\boldsymbol{\theta})$ is written as

$$\mathbf{M}_{reduced}(\boldsymbol{\theta}) = \begin{bmatrix} \mathbf{M_1} & \mathbf{0} \\ \mathbf{0} & \mathbf{M_3} \end{bmatrix} \tag{2.27}$$

where $\mathbf{M}_1$ and $\mathbf{M}_3$ are defined in Equations 2.18 and 2.20, respectively.

As for $\mathbf{C}_{reduced}$, it is defined as

$$\mathbf{C}_{reduced} = \begin{bmatrix} c_1 & 0 & 0 & 0 \\ 0 & c_2 & 0 & 0 \\ 0 & 0 & c_3 & 0 \\ 0 & 0 & 0 & c_4 \end{bmatrix}, \tag{2.28}$$

which can be separated into two damping matrices corresponding to links and motor. They are written as

$$\mathbf{C}_{link} = \begin{bmatrix} c_1 & 0 \\ 0 & c_2 \end{bmatrix}, \qquad \mathbf{C}_{motor} = \begin{bmatrix} c_3 & 0 \\ 0 & c_4 \end{bmatrix}. \tag{2.29}$$

As a result, the general equation of motion for the robot expressed in Equation 2.16 becomes

$$\mathbf{M}_{reduced}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{V}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{C}_{reduced}\dot{\boldsymbol{\theta}} + \mathbf{D}(\dot{\boldsymbol{\theta}}) + \mathbf{K}\boldsymbol{\theta} = \mathbf{T} \tag{2.30}$$

which can be rearranged and divided into link subsystem and motor subsystem with the following equations:

$$\mathbf{M}_1(\boldsymbol{\theta}_{link})\ddot{\boldsymbol{\theta}}_{link} + \mathbf{V}_{link}(\boldsymbol{\theta}_{link}, \dot{\boldsymbol{\theta}}_{link}) + \mathbf{C}_{link}\dot{\boldsymbol{\theta}}_{link} + \mathbf{D}_{link}(\dot{\boldsymbol{\theta}}_{link}) + \mathbf{K}_{\mathrm{S}}(\boldsymbol{\theta}_{link} - \frac{\boldsymbol{\theta}_{motor}}{r}) = \mathbf{0} \tag{2.31}$$

and

$$\mathbf{M}_3\ddot{\boldsymbol{\theta}}_{motor} + \mathbf{C}_{motor}\dot{\boldsymbol{\theta}}_{motor} + \mathbf{D}_{motor}(\dot{\boldsymbol{\theta}}_{motor}) + \mathbf{K}_{\mathrm{S}}(\frac{\boldsymbol{\theta}_{motor}}{r^2} - \frac{\boldsymbol{\theta}_{link}}{r}) = \mathbf{T}_{motor}\,, \quad (2.32)$$

where $\boldsymbol{\theta}_{link}$ and $\boldsymbol{\theta}_{motor}$ are obtained from

$$\boldsymbol{\theta} = \begin{bmatrix} \boldsymbol{\theta}_{link} \\ \boldsymbol{\theta}_{motor} \end{bmatrix} \quad (2.33)$$

and $\mathbf{K}_{\mathrm{S}}$ is defined as

$$\mathbf{K}_{\mathrm{S}} = \begin{bmatrix} k_5 & 0 \\ 0 & k_6 \end{bmatrix}\,. \quad (2.34)$$

2.4 Robot Kinematics

To quantitatively compare the effectiveness of the command shaping method on the two-link robotic arm, a performance metric needs to be defined. Since this work focuses on residual vibration, the performance metric has to include relevant features. In general, both acceleration vibration amplitude and settling time are of interest. Unlike a linear single-mode mechanical system that has a closed-form analytical expression for the settling time and the amplitude of vibration, the robot model presented in this work is a nonlinear multi-mode system. Therefore, the standard metrics introduced in [14] are adopted in this work. This method, based on robot kinematics, provides measurement of residual vibration at the end effector. With perfect design, there should be no vibration when the input profile ends. Therefore, any observed vibration after the command profile has ended is undesired and could be used to calculate the system performance. The coordinate frame shown in Figure 2.7 [16] serves as a starting point to derive the robot kinematics, followed by calculation of the residual vibration.

As seen, $X_0Y_0Z_0$ is the fixed frame attached at the origin, $X_iY_iZ_i$ is the coordinate frame attached at the end of link i, and $U_iV_iW_i$ is the coordinate frame attached on joint J_i. Note that gravity is pointing into the page. Since the robot moves only in the XY plane, there is no translation along axes Z and W, or rotations about axes

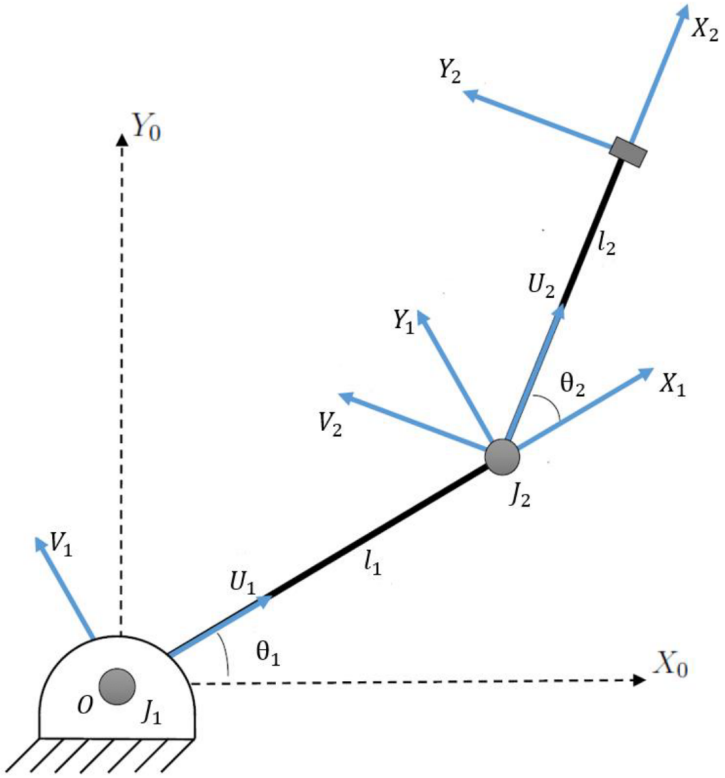


Figure 2.7. Coordinate Frame of the Robotic Arm [16]

X, Y, U, and V. The rotation and the linear translation matrices for the robot, thus, can be written as

$$\phi_i = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 & 0 \\ \sin(\theta_i) & \cos(\theta_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.35}$$

and

$$T_{trans,i} = \begin{bmatrix} 1 & 0 & 0 & l_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2.36}$$

where ϕ_i is the rotation matrix, $T_{trans,i}$ is the linear translation matrix, and $i = 1, 2$, for this robot. Therefore, the overall homogeneous transformation matrix for the manipulator is obtained by the following equation

$$T_m = \phi_1 T_{trans,1} \phi_2 T_{trans,2} \tag{2.37}$$

and written as

$$T_m = \begin{bmatrix} C(\theta_1)C(\theta_2) - S(\theta_1)S(\theta_2) & -S(\theta_1)C(\theta_2) - C(\theta_1)S(\theta_2) & 0 & P_x \\ S(\theta_1)C(\theta_2) + C(\theta_1)S(\theta_2) & C(\theta_1)C(\theta_2) - S(\theta_1)S(\theta_2) & 0 & P_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2.38}$$

where $C(\theta_i)$ and $S(\theta_i)$ are shorthand notation for $\cos(\theta_i)$ and $\sin(\theta_i)$, respectively. As for P_x and P_y, they are defined as

$$P_x = l_1 C(\theta_1) + l_2 C(\theta_1)C(\theta_2) - l_2 S(\theta_1)S(\theta_2) \tag{2.39}$$

$$P_y = l_1 S(\theta_1) + l_2 S(\theta_1)C(\theta_2) + l_2 C(\theta_1)S(\theta_2)\,. \tag{2.40}$$

To further simplify, Equation 2.38 can be written as

$$T_m = \begin{bmatrix} \cos(\theta_1 + \theta_2) & -\sin(\theta_1 + \theta_2) & 0 & P_x \\ \sin(\theta_1 + \theta_2) & \cos(\theta_1 + \theta_2) & 0 & P_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.41}$$

and Equations 2.39 and 2.40 become

$$P_x = l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2) \tag{2.42}$$

$$P_y = l_1 \sin(\theta_1) + l_2 \sin(\theta_1 + \theta_2)\,. \tag{2.43}$$

The velocity can be obtained by differentiating Equation 2.37, assuming all links are rigid with $\frac{\partial T_{trans,i}}{\partial t} = 0$, and can be written as

$$\begin{aligned} \frac{\partial T_m}{\partial t} = \frac{\partial \phi_1}{\partial t} T_{trans,1} \phi_2 T_{trans,2} + \phi_1 T_{trans,1} \frac{\partial \phi_2}{\partial t} T_{trans,2} = \\ \dot{\theta}_1 Q_R \phi_1 T_{trans,1} \phi_2 T_{trans,2} + \dot{\theta}_2 \phi_1 T_{trans,1} Q_R \phi_2 T_{trans,2}\ , \end{aligned} \tag{2.44}$$

where Q_R denotes the derivative of the homogeneous conversion of ϕ_i with respect to time. Q_R is defined as

$$Q_R = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} . \tag{2.45}$$

The translational velocity, as a result, can be obtained from Equation 2.44 and can be written as

$$\dot{P}_x = -l_1\dot{\theta}_1 \sin(\theta_1) - l_2(\dot{\theta}_1 + \dot{\theta}_2)\sin(\theta_1 + \theta_2) \tag{2.46}$$

$$\dot{P}_x = l_1\dot{\theta}_1 \cos(\theta_1) + l_2(\dot{\theta}_1 + \dot{\theta}_2)\cos(\theta_1 + \theta_2) \ . \tag{2.47}$$

Similarly, the linear acceleration for the end effector can be obtained from differentiating Equation 2.44 with respect to time, and the result can be written as

$$\begin{aligned} \frac{\partial^2 T_m}{\partial t^2} &= \ddot{\theta}_1 Q_R \phi_1 T_{trans,1} \phi_2 T_{trans,2} + \dot{\theta}_1^2 Q_R^2 \phi_1 T_{trans,1} \phi_2 T_{trans,2} \\ &+ 2\dot{\theta}_1\dot{\theta}_2 Q_R \phi_1 T_{trans,1} \phi_2 T_{trans,2} + \dot{\theta}_2^2 \phi_1 T_{trans,1} Q_R^2 \phi_2 T_{trans,2} + \ddot{\theta}_2 \phi_1 T_{trans,1} Q_R \phi_2 T_{trans,2} \ . \end{aligned} \tag{2.48}$$

Thus, the linear accelerations with respect to the fixed frame $\ddot{P}_x$ and $\ddot{P}_y$ can be written as

$$\begin{aligned} \ddot{P}_x = &-l_1\ddot{\theta}_1 \sin(\theta_1) - l_1\dot{\theta}_1^2 \cos(\theta_1) - l_2(\dot{\theta}_1^2 + 2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2)\cos(\theta_1 + \theta_2) \\ &-l_2(\ddot{\theta}_1 + \ddot{\theta}_2)\sin(\theta_1 + \theta_2) \end{aligned} \tag{2.49}$$

and

$$\begin{aligned} \ddot{P}_y = &\ l_1\ddot{\theta}_1 \cos(\theta_1) - l_1\dot{\theta}_1^2 \sin(\theta_1) - l_2(\dot{\theta}_1^2 + 2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2)\sin(\theta_1 + \theta_2) \\ &+l_2(\ddot{\theta}_1 + \ddot{\theta}_2)\cos(\theta_1 + \theta_2) \ . \end{aligned} \tag{2.50}$$

Note that the same results as Equations 2.49 and 2.50 can be directly derived from Equations 2.46 and 2.47. Getting Equation 2.48 first provides information of relative rotational speed of the end effector with respect to the fixed frame.

2.5 Performance Metrics

As mentioned before, to apply the command shaping method which aims to minimize residual vibration at the end point, namely the end effector, standardized performance metrics are required so as to compare the shaping effectiveness over different sets of parameters. Since measuring the end-point vibration amplitude is able to quantify the effectiveness, the linear accelerations, determined by Equations 2.49 and 2.50, are used. The magnitude of residual vibration is given by the following equation

$$\ddot{P}_{xy}(t) = \sqrt{\ddot{P}_x^2(t) + \ddot{P}_y^2(t)} \tag{2.51}$$

so that $\ddot{P}_{xy}(t)$ is able to capture the residual vibration magnitude regardless of its direction, and t is the time interval for settling, defined as

$$T_f \leq t \leq T_{end}\,, \tag{2.52}$$

where T_f, mentioned before, is the final time of the input profile, and T_{end} is the time when $\ddot{P}_{xy}(t)$ is smaller than a predefined threshold value. In this work, the threshold value is 0.2 ($\mathrm{m/s^2}$). Based on this, the maximum linear residual vibration is defined as

$$a_{rv} = \max(\ddot{P}_{xy}(t))\,. \tag{2.53}$$

Figure 2.8 from Wu [16] gives an example to demonstrate the definition of the residual vibration and settling time.

2.6 Application of Command Shaping to the Robot

For the purpose of demonstrating the effectiveness of the command shaping method, this subsection presents selected experimental results before and after applying the shaping method. Again, the idea of command shaping is to approximate the Bang-Bang profile to approximate the time-optimal solution while removing energy content at the associated resonant frequencies. For the two-link robotic arm, the first two natural frequencies are determined to be 3.709(rad/s) and 15.7338(rad/s). Thus the red

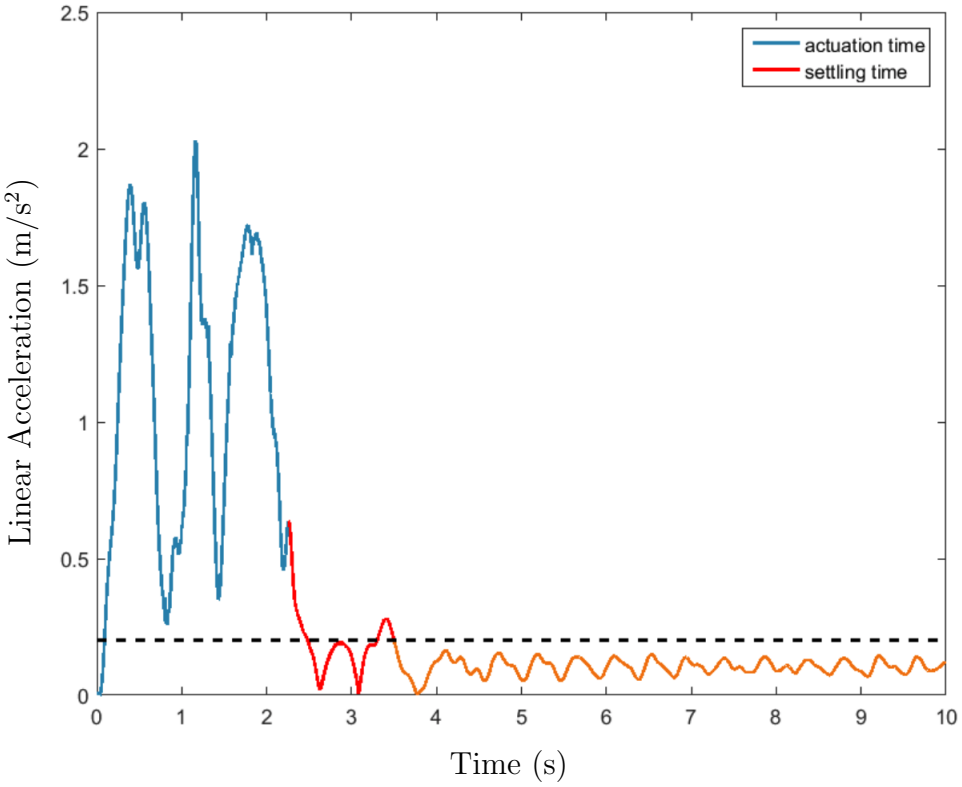


Figure 2.8. Command profile with labeled settling time [16]

dotted lines in Figure 2.9 indicate the energy at which frequencies should be taken away when approximating the Bang-Bang profile. Note that the natural frequencies are determined from the closed-loop system, which is covered in Chapter 4. Figure 2.10 from [17] shows the robot response due to the Bang-Bang profile. The first row of Figure 2.10, from left to right, presents the Bang-Bang profile and its corresponding frequency spectrum. The second row shows the torque inputs for the system generated by motor one and motor two, respectively. The third row gives actual angular position responses of the shoulder link and the elbow link, respectively. The fourth row is angular positions of motor one and motor two, respectively. The last row shows the actual angular accelerations of link one and link two, respectively. Note that dotted lines in each subplot represent the desired response. As seen, after the input has ended at $t = 3$(s), the system shows non-zero residual vibration measured

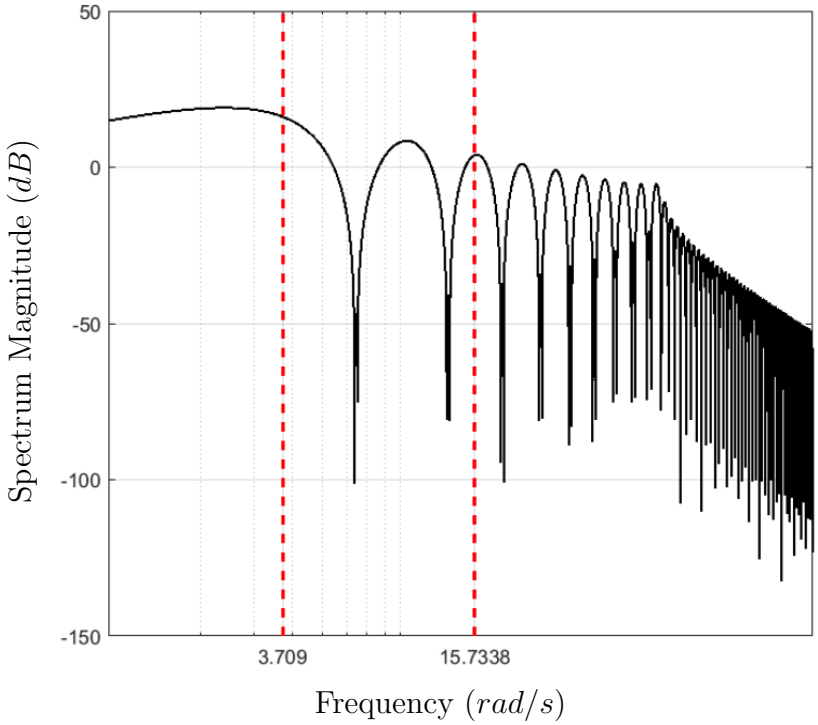


Figure 2.9. Spectrum of the Bang-Bang Profile [16]

by Equation 2.51, and the peak vibration is 1.2376(m/s^2) while settling time is about 2.248 seconds. To compare, a shaped profile using the numerical approach for the Segmented Versine is chosen, and the experimental results are given in Figure 2.11 from [17]. As seen, after shaping the profile, energy in the corresponding resonant frequencies has been decreased and the resulting peak residual vibration becomes 0.211(m/s^2), and the settling time is about 0 seconds. In a nutshell, the command shaping method has demonstrated its effectiveness in removing energy content associated with resonant frequencies, and thus minimizing both residual vibration and the settling time.

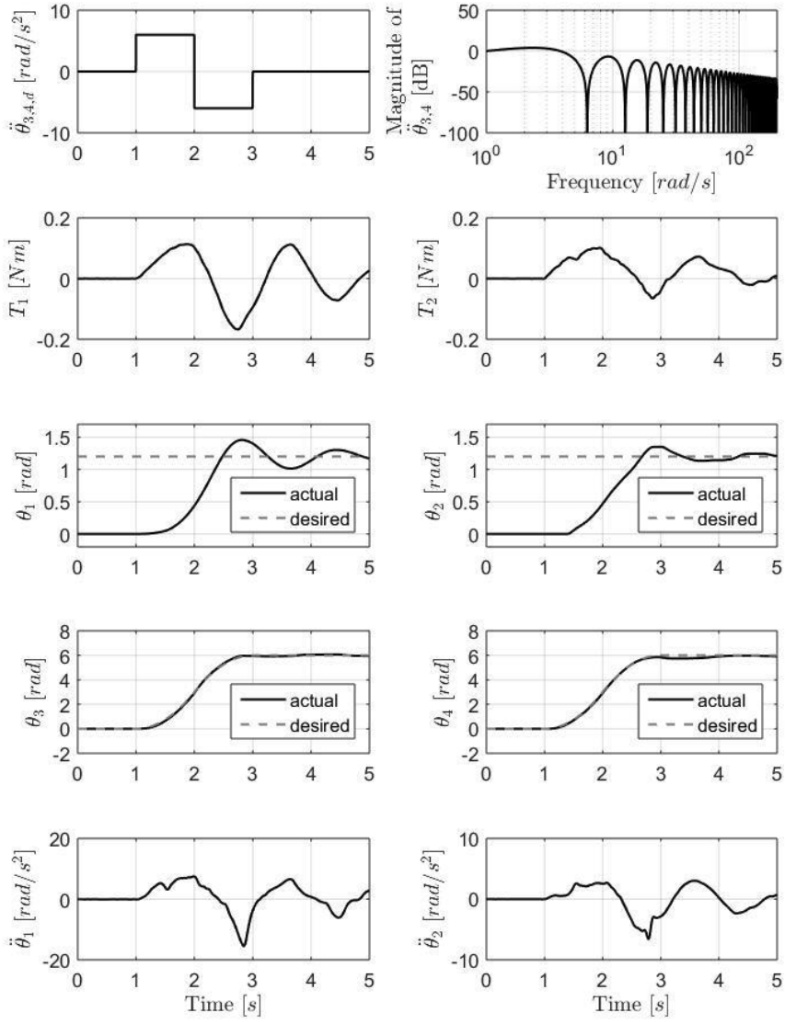


Figure 2.10. Experimental Results for the Bang-Bang Profile [17]

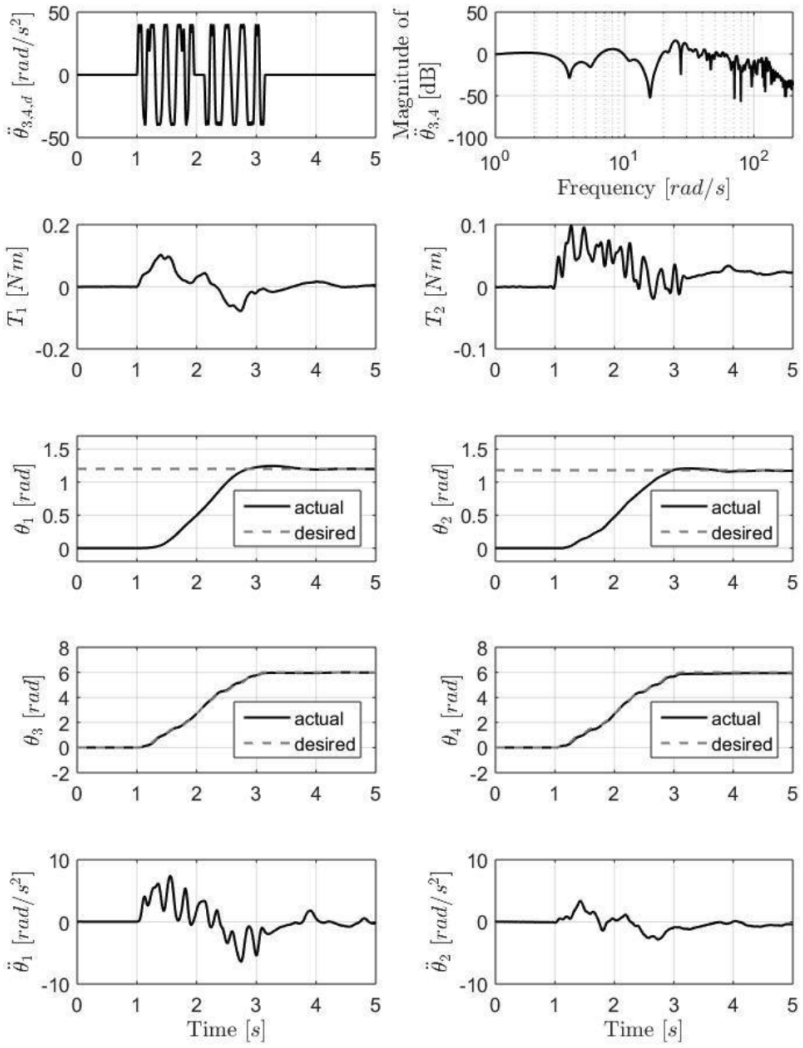


Figure 2.11. Experimental Response for Segmented Versine with Numerical Approach with Constraint Peak Acceleration $a_{max} = 40$, and Weighting Factor $\rho = 8750$ [17]

3. SYSTEM IDENTIFICATION

With the framework of the command shaping method and robot modeling discussed in Chapter 2, this chapter focuses on system identification of the two-link robotic arm with joint-flexibility. Since both the command shaping method and computed torque control, which is covered in Chapter 4.1, depends on models, it is necessary to precisely estimate each set of parameters that are essential in the shaping and control processes. Previous work in system identification done by Nho [37] used the classic Ordinary Least Squares regression approach. This open-loop approach estimates all parameters simultaneously. The OLS method, in general, leads to bias when its assumption does not hold in practice. Later, Lee [20] presented the framework for a new system identification method called nonlinear autoregressive moving average with exogenous inputs, or NARMAX. Better estimation was obtained based on this method but still limited to a range of torque inputs and working region for the robot. Scheel [22] used a new method to perform system identification on the robot. This approach has been widely called sequential system identification (ID) [21]. Instead of estimating all parameters at the same time, sequential system ID separates the process into several sequential stages and each of them depends on the results from the previous stage. In this way, parameters could be isolated and estimated separately so that minimum bias is produced. However, this approach relied on the classical ordinary least squares (OLS) method during the estimation. One strong assumption that OLS requires to avoid bias estimation is that noise in the measurement signals be normally distributed. In our case, the position signal is measured by an incremental encoder. Uncertainty between pulses that generate the position measurement is not necessarily normally distributed. Thus, the work in this Chapter presents an alternate approach that is applied from literature. It is called Instrumental Variable (IV) method.

3.1 Introduction of the IV approach

After applying the OLS method, a potential improvement in estimation can be obtained because a precise mathematical representation of the robot is known. Therefore, the IV approach utilizes this information and simulates the system response based on the estimated parameters from the OLS. Comparing the simulated response with the "real" system response, a new and improved set of estimates is produced by the IV approach. Usually the IV algorithm can be improved by iterating its process with previous estimated parameters but literature has shown that two to five iterations are enough for convergence. Figure 3.1 shows the process of integrating the IV algorithm with the classic OLS method. Note that the RLS is used in this work, which stands for recursive least squares. It is equivalent to the OLS method but it speeds up the estimation process.

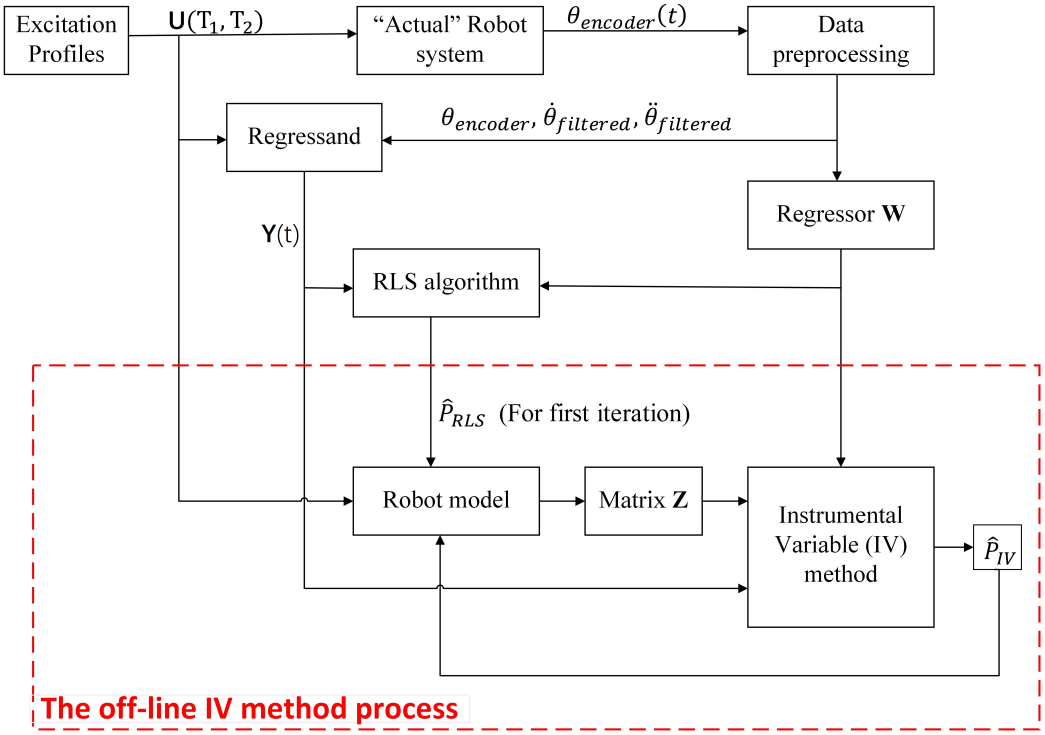


Figure 3.1. Integration of IV Approach and RLS Algorithm

3.2 Sequential Identification Process and Regression Model Formulation

As mentioned, the identification process involves sequential stages. Scheel [22] divided the entire estimation into three stages. The first stage would be purely identifying each motor's inertia and damping ratio. The second stage, with estimated parameters from stage one, requires fixing the shoulder link so that only the elbow is excited. Since only link two is movable, the robot dynamics, based on Equation 2.16, would be simplified. In the last stage, with estimates from both stage one and two, the entire robot would be excited, so that those dynamic properties from the elbow link could be estimated. Regression models are provided in Sections 3.2.1, 3.2.2, and 3.2.3, respectively, so as to initiate the RLS algorithm. The general form of the regression model can be written as

$$\mathbf{W}_{Sj}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \ddot{\boldsymbol{\theta}})\mathbf{P}_{Sj} = \mathbf{Y}_{Sj} \ , \tag{3.1}$$

where $\mathbf{W}_{Sj}$ denotes regressor matrix, $\mathbf{P}_{Sj}$ is the column vector of parameters to be estimated, and $\mathbf{Y}_{Sj}$ represents the regressand vector. Notation of Sj stands for j^{th} stage with $j = 1, 2, 3$ in this work. Simulation as well as experiment results from RLS and IV methods are given at the end of this Chapter. Appendix D provides more details and results.

3.2.1 Stage One: Motor Model

In stage one, only the motors are excited to estimate their dynamic properties. The current dynamics are assumed to be so fast as to be ignored. A simple model of a DC motor is given by the following equation

$$J_i\ddot{\theta}_i + c_i\dot{\theta}_i = T_j \, . \tag{3.2}$$

After rearranging the above equation, and converting it into the regression form, it becomes

$$\mathbf{W}_{S1}(\theta, \dot{\theta}, \ddot{\theta})\mathbf{P}_{S1} = \begin{bmatrix} \ddot{\theta}_i & \dot{\theta}_i \end{bmatrix} \begin{bmatrix} J_i \\ c_i \end{bmatrix} = \mathbf{Y}_{S1} = T_j \tag{3.3}$$

where $i = 3, 4$ denote the dynamic properties of motor one and motor two, respectively. To be consistent with the overall robot model, $j = 1, 2$ for torque input due to motor one and motor two, respectively. Again, notation of the generalized variable θ_i is consistent with the robot model. It is important to note that, during the experiment, measurements for motor one and motor two need to be conducted separately in order to isolate any unnecessary interference.

3.2.2 Stage Two: Elbow Link Model

As mentioned, with identified parameters from stage one, stage two needs to fix the shoulder link, that is, set the acceleration of the shoulder link to zero and motor one would not be included in the robot dynamics. As a result, based on Equation 2.16, a simplified equation of motion is given as

$$\mathbf{M}_{S2}\ddot{\boldsymbol{\theta}}_{S2} + \mathbf{C}_{S2}\dot{\boldsymbol{\theta}}_{S2} + \mathbf{K}_{S2}\boldsymbol{\theta}_{S2} = \mathbf{T}_{S2} \ , \tag{3.4}$$

where

$$\boldsymbol{\theta}_{S2} = \begin{bmatrix} \theta_2 \\ \theta_4 \end{bmatrix} \ , \tag{3.5}$$

the inertia matrix is

$$\mathbf{M}_{S2} = \begin{bmatrix} m_2 a_2^2 & 0 \\ 0 & J_4 + \frac{J_6}{r^2} \end{bmatrix} \ , \tag{3.6}$$

the damping matrix is

$$\mathbf{C}_{S2} = \begin{bmatrix} c_2 + c_6 & -\frac{c_6}{r} \\ -\frac{c_6}{r} & c_4 + \frac{c_6}{r^2} \end{bmatrix} \ , \tag{3.7}$$

the stiffness matrix becomes

$$\mathbf{K}_{S2} = \begin{bmatrix} k_6 & -\frac{k_6}{r} \\ -\frac{k_6}{r} & \frac{k_6}{r^2} \end{bmatrix} \ , \tag{3.8}$$

and the torque input is

$$\mathbf{T}_{S2} = \begin{bmatrix} 0 \\ T_2 \end{bmatrix} \ . \tag{3.9}$$

Note that J_3, J_4, c_3 and c_4 are treated as known variables because they have been identified in stage one. Rearranging Equation 3.4 and putting it into the regression form, one can obtain the following equation

$$\mathbf{W}_{S2}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \ddot{\boldsymbol{\theta}})\mathbf{P}_{S2} = \mathbf{Y}_{S2} \ , \tag{3.10}$$

where the regressor matrix is given by

$$\mathbf{W}_{S2} = \begin{bmatrix} \ddot{\theta}_2 & 0 & \dot{\theta}_2 & \dot{\theta}_2 - \frac{\dot{\theta}_4}{r} & \theta_2 - \frac{\theta_4}{r} \\ 0 & \frac{\ddot{\theta}_4}{r^2} & 0 & \frac{\dot{\theta}_4}{r^2} - \frac{\dot{\theta}_2}{r} & \frac{\theta_4}{r^2} - \frac{\theta_2}{r} \end{bmatrix} \ , \tag{3.11}$$

the regressand is

$$\mathbf{Y}_{S2} = \begin{bmatrix} 0 \\ T_2 - J_4\ddot{\theta}_4 - c_4\dot{\theta} \end{bmatrix} \ , \tag{3.12}$$

and the parameter matrix to be estimated is

$$\mathbf{P}_{S2} = \begin{bmatrix} m_2a_2^2 + J_2 \\ J_6 \\ c_2 \\ c_6 \\ k_6 \end{bmatrix} \ . \tag{3.13}$$

It is worth pointing out that, even though the robot model is inherently nonlinear, the system model is linear in the parameters. That is to say, the general approach of ordinary least squares is applicable here. In addition, not all of the individual parameters can be isolated and determined. Instead, some elements in matrix $\mathbf{P}_{Sj}$, for example, $m_2a_2^2 + J_2$, consist of more than one independent mechanical properties. In this case, the quantity of the combined parameter $m_2a_2^2 + J_2$ is estimated rather than obtaining an individual numerical value for each parameter in $m_2a_2^2 + J_2$.

3.2.3 Stage Three: Full Robot Model

In the last stage, with parameters estimated from stages one and two, the entire robot is actuated and excited to perform identification for the remaining parameters.

At this time, Equation 2.16 is converted into its equivalent regression form and written as

$$\mathbf{W}_{S3}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, \ddot{\boldsymbol{\theta}})\mathbf{P}_{S3} = \mathbf{Y}_{S3} \ , \tag{3.14}$$

where the regression is

$$\mathbf{W}_{S3} = \begin{bmatrix} \ddot{\theta}_1 & w_{12} & \dot{\theta}_1 & \dot{\theta}_1 - \frac{\dot{\theta}_3}{r} & \theta_1 - \frac{\theta_1}{r} & 0 \\ 0 & w_{22} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\dot{\theta}_3}{r^2} - \frac{\dot{\theta}_1}{r} & \frac{\theta_3}{r^2} - \frac{\theta_1}{r} & \frac{\ddot{\theta}_3}{r^2} \end{bmatrix} \ , \tag{3.15}$$

in which

$$w_{12} = \cos(\theta_2)\ddot{\theta}_2 - (2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2)\sin(\theta_2) + 2\cos(\theta_2)\ddot{\theta}_1 \tag{3.16}$$

$$w_{22} = \cos(\theta_2)\ddot{\theta}_1 + \sin(\theta_2)\dot{\theta}_1^2 \ . \tag{3.17}$$

The corresponding regressand vector is defined as

$$\mathbf{Y}_{S3} = \begin{bmatrix} -(m_2 a_2^2 + J_2)(\ddot{\theta}_1 + \ddot{\theta}_2) - (J_4 + \frac{J_6}{r})\ddot{\theta}_4 \\ r_{L1} \\ T_1 - c_3\dot{\theta}_3 - J_3\ddot{\theta}_3 \end{bmatrix} \ , \tag{3.18}$$

where

$$\begin{aligned} r_{L1} = &-(m_2 a_2^2 + J_2)(\ddot{\theta}_1 + \ddot{\theta}_2) - (c_2 + c_6)\dot{\theta}_2 \\ &+\frac{c_6}{r}\dot{\theta}_4 - k_6\theta_2 + \frac{k_6}{r}\theta_4 \ , \end{aligned} \tag{3.19}$$

and the parameter vector is defined as

$$\mathbf{P}_{S3} = \begin{bmatrix} p_1 \\ l_1 m_2 a_2 \\ c_1 \\ c_5 \\ k_5 \\ J_5 \end{bmatrix} \ , \tag{3.20}$$

where

$$p_1 = m_1 a_1^2 + m_2 l_1^2 + m_4 b_1^2 + m_6 l_1^2 + J_1 + J_4 + J_6 \ . \tag{3.21}$$

As seen, the first two elements of parameter vector $\mathbf{P}_{S3}$ are composed of several mechanical properties rather than a single parameter. Therefore, the overall quantity of each of these two combined parameters is estimated.

3.3 Recursive Least Squares Method

With a defined regression model for each stage, recursive least squares estimation can be applied to obtain the numerical estimation for parameter vectors $\mathbf{P}_{Sj}$. To implement, the following recursive equations are used

$$\hat{\mathbf{P}}_{Sj,RLS}(k) = \hat{\mathbf{P}}_{Sj,RLS}(k-1) + \mathbf{K}_{RLS}(k)[\mathbf{Y}_{Sj}(k) - \mathbf{W}_{Sj}(k)\hat{\mathbf{P}}_{Sj,RLS}(k-1)], \quad (3.22)$$

$$\mathbf{K}_{RLS}(k) = \mathbf{P}_{Var}(k-1)\mathbf{W}_{Sj}^{T}(k)[\lambda\mathbf{\Lambda} + \mathbf{W}_{Sj}(k)\mathbf{P}_{Var}(k-1)\mathbf{W}_{Sj}^{T}(k)]^{-1} \quad (3.23)$$

and

$$\mathbf{P}_{Var}(k) = \frac{1}{\lambda}[\mathbf{I}_{m\times m} - \mathbf{K}_{RLS}(k)\mathbf{W}_{Sj}(k)]\mathbf{P}_{Var}(k-1)\ , \quad (3.24)$$

where the dimension of regression matrix $\mathbf{W}_{Sj}$ is n by m, the estimated parameter vector $\hat{\mathbf{P}}_{Sj,RLS}$ is m by 1, the regressand vector $\mathbf{Y}_{Sj}$ is n by 1, the correcting matrix $\mathbf{K}_{RLS}$ is m by n, and the variance matrix $\mathbf{P}_{Var}$ is m by m. $\mathbf{\Lambda}$ is the weighting matrix and is set as the identity matrix with size n in this work. λ is the forgetting factor that is always used for a time-varying system. However, in this work, $\lambda = 1$ because the system parameters remain unchanged during the identification process. Subscript "RLS" of $\hat{\mathbf{P}}_{Sj,RLS}$ indicates that the parameter vector is estimated by using the RLS algorithm. Since this algorithm is a recursive process, the time stamp is denoted by $k = 1, 2, ...$ To initialize the algorithm, $\mathbf{P}_{Var}(0) = 10^{6}\mathbf{I}_{m\times m}$, and $\hat{\mathbf{P}}_{Sj,RLS}(0) = \mathbf{0}$.

3.4 Instrumental Variable Method

Because the structure of noise and its probability distribution are unknown, the RLS method yields biased estimates [28, p. 382]. To overcome this issue, a solution,

called Instrumental Variable (IV), is applied from [29, p. 486]. The IV approach is defined by the following equations

$$\hat{\mathbf{P}}_{Sj,IV} = (\mathbf{\Gamma}_{Sj}^T \mathbf{\Pi}_{Sj})^{-1} \mathbf{\Gamma}_{Sj}^T \mathbf{\Psi}_{Sj} \ , \tag{3.25}$$

where

$$\mathbf{\Gamma}_{Sj} = \left[\mathbf{Z}_{Sj}^T(k=1), ..., \mathbf{Z}_{Sj}^T(k=N\Delta T) \right]^T , \tag{3.26}$$

$$\mathbf{\Pi}_{Sj} = \left[\mathbf{W}_{Sj}^T(k=1), ..., \mathbf{W}_{Sj}^T(k=N\Delta T) \right]^T , \tag{3.27}$$

and

$$\mathbf{\Psi}_{Sj} = \left[\mathbf{Y}_{Sj}^T(k=1), ..., \mathbf{Y}_{Sj}^T(k=N\Delta T) \right]^T \ , \tag{3.28}$$

where $N\Delta T$ denotes the last time stamp, and the subscript "IV" of $\hat{\mathbf{P}}_{Sj,IV}$ indicates that the parameter vector is estimated by the IV approach. As seen in Equation 3.25, system parameters at each stage are now estimated with the help of instrumental matrix $\mathbf{\Gamma}_{Sj}$ whose columns are called instrumental variables. To have optimized estimation, it is essential to construct the matrix $\mathbf{\Gamma}_{Sj}$ such that it is highly correlated with the stacked regression matrix $\mathbf{\Pi}_{Sj}$ and uncorrelated with the estimation residual from the RLS algorithm [29, p. 486]. To fulfill these requirements, in this work, the $\mathbf{\Gamma}_{Sj}$ matrix is constructed by using uncorrupted signals simulated based on estimates $\hat{\mathbf{P}}_{Sj,RLS}$ for the first iteration and based on the previous $\hat{\mathbf{P}}_{Sj,IV}$ for the rest of the iterations. To initiate the IV method at first iteration, it is important to have stable estimates so that matrix $\mathbf{\Pi}_{Sj}$ does not contain unstable system responses. Experience and literature shows that convergence occurs very fast and usually 2 to 5 iterations are enough. Figure 3.1 shows the overall estimation process after integrating the RLS algorithm with the IV method.

While $\mathbf{W}_{Sj}$ and $\mathbf{Y}_{Sj}$ have been defined previously in each stage, the simulated regression matrix $\mathbf{Z}_{Sj}$ at each stage is written as

$$\mathbf{Z}_{S1} = \begin{bmatrix} \ddot{\theta}_{sim,i} & \dot{\theta}_{sim,i} \end{bmatrix} \ , \tag{3.29}$$

where $i = 3$ for the first motor and $i = 4$ for the second motor,

$$\mathbf{Z}_{S2} = \begin{bmatrix} \ddot{\theta}_{sim,2} & 0 & \dot{\theta}_{sim,2} & \dot{\theta}_{sim,2} - \frac{\dot{\theta}_{sim,4}}{r} & \theta_{sim,2} - \frac{\theta_{sim,4}}{r} \\ 0 & \frac{\ddot{\theta}_{sim,4}}{r^2} & 0 & \frac{\dot{\theta}_{sim,4}}{r^2} - \frac{\dot{\theta}_{sim,2}}{r} & \frac{\theta_{sim,4}}{r^2} - \frac{\theta_{sim,2}}{r} \end{bmatrix} \tag{3.30}$$

$$\mathbf{Z}_{S3} = \begin{bmatrix} \ddot{\theta}_1 & z_{sim,12} & \dot{\theta}_{sim,1} & \dot{\theta}_{sim,1} - \frac{\dot{\theta}_{sim,3}}{r} & \theta_{sim,1} - \frac{\theta_{sim,1}}{r} & 0 \\ 0 & z_{sim,22} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\dot{\theta}_{sim,3}}{r^2} - \frac{\dot{\theta}_{sim,1}}{r} & \frac{\theta_{sim,3}}{r^2} - \frac{\theta_{sim,1}}{r} & \frac{\ddot{\theta}_{sim,3}}{r^2} \end{bmatrix} , \tag{3.31}$$

where

$$\begin{aligned} z_{sim,12} = \cos(\theta_{sim,2})\ddot{\theta}_{sim,2} - (2\dot{\theta}_{sim,1}\dot{\theta}_{sim,2} \\ +\dot{\theta}^2_{sim,2})\sin(\theta_{sim,2}) + 2\cos(\theta_{sim,2})\ddot{\theta}_{sim,1} \end{aligned} \tag{3.32}$$

$$z_{sim,22} = \cos(\theta_{sim,2})\ddot{\theta}_{sim,1} + \sin(\theta_{sim,2})\dot{\theta}^2_{sim,1} \ . \tag{3.33}$$

The subscript "sim" emphasizes that the generalized variable $\theta_{sim,i}$ is obtained by simulating the system response based on the corresponding mechanical model at each stage. Strictly speaking, Equations 3.2, 3.4, and 2.16 are used to simulate $\theta_{sim,i}$ for stages one, two, and three, respectively.

3.5 Simulation Results

Selected simulation results are presented in this Section. It is worth pointing out that the noise is introduced when obtaining the angular velocity and acceleration signals, which are calculated by the finite difference method. By definition, this numerical method can be written as

$$\dot{\theta}_{1,2,3,4}(k\Delta T) = \frac{\delta\theta_{1,2,3,4}}{\delta t}|_{t=k\Delta T} \approx \frac{\theta_{1,2,3,4}(k\Delta T) - \theta_{1,2,3,4}((k-1)\Delta T)}{\Delta T} \ , \tag{3.34}$$

where $k = 1, 2, 3, ...$ denotes the discrete time stamp, and ΔT is the sampling period. Similarly, the angular acceleration can be obtained by using the following equation:

$$\ddot{\theta}_{1,2,3,4}(k\Delta T) = \frac{\delta\dot{\theta}_{1,2,3,4}}{\delta t}|_{t=k\Delta T} \approx \frac{\dot{\theta}_{1,2,3,4}(k\Delta T) - \dot{\theta}_{1,2,3,4}((k-1)\Delta T)}{\Delta T} \ . \tag{3.35}$$

Noise due to limited resolution of encoders and quantization from ADC and DAC units is also included in this simulation program. Figure 3.2 gives the excitation

profiles for motor one and motor two, respectively. For simplicity, excitation profiles are the same over the three stages. In general, the profiles are generated by using the following equation

$$T_{sine,1} = T_{sine,2} = T_{max} \sum_{k=1}^{N_{sine}} \sin(2\pi f_k t) \ , \tag{3.36}$$

where $N_{sine} = 9$ is the number of frequency components used, $T_{max} = 0.13$(Nm) is the scaling factor to limit torque output from each motor, and f_k(Hz) $\in \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ with $f_i = f_j$ iff $i = j$.

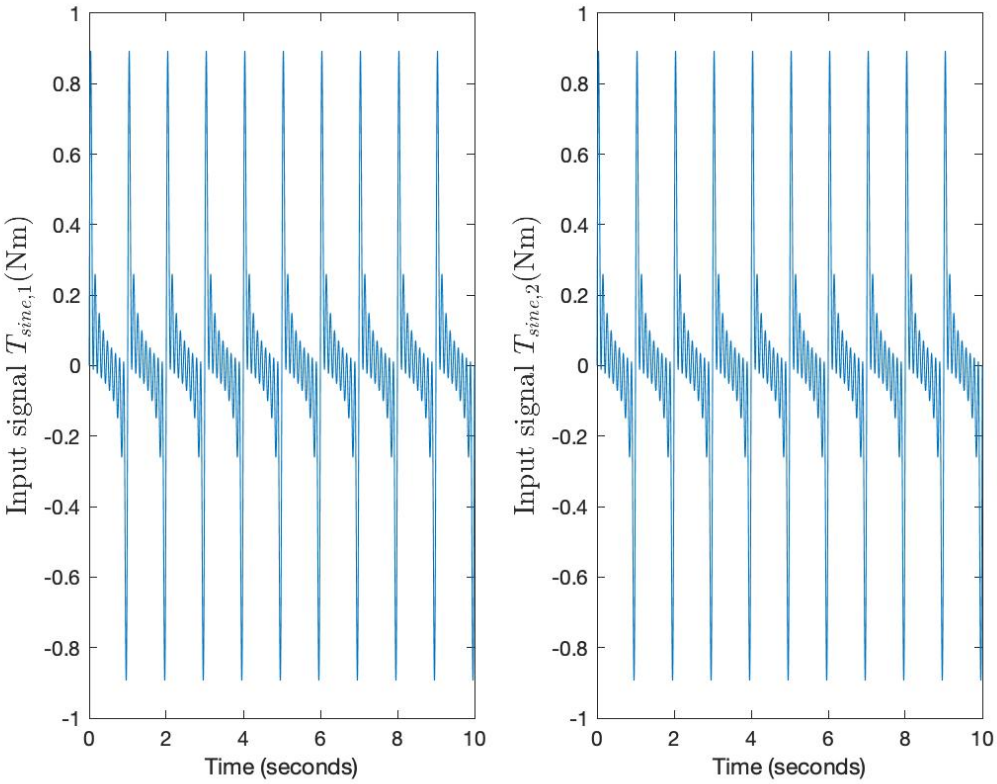


Figure 3.2. Excitation Signal Profiles for Motor One (left) and Motor Two (right)

With the defined excitation profiles, the simulated system responses from Stages One to Three are shown in Figures 3.3 to 3.5, respectively.

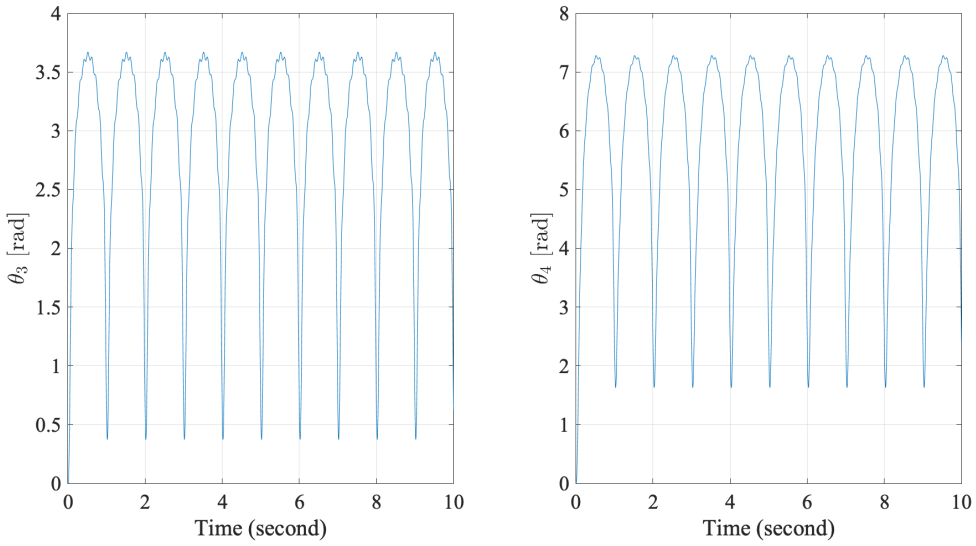


Figure 3.3. System Response at Stage One

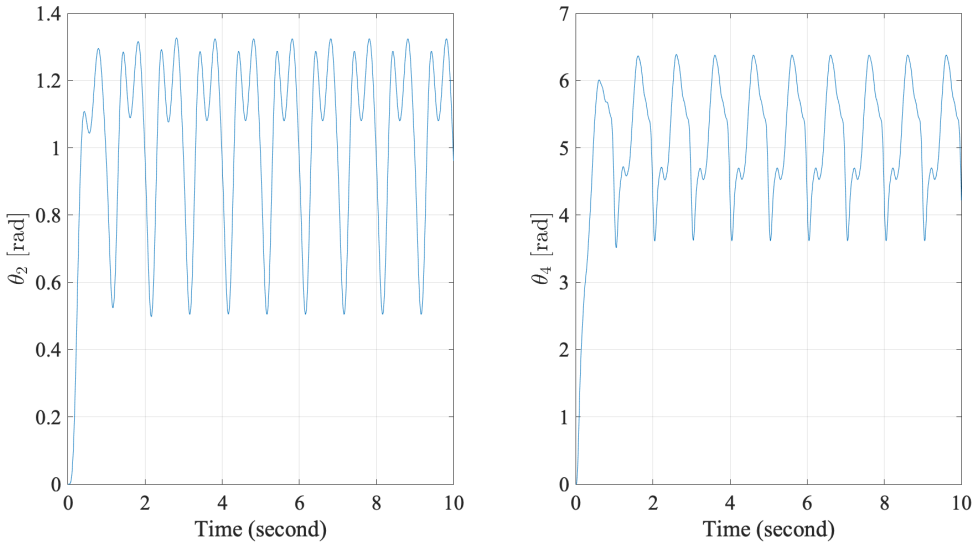


Figure 3.4. System Response at Stage Two

After applying the RLS algorithm and the IV approach, the estimated parameters are obtained. Figures 3.6 to 3.8 and Table 3.1 show the summarized results by

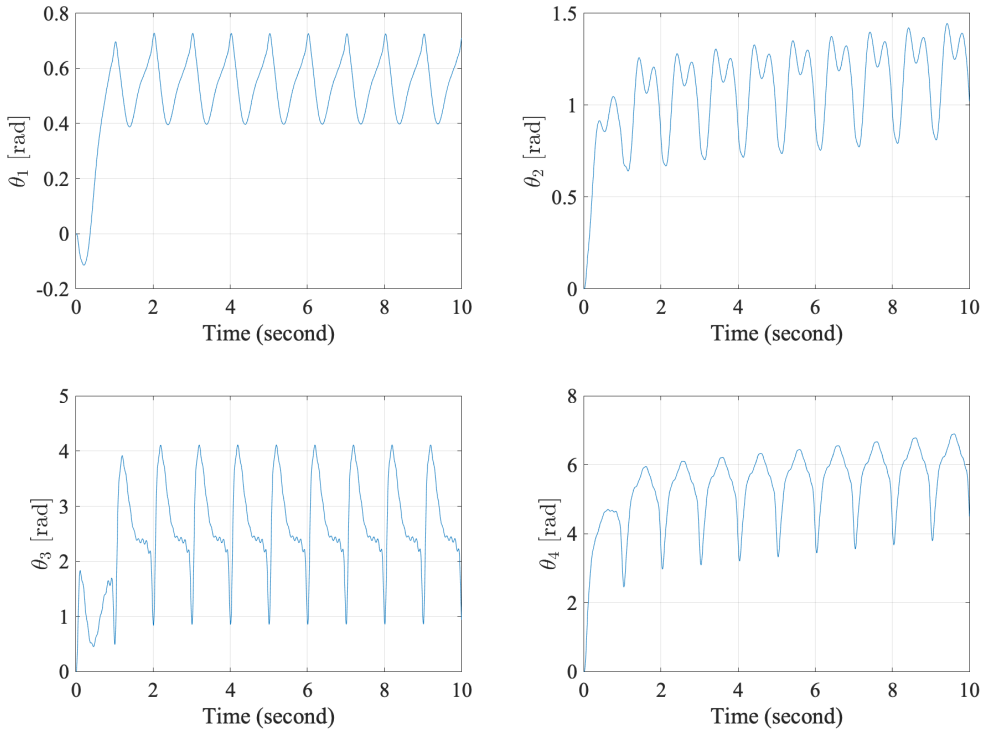


Figure 3.5. System Response at Stage Three

comparing the error percentage of estimated parameters. The error percentage is defined by the following equation

$$\text{Error Percentage} = \frac{p_{def} - p_{est}}{p_{def}} 100\% \ , \tag{3.37}$$

where p_{def} is the defined parameter value used in the simulation and listed as a title on each subplot from Figures 3.6 to 3.8, and p_{est} is the estimated parameter value either based on the RLS algorithm or the IV approach. In this work, five iterations have been run for the IV approach. Each title of the subplots defines the corresponding numerical value of a parameter that is used in the simulation program. Note that 0^{th} iteration in these figures represents error percentage due to the classic RLS algorithm, which is used to initiate the IV approach.

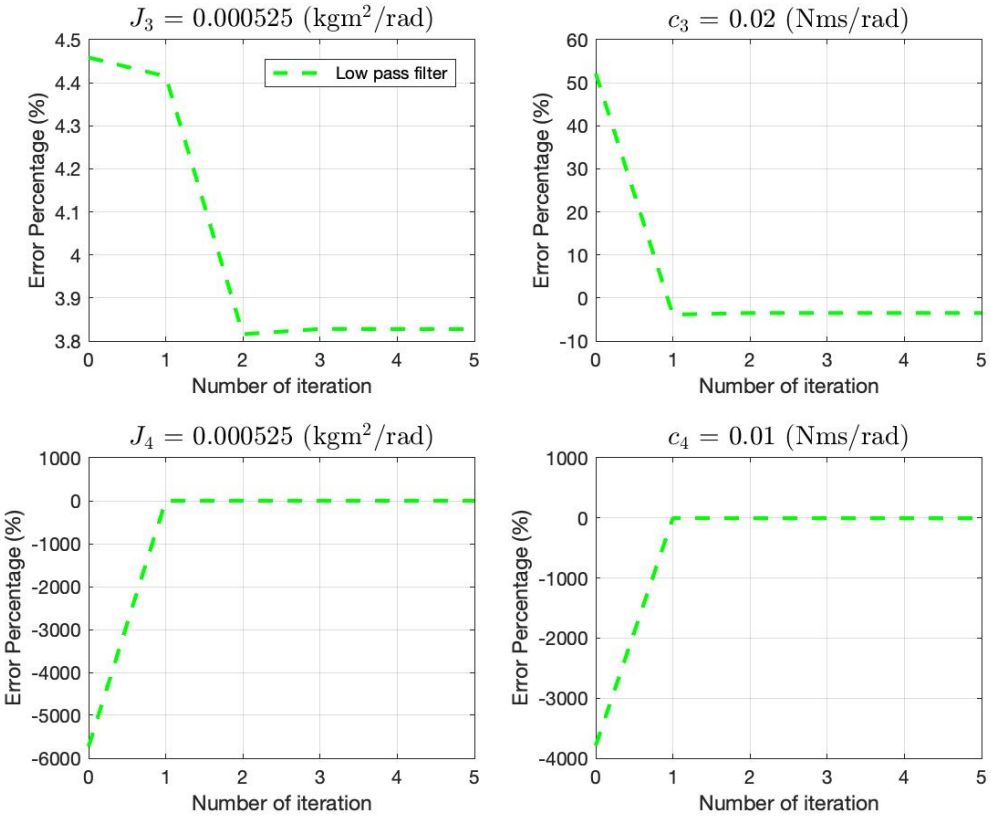


Figure 3.6. Error Percentages of Estimated Parameters for Stage One

It can be seen that convergence occurs after about five iterations, which confirms comments in the literature. In general, estimation quality has been improved significantly by the IV approach. Based on the simulated results, if RLS yields large-bias estimation, the IV approach gains relatively large momentum to correct the bias and $\hat{\mathbf{P}}_{Sj,IV}$ is able to jump into a relatively small error bound. Estimation errors for parameters c_5 and c_6 from Table 3.1 are relatively larger than for other parameter estimates. This is possibly due to their relatively small numerical values defined in the simulation. It is worth pointing out that to initiate the IV approach, it is necessary to have stable signals which form the simulated regression matrix $\mathbf{Z}_{Sj}$. Besides, both RLS and the IV are sensitive to data preprocessing. For example, to reduce the

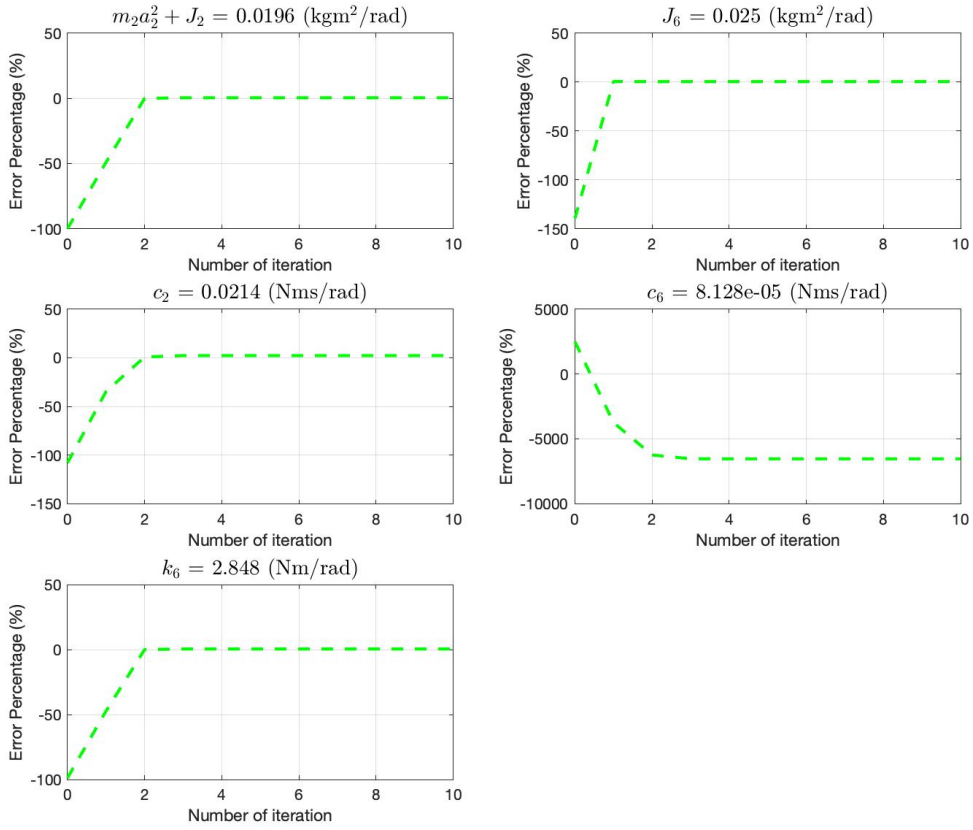


Figure 3.7. Error Percentages of Estimated Parameters for Stage Two

noise level contained in the signals, a common practice is to include filters during the data preprocessing. In this work, low-pass filters have been adopted in handling both angular velocity and acceleration signals that are obtained directly from the encoder signals. It is found that different cut-off frequencies used by the low-pass filter can greatly affect overall estimation accuracy. Again, more comparisons can be seen in Appendix D.

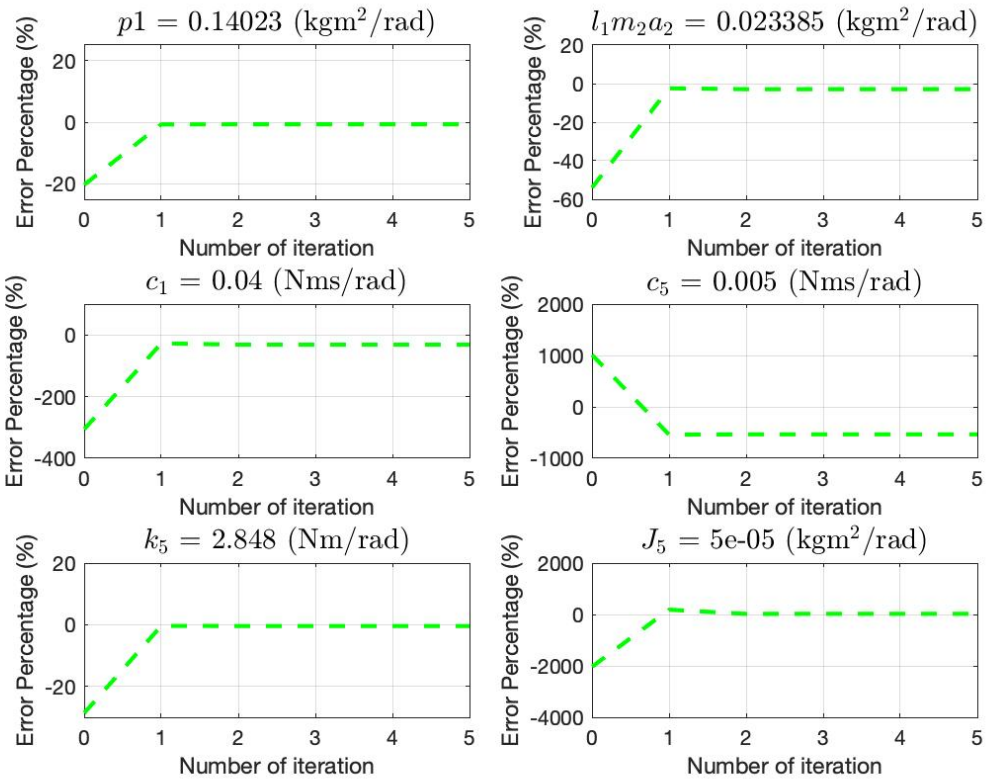


Figure 3.8. Error Percentages of Estimated Parameters for Stage Three

3.6 Experimental Results

In this Section, experimental results from Stage One are provided. Figure 3.9 shows the experimental excitation profiles for motor one and motor two, respectively. After applying the RLS algorithm and the IV approach, the estimated parameter values are shown in Figure 3.10. In this figure, similar to Figure 3.6, the 0^{th} iteration denotes the estimated parameter from the RLS algorithm which is used to initiate the IV approach. Since convergence in the IV approach is observed, only five iterations are used to obtain the final estimated parameter for the IV approach. The corresponding parameter values are listed in Table 3.2.

Table 3.1. Error Percentages of Estimated Parameters from Simulation

Stage	Parameters	RLS	IV method
Stage One	J_3	4.459%	3.828%
	c_3	52.27%	-3.498%
	J_4	−5754%	1.958%
	c_4	-3857%	-4.745%
Stage Two	$m_2a_2^2 + J_2$	-100%	0.4911%
	J_6	-139.6%	0.4951%
	c_2	-108.1%	2.186%
	c_6	2487%	-6535%
	k_6	-98.81%	0.5742%
Stage Three	p_1	-20.31%	-0.6837%
	$l_1m_2a_2$	-54%	-2.894%
	c_1	-306.1%	-31.56%
	c_5	1013%	-534.7%
	k_5	-28.64%	-0.4043%
	J_5	−2034%	25.94%

Table 3.2. Estimated Parameter Values from Experiment

Stage	Parameters	RLS	IV method
Stage One	J_3 $(\frac{\text{kgm}^2}{\text{rad}})$	3.78×10^{-5}	4.68×10^{-5}
	c_3 $(\frac{\text{Nms}}{\text{rad}})$	16.84×10^{-5}	17.95×10^{-5}
	J_4 $(\frac{\text{kgm}^2}{\text{rad}})$	72.03×10^{-5}	81.14×10^{-5}
	c_4 $(\frac{\text{Nms}}{\text{rad}})$	129.26×10^{-5}	139.09×10^{-5}

To validate the estimated parameters for each motor, based on their values, the simulated responses as well as the experimental responses are shown in Figures 3.11 and 3.12. In these figures, the solid line represents the experimental responses of each

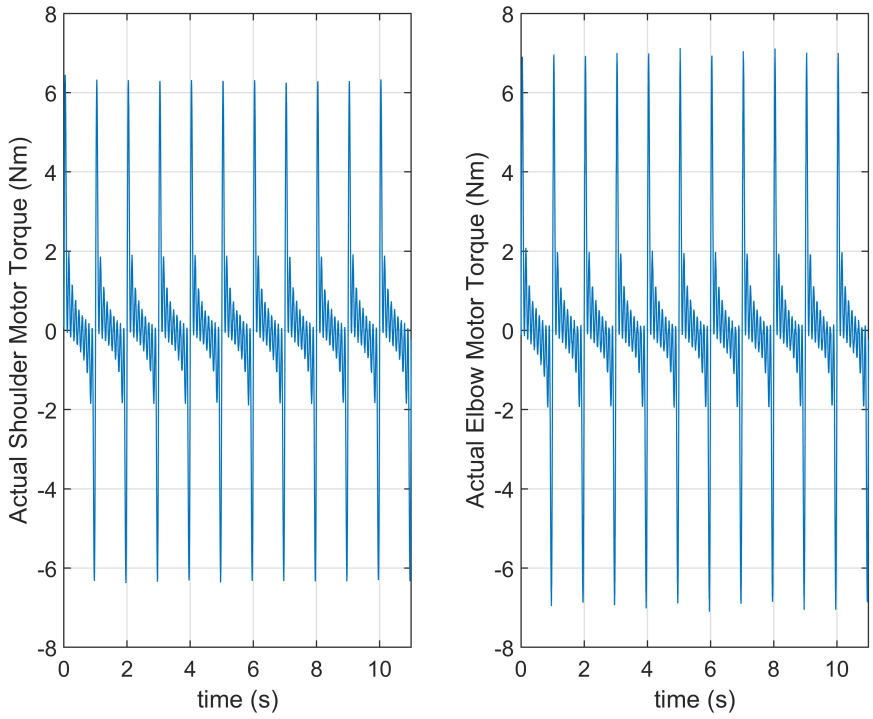


Figure 3.9. Experimental Excitation Torque Profiles for Motor One (left) and Motor Two(right)

motor while the dashed line denotes the simulated responses using the estimated parameters from the RLS algorithm or the IV approach.

To quantify the difference between the experimental response and the simulated response, the dimensionless trajectory error equation is used, defined as

$$e^* = \frac{1}{N}\sum_{i=0}^{N}\frac{|\theta_{exp}(i) - \theta_{est}(i)|}{\text{amp}(\theta_{exp})}100\% \ , \tag{3.38}$$

where θ_{exp} is the time history of the angular position measured from the experiments, θ_{est} is the simulated trajectory response based on the estimated parameters by using the RLS or the IV method, and N is the sample size. In this equation, the operation $\text{amp}(\cdot) = \max(\cdot) - \min(\cdot)$, which essentially takes the peak-to-peak value of a signal over time. Table 3.3 summarizes the dimensionless error. From this table, it can be seen that the IV method results in a smaller magnitude of the dimensionless trajectory

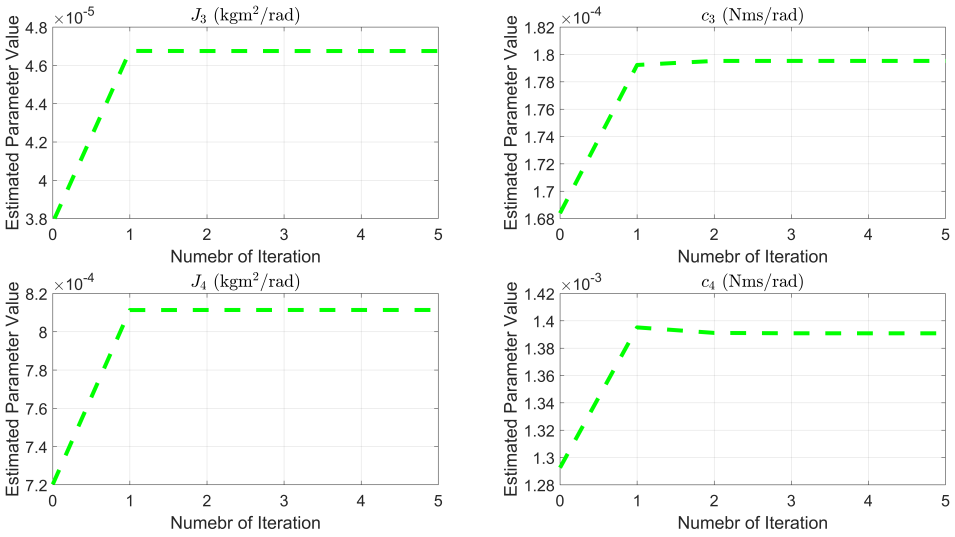


Figure 3.10. Estimated Parameter Values from Experiment for Motor One and Motor Two

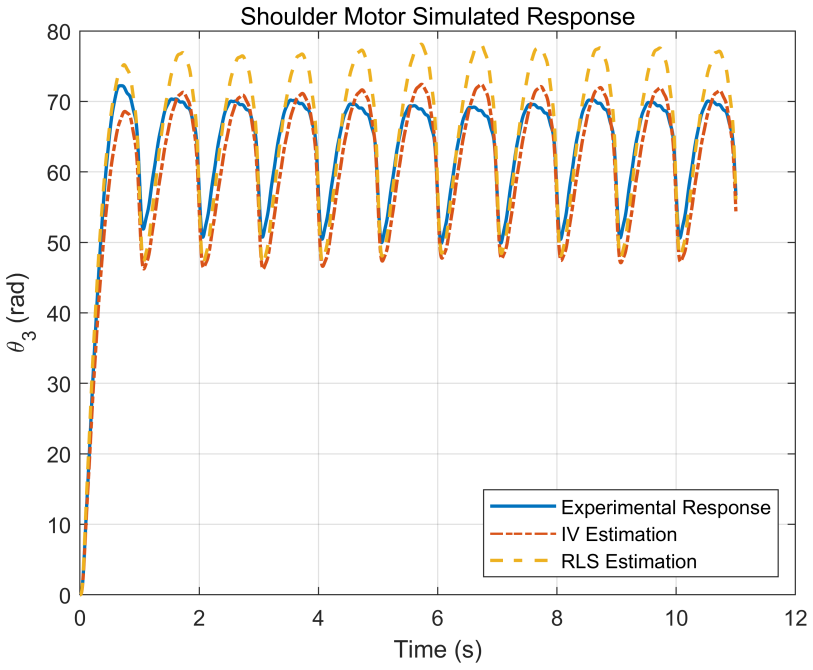


Figure 3.11. Experimental and Simulated Responses for Motor One

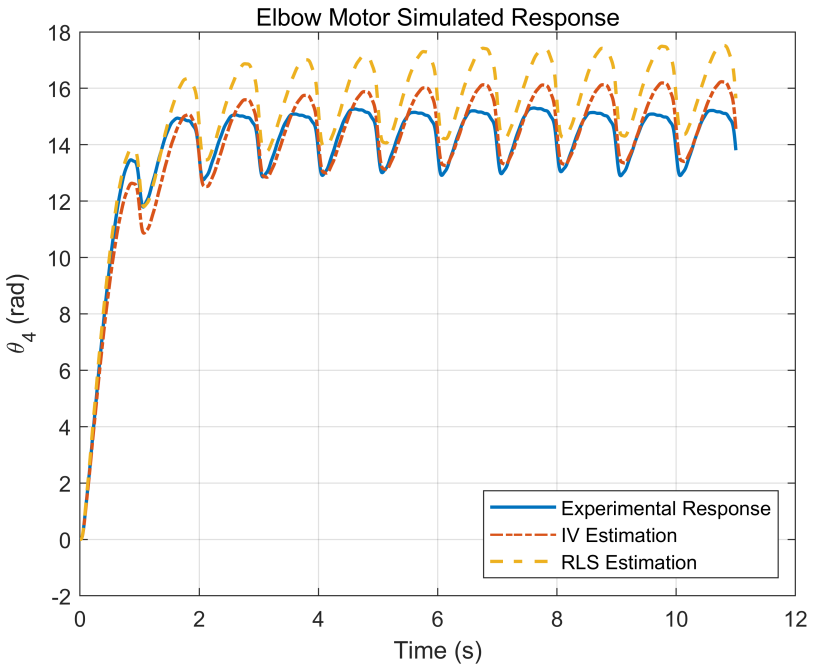


Figure 3.12. Experimental and Simulated Responses for Motor Two

Table 3.3. Dimensionless Trajectory Errors for RLS and IV methods

Stage	Dimensionless Errors	RLS	IV method
Stage One	$e^*_{\theta_3}$	5.24%	4.28%
	$e^*_{\theta_4}$	8.36%	3.13%

error for both motors. That is to say, the simulated response based on the IV method is relatively better in terms of reconstructing the motor signal. As a result, the IV approach improves the estimated parameters from the RLS algorithm.

In conclusion, simulation results have shown that integrating the IV approach with the classic RLS could improve overall estimation accuracy when the noise in measurement signals has unknown distribution. Validation based on experimental results of Stage One is provided and also shows that a considerable improvement has been achieved by using the IV method.

4. SYNTHESIS OF COMPUTED TORQUE CONTROLLER WITH COMMAND SHAPING

Feedback control has been widely used in robotics applications for its ability to ensure tracking accuracy, robustness, and to reject disturbance. Computed Torque Control is one of the popular controllers for robots and is used throughout this work. When it is combined with the command shaping method, it is of interest to understand the interaction between this controller and the quality of minimizing residual vibration using the command shaping method.

In this Chapter, Sections 4.1 and 4.2 provide the control scheme and the closed-loop stability analysis, respectively. Section 4.3 analyzes the closed-loop resonant frequency for the perfectly-decoupled system. Section 4.4 provides the calculation of the closed-loop resonant frequency when estimation error exists in the model-based compensator. At the end, simulation results are presented in Section 4.5 for both the decoupled system and the mismatched model. Summary and recommendations for choosing control gains are in Section 4.6.

4.1 Computed Torque Control

Essentially, the Computed Torque control consists of a model-based compensator and a servo-based controller. The model-based compensator removes the coupling effect between the robot links and motors based on the estimated parameters. The servo-based part, on the other hand, uses Proportional-Derivative (PD) controllers to ensure asymptotic stability and tracking accuracy. Figure 4.1 shows a block diagram when this controller is integrated with the robot system. As seen, the control law is written as

$$\mathbf{T}_{motor} = \mathbf{T}_{mb} + \mathbf{T}_{sb}\,, \tag{4.1}$$

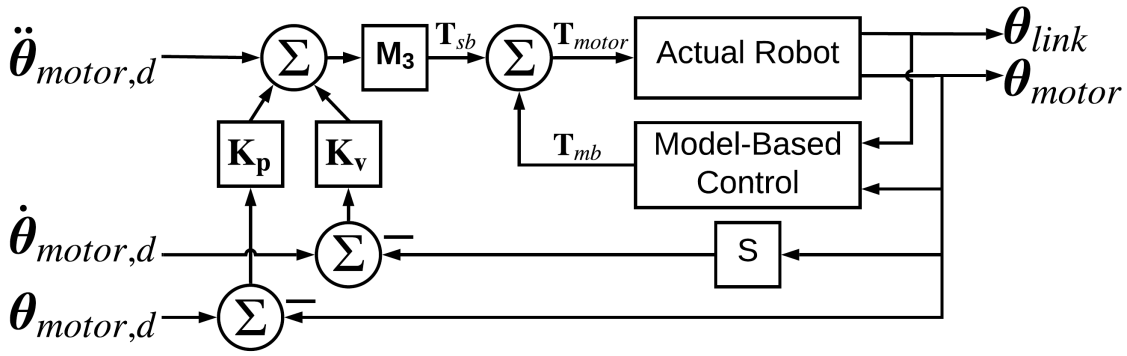


Figure 4.1. Block Diagram of the Computed Torque Controller Integrated with the Robotic Arm

where $\mathbf{T}_{motor}$, defined in Equation 2.26, is a column vector consisting of torque outputs from motor one and motor two, respectively. $\mathbf{T}_{mb}$ denotes the calculated torque amount from the model-based compensator and $\mathbf{T}_{sb}$ is the control torque from the servo-based part. By definition, model-based control torque $\mathbf{T}_{mb}$ is obtained from the following equation

$$\mathbf{T}_{mb} = \hat{\mathbf{M}}_{mb}\ddot{\hat{\boldsymbol{\theta}}}_{link} + \hat{\mathbf{C}}_{mb}\dot{\hat{\boldsymbol{\theta}}} + \hat{\mathbf{K}}_{mb}\hat{\boldsymbol{\theta}} + \hat{\mathbf{V}}_{mb}(\hat{\boldsymbol{\theta}}, \dot{\hat{\boldsymbol{\theta}}}) + \mathbf{D}_{mb}(\hat{\boldsymbol{\theta}}) \ . \tag{4.2}$$

Since $\mathbf{V}_{motor} = \mathbf{0}$ from Equation 2.21, the estimation of $\hat{\mathbf{V}}_{mb}$ becomes zero. Moreover, the Coulomb friction $\mathbf{D}_{mb}$ is assumed to be zero and hence the model-based control can be further simplified and written as

$$\mathbf{T}_{mb} = \hat{\mathbf{M}}_{mb}\ddot{\hat{\boldsymbol{\theta}}}_{link} + \hat{\mathbf{C}}_{mb}\dot{\hat{\boldsymbol{\theta}}} + \hat{\mathbf{K}}_{mb}\hat{\boldsymbol{\theta}} \tag{4.3}$$

where

$$\begin{aligned} \hat{\mathbf{M}}_{mb} &= \mathbf{M}_2^T - \tilde{\mathbf{M}}_{mb} \\ \hat{\mathbf{C}}_{mb} &= \mathbf{C}_{mb} - \tilde{\mathbf{C}}_{mb} \\ \hat{\mathbf{K}}_{mb} &= \mathbf{K}_{mb} - \tilde{\mathbf{K}}_{mb} \end{aligned} \tag{4.4}$$

and

$$\mathbf{C}_{mb} = \begin{bmatrix} -\frac{c_5}{r} & 0 & c_3 + \frac{c_5}{r^2} & 0 \\ 0 & -\frac{c_6}{r} & 0 & c_4 + \frac{c_5}{r^2} \end{bmatrix}, \tag{4.5}$$

$$\mathbf{K}_{mb} = \begin{bmatrix} -\frac{k_5}{r} & 0 & \frac{k_5}{r^2} & 0 \\ 0 & -\frac{k_6}{r} & 0 & \frac{k_6}{r^2} \end{bmatrix} . \tag{4.6}$$

For convenience, the estimated system property and estimated signals are denoted with the symbol "ˆ" while the error between the estimated and actual system is denoted by the symbol "˜". When estimation error vanishes, the estimated parameters are exactly the same as those of the actual system so that Equation 4.3 becomes

$$\mathbf{T}_{mb} = \mathbf{M}_2^T \hat{\ddot{\boldsymbol{\theta}}}_{link} + \mathbf{C}_{mb} \hat{\dot{\boldsymbol{\theta}}} + \mathbf{K}_{mb} \hat{\boldsymbol{\theta}} . \tag{4.7}$$

As for the servo-based part, $\mathbf{T}_{sb}$ is defined in the following equation

$$\mathbf{T}_{sb} = \hat{\mathbf{M}}_3 \left(\ddot{\boldsymbol{\theta}}_{motor,d} + \mathbf{K_v}(\dot{\boldsymbol{\theta}}_{motor,d} - \hat{\dot{\boldsymbol{\theta}}}_{motor}) + \mathbf{K_p}(\boldsymbol{\theta}_{motor,d} - \hat{\boldsymbol{\theta}}_{motor}) \right) , \tag{4.8}$$

with $\mathbf{K_p}$ and $\mathbf{K_v}$ being 2×2 diagonal matrices that contain proportional and derivative gains, respectively. Similarly $\hat{\mathbf{M}}_3$ is defined as

$$\hat{\mathbf{M}}_3 = \mathbf{M}_3 - \tilde{\mathbf{M}}_3 . \tag{4.9}$$

When there is no estimation error, that is $\tilde{\mathbf{M}}_3 = \mathbf{0}$, the servo-based control becomes

$$\mathbf{T}_{sb} = \mathbf{M}_3 \left(\ddot{\boldsymbol{\theta}}_{motor,d} + \mathbf{K_v}(\dot{\boldsymbol{\theta}}_{motor,d} - \hat{\dot{\boldsymbol{\theta}}}_{motor}) + \mathbf{K_p}(\boldsymbol{\theta}_{motor,d} - \hat{\boldsymbol{\theta}}_{motor}) \right) , \tag{4.10}$$

where the subscript "*d*" in $\boldsymbol{\theta}_{motor,d}$ denotes the desired trajectory profile for the motors.

4.2 Closed-Loop Stability Analysis

Evaluating the stability of the closed-loop system under the proposed computed torque control is important as it sheds light on the first step of choosing control gains in both the proportional gain matrix $\mathbf{K_p}$ and derivative gain matrix $\mathbf{K_v}$. Under the assumption of no estimation error in system parameters, Equations 4.1, 4.7 and 4.10 are used to obtain the closed-loop system dynamics with the governing Equation 2.16. The simplified closed-loop system is then obtained and can be written as

$$\begin{bmatrix} \mathbf{M_1} & \mathbf{M_2} \end{bmatrix} \ddot{\boldsymbol{\theta}} + \mathbf{V}_{link}(\boldsymbol{\theta_{link}}, \dot{\boldsymbol{\theta}}_{link}) + \mathbf{C}_{link}\dot{\boldsymbol{\theta}} + \mathbf{K}_{link}\boldsymbol{\theta} + \mathbf{D}_{link}(\dot{\boldsymbol{\theta}}) = \mathbf{0} \tag{4.11}$$

and

$$\mathbf{0} = \mathbf{M}_3\Big((\ddot{\boldsymbol{\theta}}_{motor,d} - \ddot{\boldsymbol{\theta}}_{motor}) + \mathbf{K_v}(\dot{\boldsymbol{\theta}}_{motor,d} - \dot{\boldsymbol{\theta}}_{motor}) + \mathbf{K_p}(\boldsymbol{\theta}_{motor,d} - \boldsymbol{\theta}_{motor}) \Big) \tag{4.12}$$

As seen, the closed-loop system, under the assumption of zero estimation error of system parameters, becomes two separate subsystems described by Equation 4.11 for the link subsystem and by Equation 4.12 for the motor subsystem. Furthermore, defining the motor error as

$$\boldsymbol{e_{motor}} = \boldsymbol{\theta_{motor,d}} - \boldsymbol{\theta} \ , \tag{4.13}$$

Equation 4.12 now becomes

$$\mathbf{0} = \mathbf{M}_3(\ddot{\boldsymbol{e}}_{motor} + \mathbf{K_v}\dot{\boldsymbol{e}}_{motor} + \mathbf{K_p}\boldsymbol{e}_{motor}) \, . \tag{4.14}$$

To achieve asymptotic stability, Equation 4.14, namely the error dynamics equation, needs to fulfill the Hurwitz criterion [16]. In this way, the computed torque controller guarantees asymptotic tracking and stability for the internal dynamics of the link subsystem as well.

4.3 Closed-Loop Resonance

From the standpoint of designing command profiles, the resonant frequencies of the closed-loop system need to be determined first. Finding resonant frequencies is essentially looking for the eigenvalues of the closed-loop system. However, the closed-loop system is nonlinear so that its natural frequencies are not unique and change based on the robot configuration. To deal with the nonlinearity, a typical approach is to linearize the system around an equilibrium point. In this work, the closed-loop system is linearized as a function of θ_2 by using the Taylor series expansion. By doing so, only the first-order term is preserved while the angular velocity and acceleration are set to zero. As a result, terms due to the Coriolis and Centrifugal forces are dropped. The Coulomb friction is gone as well for zero angular velocity

around the equilibrium point. Therefore, the linearized closed-loop system derived from Equations 4.11 and 4.12 can be written as

$$\mathbf{M}_{CL}\ddot{\boldsymbol{\theta}} + \mathbf{C}_{CL}\dot{\boldsymbol{\theta}} + \mathbf{K}_{CL}\boldsymbol{\theta} = \mathbf{0}\,, \tag{4.15}$$

where $\mathbf{M}_{CL}$ is defined as

$$\mathbf{M}_{CL}(\theta_e) = \begin{bmatrix} \mathbf{M_1} & \mathbf{M_2} \\ 0 & \mathbf{M_3} \end{bmatrix} \tag{4.16}$$

and is evaluated at the equilibrium point $\theta_2 = \theta_e$. In this work, the robot starts from $\theta_e = 0$ (rad) and stops at $\theta_e = 1.2$ (rad). The closed-loop damping matrix is obtained as

$$\mathbf{C}_{CL} = \begin{bmatrix} \mathbf{C_1} & \mathbf{C_2} \\ 0 & \mathbf{M_3 K_v} \end{bmatrix} \tag{4.17}$$

and the closed-loop stiffness matrix is

$$\mathbf{K}_{CL} = \begin{bmatrix} \mathbf{K_1} & \mathbf{K_2} \\ 0 & \mathbf{M_3 K_p} \end{bmatrix}\,. \tag{4.18}$$

With the linearized closed-loop system, a general approach described in [39] is followed to find the natural frequencies from the eigenvalues of the so-called $\mathbf{A}$ matrix, which is defined as

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I}_{4\times 4} \\ -\mathbf{M}_{CL}^{-1}\mathbf{K}_{CL} & -\mathbf{M}_{CL}^{-1}\mathbf{C}_{CL} \end{bmatrix}\,. \tag{4.19}$$

All numerical results presented in this Chapter are based on the physical parameter values shown in Table 2.2. For a lightly damped robot system, the resonant frequency ω_r is approximated by the damped natural frequency ω_d in this work.

4.3.1 Link-Property-Based and Control-Gain-Based Eigenvalues

Thanks to the computed torque controller, its model-based compensator $\mathbf{T}_{mb}$ defined in Equation 4.7 is able to decouple the link subsystem from the motors. That is to say, the eigenvalues of matrix $\mathbf{A}$, denoted as $\boldsymbol{\lambda}(\mathbf{A})$, is a set of eigenvalues due to

the link subsystem and the motor subsystem, respectively. This can be easily seen by rearranging Equations 4.11 and 4.12 to obtain the"input-output" format, shown as

$$\mathbf{M_1}\ddot{\boldsymbol{\theta}}_{link} + \mathbf{C_1}\dot{\boldsymbol{\theta}}_{link} + \mathbf{K_1}\boldsymbol{\theta}_{link} = -(\mathbf{M_2}\ddot{\boldsymbol{\theta}}_{motor} + \mathbf{C_2}\dot{\boldsymbol{\theta}}_{motor} + \mathbf{K_2}\boldsymbol{\theta}_{motor}) \tag{4.20}$$

and

$$\mathbf{M}_3(\ddot{\boldsymbol{\theta}}_{motor} + \mathbf{K_v}\dot{\boldsymbol{\theta}}_{motor} + \mathbf{K_p}\boldsymbol{\theta}_{motor}) = \mathbf{M}_3(\ddot{\boldsymbol{\theta}}_{motor,d} + \mathbf{K_v}\dot{\boldsymbol{\theta}}_{motor,d} + \mathbf{K_p}\boldsymbol{\theta}_{motor,d}) \tag{4.21}$$

with the right-hand sides being the inputs of both subsystems. With the assistance of computed torque control, the coupling effect between links and motors is transformed into a sequential "input-output" relationship. The desired motor profile is the input of the motor subsystem while the actual motor response is the output. For the link subsystem, the actual motor response serves as subsystem input while the actual link response is the link subsystem output.

As a result, the eigenvalues of the link subsystem, $\boldsymbol{\lambda}(\mathbf{A}_{link})$, can be obtained from the matrix $\mathbf{A}_{link}$ defined as

$$\mathbf{A}_{link} = \begin{bmatrix} \mathbf{0} & \mathbf{I}_{2\times 2} \\ -\mathbf{M}_1^{-1}\mathbf{K}_1 & -\mathbf{M}_1^{-1}\mathbf{C}_1 \end{bmatrix} . \tag{4.22}$$

To better analyze the closed-loop eigenvalues due to the motor subsystem, the controller damping ratio matrix $\boldsymbol{\zeta}_{ctrl}$ and the controller natural frequency matrix $\boldsymbol{\omega}_{n,ctrl}$ are introduced to obtain the following relationships,

$$\mathbf{K_v} = 2\boldsymbol{\zeta}_{ctrl}\boldsymbol{\omega}_{n,ctrl} \tag{4.23}$$

and

$$\mathbf{K_p} = \boldsymbol{\omega}_{n,ctrl}\boldsymbol{\omega}_{n,ctrl} . \tag{4.24}$$

For simplicity, both $\boldsymbol{\zeta}_{ctrl}$ and $\boldsymbol{\omega}_{n,ctrl}$ are diagonal matrices that can be written as

$$\boldsymbol{\zeta}_{ctrl} = \begin{bmatrix} \zeta_{3,ctrl} & 0 \\ 0 & \zeta_{4,ctrl} \end{bmatrix} \qquad \boldsymbol{\omega}_{n,ctrl} = \begin{bmatrix} \omega_{3,ctrl} & 0 \\ 0 & \omega_{4,ctrl} \end{bmatrix} \tag{4.25}$$

so that $\zeta_{3,ctrl}$ and $\omega_{3,ctrl}$ together determine the controller for motor one and $\zeta_{4,ctrl}$ and $\omega_{4,ctrl}$ for motor two. Thus, the control-gain-based eigenvalues are found by

$$\begin{aligned} \boldsymbol{\lambda}_{3,ctrl} &= -\zeta_{3,ctrl}\,\omega_{3,ctrl} \pm i\,\omega_{3,ctrl}\sqrt{1-\zeta_{3,ctrl}^2} \\ \boldsymbol{\lambda}_{4,ctrl} &= -\zeta_{4,ctrl}\,\omega_{4,ctrl} \pm i\,\omega_{4,ctrl}\sqrt{1-\zeta_{4,ctrl}^2}\ , \end{aligned} \tag{4.26}$$

where i is the imaginary number, and both $\zeta_{3,ctrl}$ and $\zeta_{4,ctrl}$ are not larger than 1.

Simply put, even though controller gains are presented in matrix $\mathbf{A}$, once the system is decoupled, the resonant frequencies are purely a function of θ_2, which is illustrated in Figure 4.2. In this figure, the upper subplot shows the first mode of the closed-loop system, and the lower subplot shows the second mode. It can be seen that as θ_2 changes, the resonant frequencies change as well. This figure shows results when the controller parameters $\zeta_{3/4,ctrl} = 1$ and $\omega_{3/4,ctrl} = 40$. To demonstrate that the resonant frequencies are independent of the values of control gains, more numerical results obtained from different values of control gains are provided in Appendix A. Note that numerical results are calculated based on the robot parameter values listed in Table 2.2.

4.3.2 Link-Property-Based Command Shaping

With the framework of the separation of link-based eigenvalues and control-gain-based eigenvalues, the command shaping method can shape the profile purely based on the resonant frequencies found from the link subsystem because controller parameters do not affect the closed-loop system resonant frequencies. As mentioned in Section 2.1.2, the basis function formed by the segmented Versine profile is able to contain different target frequencies in each segment so as to remove more relevant energy that aggravates residual vibration. For robustness purposes, each segment adds a set of shaping bounds so that energy can be removed from a specific range of frequencies. Figure 4.3, based on Figure 4.2, adds two sets of $\pm 5\%$ shaping bounds for each mode separately. In each mode, the first set of bounds removes energy in the acceleration segment based on half of the final configuration, which is $\theta_2 = 0.6$(rad). The second

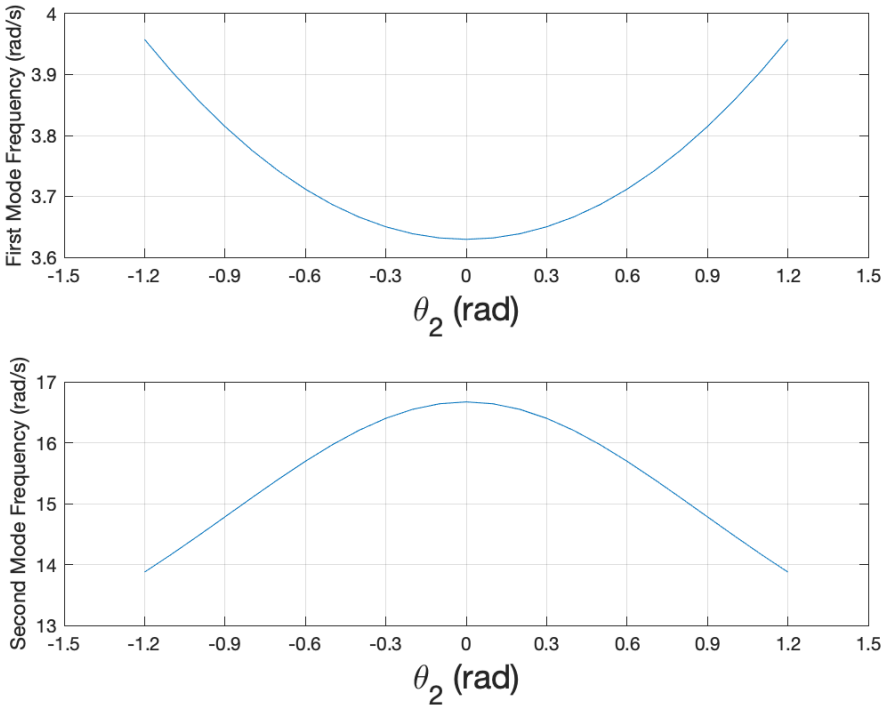


Figure 4.2. Resonant Frequencies of the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 40$

set of bounds is added in the deceleration segment based on the final configuration which is $\theta_2 = 1.2$(rad). As a result, for the first mode, the first set of bounds is added between 3.5262(rad/s) and 3.8974(rad/s), and the second set is added between 3.7597(rad/s) and 4.1555(rad/s). For the second mode, the first set of bounds is added between 14.9015(rad/s) and 16.4801(rad/s) while the second set is added between 13.1816(rad/s) and 14.5692(rad/s). Simulation results that incorporate the shaping bounds of ±5% with different control gain values are shown in Section 4.5.1.

Since the width of shaping bounds, from the standpoint of the command shaping method, is used for robustness as well as for covering more resonant frequencies during the point-to-point motion, this method is able to change the width of the shaping bounds to achieve optimized results. Figure 4.4 indicates the ±10% shaping bounds. Comparing with Figure 4.3, which uses the ±5% shaping bounds, Figure 4.4

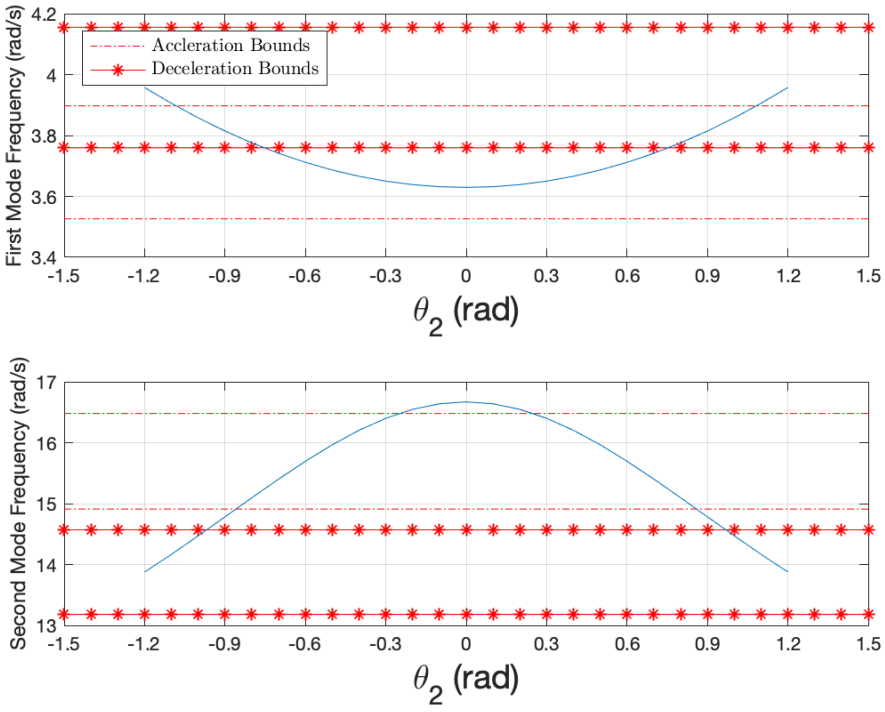


Figure 4.3. $\pm 5\%$ Shaping Bounds of Resonant Frequencies for the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 40$

shows more region could be covered in the second mode frequency. Simulation results comparing the effectiveness of different sets of shaping bounds are given in Figure 4.11 in Section 4.5.1.

4.4 Mismatched Closed-Loop System

In reality, estimation error exists so that the model-based compensator in computed torque control can not decouple the robot system perfectly. To mathematically capture this, the closed-loop system given in Equation 4.15 needs to change so that

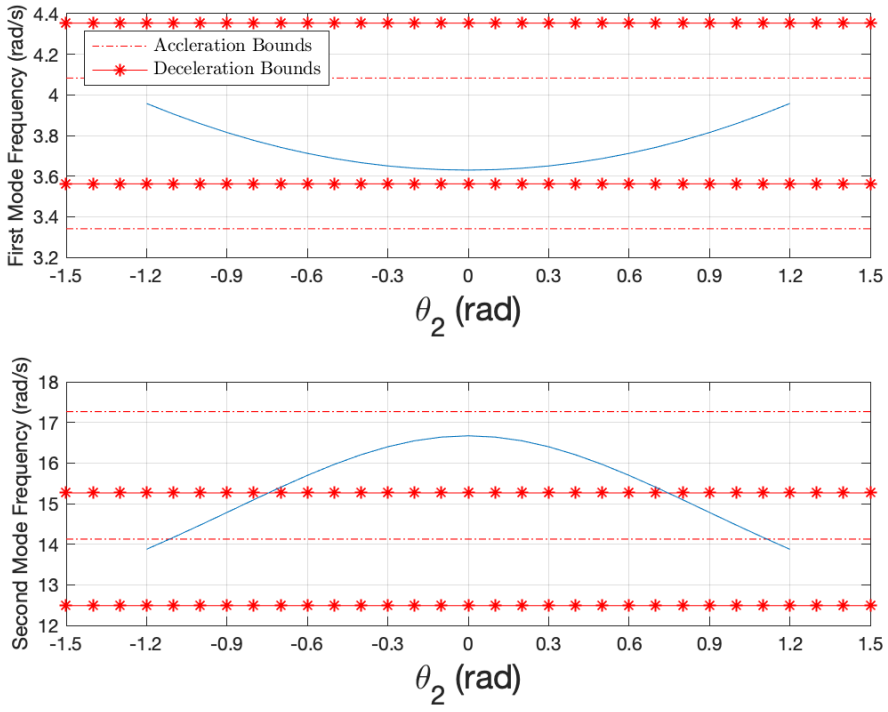


Figure 4.4. ±10% Shaping Bounds of Resonant Frequencies for the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 40$

the influence owing to the mismatched model can appear in the equations. As a result, the closed-loop equation is written as

$$\mathbf{M}_{CL,mis}\ddot{\boldsymbol{\theta}} + \mathbf{C}_{CL,mis}\dot{\boldsymbol{\theta}} + \mathbf{K}_{CL,mis}\boldsymbol{\theta} = \mathbf{0} \ , \tag{4.27}$$

where $\mathbf{M}_{CL,mis}$ is defined as

$$\mathbf{M}_{CL,mis}(\theta_e) = \begin{bmatrix} \mathbf{M_1} & \mathbf{M_2} \\ \tilde{\mathbf{M}}_{\mathbf{mb}} & \mathbf{M_3} \end{bmatrix} \tag{4.28}$$

and evaluated at the equilibrium point $\theta_2 = \theta_e$. The closed-loop damping matrix is obtained as

$$\mathbf{C}_{CL,mis} = \begin{bmatrix} \mathbf{C_1} & \mathbf{C_2} \\ \tilde{\mathbf{C}}_{\mathbf{2}} & \mathbf{M_3 K_v} \end{bmatrix} \tag{4.29}$$

and the closed-loop stiffness matrix is

$$\mathbf{K}_{CL,mis} = \begin{bmatrix} \mathbf{K_1} & \mathbf{K_2} \\ \tilde{\mathbf{K}}_\mathbf{2} & \mathbf{M_3 K_p} \end{bmatrix} . \tag{4.30}$$

It is essential to know in the presence of estimation error, the closed-loop system is no longer decoupled such that all matrices in Equations 4.28 to 4.30 no longer have zero terms at their lower-left corner, respectively. As before, resonant frequencies for the mismatch-model closed-loop system are obtained from the eigenvalues of matrix $\mathbf{A}_{mis}$

$$\mathbf{A}_{mis} = \begin{bmatrix} \mathbf{0} & \mathbf{I}_{4\times4} \\ -\mathbf{M}_{CL,mis}^{-1}\mathbf{K}_{CL,mis} & -\mathbf{M}_{CL,mis}^{-1}\mathbf{C}_{CL,mis} \end{bmatrix} . \tag{4.31}$$

Since the system is no longer decoupled, resonant frequencies become a function of both robot configuration and control gains. This can be seen from Figures 4.5 to 4.8, each having different control gain values. In these figures, a dashed line indicates the actual frequency of the mismatch-model closed-loop system while a solid line stands for the frequency of the nominal system, which has been perfectly decoupled. Same as those in Section 4.3.2, the shaping bounds are obtained based on the nominal frequencies at different configurations. Note that numerical results due to estimation errors are obtained by setting the "actual robot" with higher stiffness values, so that estimated stiffness is 95% of the"actual robot" stiffness. For convenience, it is labeled as $\mathrm{MF}_{\mathbf{K}} = 95\%$, where "MF" denotes mismatch factor, and the subscript $\mathbf{K}$ stands for stiffness.

As seen, the actual frequencies change when control gain changes. The mismatch-model simulation results that use the nominal frequencies and the corresponding shaping bounds in segmented Versine command profiles are presented in Section 4.5.2.

4.5 Simulation Analysis

In this Section, simulation results are presented. Section 4.5.1 shows results based on a perfectly decoupled system, that is to say, no estimation error exists for the

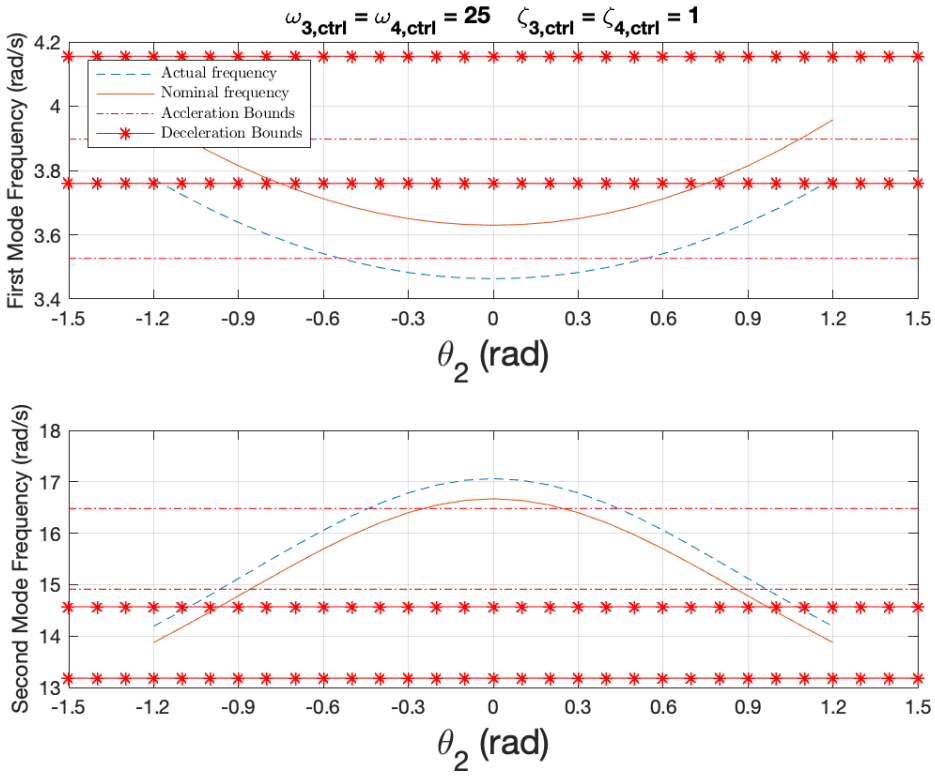


Figure 4.5. ±5% Shaping Bounds of Resonant Frequencies for the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 25$, and Mismatch factor $\mathrm{MF}_{\mathbf{K}} = 95\%$

closed-loop system. Based on results and observations from the decoupled system, focus narrows to a smaller set of control gains for which all controller damping ratios are set to 1. Section 4.5.2 provides results for the mismatch-model closed-loop system with estimation error.

4.5.1 Influence of Control Gains

This subsection focuses on the servo-based PD controller, which is one of the essential building blocks for the computed torque controller. For simplicity, controller parameters for motor one and motor two are set to the same values in the simulation such that $\zeta_{3,ctrl} = \zeta_{4,ctrl}$ and $\omega_{3,ctrl} = \omega_{4,ctrl}$. Figure 4.9 shows the influence on peak-

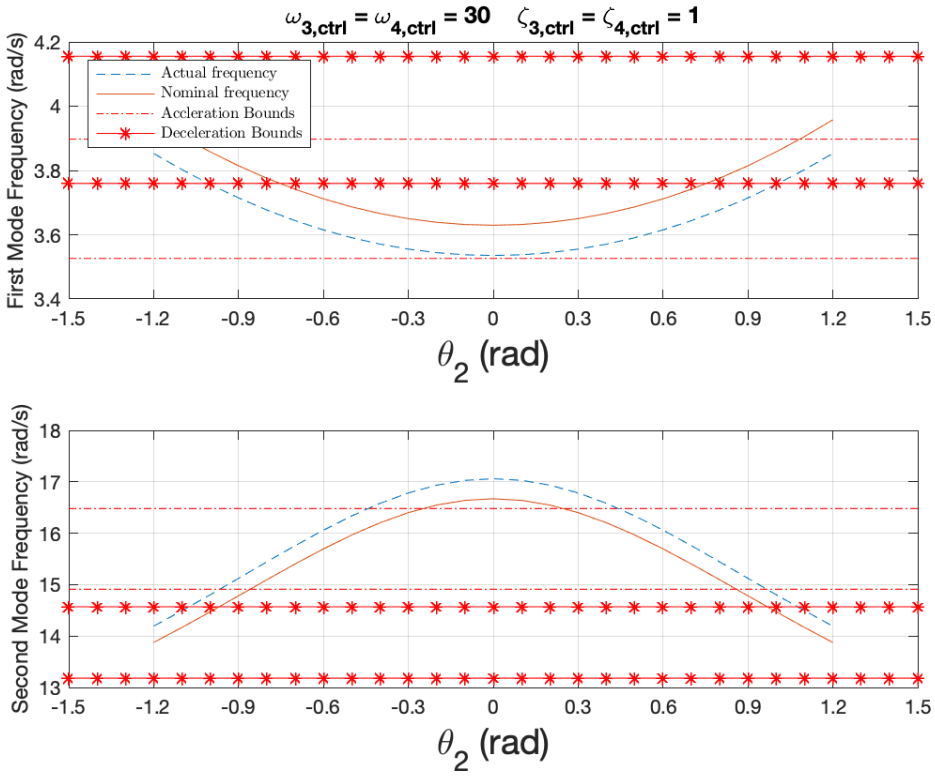


Figure 4.6. ±5% Shaping Bounds of Resonant Frequencies for the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 30$, and Mismatch factor $\mathrm{MF}_{\mathbf{K}} = 95\%$

to-peak value of the residual vibration defined by Equation 2.51, when control gains in the servo-based PD controllers change. In this figure, the X − Y plane denotes controller pole location with X axis being the real part for the pole location and Y axis the imaginary part. The eigenvalues of the controller are determined by using Equation 4.26. For the Z axis, it is used to indicate the amplitude of the peak-to-peak value of the residual vibration. For each set of control gains there is one corresponding value of the vibration amplitude. For example, when $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 20$, the corresponding peak-to-peak $\ddot{P}_{xy}$ is 0.2248(m/s^2). For this set of control gains, the repeated pole locations are $-20 \pm i0$; therefore, in this figure, the simulation result is plotted at the point (-20, 0, 0.2248). When the control gains

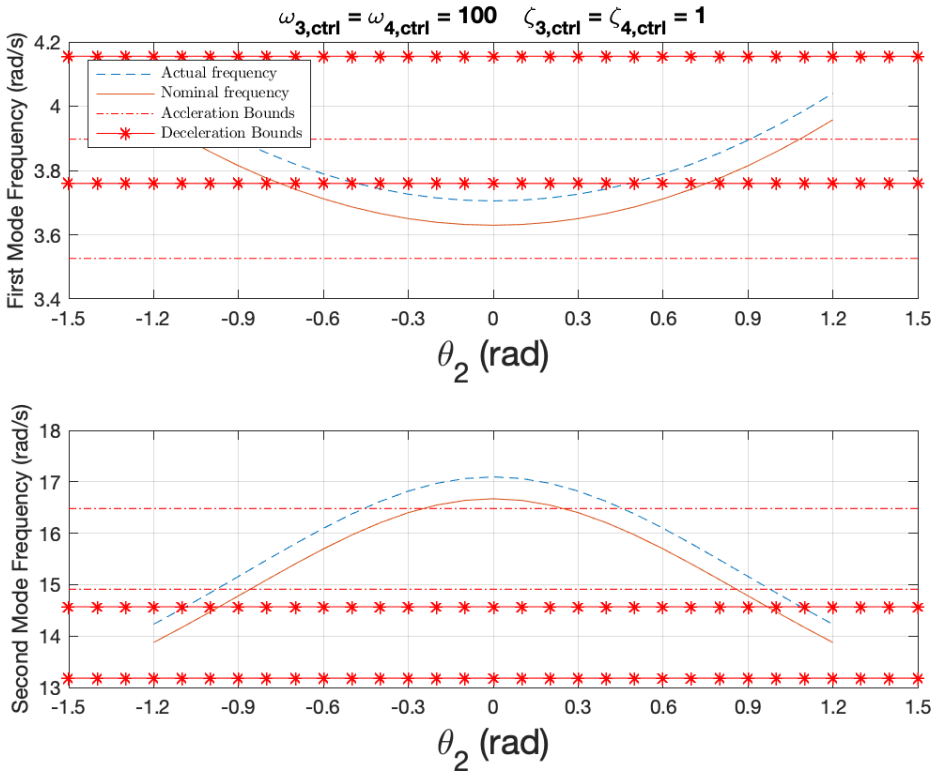


Figure 4.7. ±5% Shaping Bounds of Resonant Frequencies for the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 100$, and Mismatch factor $\mathrm{MF}_{\mathbf{K}} = 95\%$

are set to have complex eigenvalues, the peak-to-peak residual vibration is plotted on the eigenvalue with positive imaginary part. For example, when $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 0$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 3$, the corresponding peak-to-peak $\ddot{P}_{xy}$ is 0.6046(m/s^2). Since this set of control gains has complex eigenvalues of $0 \pm i3$, the simulation result is plotted only at (0, 3, 0.6046). A re-scaled figure of Figure 4.9 is shown in Figure 4.10 in order to better demonstrate the impact of control gains.

As seen, in general, residual vibration amplitude decreases when both controllers have larger values of $\omega_{3/4,ctrl}$. Relatively large residual vibration is observed around the pole location of $0 \pm i4$ and peaks at $0 \pm i4$ with peak-to-peak $\ddot{P}_{xy}$ value of 2.274(m/s^2). It is worth noting that such pole location is close to the first mode of resonant fre-

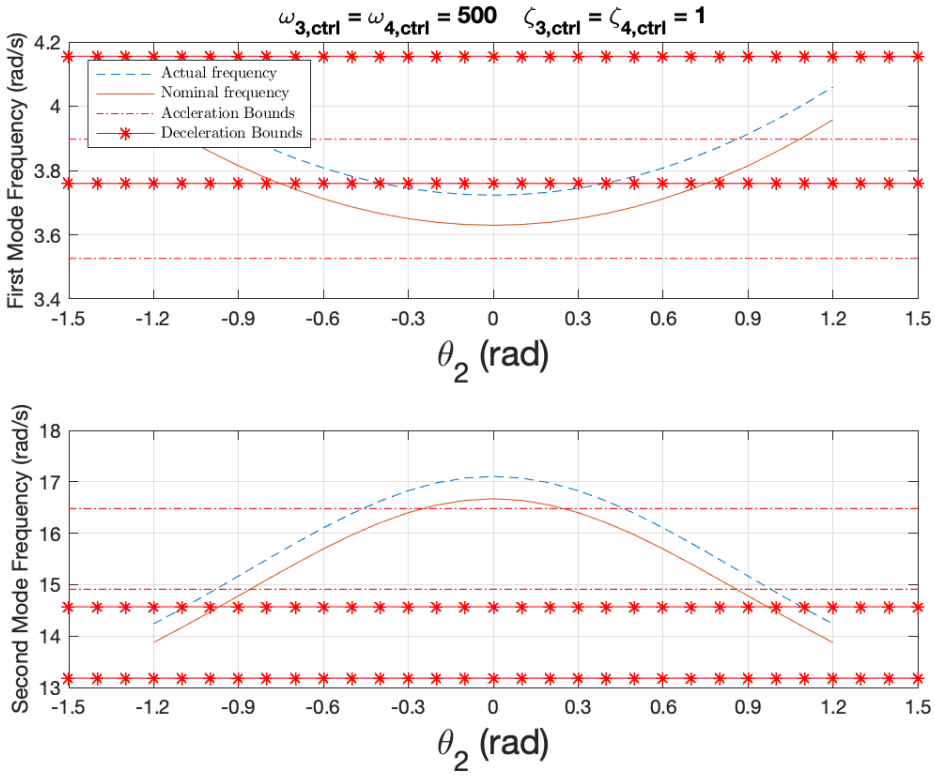


Figure 4.8. ±5% Shaping Bounds of Resonant Frequencies for the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 500$, and Mismatch factor $\mathrm{MF}_{\mathbf{K}} = 95\%$

quency, which is about 3.9576(rad/s). This phenomenon, in general, could be explained by the frequency response of a second-order system. Since the motor control behaves as a second-order system, when the controller damping ratio is less than 0.707, its frequency response is greater than 0(dB) at its natural frequency. As a result, more energy is injected into the robot system when the controller's eigenvalues are closer to those of the link subsystem.

Even though resonant frequencies are independent of control gain values for the perfectly decoupled system, results covering a large number of different control gain values indicate that residual vibration is still affected by the controller. This can be possibly explained by using a simple input/output transfer function. The response of

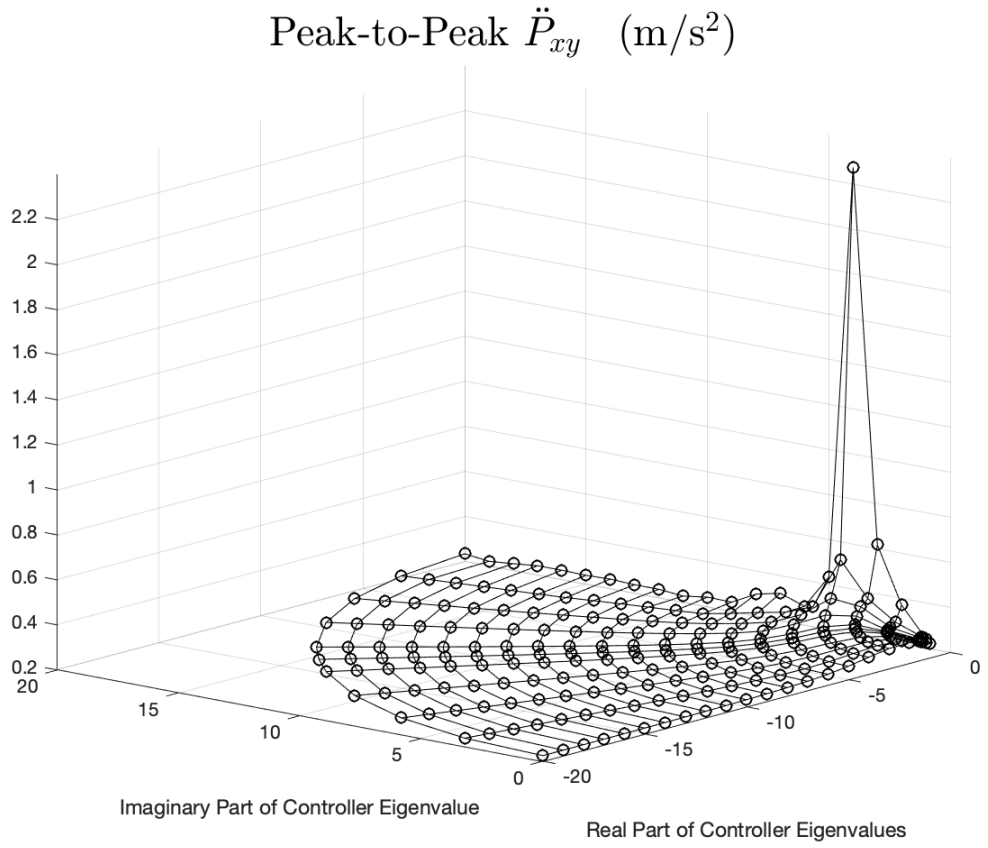


Figure 4.9. Peak-to-Peak Residual Vibration over the Locations of Controller Eigenvalues for Perfectly Decoupled Robotic Arm

the system is not only characterized by the system poles but are also affected by the system zeros. If the control gains affect the numerator, it indicates that the controller still influences the residual vibration.

Instead of merely looking at the influence of different control gains, it is worthwhile to evaluate the effectiveness of shaping bounds in minimizing residual vibration. When the system is perfectly-decoupled, Figure 4.11 shows the effectiveness of minimizing residual vibration with the built-in shaping bounds. It can be seen that as the width of the shaping bounds increases, residual vibration amplitude gets smaller. This can be explained by the range of resonant frequencies being covered, which is shown in Figure 4.3 for a ±5% shaping bounds and in Figure 4.4 for ±10%. Since the

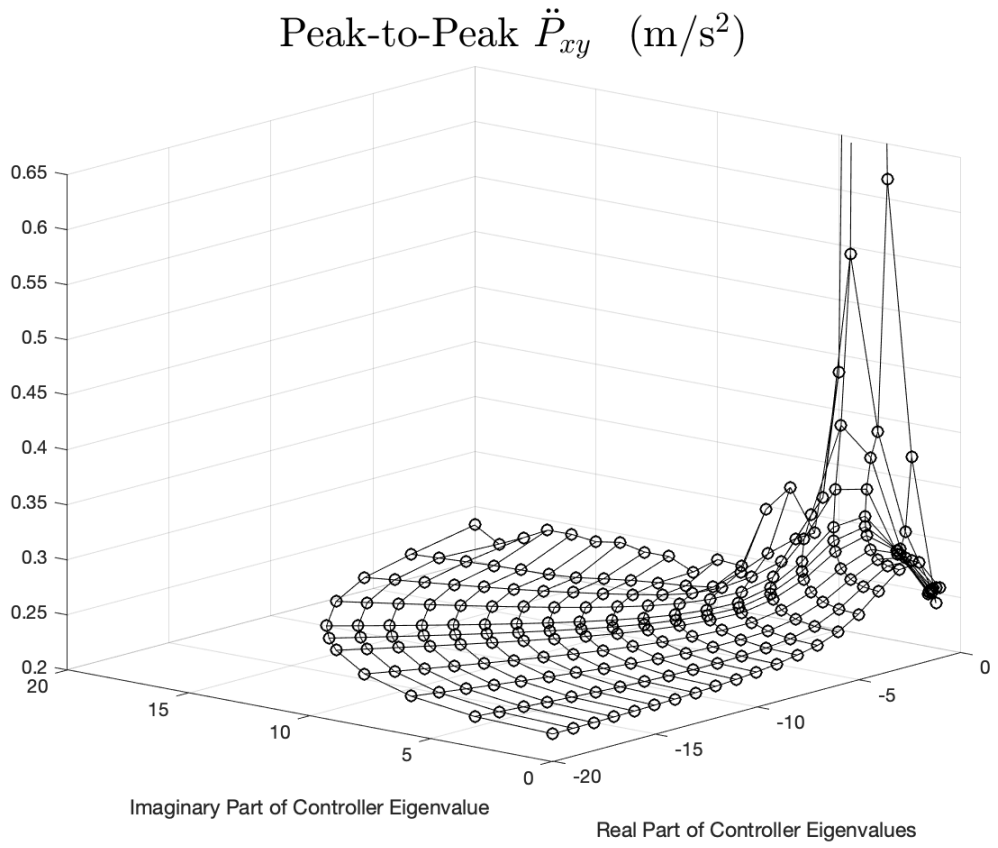


Figure 4.10. Rescaled Plot of Peak-to-Peak Residual Vibration over the Locations of Controller Eigenvalues for Perfectly Decoupled Robotic Arm

second mode frequency is completely covered by the ±10% shaping bounds while this is not the case for the ±5% shaping bounds, more critical energy that is relevant to residual vibration has been removed during the shaping process. Thus, less vibration is observed for each set of control gains. A case for shaping bounds of 0% width is included to better illustrate this idea.

Based on these sets of simulation results, when choosing the controller gains, it is better to avoid placing control poles around the system resonant frequencies, and also to have control gains as large as possible. Moreover, at this stage, the controller

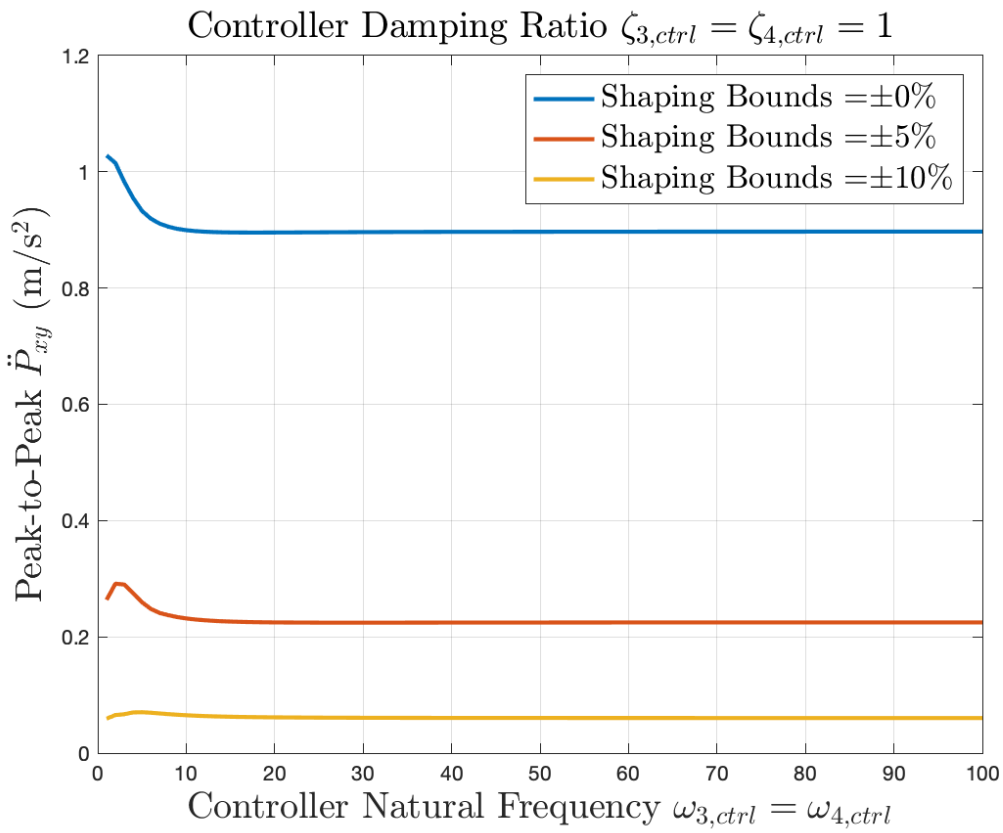


Figure 4.11. Influence of the Width of Shaping Bounds on Peak-to-Peak Residual Vibration with $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ for the Perfectly Decoupled Robotic Arm

works quite independent of the width of shaping bounds so that choosing a large control gain does not degrade the command profile.

4.5.2 Impact of Model Mismatch

With the information and numerical observations from the perfectly-decoupled system response, focus in this subsection shifts to results generated by setting the controller damping ratios to 1. By doing so, each motor in the robot system introduces two repeated poles on the real axis so that they are not close to those lightly-damped poles for the link dynamics. Figure 4.12 shows the quality of shaping bounds when

the system is not perfectly decoupled. In this figure, the solid line denotes results for a perfectly-decoupled system, and the dashed line for a system with estimation error of $\mathrm{MF}_{\mathbf{K}} = 95\%$. It can be seen that, with wider shaping bounds, residual vibration

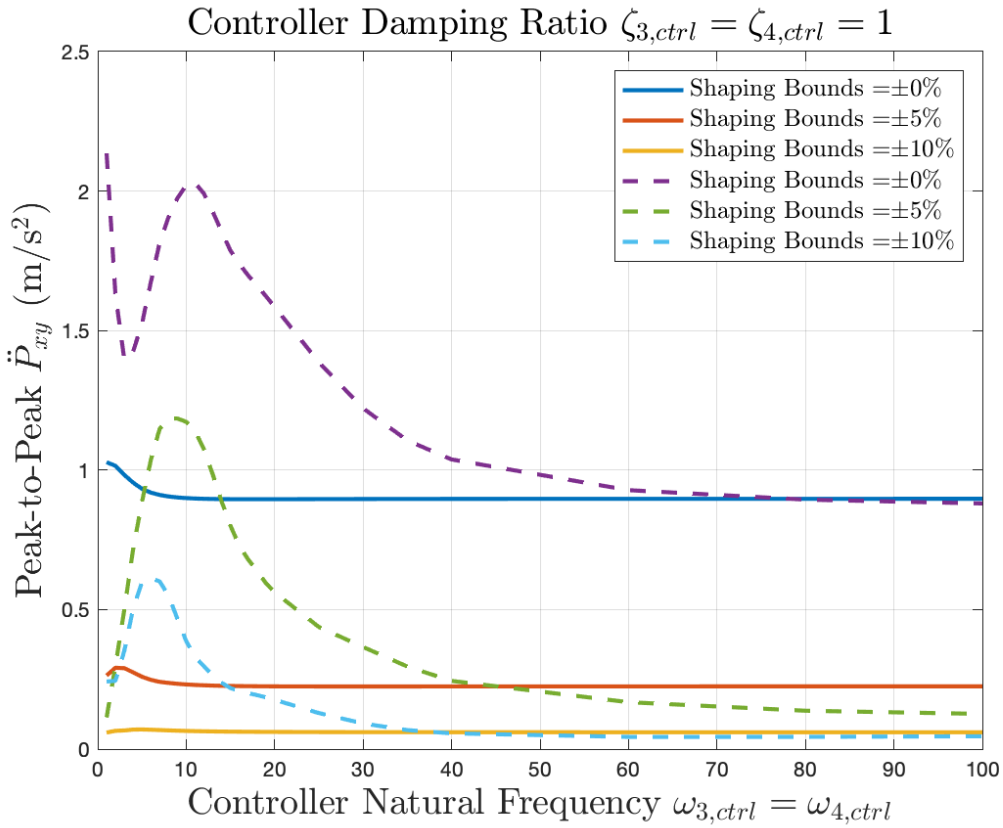


Figure 4.12. Influence of the Width of Shaping Bounds on Peak-to-Peak Residual Vibration with Solid Line for Perfectly-Decoupled System and Dash Line for Mismatch Model of $\mathrm{MF}_{\mathbf{K}} = 95\%$

tends to decrease. This confirms the idea of robustness when the model is not perfectly decoupled. Compared with the decoupled system, the mismatched model yields lower vibration as control gain increases. This is mainly because the resonant frequency, in mismatched cases, is not simply a function of robot configuration. It depends on both robot configuration and the control gains. This has been demonstrated in Figures 4.5 to 4.8. As the control gain increases, the curve of actual resonant frequency is

gradually being pushed into the region bounded by the shaping bounds, and this curve finally stays in a certain region regardless of the control gain magnitude.

Results in Figures 4.13 to 4.20 present the impact on residual vibration when $\mathrm{MF}_{\mathbf{K}}$ decreases from 95% to 50%, meaning that the actual system has higher and higher stiffness relative to the model. Figure 4.21 summarizes the results in one plot.

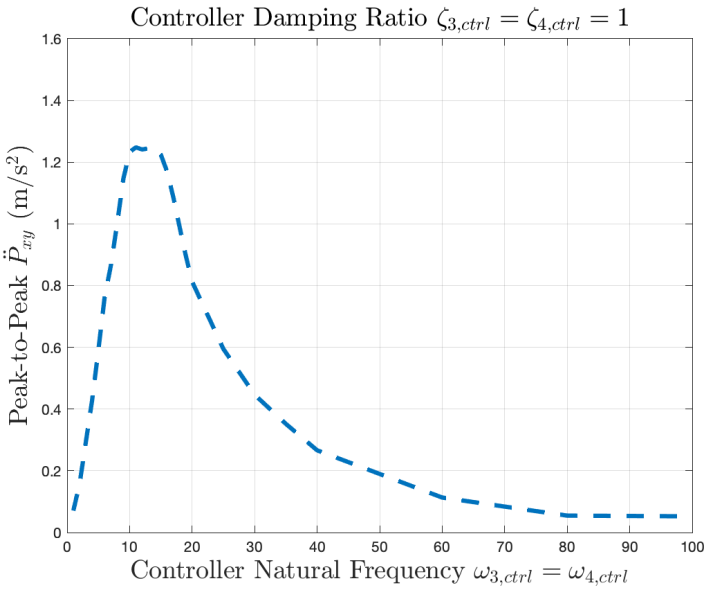


Figure 4.13. Peak-to-Peak Residual Vibration with $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\mathrm{MF}_{\mathbf{K}} = 90\%$ under $\pm 5\%$ Shaping Bounds.

Based on the results, it can be seen that each set of results peaks around $\omega_{3/4,ctrl} = 13$(rad/s) before vibration starts decreasing. This is mainly because, when estimation error exists in the model-based compensator, performance in tracking the command profile degrades. When the control gain becomes sufficiently large, it is able to better track the command profile, and starts lowering the vibration. After that, optimal control gain occurs for cases of $\mathrm{MF}_{\mathbf{K}} < 90\%$, indicating a local minimum of residual vibration has been reached. Comparing different values of $\mathrm{MF}_{\mathbf{K}}$, optimal control gain increases when $\mathrm{MF}_{\mathbf{K}}$ gets closer to 100%. Besides, at each optimal control gain, higher residual vibration is seen when $\mathrm{MF}_{\mathbf{K}}$ goes down. Table 4.1 provides residual vibration amplitude at each optimal control gain for different values of $\mathbf{MF}_{\mathbf{K}}$. It is

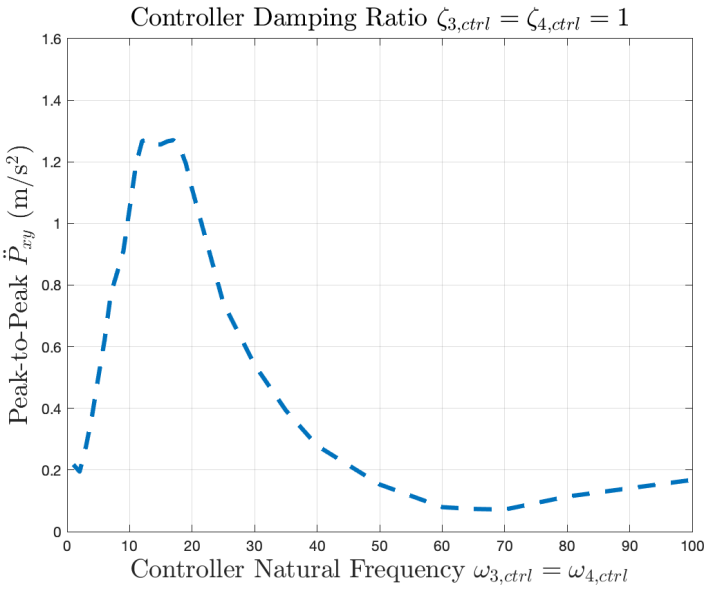


Figure 4.14. Peak-to-Peak Residual Vibration with $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\text{MF}_{\mathbf{K}} = 85\%$ under $\pm 5\%$ Shaping Bounds.

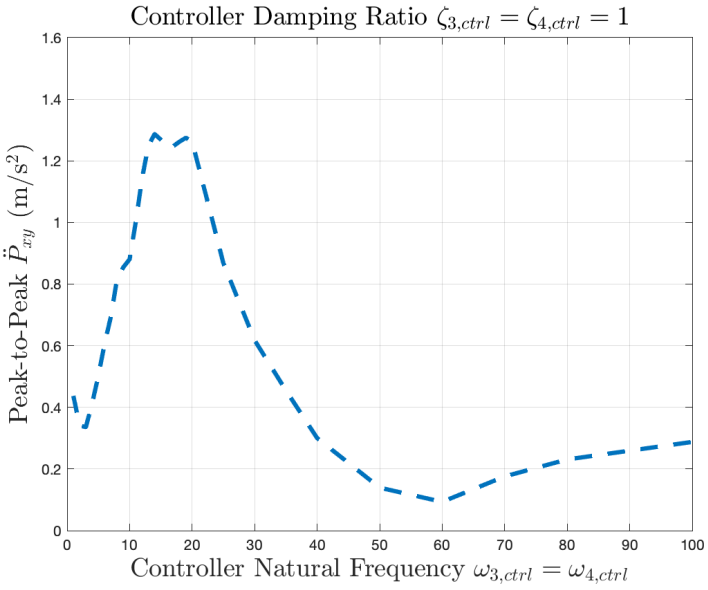


Figure 4.15. Peak-to-Peak Residual Vibration with $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\text{MF}_{\mathbf{K}} = 80\%$ under $\pm 5\%$ Shaping Bounds.

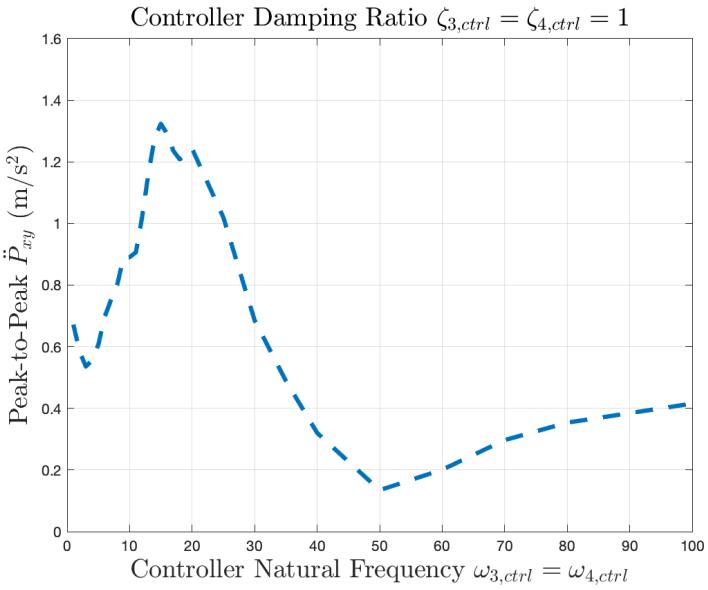


Figure 4.16. Peak-to-Peak Residual Vibration with $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\text{MF}_{\mathbf{K}} = 75\%$ under $\pm 5\%$ Shaping Bounds.

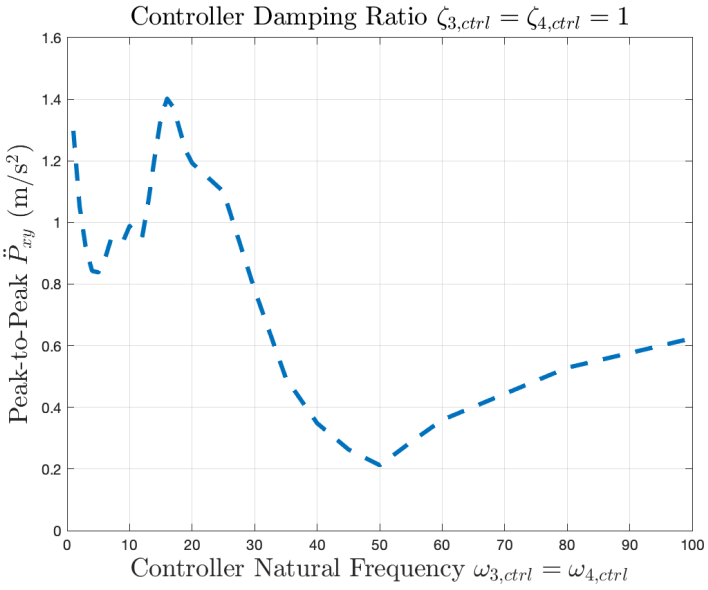


Figure 4.17. Peak-to-Peak Residual Vibration with $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\text{MF}_{\mathbf{K}} = 70\%$ under $\pm 5\%$ Shaping Bounds.

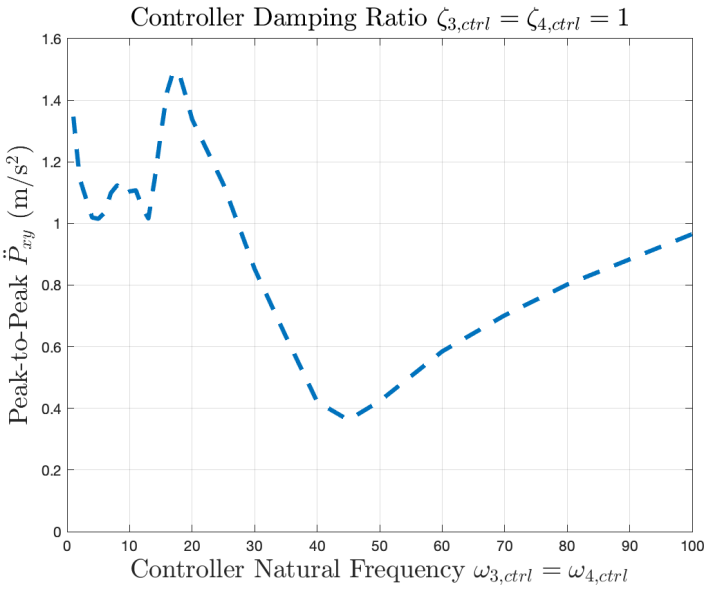


Figure 4.18. Peak-to-Peak Residual Vibration with $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\text{MF}_{\mathbf{K}} = 65\%$ under $\pm 5\%$ Shaping Bounds.

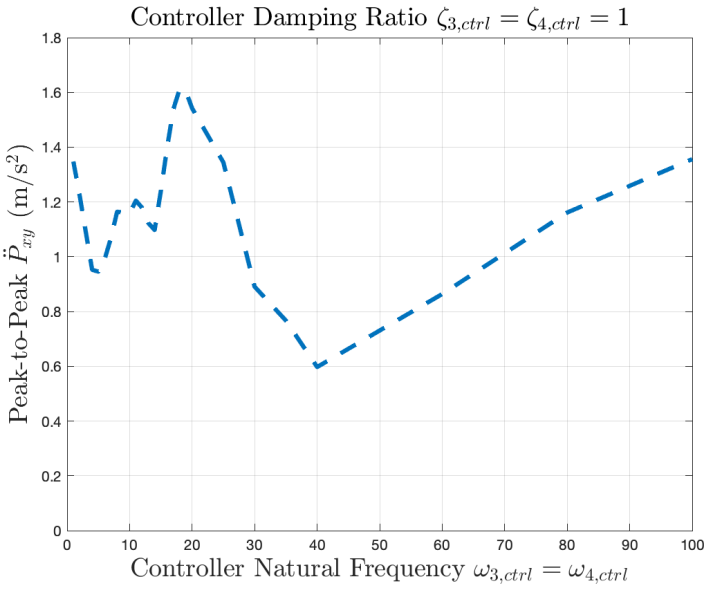


Figure 4.19. Peak-to-Peak Residual Vibration with $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\text{MF}_{\mathbf{K}} = 60\%$ under $\pm 5\%$ Shaping Bounds.

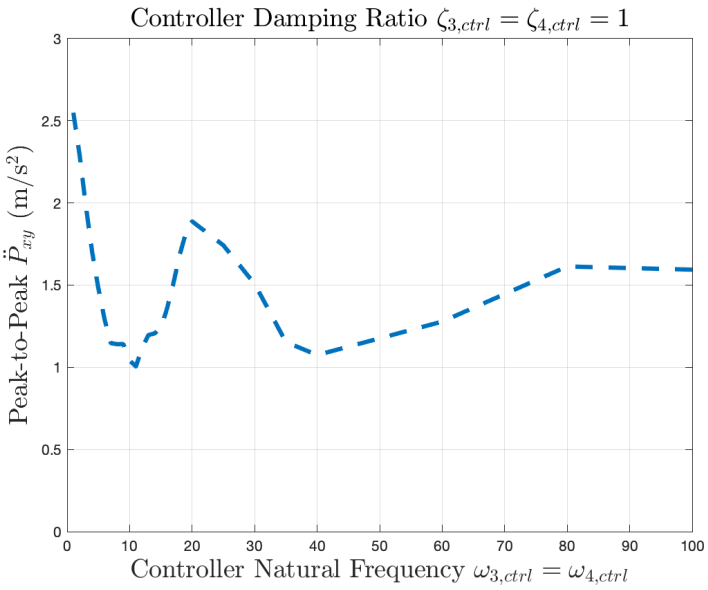


Figure 4.20. Peak-to-Peak Residual Vibration with $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\text{MF}_{\mathbf{K}} = 50\%$ under $\pm 5\%$ Shaping Bounds.

interesting to investigate the corresponding actual resonant frequency so as to build the connections between the shaping bounds, control gain, and mismatch factor. Appendix B shows the actual resonant frequencies for $\mathbf{MF_K} = 70\%$ cases. It is found that whenever the actual resonant frequency curve pushed by the control gain enters the shaping bounds, less residual vibration is observed. However, for cases that have large estimation error, the controller could "over-push" the curve, and as a result, the curve could go outside the shaping bounds, resulting in stronger vibration. To better demonstrate this idea, an animation with $\mathbf{MF_K} = 70\%$ is provided and shown in Animation 4.22. In this animation, the actual closed-loop resonant frequencies are shown in the left subplots while the amplitude of residual vibration is plotted on the right subplot. As the controller parameter $\omega_{3/4,ctrl}$ increases, it can be seen that both resonant frequency curves and residual vibration amplitudes change correspondingly. More animations with different estimation errors are shown in Appendix C.

Efforts were also made in evaluating the cases having estimation error in the viscous damping matrix instead of the stiffness matrix. For convenience, $\mathbf{MF_C}$ denotes

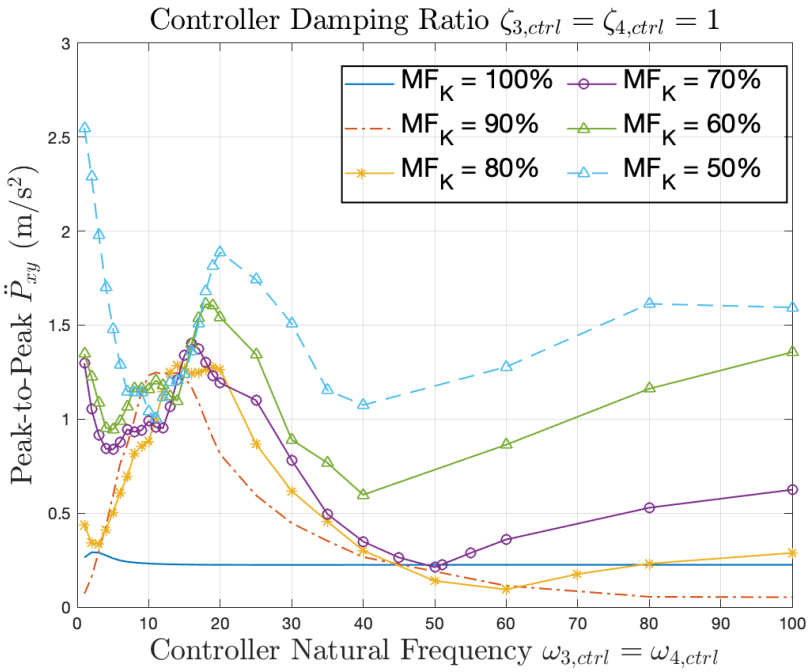
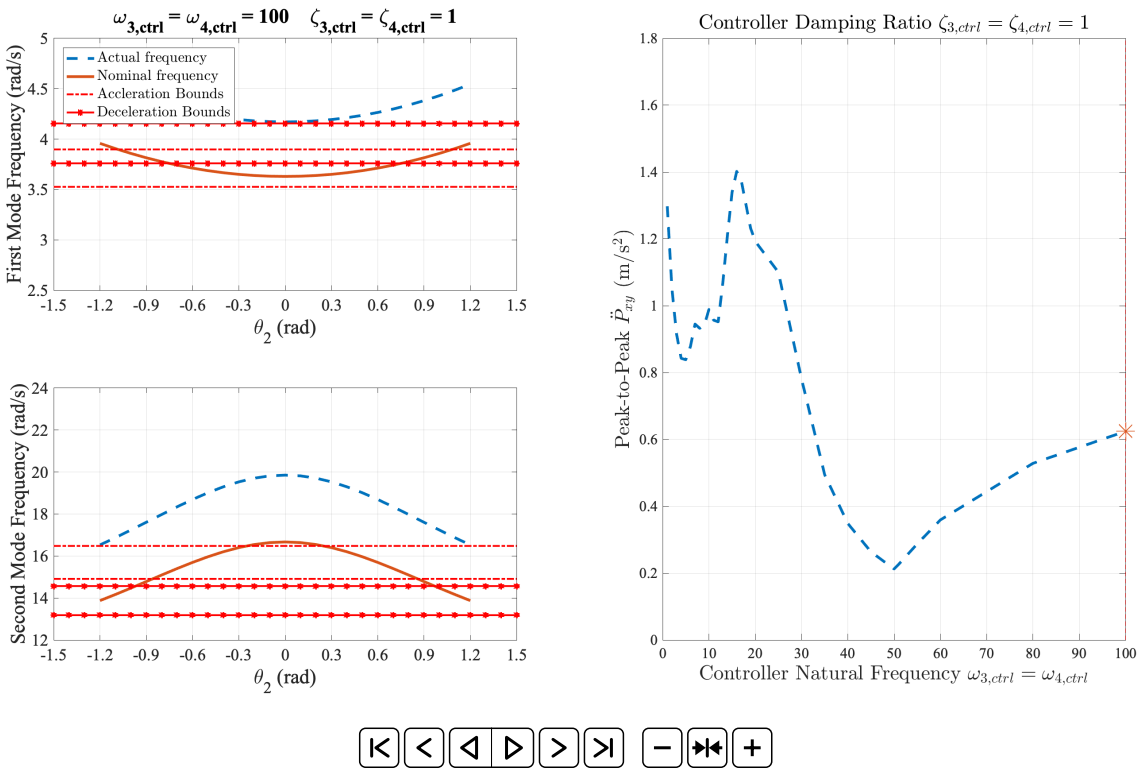


Figure 4.21. Peak-to-Peak Residual Vibration with $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and different $\mathbf{MF_K}$ values under $\pm 5\%$ Shaping Bounds.

Table 4.1. Optimal Control Gain for Mismatch Model

$\mathbf{MF_K}$	Optimal $\omega_{3/4,ctrl}$ (rad/s)	$\ddot{P}_{xy}$(m/s^2)
85%	65	0.07
80%	60	0.09
75%	50	0.13
70%	50	0.21
65%	45	0.36
60%	40	0.60
50%	40	1.08

the mismatch percentage of estimated damping coefficients compared to that of the actual robot system. For example, $\mathbf{MF_C} = 95\%$ says that the estimated viscous



Animation 4.22. Actual Closed-loop Resonant Frequencies with ±5% Shaping Bounds(left) and Residual Vibration of the Robotic Arm(right) with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and variable $\omega_{3,ctrl} = \omega_{4,ctrl}$, and Mismatch Factor of Stiffness $\mathbf{MF_K} = 70\%$

damping matrix is 95% of the actual system. Figures 4.23 to 4.25 show simulation results under various values of $\mathbf{MF_C}$. Figure 4.26 summarizes the results in one plot. As seen, vibration peaks at relatively small control gain, as mentioned before, because the controller can not perfectly track the command profile when there is estimation error in the model-based portion of computed torque control. However, as the control gain goes up, vibration stays around 0.219(m/s^2) regardless of the control gain magnitude or the value of $\mathrm{MF}_\mathbf{C}$. This can also be explained by the actual resonant frequencies presented in Appendix B. The controller pushes the curve of resonant frequency into the shaping bounds designed based on the nominal system. Unlike cases that have estimation error in the stiffness matrix, these cases show that the

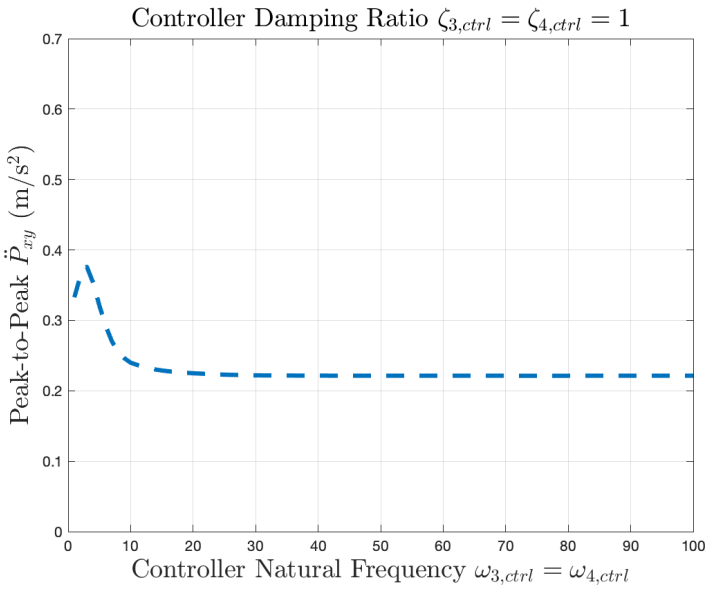


Figure 4.23. Peak-to-Peak Residual Vibration with $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\mathrm{MF}_{\mathbf{C}} = 95\%$ under $\pm 5\%$ Shaping Bounds.

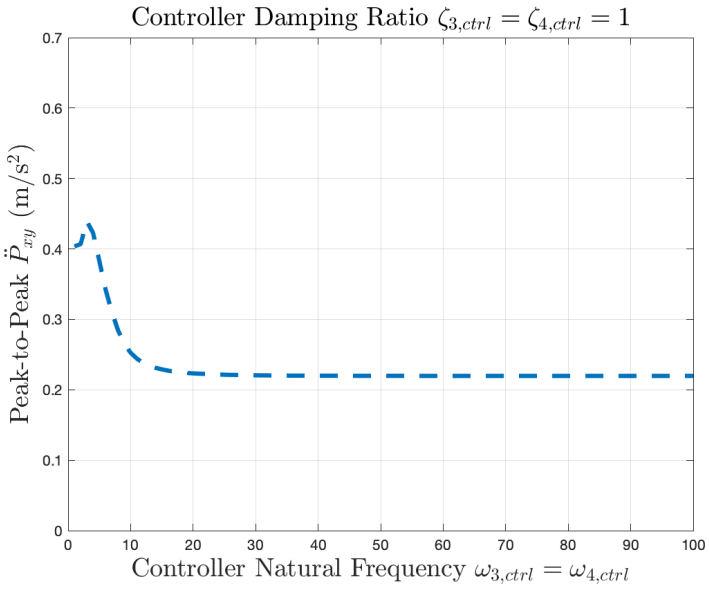


Figure 4.24. Peak-to-Peak Residual Vibration with $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\mathrm{MF}_{\mathbf{C}} = 90\%$ under $\pm 5\%$ Shaping Bounds.

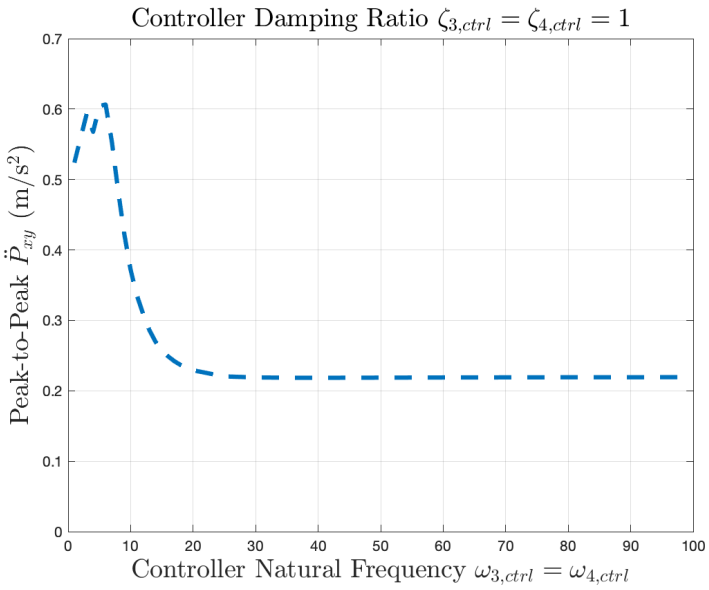


Figure 4.25. Peak-to-Peak Residual Vibration with $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\mathrm{MF_C} = 65\%$ under $\pm 5\%$ Shaping Bounds.

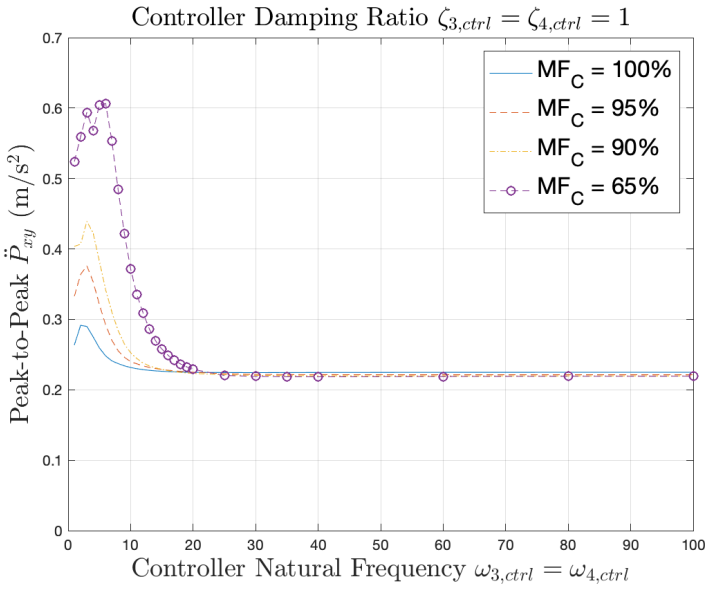


Figure 4.26. Peak-to-Peak Residual Vibration with $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\mathrm{MF_C} = 65\%$ to 100% under $\pm 5\%$ Shaping Bounds.

curve approaches the nominal frequency curve as control gain increases, and becomes insensitive to the controller.

4.6 Recommendations for Control Gain Selection

At this stage, some basic interactions between the shaping bounds from the command shaping method and the control gains from computed torque control have been observed, and can be summarized as follows:

1. For a perfectly-decoupled system, control gain and command shaping work quite independently except for cases in which the controller places its poles around the link-property-based pole location, injecting critical energy back to the robot system.

2. When model estimation error exists in the computed torque control, control gains induce a deviation of the actual resonant frequency curve from the nominal resonant frequency curve that is used to generate the shaping bounds in the command shaping method.

3. The amount of overlap area between the actual resonant frequency curve and the shaping bounds greatly affects the residual vibration in the sense that more critical energy which contributes to the residual vibration can be removed with larger overlap area.

In light of these observations, it is recommended to firstly choose critically-damped controller poles in order to avoid injecting energy into the robot system. Therefore, $\zeta_{3/4,ctrl} = 1$. Besides, it is important for tracking the command profile. From the standpoint of the command shaping method, it is better to accurately remove critical energy that is associated with the residual vibration. Therefore, two different approaches can be adopted to achieve this goal by the computed torque controller. One is to correct the model-based compensator such that it has less estimation error. With that, higher control gain is able to "push" the resonant frequency curve into

the shaping bounds. It is important to know that, based on eigenvalue analysis, it is better to underestimate the system than to overestimate it for the sake of stability in the presence of estimation error. That is to say the nominal values of both viscous matrix $\mathbf{C}_{mb}$ and stiffness matrix $\mathbf{K}_{mb}$ are better to be smaller than those of the actual system. Another approach is to tune the control gains $\omega_{3/4,ctrl}$ inside the servo-based PD controller, since, in the presence of estimation error, control gain is able to "push" the actual resonant frequency curve. When the curve enters into the shaping bounds region, the quality of the command profile is improved, since, for a particular range of estimation error due to the stiffness, an optimal control gain exists such that the resonant frequency curve is inside the shaping bounds, and more critical energy is removed.

5. CONCLUSIONS AND FUTURE WORK

5.1 Summary and Conclusion

In this work, a system identification algorithm called Instrumented Variable (IV) is applied from the literature to improve the estimation of mechanical parameters. Based on suggestions from previous work, a sequential identification process is adopted in order to minimize estimation error because of its flexibility in estimating parameters stage by stage instead of acquiring all at once. On top of this procedure, the IV approach is applied. The IV approach, based on knowledge of the robot model, essentially compares the corrupted actual signal against the simulated noise-free data so as to improve the estimation quality. Unlike the classic Ordinary Least Square method that assumes normally-distributed noise in order to avoid estimation bias, the IV algorithm is able to relax this assumption. This is important in practice as the uncertainty between encoder pulses, which are usually used to take position measurement, is not necessarily normally distributed. From the simulation results, the IV approach has successfully demonstrated its ability to improve the estimation accuracy.

In addition to the accuracy of parameter estimation, the controller used in the robotic arm may affect the quality of the command profile. In the previous work, a feedback controller called Computed Torque Control has been implemented to track the command profile, and to reject disturbances. However, it remains uncertain if this controller can potentially inject energy which has been taken out by the command profile back into the system. To analyze, a simulation starts from a perfectly-decoupled closed-loop system. From the results, it is found that residual vibration peaks when the controller places poles around the resonant frequency. This is mainly due to the frequency response of a second-order controller that leaks energy around

its natural frequency. Since, from the mathematical derivation, the coupling effect is transformed into the input/output relation between the link subsystem and the motor subsystem, energy that goes into the motor subsystem at its natural frequency will also enter the link subsystem. As a result, when controller poles are near the system resonant frequencies, strong residual vibration is observed. Therefore, it is suggested to have critically-damped PD controllers in the servo-based part of the computed torque control to avoid injecting energy back to the robot system.

In practice, estimation error exists; thus, the impact from a mismatched model has been analyzed. In the command shaping method, shaping bounds are generated based on the nominal model and aim to incorporate the robustness into the command profile. On the other hand, in the presence of estimation error, the closed-loop dynamic is different. Based on the mathematical derivation, it is found that the actual resonant frequency is not only a function of robot configuration but also a function of control gains. Affected by the control gains, the resonant frequency curve enters or leaves the shaping bounds. This result becomes interesting as it indicates that, for a certain range of model estimation error, there exists a set of optimal gains so that the actual resonant frequency stays inside the shaping bounds, and the command shaping method can remove most of the critical energy.

To conclude, simulation results show that the IV algorithm is able to improve the accuracy of parameter estimation. Based on the simulation results from the perfectly-decoupled system as well as observations from the mismatched model, recommendations are provided at the end of Chapter 4 to choose control gains.

5.2 Contributions

This work has several unique contributions. A system identification approach called Instrumental Variable is applied from the literature. A framework for combining the Ordinary Least Square approach with the IV method is provided. Simulation results have shown significant improvement in the accuracy of parameter estimation.

Flexibility in terms of data preprocessing is also built into the IV approach to achieve better results. In general, this part of the work not only validates the model-based system ID approach but also provides the command shaping method a more accurate model when developing the command profile.

In a two-link robot system, the resonant frequencies are configuration-dependent. Thus, another contribution herein lies in the analysis of combining the computed torque control with the command shaping method when estimation error exists. Based on numerical observations, eigenvalues due to the controller interact with those coming from the link subsystem. As a result, the closed-loop resonant frequency depends on both the control gains as well as the robot configuration. Simulation results have shown that, for a certain range of estimation error, the connection between the shaping bounds and the controller leads to an optimal control gain that minimizes residual vibration. In light of all observations, recommendations for choosing control gains are given.

The MATLAB code that uses the IV approach to perform system identification has been generated based on the robot model so that it can be used for experimental data. The lookup table of the resonant frequency has been replaced with a new function called "find_freq.m" which incorporates the controller information and estimation error to find the closed-loop resonant frequency.

5.3 Future Work

There are some areas of research that could be examined in the future. Since results from the new system identification approach are based on simulation, and experiment of Stage One, it would be worthwhile to collect experimental data of Stage Two and Stage Three. A comparison could be made between the OLS method and the IV approach, in order to validate its improvement. Since the IV approach in this work is "off-line", its "on-line" version would be appealing because it could potentially

improve the model-based controller. That is to say, the computed torque controller could incorporate the on-line IV approach into its model-based compensator.

Estimation improvement could also be achieved at different stages. Thanks to the flexibility of the sequential identification procedure, it is possible to include a more detailed motor model during the identification process. With the help of this "upgraded" motor model, precise estimation of the motor inertia and damping ratio could be obtained and used at the second stage.

As mentioned, when estimation error exists, the width of shaping bounds and the control gain simultaneously affect the residual vibration. Therefore, it is possible to find an optimal set of gains so that more critical energy associated with the vibration could be taken out by using the command shaping method. Moreover, even when the system is perfectly-decoupled, the coupling effect still exists between the robot links; therefore, it implies that the controller implemented on one link is able to affect the rest of the link system. In this work, simulation results used a controller with damping ratio $\zeta_{3/4,ctrl} = 1$ and the same $\omega_{3/4,ctrl}$ for both motors. Thus, it would be important to study the influence due to various sets of control gains. It is also found that the shaping bounds that are based on different robot configurations can affect the residual vibration. Thus, a methodology to locate the shaping bounds is attractive. Based on the idea that the controller could push actual closed-loop resonant frequency into the shaping bounds region, when applying other types of controllers rather than the computed torque control, similar analysis could be conducted in order to limit the impact on the command profile.

REFERENCES

REFERENCES

[1] William Singhose. Command shaping for flexible systems: A review of the first 50 years. *International journal of precision engineering and manufacturing*, 10(4):153–168, 2009.

[2] Jun Wu, Jinsong Wang, and Zheng You. An overview of dynamic parameter identification of robots. *Robotics and computer-integrated manufacturing*, 26(5):414–419, 2010.

[3] Syed Ali Ajwad, Jamshed Iqbal, Muhammad Imran Ullah, and Adeel Mehmood. A systematic review of current and emergent manipulator control approaches. *Frontiers of mechanical engineering*, 10(2):198–210, 2015.

[4] P. H. Meckl. *Control of Vibration in Mechanical Systems Using Shaped Reference Inputs*. PhD thesis, Massachusetts Institute of Technology, 1988.

[5] S. P. Bhat and D. K. Miu. Precise point-to-point positioning control of flexible structures. *Journal of Dynamic Systems, Measurement, and Control*, 112(4):667–674, 12 1990.

[6] Otto JM Smith. Posicast control of damped oscillatory systems. *Proceedings of the IRE*, 45(9):1249–1255, 1957.

[7] N. C. Singer and W. P. Seering. Preshaping command inputs to reduce system vibration. *Journal of Dynamic Systems, Measurement, and Control*, 112(1):76–82, 03 1990.

[8] Z Mohamed and MO Tokhi. Command shaping techniques for vibration control of a flexible robot manipulator. *Mechatronics*, 14(1):69–90, 2004.

[9] P. H. Meckl. *Minimizing Residual Vibration of a Linear System Using Appropriately Shaped Forcing Functions*. Master's thesis, Massachusetts Institute of Technology, 1984.

[10] MO Tokhi and AKM Azad. Flexible robot manipulators. In *Modelling, Simulation and Control*, volume 68. The Institution of Engineering and Technology London, 2008.

[11] Victor Maxwell Beazel. *Command shaping applied to nonlinear systems with configuration-dependent resonance*. PhD thesis, Purdue University, 2004.

[12] Withit Chatlatanagulchai, Victor M Beazel, and Peter H Meckl. Command shaping applied to a flexible robot with configuration-dependent resonance. In *2006 American Control Conference*, page 6. IEEE, 2006.

[13] Lucy Y Pao and Mark A Lau. Robust input shaper control design for parameter variations in flexible structures. *J. Dyn. Sys., Meas., Control*, 122(1):63–70, 2000.

[14] Alok Agrawal. *Constrained Optimized Command Shaping for Minimizing Residual Vibration in a Flexible-Joint Robot*. Master's thesis, Purdue University, 2015.

[15] Yumeng Wu. *Command shaping with constrained peak input acceleration to minimize residual vibration in a flexible-joint robot*. Master's thesis, Purdue University, 2016.

[16] Bowen Wu. *Numerical Optimized Command Shaping with Constrained Peak Input Acceleration for Minimizing Residual Vibration*. Master's thesis, Purdue University, 2017.

[17] Pratheek Patil. *Input Command Shaping Using the Versine Function with Peak Acceleration Constraint and Numerical Optimization to Minimize Residual Vibration*. Master's thesis, Purdue University Graduate School, 2019.

[18] Sigeru Futami, Nobuhiro Kyura, and Shujiro Hara. Vibration absorption control of industrial robots by acceleration feedback. *IEEE transactions on Industrial Electronics*, (3):299–305, 1983.

[19] L. M. Sweet and Malcolm Good. Re-definition of the robot motion control problem: Effects of plant dynamics, drive system constraints, and user requirements. *The 23rd IEEE Conference on Decision and Control*, pages 724–732, 1984.

[20] Ryan Sangjun Lee. *Optimal parameter estimation for long-term prediction in the presence of model mismatch applied to a two-link flexible joint robot*. PhD thesis, Purdue University, 2011.

[21] Pradeep K Khosla and Takeo Kanade. Parameter identification of robot dynamics. In *1985 24th IEEE Conference on Decision and Control*, pages 1754–1760. IEEE, 1985.

[22] Alexander Scheel. Identification of a robot with joint flexibility, course project report, 2012.

[23] Andrès Vivas, Philippe Poignet, Frédéric Marquet, François Pierrot, and Maxime Gautier. Experimental dynamic identification of a fully parallel robot. In *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, volume 3, pages 3278–3283. IEEE, 2003.

[24] Maxime Gautier. Dynamic identification of robots with power model. In *Proceedings of International Conference on Robotics and Automation*, volume 3, pages 1922–1927. IEEE, 1997.

[25] Maxime Gautier, Alexandre Janot, and Pierre-Olivier Vandanjon. A new closed-loop output error method for parameter identification of robot dynamics. *IEEE Transactions on Control Systems Technology*, 21(2):428–444, 2012.

[26] Jingfu Jin and Nicholas Gans. Parameter identification for industrial robots with a fast and robust trajectory design approach. *Robotics and Computer-Integrated Manufacturing*, 31:21–29, 2015.

[27] L. Ljung. *System Identification: Theory for the User*. Prentice Hall information and system sciences series. Prentice Hall PTR, 1999.

[28] Rolf Isermann. *Identification of Dynamic Systems An Introduction with Applications*. Advanced textbooks in control and signal processing. Springer Berlin Heidelberg, 2011.

[29] Oliver Nelles. *Nonlinear System Identification From Classical Approaches to Neural Networks and Fuzzy Models*. Springer Berlin Heidelberg : Imprint: Springer, 2001.

[30] H. Moulin and E. Bayo. On the Accuracy of End-Point Trajectory Tracking for Flexible Arms by Noncausal Inverse Dynamic Solutions. *Journal of Dynamic Systems, Measurement, and Control*, 113(2):320–324, 06 1991.

[31] P. Tomei. A simple pd controller for robots with elastic joints. *IEEE Transactions on Automatic Control*, 36(10):1208–1213, Oct 1991.

[32] Mathukumalli Vidyasagar. *Nonlinear systems analysis*, volume 42. Siam, 2002.

[33] Jean-Jacques E Slotine, Weiping Li, et al. *Applied nonlinear control*, volume 199. Prentice hall Englewood Cliffs, NJ, 1991.

[34] Alessandro De Luca. Dynamic control of robots with joint elasticity. In *Proceedings. 1988 IEEE International Conference on Robotics and Automation*, pages 152–158. IEEE, 1988.

[35] John J. Craig. *Introduction to robotics : mechanics and control*. Pearson/Prentice Hall, Upper Saddle River, N.J., 3rd ed. edition, 2005.

[36] Withit Chatlatanagulchai. *Backstepping intelligent control applied to a flexible-joint robot manipulator*. PhD thesis, Purdue University, 2006.

[37] Hyuk C Nho. *Precise motion control of flexible-joint robot manipulators with an intelligent payload estimator*. PhD thesis, Purdue University, 2004.

[38] MW Spong. Modeling and control of elastic joint robots. *Mathematical and Computer Modelling*, 12(7):912, 1989.

[39] Gerard Lallement and Daniel J Inman. A tutorial on complex eigenvalues. In *Proceedings-SPIE the International Society for Optical Engineering*, pages 490–490. SPIE International Society for Optical, 1995.

APPENDICES

A. RESONANT FREQUENCY FOR PERFECTLY DECOUPLED ROBOTIC ARM

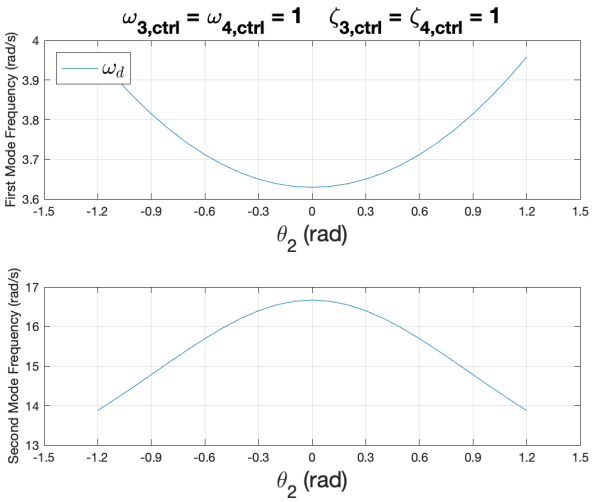


Figure A.1. Resonant Frequencies of the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 1$

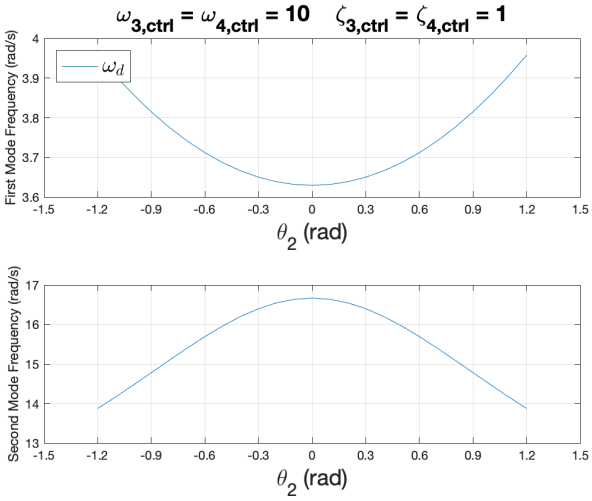


Figure A.2. Resonant Frequencies of the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 10$

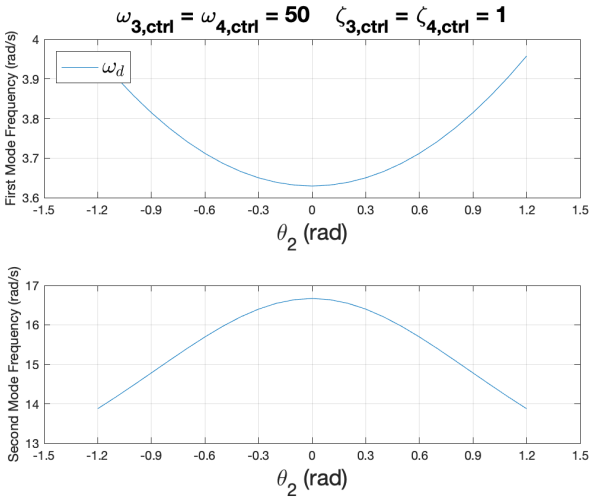


Figure A.3. Resonant Frequencies of the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 50$

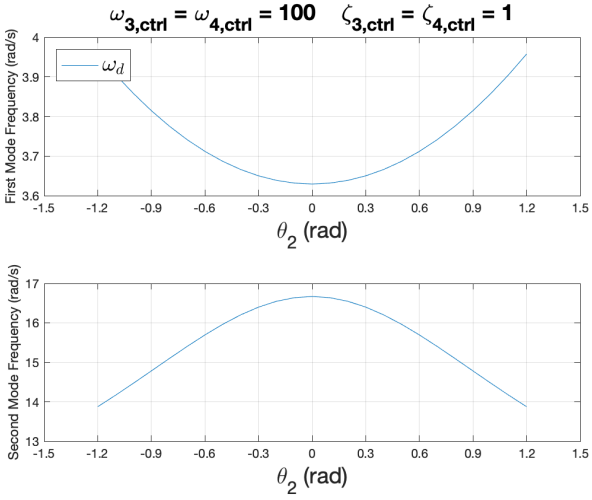


Figure A.4. Resonant Frequencies of the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 100$

B. RESONANT FREQUENCY OF ROBOTIC ARM WITH MISMATCH MODEL

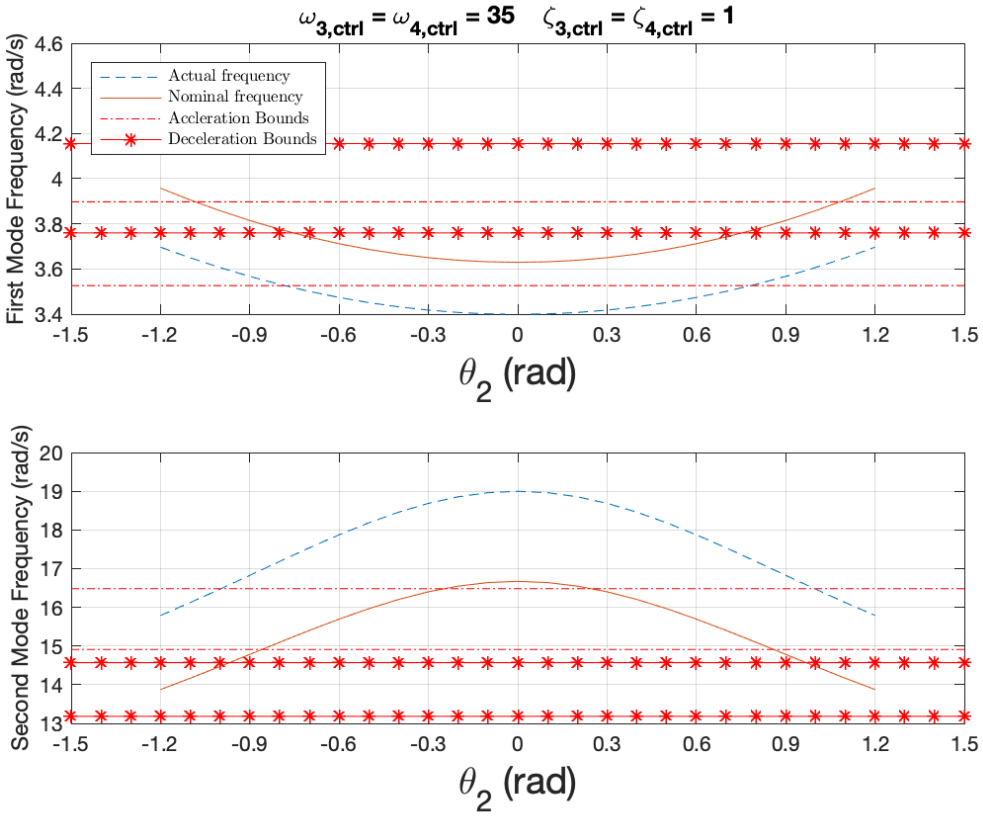


Figure B.1. ±5% Shaping Bounds of Resonant Frequencies for the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 35$, and Mismatch Factor of Stiffness $\mathbf{MF_K} = 75\%$

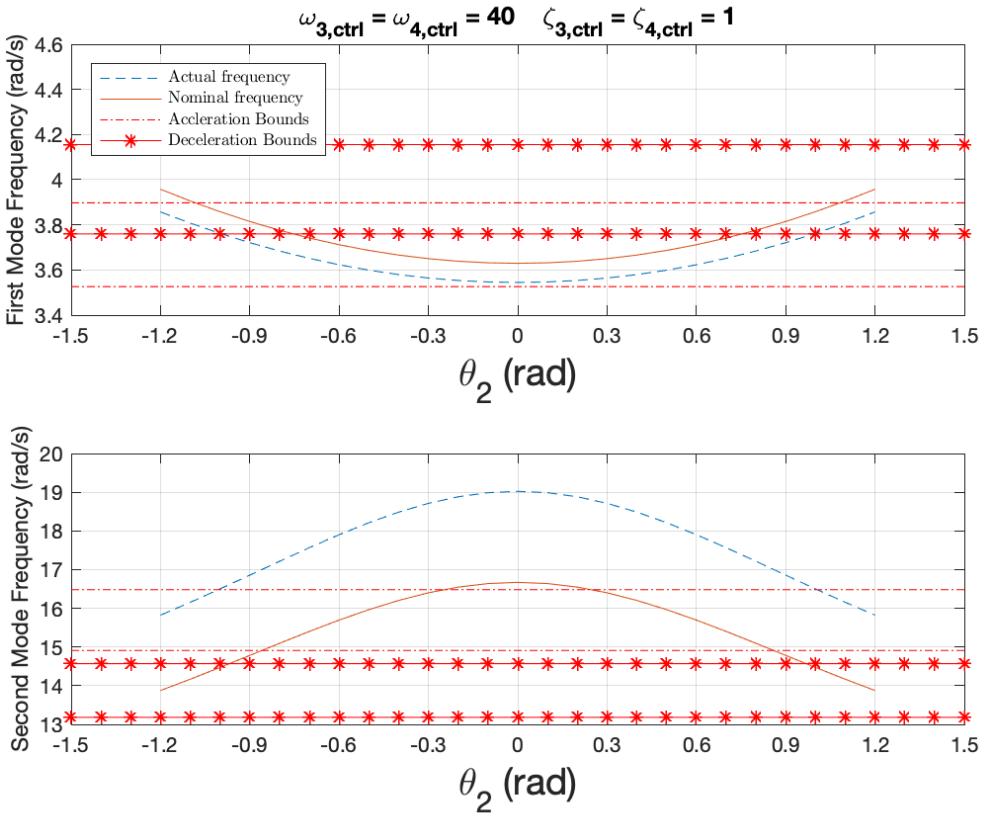


Figure B.2. ±5% Shaping Bounds of Resonant Frequencies for the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 40$, and Mismatch Factor of Stiffness $\mathbf{MF_K} = 75\%$

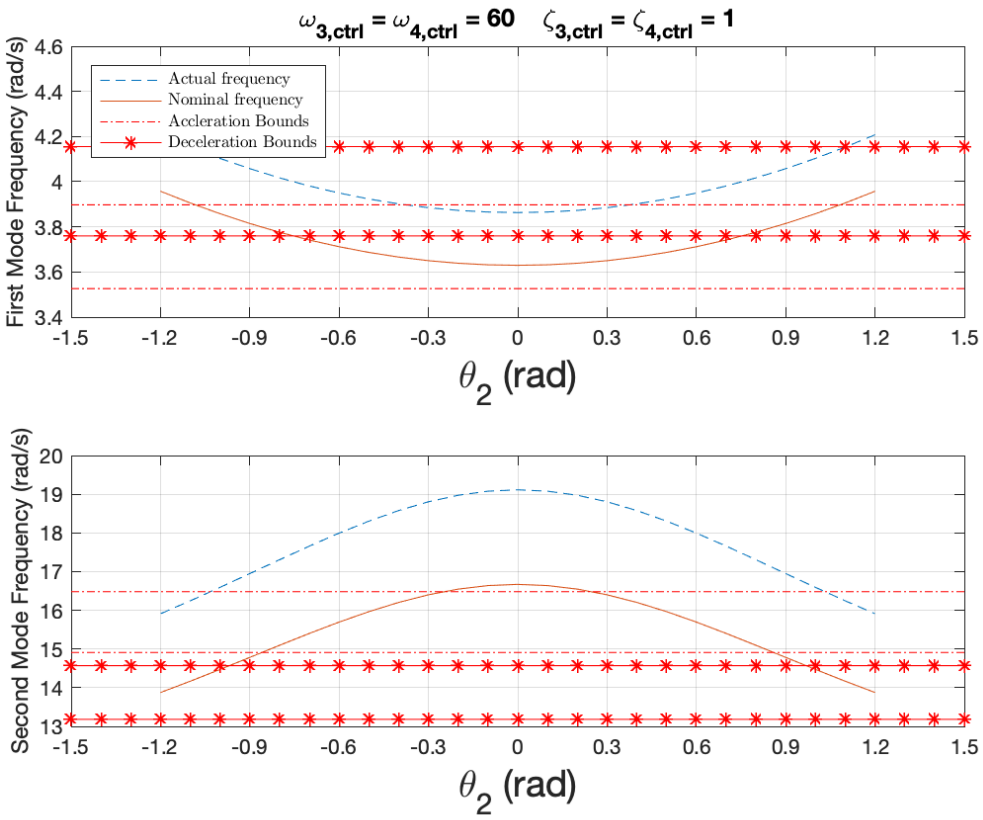


Figure B.3. ±5% Shaping Bounds of Resonant Frequencies for the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 60$, and Mismatch Factor of Stiffness $\mathbf{MF_K} = 75\%$

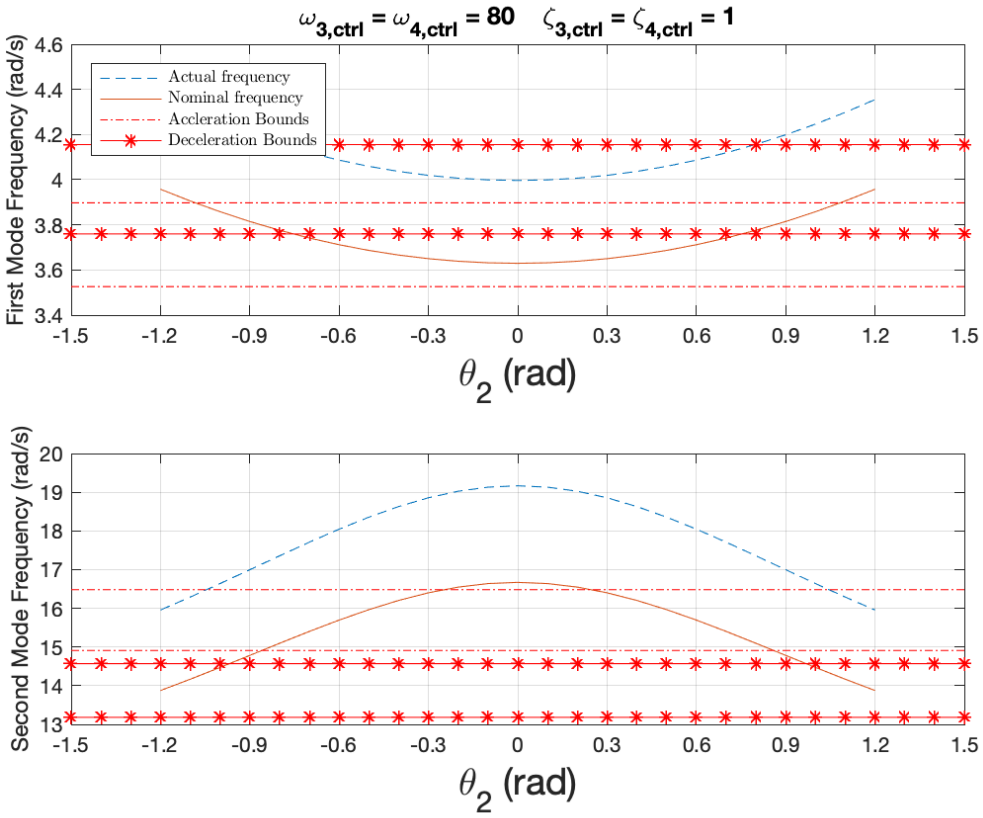


Figure B.4. ±5% Shaping Bounds of Resonant Frequencies for the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 80$, and Mismatch Factor of Stiffness $\mathbf{MF_K} = 75\%$

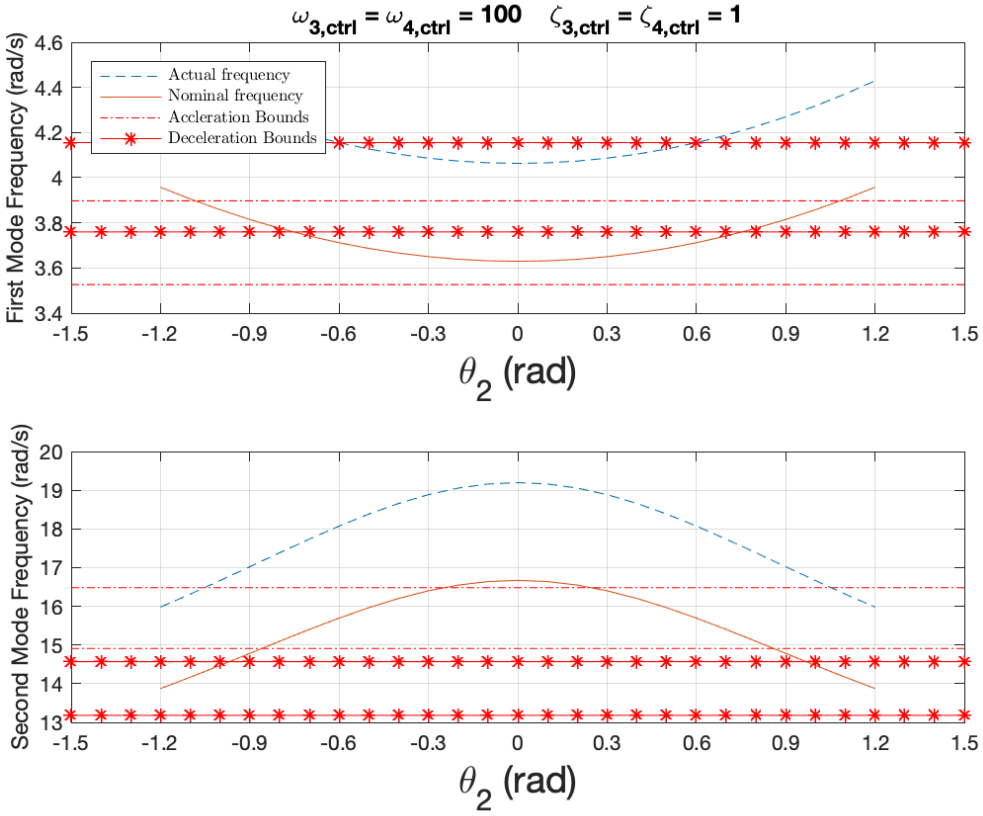


Figure B.5. ±5% Shaping Bounds of Resonant Frequencies for the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 100$, and Mismatch Factor of Stiffness $\mathbf{MF_K} = 75\%$

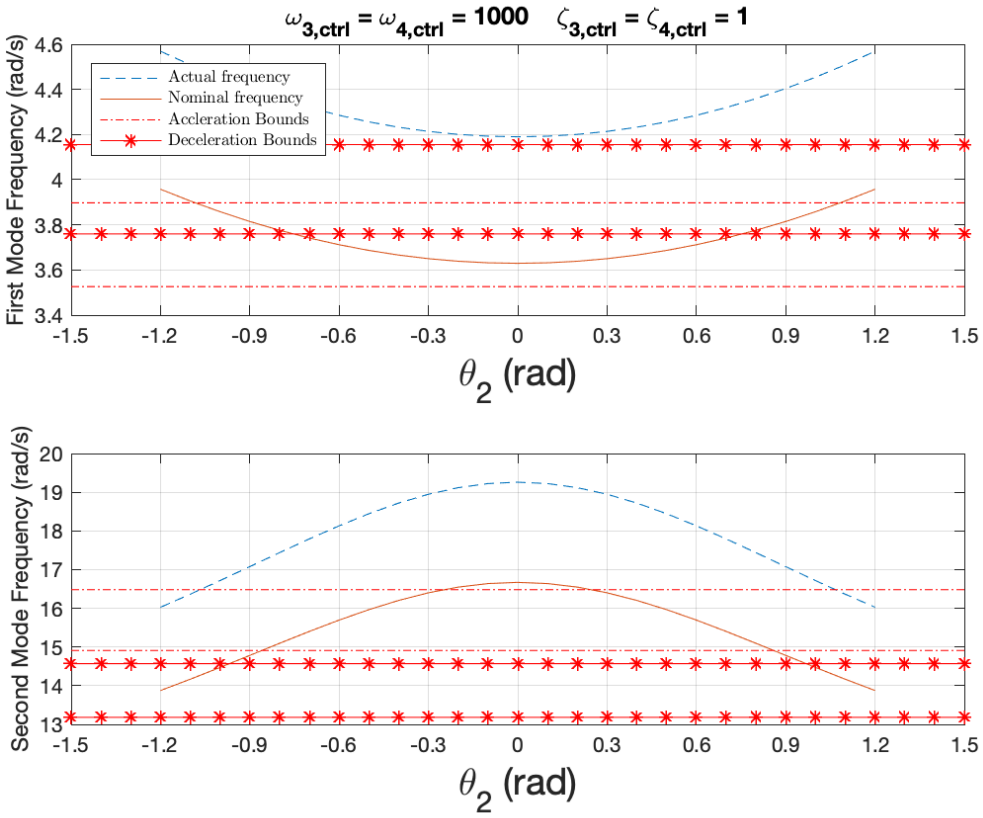


Figure B.6. ±5% Shaping Bounds of Resonant Frequencies for the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 1000$, and Mismatch Factor of Stiffness $\mathbf{MF_K} = 75\%$

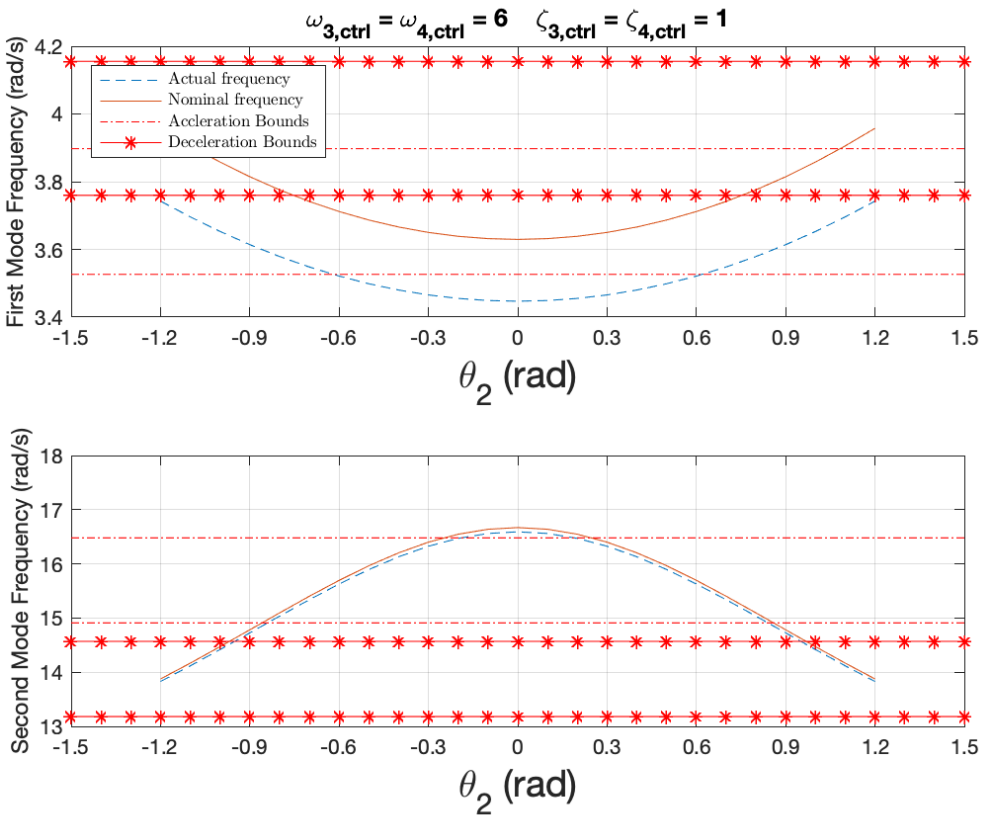


Figure B.7. ±5% Shaping Bounds of Resonant Frequencies for the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 6$, and Mismatch Factor of Stiffness $\mathbf{MF_C} = 65\%$

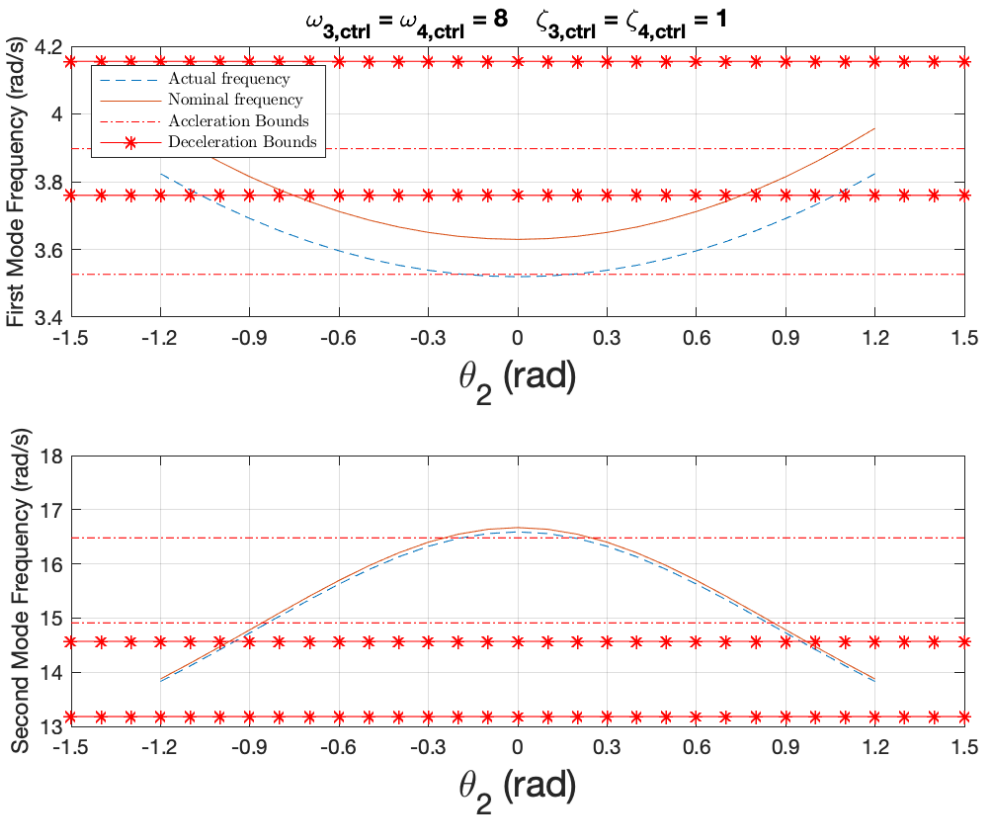


Figure B.8. ±5% Shaping Bounds of Resonant Frequencies for the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 8$, and Mismatch Factor of Stiffness $\mathbf{MF_C} = 65\%$

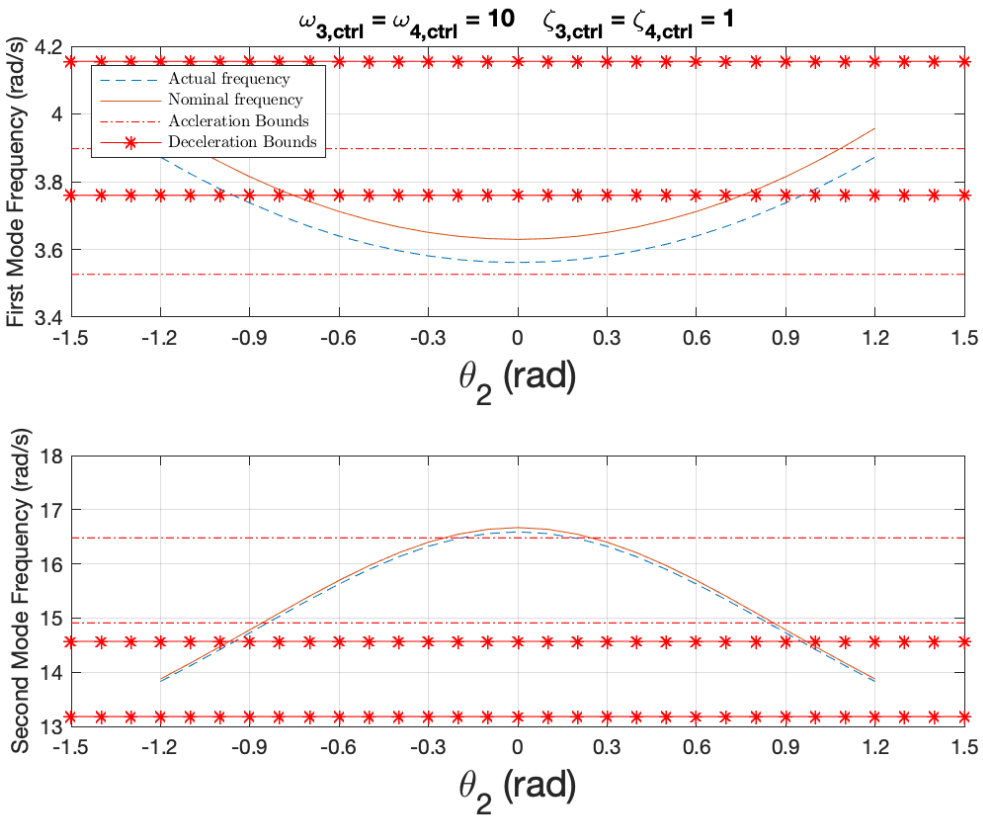


Figure B.9. ±5% Shaping Bounds of Resonant Frequencies for the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 10$, and Mismatch Factor of Stiffness $\mathbf{MF_C} = 65\%$

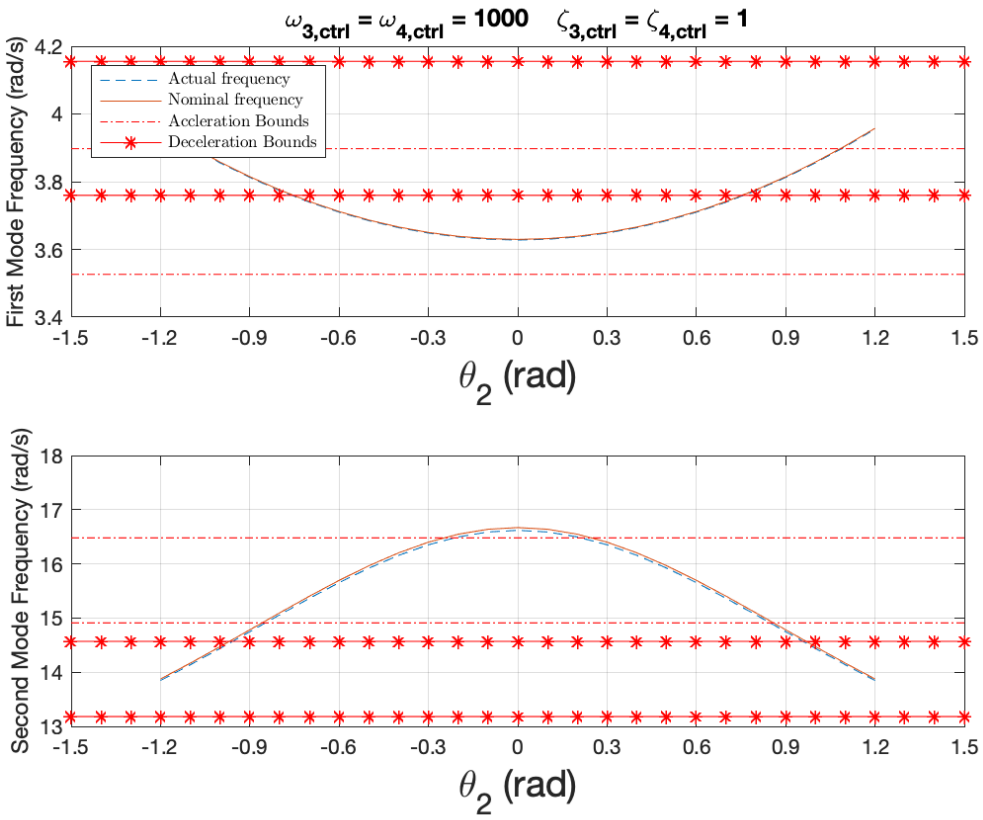
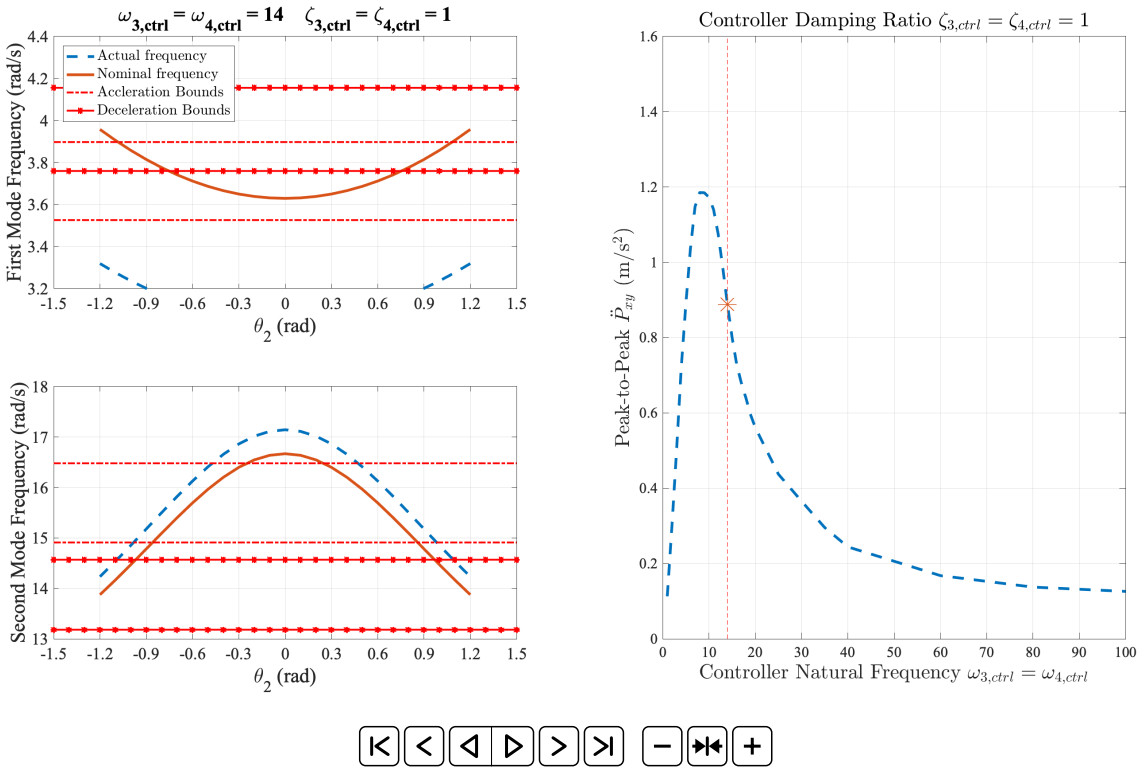
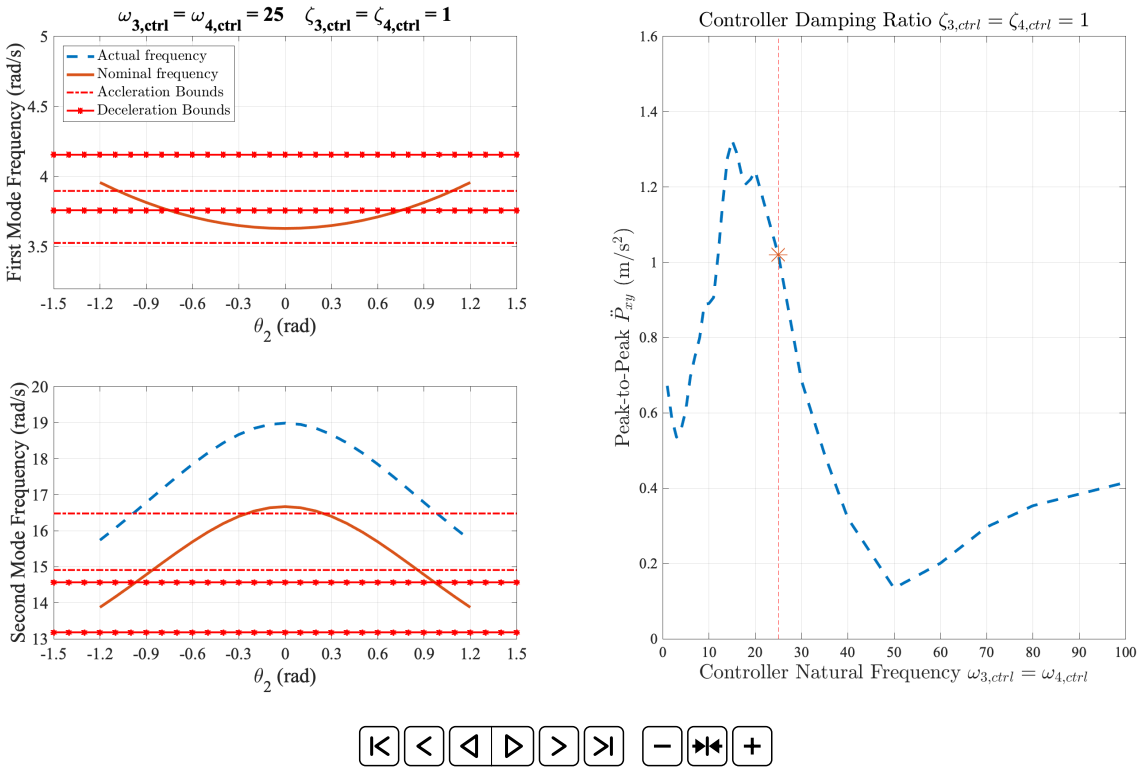


Figure B.10. ±5% Shaping Bounds of Resonant Frequencies for the Robotic Arm with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and $\omega_{3,ctrl} = \omega_{4,ctrl} = 1000$, and Mismatch Factor of Stiffness $\mathbf{MF_C} = 65\%$

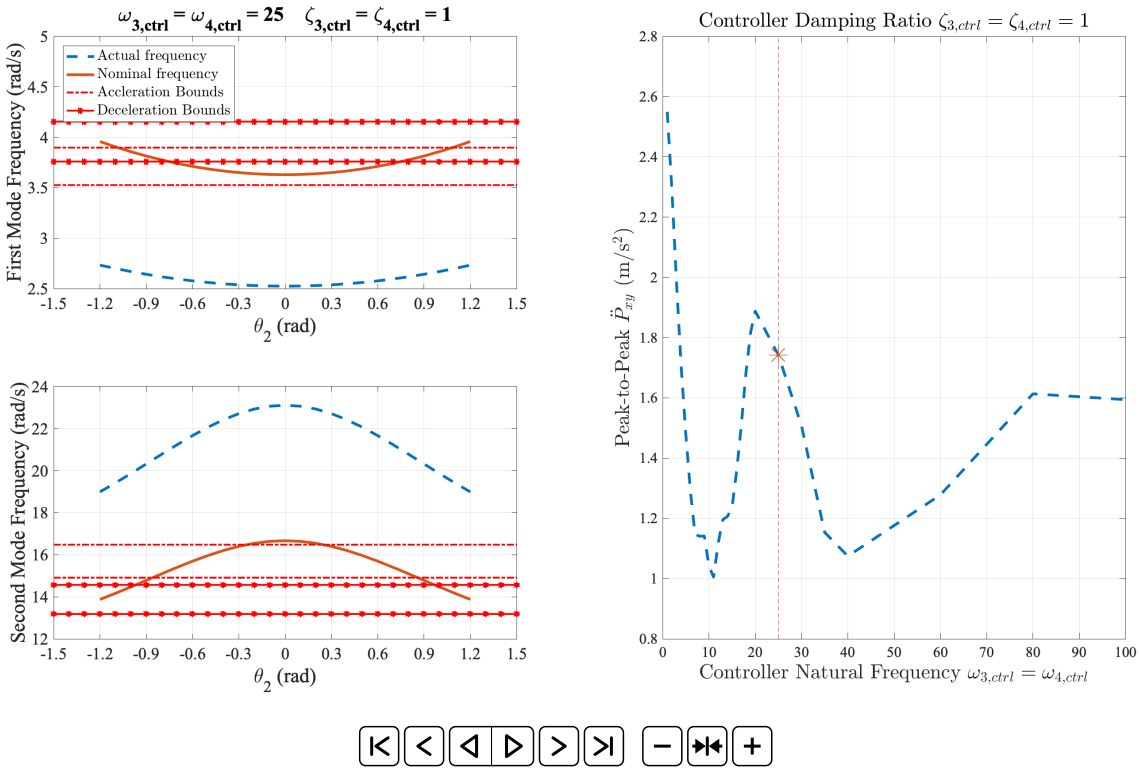
C. ANIMATION



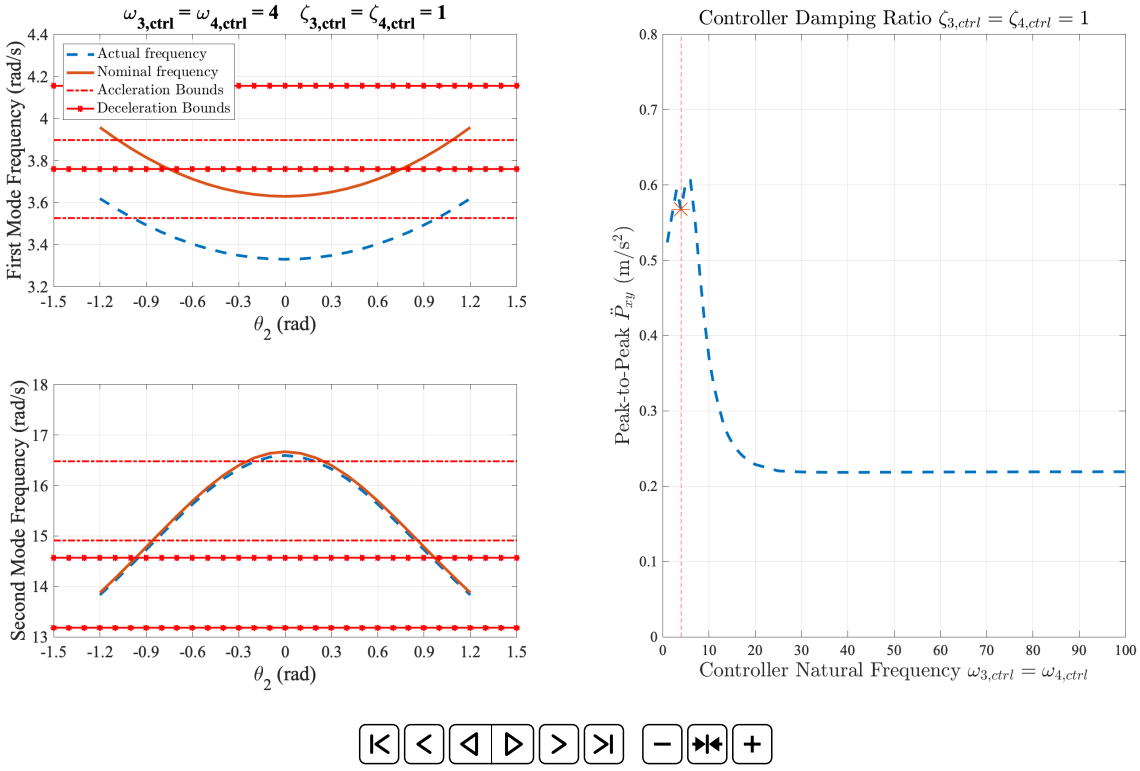
Animation C.1. Actual Closed-loop Resonant Frequencies with ±5% Shaping Bounds(left) and Residual Vibration of the Robotic Arm(right) with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and variable $\omega_{3,ctrl} = \omega_{4,ctrl}$, and Mismatch Factor of Stiffness $\mathbf{MF_K} = 95\%$



Animation C.2. Actual Closed-loop Resonant Frequencies with ±5% Shaping Bounds(left) and Residual Vibration of the Robotic Arm(right) with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and variable $\omega_{3,ctrl} = \omega_{4,ctrl}$, and Mismatch Factor of Stiffness $\mathbf{MF_K} = 75\%$



Animation C.3. Actual Closed-loop Resonant Frequencies with ±5% Shaping Bounds(left) and Residual Vibration of the Robotic Arm(right) with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and variable $\omega_{3,ctrl} = \omega_{4,ctrl}$, and Mismatch Factor of Stiffness $\mathbf{MF_K} = 50\%$



Animation C.4. Actual Closed-loop Resonant Frequencies with ±5% Shaping Bounds(left) and Residual Vibration of the Robotic Arm(right) with Controller Parameters $\zeta_{3,ctrl} = \zeta_{4,ctrl} = 1$ and variable $\omega_{3,ctrl} = \omega_{4,ctrl}$, and Mismatch Factor of Stiffness $\mathbf{MF_C} = 65\%$

D. SYSTEM IDENTIFICATION PAPER

INNOVATIVE SYSTEM IDENTIFICATION APPROACH FOR TWO-LINK ROBOTIC ARM WITH JOINT FLEXIBILITY

Ruiwen Wei
School of Mechanical Engineering
Purdue University
West Lafayette, Indiana 47906
e-mail: wei144@purdue.edu

Peter Meckl
Assistant Head Professor
Mechanical Engineering
Purdue University
585 Purdue Mall, West Lafayette, IN 47907
e-mail: meckl@purdue.edu

ABSTRACT

A robotic arm has become a popular choice not only in research labs but also in traditional industry, more research has focused in this area these days. Unlike a rigid-body robot, a robot with joint flexibility could introduce more vibration at its end point. Many strategies have been proposed to solve this issue but most of them require an accurate estimation of model parameters. In this paper, an innovative system identification method, called Instrumental Variable, is applied from literature to improve the estimation obtained from the classic method-Ordinary Least Square. Simulation results that are based on a two-link joint-flexible robotic arm model are provided at the end, and show that the IV algorithm is able to obtain better estimation.

INTRODUCTION

Robotic arm has been widely used in modern mass production line because, with high programmability and mobility, robotic arm is able to perform various kinds of work at lower cost comparing to traditional machine. Therefore, industry such as automation manufacturing utilizes high-precision robotic arms to perform welding, assembly, and so on with different end effectors. Depending on specific types used, robotic arm can has two to six links on its body, providing corresponding degrees of freedom in terms of mobility in space. While higher degrees of freedom means more sophisticate trajectories that can be achieved, it also brings more parameter uncertainty from each link into the system, limiting overall performance in terms of motion accuracy and settling time from the control perspective.

To identify the system parameter, different approaches for robots have been reviewed by Jun Wu [1], pointing out that among physical experiments, computer aided design, and input/output identification methods, the last one gives more accurate estimation as well as simpler experiment. Furthermore, [2–5] provides modified estimation methods based on the classic Ordinary Lease Square (OLS), either improving or simplifying the identification process. However, these work assume normal-distributed noise presenting in the measurement data, which theoretically leads to bias estimation if this assumption is violated in practice [6–8]. Therefore, it is of interest to investigate and find a practical approach that theoretically improves and even avoids bias estimation.

A method called Instrumental Variable (IV) is re-discovered in [8], and found that it requires no prior knowledge on noise structure during the estimation process. This concept was first derived by Philip Wright and Sewall Wright in 1920s, targeting statistics estimation problems. P. Young later applied this method in solving engineering estimation problems [9, 10]. Recently, Alexandre [11] and Mathieu [12] applied the IV approach to estimate parameters of rigid-body-model robots. However, assuming rigid-body model without joint flexibility for robots during the identification process may results in bias estimation due to model mismatch, and limits the model-based controller performance as flexibility could introduce residual vibration onto the end effector in fast point-to-point motion [13].

This paper applies the Instrumental Variable method in order to optimize the accuracy of parameters estimated by the classic OLS approach. To simplify, a custom-built two-link robot with joint flexibility at Ruth and Joel Spira Laboratory for Electromechanical Systems in the School of Mechanical Engineering at Purdue University is used. To focus more on effects brought by the joint flexibility, the robot is operated in a horizontal plane where gravity influence is minimized. [14, 15] provide different procedures to perform system identification. One idea among

them is to identify each link and motor stage by stage. This sequential identification procedure is adopted as it is able to capture essential dynamic behavior in each stage while minimizing the coupled effect brought by different mechanical parts.

The rest of the paper is organized as follows. Modeling section provides mathematical model to represent the robotic arm being simulated, and sets up corresponding estimation model for identification at each stage. Simulation Design section shows necessary setups for the entire simulation process and performs mode validation. Estimation Optimization section first uses the classic OLS method for estimation and then combines the IV approach for improvement, followed by the Results section showing analysis from different perspectives.

MODELING

In order to understand the system dynamic response, a model of the two-link, flexible-joint manipulator is provided in this section. Detailed schematic is shown in Figure 1[ADD REF?]. To derive the equation of motion of this system, Lagrange method is applied and derived by Nho [16]. The governing equation is given by Equation (1) below. According to the Lagrange method, independent generalized coordinates are selected so that the whole system could be represented in terms of kinetic energy (K), potential energy (P), dissipative power (D), and the generalized force (Q_i). In this system, θ_1 and θ_2 are generalized coordinates for link 1 and link 2 respectively while θ_3 and θ_4 are generalized coordinates of motors. Note that θ_5 and θ_6 are not independent as they are related to the generalized motor coordinates by the gear ratio r.

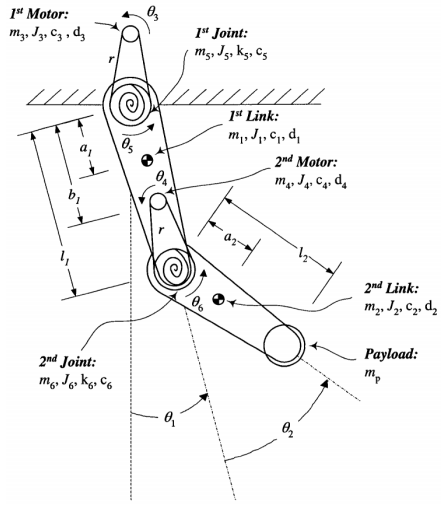


FIGURE 1. Schematic of two link, flexible-joint manipulator

$$\frac{d}{dt}\left(\frac{\delta K}{\delta \dot{\theta}_i}\right) - \frac{\delta K}{\delta \theta_i} + \frac{\delta P}{\delta \theta_i} + \frac{\delta D}{\delta \dot{\theta}_i} = Q_i \quad (i = 1,2,3,4) \tag{1}$$

The kinetic energy of the robotic arm is given in Equation (2).

$$\begin{aligned} K = {} & \frac{1}{2}(m_1 a_1^2 + m_2 l_1^2 + m_4 b_1^2 + m_6 l_1^2 + m_p l_1^2 + J_1)\dot{\theta}_1^2 \\ & + \frac{1}{2}(J_3 + \frac{J_5}{r^2})\dot{\theta}_3^2 + \frac{1}{2}(m_2 a_2^2 + J_2 + m_p l_2^2)(\dot{\theta}_1 + \dot{\theta}_2)^2 \\ & + \frac{1}{2}J_4(\dot{\theta}_1 + \dot{\theta}_4)^2 + \frac{1}{2}J_6(\frac{\dot{\theta}_4}{r} + \dot{\theta}_1)^2 \\ & + l_1(m_2 a_2 + m_p l_2)\dot{\theta}_1(\dot{\theta}_1 + \dot{\theta}_2)cos(\theta_2) \end{aligned} \tag{2}$$

The potential energy of the robotic arm is given in Equation (3).

$$P = \frac{1}{2}k_5(\frac{\theta_3}{r} - \theta_1)^2 + \frac{1}{2}k_6(\frac{\theta_4}{r} - \theta_2)^2 \tag{3}$$

In this equation, k_i represents coefficient of torsional spring. The dissipative energy of the robotic arm is given in Equation (4).

$$\begin{aligned} D = {} & \frac{1}{2}c_3\dot{\theta}_3^2 + \frac{1}{2}c_1\dot{\theta}_1^2 + \frac{1}{2}c_5(\frac{\dot{\theta}_3}{r} - \dot{\theta}_2)^2 \\ & + \frac{1}{2}c_4\dot{\theta}_4^2 + \frac{1}{2}c_2\dot{\theta}_2^2 + \frac{1}{2}c_3(\frac{\dot{\theta}_4}{r} - \dot{\theta}_2)^2 \end{aligned} \tag{4}$$

In this equation, c_i represents coefficient of friction that exists in connections. The generalized force of the robotic arm are given in Equation (5).

$$\begin{aligned} Q_1 &= -d_1 sign(\dot{\theta}_1) \\ Q_2 &= -d_2 sign(\dot{\theta}_2) \\ Q_3 &= T_1 - d_3 sign(\dot{\theta}_3) \\ Q_4 &= T_2 - d_4 sign(\dot{\theta}_4) \end{aligned} \tag{5}$$

In this equation, d_i represents coefficient of Coulomb friction, and "sign" for the Coulomb friction function.

Matrix form

After substituting Equations (2) to (5) into Equation (1), the result can be represented in a matrix form shown in Equation (6).

$$\mathbf{M}(\theta)\ddot{\theta} + \mathbf{V}(\theta, \dot{\theta}) + \mathbf{C}\dot{\theta} + \mathbf{D}(\theta) + \mathbf{K}\theta = \mathbf{T} \tag{6}$$

where

$$\mathbf{M} = \begin{bmatrix} \mathbf{M_1} & \mathbf{M_2} \\ \mathbf{M_2^T} & \mathbf{M_3} \end{bmatrix} \tag{7}$$

$$\mathbf{M_1} = \begin{bmatrix} \begin{pmatrix} m_1a_1^2 + m_2(l_1^2 + a_2^2) \\ +m_4b_1^2 + m_6l_1^2 + J_1 + J_2 \\ +J_4 + J_6 + m_p(l_1^2 + l_2^2) \\ +2l_1(m_2a_2 + m_pl_2)cos(\theta_2) \end{pmatrix} & \begin{pmatrix} m_2a_2^2 + J_2 \\ +m_pl_2^2 + l_1(m_2a_2 \\ +m_pl_2)cos(\theta_2) \end{pmatrix} \\ \begin{pmatrix} m_2a_2^2 + J_2 + m_pl_2^2 \\ +l_1(m_2a_2 + m_pl_2)cos(\theta_2) \end{pmatrix} & \left(m_2a_2^2 + J_2 + m_pl_2^2\right) \end{bmatrix} \tag{8}$$

$$\mathbf{M_2} = \begin{bmatrix} 0 & (J_4 + \frac{J_6}{r}) \\ 0 & 0 \end{bmatrix} \tag{9}$$

and

$$\mathbf{M_3} = \begin{bmatrix} (J_3 + \frac{J_5}{r^2}) & 0 \\ 0 & (J_4 + \frac{J_6}{r^2}) \end{bmatrix} \tag{10}$$

$$\mathbf{V}(\theta, \dot{\theta}) = \begin{bmatrix} -l_1(m_2a_2 + m_pl_2)(2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2)sin(\theta_2) \\ l_1(m_2a_2 + m_pk_2)\dot{\theta}_1^2 sin(\theta_2) \\ 0 \\ 0 \end{bmatrix} \tag{11}$$

$$\mathbf{C} = \begin{bmatrix} c_1 + c_5 & 0 & -\frac{c_5}{r} & 0 \\ 0 & c_2 + c_6 & 0 & -\frac{c_6}{r} \\ -\frac{c_5}{r} & 0 & c_3 + \frac{c_5}{r^2} & 0 \\ 0 & -\frac{c_6}{r} & 0 & c_4 + \frac{c_5}{r^2} \end{bmatrix} \tag{12}$$

$$\mathbf{D}(\dot{\theta}) = \begin{bmatrix} d_1sign(\dot{\theta}_1) \\ d_2sign(\dot{\theta}_2) \\ d_3sign(\dot{\theta}_3) \\ d_4sign(\dot{\theta}_4) \end{bmatrix} \tag{13}$$

$$\mathbf{K} = \begin{bmatrix} k_5 & 0 & -\frac{k_5}{r} & 0 \\ 0 & k_6 & 0 & -\frac{k_6}{r} \\ -\frac{k_5}{r} & 0 & \frac{k_5}{r^2} & 0 \\ 0 & -\frac{k_6}{r} & 0 & \frac{k_6}{r^2} \end{bmatrix} \tag{14}$$

$$\mathbf{T} = \begin{bmatrix} 0 \\ 0 \\ T_1 \\ T_2 \end{bmatrix} \tag{15}$$

Regression Model

In general, identification for the two-link robot system could be divided into three stages [17]. The first stage is to identify parameters of motors. Carrying estimated parameters from stage one, the second stage is only to estimate parameters on the second link while fixing the first link. The remaining stage would then be estimating parameters of the first link with the existence of coupling due to the second link. At this stage, estimated parameters from previous stages are treated as known variables. Ordinary least squares(OLS) regression method is used for system identification because the system is linear in parameter. To apply this method, rearrangement of Equation (6) is made and has the form as Equation (16). Note that the influence coming from dissipative energy of robotic arm is negligible so that the matrix **D** is dropped. To implement the estimation, standard RLS algorithm shown by Equation (17) to (19), the recursive version of OLS, is used instead, given that RLS consumes less computational power than OLS method.

$$\mathbf{W}_{Sj}(\theta, \dot{\theta}, \ddot{\theta}) * \mathbf{P}_{Sj} = \mathbf{Y}_{Sj}; \tag{16}$$

where $\mathbf{W}_{Sj}$ is the regressor matrix, $\mathbf{P}_{Sj}$ the matrix of parameters to be estimated, and $\mathbf{Y}_{Sj}$ the regressands matrix for stage one, two and three respectively when $Sj = S1, S2, S3$.

$$\hat{\mathbf{P}}_{Sj,RLS}(k) = \hat{\mathbf{P}}_{Sj,RLS}(k-1) + \mathbf{K}_{RLS}(k)[\mathbf{Y}(k) - \mathbf{W}(k)\hat{\mathbf{P}}_{Sj,RLS}(k-1)] \tag{17}$$

$$\mathbf{K}_{RLS}(k) = \frac{\mathbf{P}_{Var}(k-1)\mathbf{W}(k)}{\lambda + \mathbf{W}(k)^T\mathbf{P}_{Var}(k-1)\mathbf{W}(k)} \tag{18}$$

$$\mathbf{P}_{Var}(k) = \frac{1}{\lambda}[\mathbf{I} - \mathbf{K}_{RLS}(k)\mathbf{W}(k)^T]\mathbf{P}_{Var}(k-1) \tag{19}$$

where Equation (18) and (19) are used to calculate updates for next estimation iteration. k is the time stamp. The forgetting factor, λ is set to 1 given that the mechanical system could be considered as a deterministic system during the identification process. In general, $\mathbf{P}_{Var}(0)$ is set to $10^6 * \mathbf{I}$(identity matrix) and $\hat{\mathbf{P}}_{RLS}(0)$ to 0 for all parameters to initiate the recursive calculation.

Stage One: Identification of motors

The dynamic response of motor 1 and motor 2 is governing by the following equations respectively

$$\begin{aligned} J_3\ddot{\theta}_3 + c_3\dot{\theta}_3 &= T_1 \\ J_4\ddot{\theta}_4 + c_4\dot{\theta}_4 &= T_2 \end{aligned} \tag{20}$$

After rearranging the above equations, they could be written as

$$\mathbf{W}_{S1}(\theta,\dot{\theta},\ddot{\theta}) * \mathbf{P}_{S1} = \begin{bmatrix} \ddot{\theta}_i & \dot{\theta}_i \end{bmatrix} \begin{bmatrix} J_i \\ C_i \end{bmatrix} = \mathbf{Y}_{S1} = T_j \tag{21}$$

where i=3, 4 and j=1, 2 respectively.

Stage Two: Identification of the second link

The equation of motion for the second link is given in Equation (22) by fixing the first link.

$$\mathbf{M}_{L2}\ddot{\theta}_{L2} + \mathbf{C}_{L2}\dot{\theta}_{L2} + \mathbf{K}_{L2}\theta_{L2} = \mathbf{T}_{L2} \tag{22}$$

where

$$\theta_{L2} = \begin{bmatrix} \theta_2 \\ \theta_4 \end{bmatrix} \tag{23}$$

$$\mathbf{M}_{L2} = \begin{bmatrix} m_2 a_2^2 & 0 \\ 0 & J_4 + \frac{J_6}{r^2} \end{bmatrix} \tag{24}$$

$$\mathbf{C}_{L2} = \begin{bmatrix} c_2 + c_6 & -\frac{c_6}{r} \\ -\frac{c_6}{r} & c_4 + \frac{c_6}{r^2} \end{bmatrix} \tag{25}$$

$$\mathbf{K}_{L2} = \begin{bmatrix} k_6 & -\frac{k_6}{r} \\ -\frac{k_6}{r} & \frac{k_6}{r^2} \end{bmatrix} \tag{26}$$

$$\mathbf{T}_{L2} = \begin{bmatrix} 0 \\ T_2 \end{bmatrix} \tag{27}$$

Since J_4 and c_4 are estimated from identification on motors, they are taken cared by putting into the regressand matrix. The completely rearranged equation is given by

$$\mathbf{W}_{S2}(\theta,\dot{\theta},\ddot{\theta}) * \mathbf{P}_{S2} = \mathbf{Y}_{S2} \tag{28}$$

where the regressor matrix is given by

$$\mathbf{W}_{S2} = \begin{bmatrix} \ddot{\theta}_2 & 0 & \dot{\theta}_2 & \dot{\theta}_2 - \frac{\dot{\theta}_4}{r} & \theta_2 - \frac{\theta_4}{r} \\ 0 & \frac{\ddot{\theta}_4}{r^2} & 0 & \frac{\dot{\theta}_4}{r^2} - \frac{\dot{\theta}_2}{r} & \frac{\theta_4}{r^2} - \frac{\theta_2}{r} \end{bmatrix} \tag{29}$$

the regressand is

$$\mathbf{Y}_{S2} = \begin{bmatrix} 0 \\ T_2 - J_4\ddot{\theta}_4 - c_4\dot{\theta} \end{bmatrix} \tag{30}$$

and the parameter matrix to be estimated is

$$\mathbf{P}_{S2} = \begin{bmatrix} m_2 a_2^2 + J_2 \\ J_6 \\ c_2 \\ c_6 \\ k_6 \end{bmatrix} \tag{31}$$

Stage Three: Identification of the first link

Since parameters estimated from previous two sections would be used directly in this section, they would be treated as given numbers and placed into the regressor matrix and the output vector accordingly. Following are the equations of regression form for the first link system.

$$\mathbf{W}_{S3}(\theta,\dot{\theta},\ddot{\theta}) * \mathbf{P}_{S3} = \mathbf{Y}_{S3} \tag{32}$$

$$\mathbf{W}_{S3} = \begin{bmatrix} \ddot{\theta}_1 & \omega_{12} & \dot{\theta}_1 & \dot{\theta}_1 - \frac{\dot{\theta}_3}{r} & \theta_1 - \frac{\theta_1}{r} & 0 \\ 0 & \omega_{22} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\dot{\theta}_3}{r^2} - \frac{\dot{\theta}_1}{r} & \frac{\theta_3}{r^2} - \frac{\theta_1}{r} & \frac{\ddot{\theta}_3}{r^2} \end{bmatrix} \tag{33}$$

where

$$\begin{aligned} \omega_{12} &= cos(\theta_2)\ddot{\theta}_2 - (2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2)sin(\theta_2) + 2cos(\theta_2)\ddot{\theta}_1 \\ \omega_{22} &= cos(\theta_2)\ddot{\theta}_1 + sin(\theta_2)\dot{\theta}_1^2 \end{aligned} \tag{34}$$

For the regressand vector, it is defined as

$$\mathbf{Y}_{S3} = \begin{bmatrix} -(m_2 a_2^2 + J_2)(\ddot{\theta}_1 + \ddot{\theta}_2) - (J_4 + \frac{J_6}{r})\ddot{\theta}_4 \\ r_{L1} \\ T_1 - c_3\dot{\theta}_3 - J_3\ddot{\theta}_3 \end{bmatrix} \tag{35}$$

where

$$r_L 1 = -(m_2 a_2^2 + J_2)(\ddot{\theta}_1 + \ddot{\theta}_2) - (c_2 + c_6)\dot{\theta}_2 + \frac{c_6}{r}\dot{\theta}_4 - k_6\theta_2 + \frac{k_6}{r}\theta_4 \quad (36)$$

$$\mathbf{P}_{S3} = \begin{bmatrix} p_1 \\ l_1 m_2 a_2 \\ c_1 \\ c_5 \\ k_5 \\ J_5 \end{bmatrix} \quad (37)$$

where

$$p_1 = m_1 a_1^2 + m_2 l_1^2 + m_4 b_1^2 + m_6 l_1^2 + J_1 + J_4 + J_6 \quad (38)$$

SIMULATION DESIGN

Excitation signal profile

To have unbiased parameter estimation, the system should be persistently excited [7, p.250]; thus it is essential to select the input signals. In this paper, different sets of input signal $\mathbf{U}_{Sj} = [T_1, T_2]^T$, consisting of torques of the first and second motor respectively are used to excite the system. Chirp signal profile from 0.2 Hz to 4 Hz over 60 seconds with time step of 0.001 seconds was used to excite the system at first as recommend in [17]. However, due to nonlinearity in the robot dynamics, chirp signal may lead to under excitation at certain frequency region, depending on the robot configuration. Moreover, when noise is included, such as limited encoder resolution, low Signal to Noise Ratio (SNR) tends to appear in the high-frequency region, leading to worse angular speed and acceleration estimation, and so are the parameter estimates. Instead of chirp signal, integrated sinusoid signals with different frequencies are used. More specifically, integrated sinusoid signals $T_{sine,1}$ and $T_{sine,2}$ are defined by Equation (39) and shown in Figure 2

$$T_{sine,1} = T_{sine,2} = T_{max} * \sum_{k=1}^{N_{sine}} sin(2\pi f_k t) \quad (39)$$

where $N_{sine} = 9$ is the number of frequency components used, $T_{max} = 0.13(Nm)$ the scaling factor to limit torque output from each motor, $f_k(\text{Hz}) \in \{1,2,3,4,5,6,7,8,9\}$ and $f_i = f_j$ iff $i = j$.

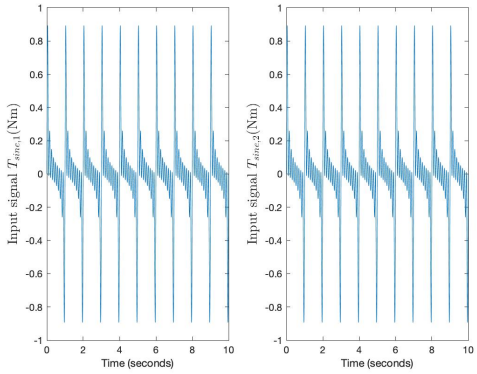


FIGURE 2. Excitation signal profiles

Data Preprocessing

In the experiment setup, only four encoders data could be obtained; thus it would be necessary to include the encoder resolution effect on angular position measurement in the simulation. Note that each encoder has same resolution of $0.00157[rad]$. To obtain both angular speed and angular acceleration, numerical approximation is made so that they are derived from angular position measurement by applying the backward difference method, as shown in Equation (40) and (41).

$$\dot{\theta}_{1,2,3,4}(k) = \frac{\delta\theta_{1,2,3,4}}{\delta t}|_{t=k} \approx \frac{\theta_{1,2,3,4}(k) - \theta_{1,2,3,4}(k-1)}{\Delta T} \quad (40)$$

$$\ddot{\theta}_{1,2,3,4}(k) = \frac{\delta\dot{\theta}_{1,2,3,4}}{\delta t}|_{t=k} \approx \frac{\dot{\theta}_{1,2,3,4}(k) - \dot{\theta}_{1,2,3,4}(k-1)}{\Delta T} \quad (41)$$

where $k = 1 * \Delta T, 2 * \Delta T, 3 * \Delta T...$ represents for discrete time stamp, and ΔT (second) for sampling period.

Estimating angular speed and acceleration by using numerical approximation, however, contain high-frequency noise that corrupts the signals; therefore, Butterworth low pass filters that are designed to filter out the high-frequency noise and Kalman Filter that provides best angular position, speed and acceleration estimation are used separately. To be consistent, all the Butterworth low pass filters are set to be order of three throughout the simulation. In terms of the Kalman Filter, the All-Integrator Model stated in [18] is used. To have a comprehensive investigation, within each stage, there are six simulation cases that each has different cut-off frequency, f_{cf}(Hz) for the low pass filter

TABLE 1. Simulation Cases

Stage	Case number	Excitation profile	f_{cf}	Q
Stage One	1	$\mathbf{U}_{S1} = [T_{sine,1}, T_{sine,2}]^T$	50	N/A
	2	$\mathbf{U}_{S1} = [T_{sine,1}, T_{sine,2}]^T$	100	N/A
	3	$\mathbf{U}_{S1} = [T_{sine,1}, T_{sine,2}]^T$	150	N/A
	4	$\mathbf{U}_{S1} = [T_{sine,1}, T_{sine,2}]^T$	200	N/A
	5	$\mathbf{U}_{S1} = [T_{sine,1}, T_{sine,2}]^T$	250	N/A
	6	$\mathbf{U}_{S1} = [T_{sine,1}, T_{sine,2}]^T$	300	N/A
Stage Two	1	$\mathbf{U}_{S2} = [\mathbf{0}, T_{sine,2}]^T$	50	20
	2	$\mathbf{U}_{S2} = [\mathbf{0}, T_{sine,2}]^T$	100	50
	3	$\mathbf{U}_{S2} = [\mathbf{0}, T_{sine,2}]^T$	150	100
	4	$\mathbf{U}_{S2} = [\mathbf{0}, T_{sine,2}]^T$	200	150
	5	$\mathbf{U}_{S2} = [\mathbf{0}, T_{sine,2}]^T$	250	160
	6	$\mathbf{U}_{S2} = [\mathbf{0}, T_{sine,2}]^T$	300	170
Stage Three	1	$\mathbf{U}_{S3} = [T_{sine,1}, T_{sine,2}]^T$	15	20
	2	$\mathbf{U}_{S3} = [T_{sine,1}, T_{sine,2}]^T$	20	50
	3	$\mathbf{U}_{S3} = [T_{sine,1}, T_{sine,2}]^T$	25	100
	4	$\mathbf{U}_{S3} = [T_{sine,1}, T_{sine,2}]^T$	30	150
	5	$\mathbf{U}_{S3} = [T_{sine,1}, T_{sine,2}]^T$	35	160
	6	$\mathbf{U}_{S3} = [T_{sine,1}, T_{sine,2}]^T$	40	170

and different variance of process noise, $Q(rad/s^2)^2$ applied in the Kalman Filter. Case detail is shown in Table 1.

Model Validation

To validate both the models and the RLS algorithm, it is necessary to check the convergence and consistency of estimates by performing simulation first before conducting experiments. For convergence, estimates should behave stable and show the ability to stay in a bounded region as sample size increases. To have consistent estimates is to have unbiased estimation, and estimates are able to converge to their true valued defined in the simulation as sample size becomes infinitely large. For the sake of model and algorithm validation, it is assumed that physical-noise-free signals including angular speed and acceleration are accessible, such that only computational noise is presented in the data and it is assumed to be normal distributed. This assumption is relaxed after validation. A tabulated estimation result showing consistency is shown at Table 2. Figure 3 to 5 show the convergency of the estimation process, validating the models used in the simulation and RLS algorithm for parameter estimation.

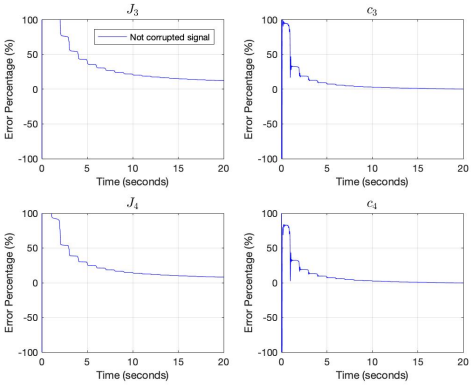


FIGURE 3. Convergence of stage one estimates

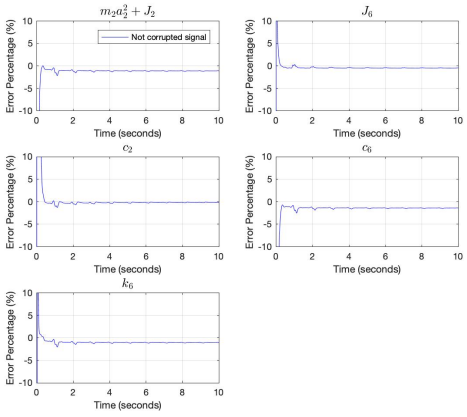


FIGURE 4. Convergence of stage two estimates

ESTIMATION OPTIMIZATION

RLS Estimation

Upon satisfying both conditions, the simulation is extended to include noise coming from angular speed and acceleration estimation due to encoder resolution, numerical approximation and sampling rate issues. Figure 6 to 8 show estimation results of case 6 by using the extended model with low pass filter and Kalman Filter. For simplicity, in stage one, only low pass filter is used in data preprocessing.

As seen, even if low pass filters and Kalman filter are used to get rid of high-frequency noise, parameter estimation are not

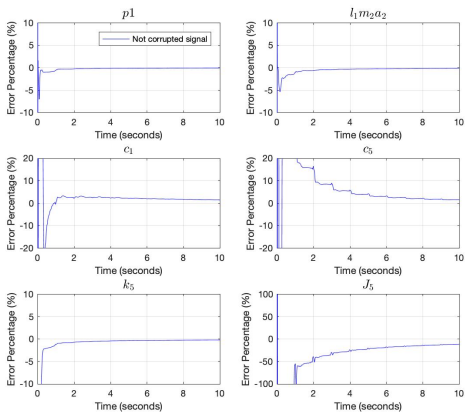


FIGURE 5. Convergence of stage three estimates

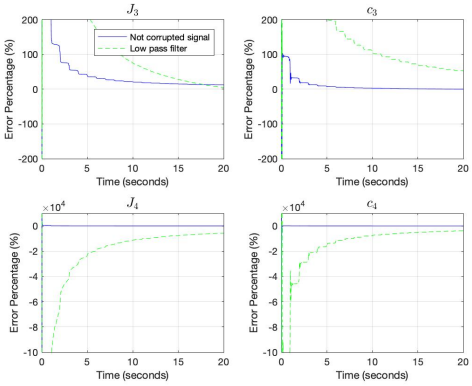


FIGURE 6. Error percentage of stage one estimates based on RLS algorithm

TABLE 2. Parameter Estimations for models and algorithm validation

Stage	Parameters	Definition	Estimation	Unit
Stage One	J_3	$5.25*10^{-4}$	$5.9034*10^{-4}$	$[Kg*m^2/rad]$
	c_3	0.020000	0.0201	$[N*m*s/rad]$
	J_4	$5.25*10^{-4}$	$5.6762*10^{-4}$	$[Kg*m^2/rad]$
	c_4	0.010000	0.0100	$[N*m*s/rad]$
Stage Two	$m_2a_2^2+J_2$	0.0196	0.0194	$[Kg*m^2/rad]$
	J_6	0.025	0.0249	$[Kg*m^2/rad]$
	c_2	0.0214	0.0214	$[N*m*s/rad]$
	c_6	0.08128	0.00801	$[N*m*s/rad]$
	k_6	2.848	2.8186	$[N*m/rad]$
Stage Three	p_1	0.14023	0.1402	$[Kg*m^2/rad]$
	$l_1m_2a_2$	0.023385	0.0234	$[Kg*m^2/rad]$
	c_1	0.04	0.0406	$[N*m*s/rad]$
	c_5	0.005	0.0051	$[N*m*s/rad]$
	k_5	2.848	2.8428	$[N*m/rad]$
	J_5	$5*10^{-5}$	$4.4453*10^{-5}$	$[Kg*m^2/rad]$

accurate enough. This mainly results from non-Gaussian distributed noise that appears in position measurement. To overcome this issue, prior knowledge of noise is essential, and a typical approach would be identifying the form filter of noise structure, which characterized the noise mathematically [19]. However, practically speaking, it would be very hard to identify the

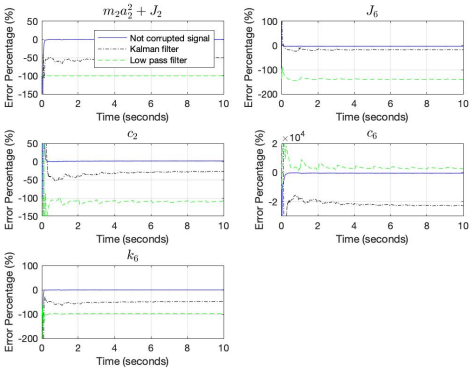


FIGURE 7. Error percentage of stage two estimates based on RLS algorithm

noise structure, and any types of assumption on its structure would potentially produce biased estimates ultimately.

IV Approach

Because of unknown noise structure and its probability distribution, RLS method yields biased estimates. [7, p.382] To overcome such an issue, an innovative solution, called Instrumental Variable (IV), is re-discovered in [8, p.486]. As seen in Equation (42), system parameters at each stage are now estimated with the help of matrix $\mathbf{\Gamma}_{Sj}$ whose columns are called

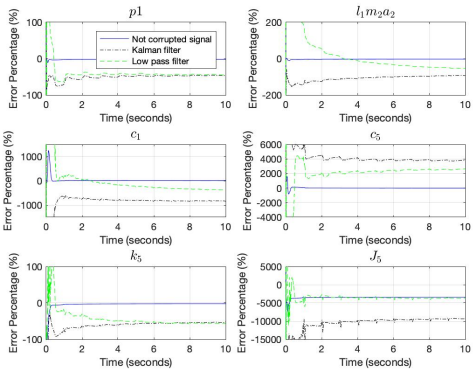


FIGURE 8. Error percentage of stage three estimates based on RLS algorithm

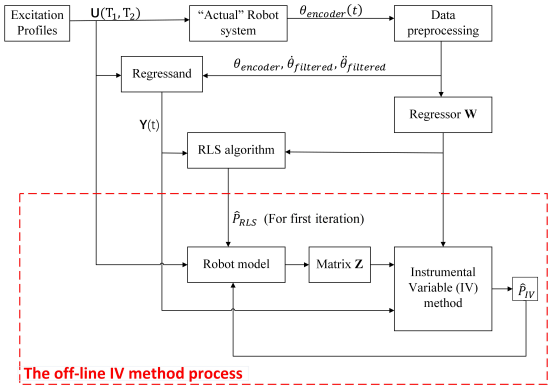


FIGURE 9. Overall estimation process with aid from Instrumental Variable method

instrumental variables. To have optimized estimation, it is essential to construct the matrix $\mathbf{\Gamma}_{Sj}$ such that it is highly correlated with the regression matrix $\mathbf{\Pi}_{Sj}$ and uncorrelated with the estimation residual from RLS algorithm. [8, p.486] To fulfill these requirements, in this work, the $\mathbf{\Gamma}_{Sj}$ matrix is constructed by using uncorrupt signals simulated based on estimates $\hat{\mathbf{P}}_{Sj,RLS}$ at the first iteration and $\hat{\mathbf{P}}_{Sj,IV}$ for the rest of iterations. To initiate the IV method at first iteration, it is important to have stable estimates so that matrix $\mathbf{\Pi}_{Sj}$ does not contain unstable system responses. Experience and literature showed that convergency occurs very fast and usually 2 to 5 iterations are enough. Figure.9 shows overall estimation process after integrating RLS algorithm with the IV method.

$$\begin{aligned}
\hat{\mathbf{P}}_{Sj,IV} &= (\mathbf{\Gamma}_{Sj}^T \mathbf{\Pi}_{Sj})^{-1} \mathbf{\Gamma}_{Sj}^T \mathbf{\Psi}_{Sj} \\
\mathbf{\Gamma}_{Sj} &= \left[\mathbf{Z}_{Sj}^T(k=\Delta T), ..., \mathbf{Z}_{Sj}^T(k=N*\Delta T)\right]^T \\
\mathbf{\Pi}_{Sj} &= \left[\mathbf{W}_{Sj}^T(k=\Delta T), ..., \mathbf{W}_{Sj}^T(k=N*\Delta T)\right]^T \\
\mathbf{\Psi}_{Sj} &= \left[\mathbf{Y}_{Sj}^T(k=\Delta T), ..., \mathbf{Y}_{Sj}^T(k=N*\Delta T)\right]^T
\end{aligned} \tag{42}$$

Equation (43) to (45) define matrix $\mathbf{Z}_{Sj}$ at each time step that is used from stage one to stage three respectively.

$$\mathbf{Z}_{S1} = \begin{bmatrix} \ddot{\theta}_{sim,i} & \dot{\theta}_{sim,i} \end{bmatrix} \tag{43}$$

where i=3 for the first motor and i=4 for the second motor.

$$\mathbf{Z}_{S2} = \begin{bmatrix} \ddot{\theta}_{sim,2} & 0 & \dot{\theta}_{sim,2} & \dot{\theta}_{sim,2} - \frac{\dot{\theta}_{sim,4}}{r} & \theta_{sim,2} - \frac{\theta_{sim,4}}{r} \\ 0 & \frac{\ddot{\theta}_{sim,4}}{r^2} & 0 & \frac{\dot{\theta}_{sim,4}}{r^2} - \frac{\dot{\theta}_{sim,2}}{r} & \frac{\theta_{sim,4}}{r^2} - \frac{\theta_{sim,2}}{r} \end{bmatrix} \tag{44}$$

$$\mathbf{Z}_{S3} = \begin{bmatrix} \ddot{\theta}_1 & \omega_{sim,12} & \dot{\theta}_{sim,1} & \dot{\theta}_{sim,1} - \frac{\dot{\theta}_{sim,3}}{r} & \theta_{sim,1} - \frac{\theta_{sim,1}}{r} & 0 \\ 0 & \omega_{sim,22} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\dot{\theta}_{sim,3}}{r^2} - \frac{\dot{\theta}_{sim,1}}{r} & \frac{\theta_{sim,3}}{r^2} - \frac{\theta_{sim,1}}{r} & \frac{\ddot{\theta}_{sim,3}}{r^2} \end{bmatrix} \tag{45}$$

where

$$\begin{aligned}
\omega_{sim,12} = cos(\theta_{sim,2})\ddot{\theta}_{sim,2} - (2\dot{\theta}_{sim,1}\dot{\theta}_{sim,2} \\
+\dot{\theta}_{sim,2}^2)sin(\theta_{sim,2}) + 2cos(\theta_{sim,2})\ddot{\theta}_{sim,1}
\end{aligned} \tag{46}$$

$$\omega_{sim,22} = cos(\theta_{sim,2})\ddot{\theta}_{sim,1} + sin(\theta_{sim,2})\dot{\theta}_{sim,1}^2 \tag{47}$$

RESULT

As mentioned previously, Butterworth low pass filters and Kalman filter are used separately to handle collected encoder data so as to get low-noise angular speed and angular acceleration. In addition, Kalman Filter provides best estimation on angular position. Since in each stage, the robot dynamic model is inherently different than each other; therefore, by following the guideline that would be provided later in this paper, both cut-off

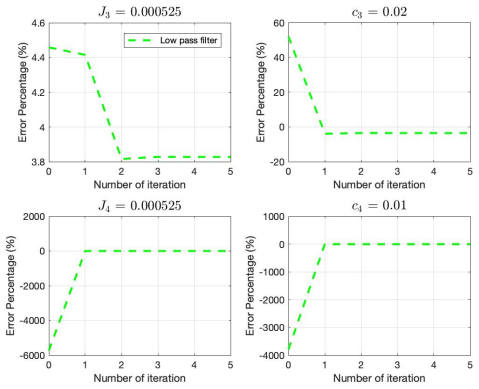


FIGURE 10. Error percentage of case 6 at stage one estimates after applying the IV method

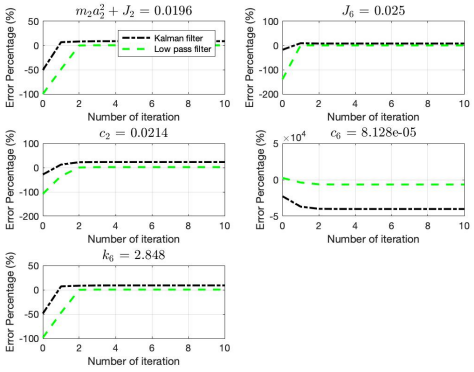


FIGURE 11. Error percentage of case 6 at stage two estimates after applying the IV method

frequency and the variance of process noise are tuned to achieve accurate parameter estimation. Figures 10 to 12 show the estimated parameters after employing the IV method while Table 3 makes a comparison between RLS algorithm and IV method on estimation accuracy for case 6 that is processed by low pass filter. It is interesting to point out that even though Kalman Filter is able to provide better estimation on angular position, velocity and acceleration, it does not yield as accurate results as "low pass filter case" dose after integrating the RLS algorithm with the IV method. Thus, analysis hereafter wold focus on low pass cases only.

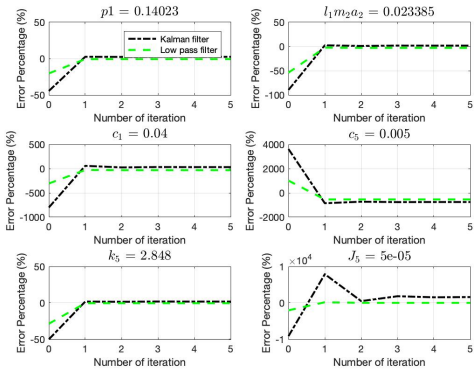


FIGURE 12. Error percentage of case 6 at stage three estimates after applying the IV method

TABLE 3. Error percentage of parameter estimation of case 6

Stage	Parameters	RLS	IV method
Stage One	J_3	4.459%	3.828%
	c_3	52.27%	-3.498%
	J_4	−5754%	1.958%
	c_4	-3857%	-4.745%
Stage Two	$m_2a_2^2+J_2$	-100%	0.4911%
	J_6	-139.6%	0.4951%
	c_2	-108.1%	2.186%
	c_6	2487%	-6535%
	k_6	-98.81%	0.5742%
Stage Three	p_1	-20.31%	-0.6837%
	$l_1m_2a_2$	-54%	-2.894%
	c_1	-306.1%	-31.56%
	c_5	1013%	-534.7%
	k_5	-28.64%	-0.4043%
	J_5	−2034%	25.94%

In addition to simply comparing the estimates error percentage, it would be important to see how do all of the estimates perform in terms of simulating the robot dynamic. Since this paper values angular acceleration more than angular position or velocity, for simplicity, analysis hereafter would focus more on the acceleration. From the error plot of angular acceleration, shown in Figure 13, the IV method outweighs the classic RLS algorithm.

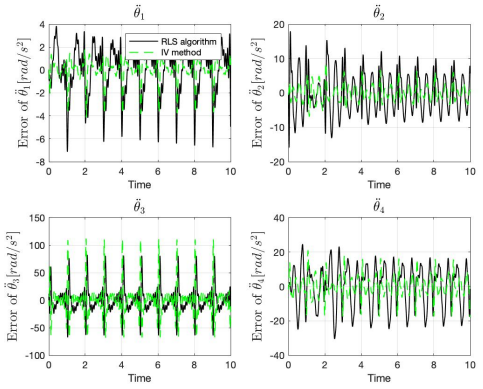


FIGURE 13. Error plot of robot angular acceleration simulated by parameters estimated by low pass filter for case 6

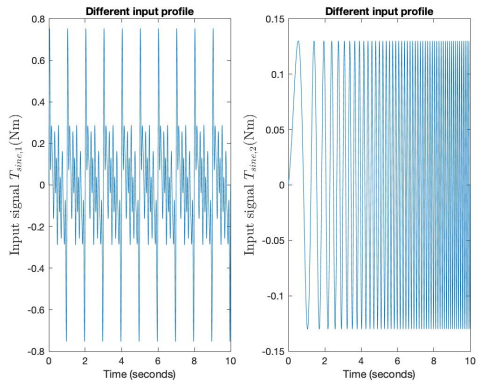


FIGURE 14. A different set of input profiles for checking estimates performance

To further compare both methods, a different set of input profiles shown in Figure 14, is generated to excite the full robot system, yielding results shown in Figure 15. As seen, acceleration error drops after employing the IV method, qualitatively maintaining its performance in terms of robot trajectory simulation. Note that effort have been made to check simulated angular position and velocity and similarly qualitative performance is observed.

In addition to comparing the method-wide performance, within each method, Table 1 gives details of 6 different cases that are simulated and studied. A dimensionless trajectory acceleration error is defined by Equation (48) for further anlysis.

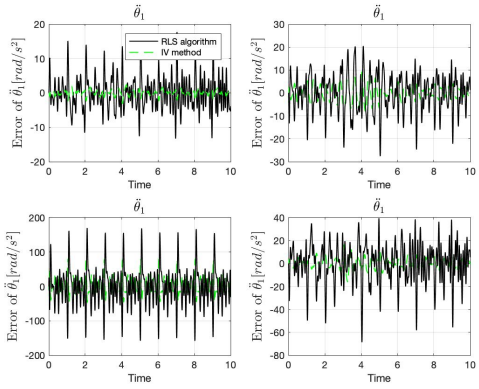


FIGURE 15. Error plot of robot angular acceleration simulated by parameters estimated by low pass filter for case 6 with different input profiles

$$e^* = \sum_{i=0}^{N} \frac{|\ddot{\theta}_{defined}(i) - \ddot{\theta}_{est}(i)|}{amp(\ddot{\theta}_{defined})} * \frac{1}{N} \tag{48}$$

$$amp(\cdot) = max(\cdot) - min(\cdot)$$

where $\ddot{\theta}_{defined}$ is the time history of robot angular acceleration simulated by the defined values, and $\ddot{\theta}_{est}$ by the estimates; N is the sample size.

As seen from Figure 16, as the cut-off frequency of low pass filter increases, the dimensionless trajectory error shrinks. Similar trend of dimensionless error, shown in Figure 17 is obtained in the event of using a different set of input profiles. Note that, in Figure 16 and 17, the blue-solid curve of "Perfect signal" represents estimation cases that use system response without measurement noise nor numerical approximation on joint velocity and acceleration; therefore, this curve intuitively denotes the "best" results that one could possibly obtain out of the simulation when all noise-free states of the system are accessible.

Therefore, results above, to some extend, shed light on the guideline of applying the IV method for obtaining high accuracy estimates. To initiate the IV method for its first iteration, it is important to construct the matrix $\mathbf{\Gamma}_{sj}$ such that all of its elements are stable signals. Secondly, if low pass filters are chosen for data preprocessing, it is recommended to have larger cut-off frequency whenever possible.

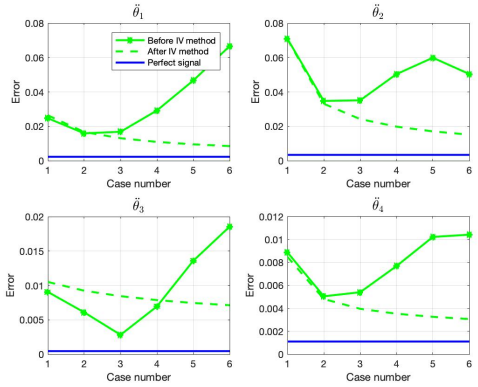


FIGURE 16. Dimensionless trajectory error before and after applying the IV method by using low pass filters

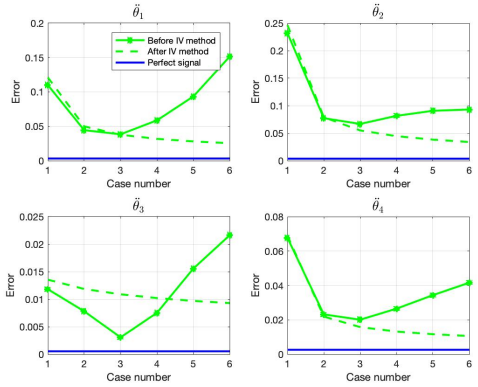


FIGURE 17. Dimensionless trajectory error before and after applying the IV method by using low pass filters with different input profiles

REFERENCES

[1] Wu, J., Wang, J., and You, Z., 2010. "An overview of dynamic parameter identification of robots". *Robotics and computer-integrated manufacturing,* ***26***(5), pp. 414–419.

[2] Vivas, A., Poignet, P., Marquet, F., Pierrot, F., and Gautier, M., 2003. "Experimental dynamic identification of a fully parallel robot". In 2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422), Vol. 3, IEEE, pp. 3278–3283.

[3] Gautier, M., 1997. "Dynamic identification of robots with power model". In Proceedings of International Conference on Robotics and Automation, Vol. 3, IEEE, pp. 1922–1927.

[4] Gautier, M., Janot, A., and Vandanjon, P.-O., 2012. "A new closed-loop output error method for parameter identification of robot dynamics". *IEEE Transactions on Control Systems Technology,* ***21***(2), pp. 428–444.

[5] Jin, J., and Gans, N., 2015. "Parameter identification for industrial robots with a fast and robust trajectory design approach". *Robotics and Computer-Integrated Manufacturing,* ***31***, pp. 21–29.

[6] Ljung, L., 1999. *System Identification: Theory for the User*. Prentice Hall information and system sciences series. Prentice Hall PTR.

[7] Isermann, R., 2011. *Identification of Dynamic Systems An Introduction with Applications*. Advanced textbooks in control and signal processing. Springer Berlin Heidelberg.

[8] Nelles, O., 2001. *Nonlinear System Identification From Classical Approaches to Neural Networks and Fuzzy Models*. Springer Berlin Heidelberg : Imprint: Springer.

[9] YOUNG, P., 1976. "Some observations on instrumental variable methods of time-series analysis". *International Journal of Control,* ***23***(5), pp. 593–612.

[10] Young, P., 1985. "The instrumental variable method: a practical approach to identification and system parameter estimation". *IFAC Proceedings Volumes,* ***18***(5), pp. 1–15.

[11] Janot, A., Vandanjon, P.-O., and Gautier, M., 2014. "A generic instrumental variable approach for industrial robot identification". *IEEE Transactions on Control Systems Technology,* ***22***(1), pp. 132–145.

[12] Brunot, M., Janot, A., Young, P., and Carrillo, F., 2018. "An instrumental variable method for robot identification based on time variable parameter estimation".

[13] Meckl, P. H., 1988. "Control of Vibration in Mechanical Systems Using Shaped Reference Inputs". PhD Thesis, Massachusetts Institute of Technology, Cambridge, MA.

[14] Lee, R. S., 2011. "Optimal parameter estimation for long-term prediction in the presence of model mismatch applied to a two-link flexible joint robot". PhD thesis, Purdue University.

[15] Khosla, P. K., and Kanade, T., 1985. "Parameter identification of robot dynamics". In 1985 24th IEEE Conference on Decision and Control, IEEE, pp. 1754–1760.

[16] Nho, H. C., 2004. "Precise motion control of flexible -joint robot manipulators with an intelligent payload estimator". PhD thesis, Purdue University.

[17] Scheel, A., 2012. "Identification of a robot with joint flexibility".

[18] Belanger, P., Dobrovolny, P., Helmy, A., and Zhang, X., 1998. "Estimation of angular velocity and acceleration from shaft-encoder measurements". *The International Journal of Robotics Research,* ***17***(11), pp. 1225–1233.

[19] Tomizuka, M., and Takahashi, Y., 1977. "Deterministic disturbance rejection in linear identification". *Journal of Dynamic Systems, Measurement, and Control,* ***99***(4), pp. 307–310.