Title

Learning and Optimization Methods for Robust Control of Hard Disk Drives and Geometric Control of Fully Actuated Mechanical Systems

Author

Potu Surya Prakash, Nikhil

Publication Date

2024

Peer reviewed|Thesis/dissertation

Learning and Optimization Methods for Robust Control of Hard Disk Drives and Geometric Control of Fully Actuated Mechanical Systems

By

Nikhil Potu Surya Prakash

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering - Mechanical Engineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Roberto Horowitz, Chair
Associate Professor Koushil Sreenath
Associate Professor Gireeja Ranade

Summer 2024

Learning and Optimization Methods for Robust Control of Hard Disk Drives and Geometric Control of Fully Actuated Mechanical Systems

Abstract

Learning and Optimization Methods for Robust Control of Hard Disk Drives and Geometric Control of Fully Actuated Mechanical Systems

by

Nikhil Potu Surya Prakash

Doctor of Philosophy in Engineering - Mechanical Engineering

University of California, Berkeley

Professor Roberto Horowitz, Chair

The use of Machine Learning (ML) and optimization for control applications in enhancing performance, replicating expert behaviors, and addressing other complex challenges has emerged as a focal point in contemporary research. This has become possible due to ML techniques' ability to find patterns, approximate complex functions and make decisions from data. Although promising, learning-based controllers often encounter difficulties in maintaining stability guarantees due to their reliance solely on data. This data is often noisy or incomplete, which may lead to unpredicted and in many cases unstable system responses under previously unseen scenarios. A typical ML problem is formulated by designing a loss function and the parameters that minimize the loss are used to obtain the optimal model. For control applications, a constraint on the stability needs to be imposed both during training and inference. But, most ML models do not impose such conditions as hard constraints and only encourage through soft constraints in the loss function. These constraints might not be satisfied during inference and the ML control laws can destabilize the system.

This dissertation addresses the challenge of designing stabilizing controllers utilizing ML and optimization techniques for two distinct classes of systems. In the first part, we explore designing controllers through Neural Network potential functions for fully actuated mechanical systems that evolve on manifolds with well-defined dynamics in the state space. The control laws incorporate the concept of invariance for data efficient training and easy transferability between robots with similar kinematic structure. The design methodology will be emphasized on an application to variable impedance control of mechanical manipulators. In the second part, we discuss the robust control of Multi-Input Single Output (MISO) systems, with a particular focus on Multi-Actuator Hard Disk Drives (HDDs). A design methodology in frequency domain based on available frequency response data to ensure stability and robustness against disturbances and model uncertainties is presented. An unsupervised ML technique to cluster the plant transfer functions and frequency responses into subgroups

is presented. Clustering helps us design common controllers within each cluster to both maintain robustness and improve performance. Lastly, identification methods for obtaining dynamical models of disturbance processes with colored noises and necessary filters that satisfy Strictly Positive Real (SPR) conditions for stability of adaptive control algorithms are presented. Throughout the dissertation, we will delve into the theoretical underpinnings of these methodologies, complemented by simulation results that highlight significant improvements in system responsiveness and efficiency achieved through these innovative control strategies.

This dissertation shows that integrating learning-based mechanisms into mechanical system controllers is both feasible and effective. It also offers guidance for future research to address the challenges of these technologies. By combining theoretical analysis and simulation studies, this work demonstrates how data-driven approaches can improve control systems.

To my mother Bujji, for all her sacrifices

Contents

List of Figures

Acknowledgments

Being a graduate student at Berkeley has been nothing short of a roller coaster ride both academically and emotionally. I was bent, stretched, compressed, sheared and twisted yet was unyielded during this ride. This ride only made me tougher instead of fatiguing or fracturing me and has prepared me to navigate through any future challenges with confidence. First and foremost, I would like to express my deepest gratitude to my advisor, Prof. Roberto Horowitz, for giving me the opportunity to even get on this roller coaster and experience the ride by accepting me into his group and be at Berkeley. His invaluable guidance, support, and mentorship throughout the course of my PhD journey has helped me to become the person I am today. His expertise in the field of control systems and relentless dedication to novelty, utility and excellence have not only shaped this research but have also profoundly influenced my professional and personal development. The opportunity to work under such a distinguished scholar was nothing short of a privilege, and for that, I am eternally grateful. I should particularly mention about his willingness to become a student and let his students be the teachers during our meetings which led all of us to become coherent and complete in terms of the concepts and thinking. His idea to have open lab meetings with all the members together has expanded our research areas and made us help and collaborate with each other fostering our bonds and friendships. I would next like to thank Prof. Jonguen Choi for being extremely supportive and encouraging me with my research. He has been a powerhouse of ideas and has guided me in identifying interesting and meaningful problems to solve. Some of the research wouldn't even have been possible if not for his visit to Berkeley. I am eternally grateful to him for this. I would like to thank ASRC (Fred Hong and Raye Sosseh) and HKCRC for funding my research without which none of these acknowledgements would have been possible.

I extend my sincere thanks to my other dissertation committee members, Prof. Koushil Sreenath and Prof Gireeja Ranade for their insightful feedback and constructive critiques that significantly enhanced the quality of my work. Their expertise and thoughtful guidance have been crucial in refining my research focus and methodologies. Prof. Sreenath's ever welcoming smile and energy have made him more approachable both during and outside the multiple courses I took with him which culminated in multiple discussions that helped me understand many rudimentary yet non-trivial concepts in geometry and control. If I had to pick one thing I would miss the most about Berkeley, it has to be attending his lectures. Though I did not have the opportunity to take in-person courses with Prof. Ranade, her lecture recordings of EECS 16A, 16B and 227AT helped me come to ease with many topics in Linear Algebra and Optimization. Her teaching and course structures have helped me get an idea on how to design courses and have inspired me to become a teacher. Her insightful questions and comments during and after my qualifying exam have helped me become better at presenting my research. I would also thank my other qualifying exam members Prof. Mark Mueller and Prof. Kameshwar Poolla for their support and insights. I would forever be indebted to Prof. Mueller for making me feel so comfortable with probability and statistics.

My journey through the PhD program was also made possible by the camaraderie and support of my colleagues in the Computer Mechanics Lab. I am thankful to Joohwan Seo whose collaboration and friendship made the countless hours of research and discussion both productive and enjoyable. I would particularly like to thank him for introducing me to the concepts of invariance and equivariance which made path for new and exciting research problems. I would also like to thank Roshan Mathew Tom for being a wonderful colleague, flat mate and more than these, a lovely friend. I am also very thankful to Ruolin Li for bringing so much cheer into the lab with all her interesting stories and humor. I guess I should thank her more for her indirect help in easing the at times stressed environment than for her help in my research even which was not by any means insignificant. I would also thank my other lab mates and visitors during the course of my PhD Prateek Shah, Zhi Chen, Alexander Rose and Yiheng Zhu for all the insightful discussions, collaborations and help.

I also want to thank all the wonderful teachers for making me learn the way I should. I would particularly like to thank professors (in the chronological order) Arun Menon, Saravanan, Manu Santhanam, Ashwin Mahalingam, Dennis Bernstein, Kon Well Wang, Carl Simon, David Remy, Gabor Orosz, Dimitra Panagou, Anthony Bloch, Roberto Horowitz, Mark Mueller, Koushil Sreenath, Alexandre Bayen, Francesco Borrelli, Paniyotis Papadopoulos, Gireeja Ranade, Anca Dragan, Murat Arcak and Lawrence Evans.

Last but certainly not least, I must acknowledge the unwavering support and love of my family and friends. To my parents, Anuradha and Surya Prakash, kins Saroja, Suresh, Srinivasulu, Sailaja, Mohita, Samhita and Madhavi, my friends Ashwath, Sayan, Anurag, Jaeseung, Bala, Prasanth, Sai Kumar, Swetha, Suryas, Vivek, Rohith, Saketh, Sowmith, Abhishek, Tejaswi, Manuha, Sravya, M.Vaishnavi, Vijaya, N.Vaishnavi, Sruthi and Devanshi who provided endless encouragement and were always there with a comforting word during the inevitable ups and downs of this rigorous academic pursuit. Your belief in my capabilities has been a constant source of motivation and resilience.

To all of you who have been part of this journey, directly or indirectly, your roles have been instrumental in achieving this milestone. I am deeply grateful for each and every contribution that helped make this dissertation possible.

Chapter 1

Introduction

1.1 Basic Introduction

The advent of Machine Learning (ML) is revolutionizing numerous fields, including control systems, by offering powerful tools to learn complex patterns and make decisions based on data. Machine learning algorithms excel at handling high-dimensional data, identifying intricate relationships, and optimizing performance in tasks where traditional methods might struggle due to their complexity or the sheer volume of data. In the context of control systems, ML techniques such as reinforcement learning, neural networks, and Bayesian learning have shown great promise in designing controllers that can adapt to varying conditions and optimize performance.

However, despite the significant advantages of machine learning, it also has notable limitations when applied directly to designing control policies. One of the critical shortcomings is that ML models, trained solely on data, might not guarantee stability when faced with unseen environments or unexpected conditions. This is particularly problematic in control applications, where stability and reliability are extremely important. Pure ML-based controllers can perform exceptionally well within the scope of their training data but might fail to generalize or ensure safe operation beyond those scenarios.

To address these limitations, it is essential to augment traditional control methods with machine learning. Traditional control techniques, grounded in well-established principles of control theory, offer robust guarantees of stability and performance. By integrating machine learning on top of these methods, we can leverage the adaptive and predictive capabilities of ML while maintaining the foundational stability provided by classical control approaches. This hybrid approach allows for the creation of controllers that are not only robust and stable but also capable of adapting to complex and dynamic environments, thereby enhancing overall system performance.

In this dissertation, we focus on applying this hybrid approach to systems evolving on manifolds, with particular emphasis on the impedance control of robotic manipulators and the robust controller design for Hard Disk Drives (HDDs). By combining traditional control

methodologies with ML techniques, we aim to ensure stability while optimizing performance in these specific applications.

In the upcoming sections, a detailed overview of robotic manipulators and HDDs, highlighting their specific control requirements and the challenges associated with their operation will be presented. The discussion will set the stage for understanding how this hybrid control approach can be effectively applied to these systems, ensuring both stability and enhanced performance.

1.2 Robotic Manipulators

Robotic manipulators are mechanical systems designed to perform tasks that require precision, dexterity, and control. These machines are fundamental in the field of robotics, integrating mechanical, electrical, and computational systems to interact with objects and their environments effectively. Robotic manipulators vary in complexity, from simple two-degree-of-freedom (DOF) arms to sophisticated multi-DOF systems that replicate the versatility of a human arm.

A typical robotic manipulator is composed of several key components. The structure includes rigid links connected by joints, which can be either rotational (revolute) or linear (prismatic). These joints provide the necessary degrees of freedom for the manipulator to position and orient its end-effector. Actuators, such as electric motors, hydraulic cylinders, or pneumatic systems, supply the force and motion required to move the joints. The choice of actuators depends on specific application requirements, including speed, precision, and load capacity. The end-effector, the part of the manipulator that interacts directly with the environment, can take various forms, such as a gripper for picking up objects, a welding tool, or a camera for inspection. Sensors embedded within the manipulator provide critical feedback to the control system about the manipulator's position, orientation, and interaction with the environment. These sensors can include encoders, force/torque sensors, and vision systems. The control system, which processes sensor data and sends commands to the actuators, ensures that the manipulator performs tasks accurately and efficiently. This system often incorporates advanced control algorithms from control theory, machine learning, and artificial intelligence to achieve optimal performance.

Robotic manipulators have diverse applications across various industries due to their versatility and precision. In manufacturing, they perform essential tasks such as assembly, welding, painting, and material handling. These robots significantly boost productivity and consistency while ensuring safety by executing repetitive and hazardous tasks with high accuracy. In the healthcare sector, robotic manipulators are transformative, aiding in surgeries, rehabilitation, and patient care. Surgical robots, for instance, enable minimally invasive procedures, enhancing precision and control, thereby reducing recovery times and improving patient outcomes. The logistics and warehousing industry also greatly benefits from robotic manipulators. Automated storage and retrieval systems (AS/RS) utilize these robots to efficiently manage and transport goods, increasing throughput and lowering op-

erational costs. In agriculture, robotic manipulators undertake tasks such as harvesting, pruning, and planting, enhancing efficiency and minimizing the physical labor required, particularly in challenging environments. Robotic manipulators are indispensable in exploration and research, particularly in space and underwater missions. They handle delicate instruments, collect samples, and perform repairs in hazardous or otherwise inaccessible locations. Furthermore, the service and entertainment industries are increasingly utilizing robotic manipulators. From cleaning and serving food to enhancing interactive experiences through animatronics and robotic performers, these robots are expanding their footprint, showcasing their broad applicability and transformative impact across diverse fields.

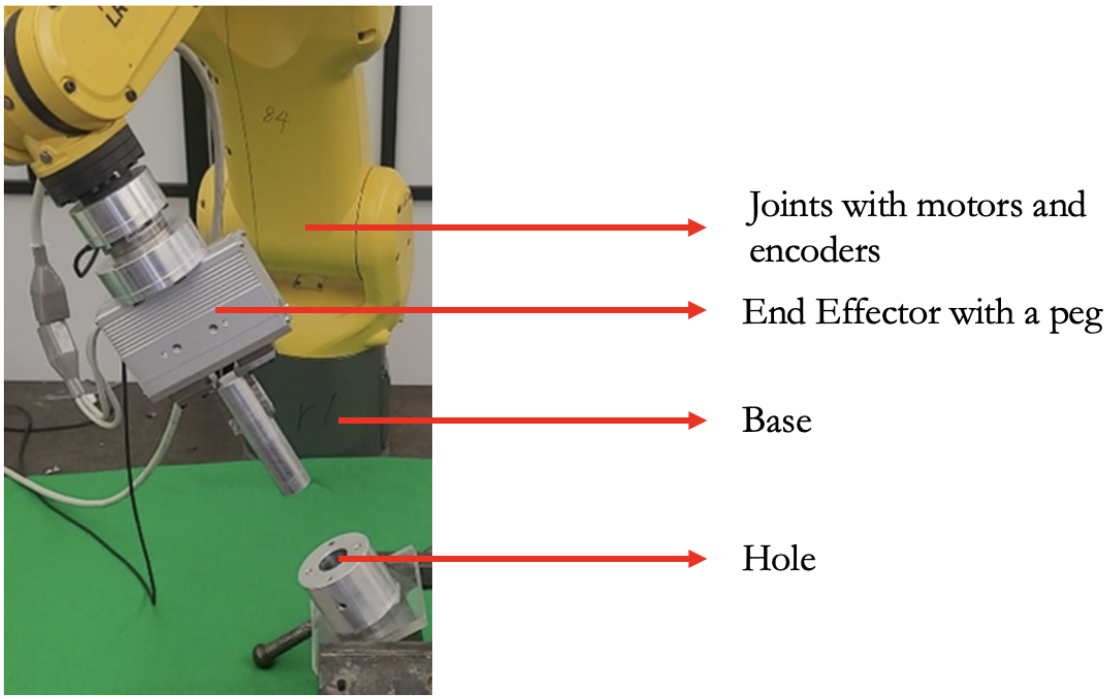

Figure 1.1: An image of a Fanuc 200iD/7L robotic manipulator with a peg as an end effector for demonstrating a "peg in a hole task".

The operation of robotic manipulators involves several well-coordinated steps. The control system begins by generating a trajectory or path for the manipulator to follow, calculating the required positions, velocities, and accelerations at each joint to achieve the desired end-effector movement. Sensors continuously monitor the manipulator's state and environment, providing feedback for real-time adjustments and ensuring the manipulator remains on its intended path. Actuators, responding to control signals, move the joints accordingly, with their precision and responsiveness being critical for the manipulator's performance. Algorithms ranging from something as simple as proportional-integral-derivative (PID) con-

trollers to advanced algorithms like model predictive control (MPC), or neural network-based controllers, are employed to minimize errors and adapt to dynamic conditions. Finally, the end-effector interacts with the environment, performing the desired task. The manipulator must handle variations and uncertainties in the task environment, such as changes in object positions or unexpected obstacles, to accomplish its objectives effectively.

An emerging and critical area of application for robotic manipulators is contact-rich manipulation, where the robot interacts closely with objects and surfaces in a manner that requires nuanced control and feedback. This type of manipulation is particularly relevant in tasks that involve assembling parts, handling delicate materials, or performing operations that require a high degree of tactile feedback. Contact-rich manipulation leverages force/torque sensors at the end effector and sometimes cleverly designed impedance control without force/torque sensors to ensure that the manipulator can adapt to varying levels of resistance and friction, making real-time adjustments to maintain precision and avoid damage.

In the construction industry, robotic manipulators are increasingly being utilized for their ability to perform contact-rich manipulation tasks. These applications include brick laying, welding steel structures, painting large surfaces, and installing fixtures. The ability to automate such labor-intensive and precise tasks not only enhances productivity but also improves safety by reducing the need for human workers to operate in potentially hazardous environments. The integration of robotic manipulators in construction allows for continuous operation, higher quality workmanship, and the ability to work in environments that might be dangerous or challenging for human laborers. These advancements are driving significant transformations in construction methodologies, pushing the boundaries of what can be achieved through automation.

1.3 Multi Actuator Hard Disk Drives

Based on current trends and projections, the total amount of data created, captured, copied, and consumed globally is expected to reach approximately 610 zettabytes by 2030. This estimate is derived from the exponential growth patterns observed in recent years and forecasts for the near future. Projections indicate that by 2025, the global datasphere will increase to around 181 zettabytes. The growth rate of data generation has been consistently high, with year-over-year increases ranging from 20% to 30% in recent years. Assuming this trend continues, we can expect the amount of data to more than triple between 2025 and 2030.

This massive growth in data generation necessitates robust and scalable storage solutions, driving the need for advanced data storage devices and high-capacity data centers. Data centers must ensure high availability, fast access speeds, reliability, and efficient power consumption to meet the growing needs of various industries. The types of data storage devices can be broadly classified into the following types each with unique characteristics and use cases:

- *Hard Disk Drives (HDDs)*: These are magnetic storage devices widely used in data centers, personal computers, and enterprise storage solutions due to their large storage capacities and cost-effectiveness. They are critical in enterprise storage arrays, network-attached storage (NAS), and storage area networks (SANs) for efficient data management and storage. Due to their durability and cost-efficiency, HDDs are widely used for backup and archival purposes in both consumer and enterprise environments.

- *Solid State Drives (SSDs)*: Utilizing flash memory, SSDs offer faster data access speeds, lower latency compared to HDDs. They are increasingly used in personal computers, high-performance servers, and applications requiring rapid data access.

- *Hybrid Drives (SSHDs)*: Combining HDD and SSD technologies, SSHDs aim to balance performance and capacity, offering faster access speeds than HDDs and higher storage capacities than SSDs.

- *Optical Storage Devices*: Including CDs, DVDs, and Blu-ray discs, these are used for media distribution and archival storage.

- *Tape Storage*: Predominantly used for archival and backup purposes due to their high storage density and longevity.

Despite the advancements in SSD technology, HDDs continue to dominate the storage industry, especially in data centers and enterprise storage systems. The primary advantages of HDDs over SSDs include:

- *Cost per Gigabyte (GB)*: HDDs offer significantly lower cost per gigabyte, making them ideal for large-scale storage requirements. Data storage on an SSD can cost $0.08–0.10 per GB, while an HDD only costs $0.03–0.06 per GB

- *Storage Capacity*: HDDs are available in much larger capacities compared to SSDs, with enterprise HDDs reaching up to 20TB per drive.

- *Reliability for Archival Storage*: HDDs are well-suited for archival and backup storage due to their established reliability and longer lifespan for write-heavy applications.

A typical HDD comprises several key components, including platters, which are thin disks coated on either side with magnetic material for data storage. The data is stored on concentric circluar tracks by modifying the polarity of the magnetic material. Modern HDDs typically have between one and nine platters. The spindle motor spins these platters at a constant angular speed, usually ranging from 5,400 to 15,000 RPM. Positioned at the end of a suspension arm, the read/write head, also known as a transducer, flies just above the platter surface to read and write data. The Voice Coil Motor (VCM) moves the suspension arm to accurately position the read/write head over the desired track. For fine positioning adjustments, a piezoelectric (PZT) micro-actuator works in tandem with the VCM in a dual-stage actuation system. In a typical multi-actuator HDD, there can be more than one

PZT actuator to enhance precision and performance. The VCM and PZT actuators work in parallel to move the head. There could be additional PZT actuators that can be added to improve the performance. The collection of all of these actuators that work together to move a single head is called the Multi-Stage Actuator (MSA). Having one VCM and one PZT makes it a Dual-Stage Actuator (DSA) and having one VCM and two PZTs makes it a Triple-Stage Actuator (TSA) and so on. One MSA could also be stacked on top of the other to control two different platters independently to parallelize the read/write operations. Such an HDD is called a Multi Actuator HDD. On top of the regular challenges, Multi Actuator HDDs face another disturbance transferred through the pivot from the other MSAs due to their motion as both the MSAs operate on the same pivot.

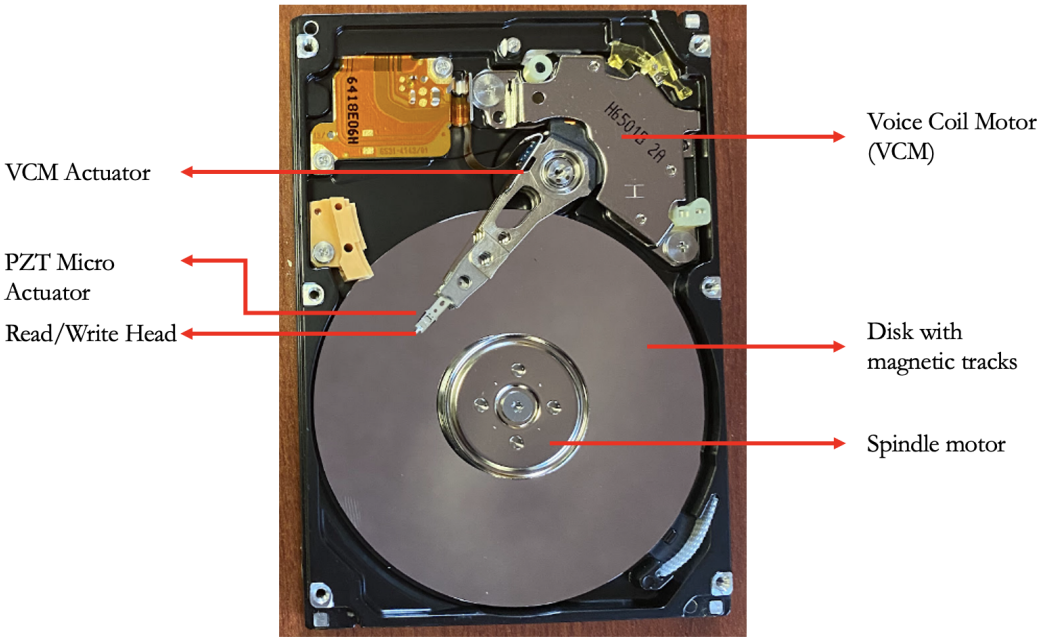

Figure 1.2: An image of a Multi Actuator Dual Stage HDD with the case removed to show the internal components.

Methods of Data Writing on HDDs

- *Perpendicular Magnetic Recording (PMR):* PMR [83] is a widely used technology in modern HDDs where data bits are aligned perpendicularly to the disk surface, rather than parallel as in traditional longitudinal recording. This method allows for a higher data density by stacking more bits closer together, increasing the storage capacity of the disk. PMR has become the standard for most consumer and enterprise HDDs due to its balance of performance and cost.

- *Shingled Magnetic Recording (SMR):* SMR [93] is a technique that further increases the data density by overlapping the tracks on the disk surface, similar to the shingles on a roof. This method allows for narrower track widths and higher areal density. However, the overlapping of tracks means that writing new data can be slower, as adjacent tracks may need to be rewritten. SMR is particularly useful for applications where data is written sequentially, such as in archival and backup storage.

- *Two-Dimensional Magnetic Recording (TDMR):* TDMR [93] is an emerging technology that utilizes multiple read heads to read data from overlapping tracks simultaneously. This method improves read reliability and data density by using sophisticated signal processing algorithms to distinguish between the closely spaced tracks. TDMR is seen as a promising solution to push the limits of data density beyond what is achievable with PMR and SMR.

- *Heat-Assisted Magnetic Recording (HAMR):* HAMR [31, 53, 84, 85, 86, 87] represents a significant leap in HDD technology by incorporating a laser to heat the magnetic media temporarily. The localized heating reduces the coercivity of the media, allowing data to be written at much higher densities without compromising stability. Once the media cools down, the written data becomes stable and less susceptible to thermal fluctuations. HAMR promises to vastly increase storage capacities, making it a key technology for future high-capacity HDDs.

- *Microwave-Assisted Magnetic Recording (MAMR):* MAMR [82] is another advanced technology that uses a spin-torque oscillator to generate a microwave field, which assists in writing data onto the magnetic media. The microwave field makes it easier to switch the magnetic bits, enabling higher data densities. MAMR offers a pathway to increase storage capacity and reliability while maintaining compatibility with existing manufacturing processes and materials used in HDD production.

Each of these methods has its unique advantages and applications, allowing HDD technology to continue evolving and meeting the growing demands for data storage in various sectors.

Control Problems in HDDs

There are two main types of control problems in HDDs namely the Track Seeking problem and the Track Following problem.

- *Track Seeking*: Involves moving the read/write head from one track to another rapidly and accurately. In this mode, the motions are large and hence disturbances don't play a big role as precision is not the priority. The most important task is to take the head to the commanded track in the least possible time while not affecting the other actuators that are in track following mode. It typically involves generating fast trajectories and

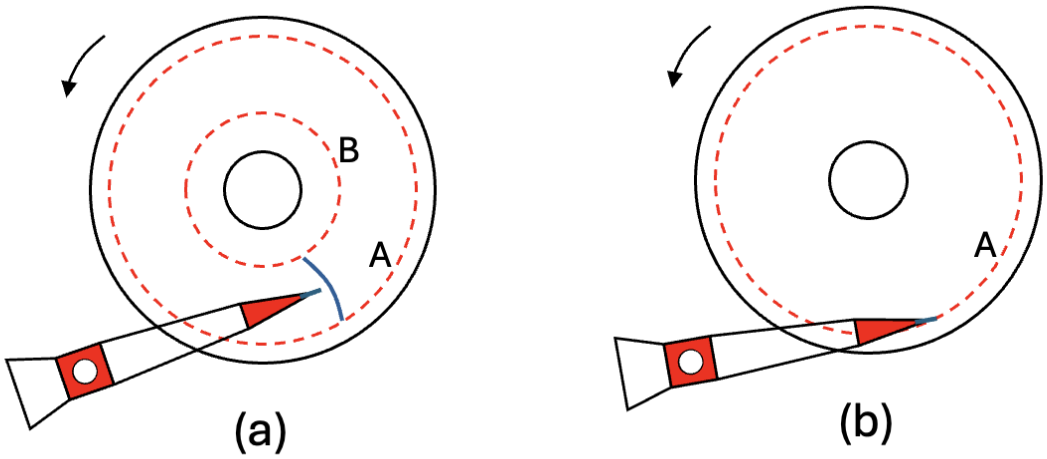

Figure 1.3: (a) Actuator moving from track A to track B (b) Actuator following track A

using feedforward control to track the trajectory. A technique called input shaping [79] is used to prevent the control inputs from containing resonant frequencies of the pivot on which the actuators operate.

- *Track Following*: Focuses on maintaining the read/write head precisely on the target track to ensure accurate data reading and writing. Precision through disturbance rejection is the top most priority as motion by even a few nanomoeters might put the head out of the track and the read/write process has to start afresh.

Disturbances

The following are the different types of disturbances that a HDD faces which makes the control of the head challenging:

- Mechanical Vibrations: Caused by spindle motor imperfections, and operational vibrations.
- Acoustic Vibrations: Due to stacking of multiple HDDs in a small space in data centers, HDDs generate a lot of heat which are handled by fans. These fan blades cause acoustic disturbances inside HDDs by generating pressure waves. There is a also a periodic component associated with it called the blade pass harmonic which occurs whenever the blade of a fan passes through a certain location.

- Thermal Variations: Resulting from changes in temperature that affect the physical dimensions of the drive components.
- Aerodynamic Forces: Generated by the high-speed rotation of platters, impacting the suspension arm and read/write head stability. The present HDDs however use Helium inside them which is less denser than air. The usage of Helium inside HDDs in the recent past has, to a large extent, reduced the aerodynamic forces on the head. Nevertheless, the disturbances still can affect the read/write capabilities given the amount of precision required today. Aerodynamic forces are called the windage in the HDD community.
- Runout: Current day HDDs use the already existing head inside them to write the magnetic tracks on a plane disk instead of an expensive process that requires precision machining. This process however comes with a downside that the tracks written are not exact circles as the actuators are disturbed by some of the above external factors. This offset from the nominal circle is periodic during operation of head and is called the runout. This sometime is specifically called the repeatable runout (RRO) due to its periodic nature.

The disturbances caused by the mechanical vibrations and the aerodynamic forces are lumped into a single category called the non-repeatable runout (NRRO). To mitigate these disturbances, HDDs are equipped with various sensing mechanisms like position error signal (PES) estimation [1, 24] to monitor the precise location of the read/write head, vibration sensors to detect the acceleration of the case and microphones to get acoustic measurements. Vibration sensors and acoustic measurements can be used in feedforward control to anticipate and compensate for any external disturbances. On top of these there is another safety layer which parks the head away from the disk when the accelerations are too high to safely carry out read/write operations.

1.4 Dissertation Outline

The outlines of each of the remaining chapters in the dissertation are presented here. Chapter 2 presents the mathematical preliminaries required to arrive at the results of this dissertation. Chapter 3 discusses designing geometric controllers for fully actuated mechanical systems evolving on smooth manifolds by building potential functions using Physics Informed Neural Networks (PINNs). Chapters 4, 5, 6 and 7 discuss identification, control and classification of Multi Stage Actuator HDD plants. The work presented in this dissertation was first presented in published form in [66, 64, 61, 62, 60, 59, 58].

Chapter 2:

In this chapter, the mathematical preliminaries required for arriving at the results in this dissertation will be presented. Crucial topics include smooth manifolds, Lie groups and

Lie algebras which will be necessary for understanding the geometric control of robotic manipulators. A section about system norms of Linear Time Invariant (LTI) systems which form preliminaries for the control of HDDs are also presented.

Chapter 3:

In this chapter, we introduce a novel approach for designing control laws for fully actuated dynamical systems evolving on manifolds, which leverages the power of neural networks to build invariant potential functions. The weights of the potential function can be tuned to shape the potential functions according to the performance requirements through minimizing a loss function.

Chapter 4:

This chapter discusses clustering stable linear dynamical systems to design robust controllers to enhance performance. It introduces a k-medoids algorithm for hard clustering of stable LTI systems and a Gaussian Mixture Models (GMM) clustering method tailored for a specific class of LTI systems, commonly found in HDD plants. By clustering, the chapter aims to design controllers that are optimal for each cluster, improving overall performance despite plant variations.

Chapter 5:

This chapter describes a data-driven feedback control framework in the frequency domain for designing track-following controllers for multi-actuator HDDs, using frequency response measurements of the plants. The framework improves robustness and avoids model mismatch by using plant measurements directly for controller design, leveraging the H_∞ and H_2 norms for a convex optimization problem. The H_∞ norm shapes the closed-loop transfer functions and ensures stability, while the H_2 norm minimizes the variance of relevant signals in the time domain. The methodology is demonstrated on a triple-stage HDD.

Chapter 6:

In this chapter, we address the interaction between DSAs in Multi-Actuator HDDs and the resulting disturbances. We use the Observer/Kalman Identification (OKID) framework to accurately estimate the disturbance cross transfer function, overcoming the biases introduced by colored runout noise.

Chapter 7:

In this chapter, we address the challenge of system identification for SPR transfer functions. We introduce a convex optimization algorithm using Generalized Orthonormal Basis

Functions (GOBFs) to approximate frequency response data, and a nonconvex optimization problem for direct SPR transfer function estimation.

Key contributions of the dissertation

1. A novel approach for designing invariant control laws from by constructing potential function for fully actuated dynamical systems evolving on manifolds is introduced. This method falls under the umbrella of Physics Informed Neural Networks (PINNs).

2. Clustering methods for stable linear dynamical systems to enhance control performance were proposed:

   a) A k-medoids algorithm for hard clustering of stable Linear Time Invariant (LTI) systems was developed.

   b) Gaussian Mixture Models (GMM) clustering tailored for a specific class of LTI systems in HDD plants was designed.

3. A data-driven feedback control framework in the frequency domain for designing track-following controllers for multi-actuator HDDs was developed:

   a) Frequency response measurements were utilized directly for controller design.

   b) The H_∞ and H_2 norms were leveraged in a convex optimization problem to improve robustness and minimize the variance of relevant signals.

4. The interaction between Dual Stage Actuators (DSAs) in Multi-Actuator HDDs and the resulting disturbances was addressed using the Observer/Kalman Identification (OKID) framework to estimate disturbance cross transfer functions accurately.

5. A convex optimization algorithm set up using Generalized Orthonormal Basis Functions (GOBFs) for approximating frequency response data and a nonconvex optimization problem for direct estimation of Strictly Positive Real (SPR) transfer functions were introduced.

Chapter 2

Mathematical Preliminaries

In this chapter, the necessary mathematical background required to understand and arrive at the results in this dissertation will be presented.

2.1 Groups

In mathematics, a **group** is a set equipped with a binary operation ($\cdot$) that combines any two elements to form a third element, in such a way that four conditions, called the group axioms, are satisfied. The group axioms are:

1. **Closure**: For every pair of elements a and b in the group, the result of the operation, $a \cdot b$, is also in the group.
2. **Associativity**: For every triplet of elements a, b, and c in the group, $(a \cdot b) \cdot c = a \cdot (b \cdot c)$.
3. **Identity element**: There exists an element e in the group such that for every element a in the group, $e \cdot a = a \cdot e = a$. This element e is called the identity element.
4. **Inverse element**: For every element a in the group, there exists an element b (often denoted as a^{-1}) in the group such that $a \cdot b = b \cdot a = e$, where e is the identity element.

Example: A common example of a group is the set of integers under addition. The set of integers ($\mathbb{Z}$) with the operation of addition ($+$) satisfies all four group axioms:

1. **Closure**: The sum of any two integers is an integer.
2. **Associativity**: For any integers a, b, and c, $(a + b) + c = a + (b + c)$.
3. **Identity element**: The identity element is 0, since $a + 0 = a$ for any integer a.
4. **Inverse element**: For any integer a, the inverse is $-a$, since $a + (-a) = 0$.

Groups are fundamental objects in abstract algebra and are used in many areas of mathematics and science to study symmetry, structure, and transformations.

2.2 Manifolds

Manifold

A **manifold** is a topological space that locally resembles Euclidean space. More formally, a topological space M is called an n-dimensional manifold if each point in M has a neighborhood that is homeomorphic to an open subset of $\mathbb{R}^n$. This means that for every point $p \in M$, there exists a neighborhood U of p and a homeomorphism $\varphi : U \to V \subset \mathbb{R}^n$, where V is an open subset of $\mathbb{R}^n$.

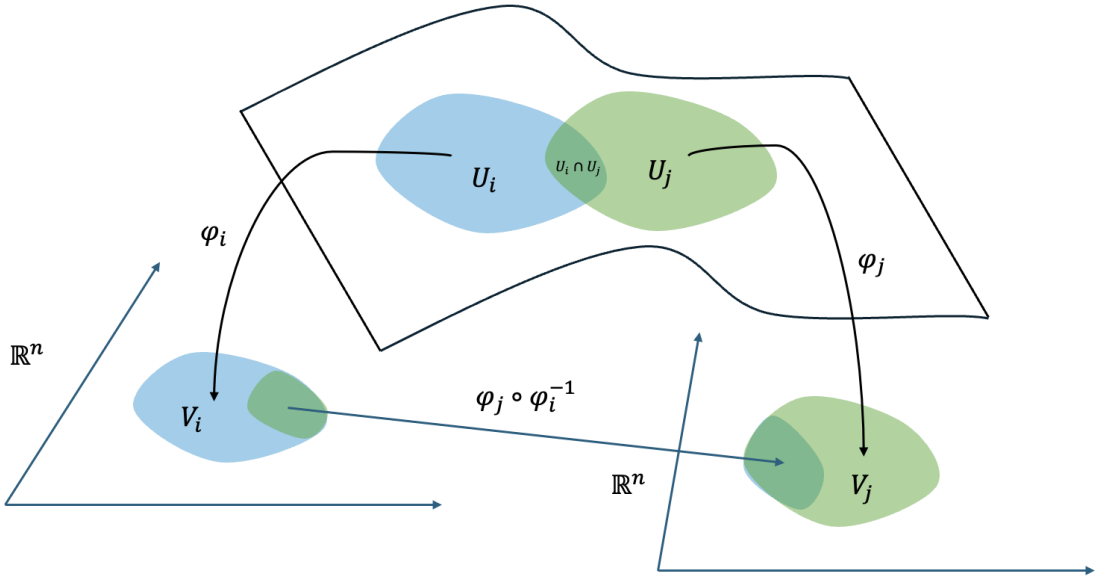

Figure 2.1: Manifold M (n-dimensional) with two open subsets U_i and U_j that map to open sets V_i and V_j through ϕ_i and ϕ_j respectively.

Charts and Atlases

A **chart** on a manifold M is a pair (U, φ) where U is an open subset of M and $\varphi : U \to \varphi(U) \subset \mathbb{R}^n$ is a homeomorphism. The map φ is called a coordinate map or coordinate chart. An **atlas** for a manifold M is a collection of charts $\{(U_i, \varphi_i)\}$ such that the $\bigcup U_i = M$. The charts in the atlas must be compatible: if $U_i \cap U_j \neq \emptyset$, then the transition map $\varphi_j \circ \varphi_i^{-1} : \varphi_i(U_i \cap U_j) \to \varphi_j(U_i \cap U_j)$ is a homeomorphism.

Smooth Manifold

A **smooth manifold** is a manifold with an atlas whose transition maps are all smooth. Formally, an n-dimensional manifold M is a smooth manifold if it has an atlas $\{(U_i, \varphi_i)\}$ such that for any overlapping charts (U_i, φ_i) and (U_j, φ_j), the transition map $\varphi_j \circ \varphi_i^{-1}$ is a smooth function, i.e., it is infinitely differentiable.

Example

Consider the 2-dimensional sphere S^2. It can be covered by two charts:

1. **Chart 1**: Excluding the south pole,

$$U_1 = S^2 \setminus \{(0, 0, -1)\}, \quad \varphi_1 : U_1 \to \mathbb{R}^2 \text{ via stereographic projection from the south pole.}$$

2. **Chart 2**: Excluding the north pole,

$$U_2 = S^2 \setminus \{(0, 0, 1)\}, \quad \varphi_2 : U_2 \to \mathbb{R}^2 \text{ via stereographic projection from the north pole.}$$

The transition maps between these charts are smooth, making S^2 a smooth manifold. The transition map between these charts can be expressed as:

$$\varphi_2 \circ \varphi_1^{-1}(x, y) = \left(\frac{x}{x^2 + y^2 + 1}, \frac{y}{x^2 + y^2 + 1} \right).$$

This map is smooth, confirming that S^2 is a smooth manifold.

Configuration on a Manifold

A **configuration** on a manifold M is a point $q \in M$ that represents a specific position of a system evolving on the manifold. The manifold M is called the **configuration space**, encompassing all possible configurations of the system.

Example

For a rigid body moving in 3D space, the configuration space is $SE(3)$, representing all possible positions and orientations. A configuration is a specific point in $SE(3)$, defined by a rotation matrix $R \in \mathrm{SO}(3)$ and a translation vector $p \in \mathbb{R}^3$.

Variation on a Manifold

A **variation** on a manifold M is a smooth one-parameter family of curves β_{ts} on M, where s is the variation parameter and t is the curve parameter. The variation describes how a curve β_t on M is smoothly perturbed as the parameter s changes.
Formally, a variation is a map $F : (-\epsilon, \epsilon) \times [0, 1] \to M$, where $F(s, t) = \beta_{ts}$, such that for each fixed s, β_{ts} is a smooth curve on M.

Example

Consider a manifold M representing the surface of a sphere S^2. A variation could be a family of great circle arcs β_{ts} on the sphere, where s parameterizes different great circles, and t parameterizes the position along each circle.

2.3 Tangent Space

A **tangent space** T_pM at a point p on a smooth manifold M can be formally defined and constructed in several ways, including using smooth curves, derivations, or local coordinates.

Via Curves

Consider a smooth manifold M and a point $p \in M$. Let $\gamma : (-\epsilon, \epsilon) \to M$ be a smooth curve such that $\gamma(0) = p$. The tangent vector to γ at p is given by the derivative of γ at 0:

$$\left.\frac{d\gamma(t)}{dt}\right|_{t=0} = \dot{\gamma}(0)$$

The tangent space T_pM is then the set of all such tangent vectors, i.e.,

$$T_pM = \{\dot{\gamma}(0) \,|\, \gamma : (-\epsilon, \epsilon) \to M \text{ is a smooth curve with } \gamma(0) = p\}$$

Via Derivations

The tangent space T_pM can also be defined as the space of derivations at p. A derivation at p is a linear map $D : C^\infty(M) \to \mathbb{R}$ that satisfies the Leibniz rule:

$$D(fg) = D(f)g(p) + f(p)D(g),$$

where $f, g \in C^\infty(M)$. The tangent space T_pM is then the set of all such derivations at p.

Via Local Coordinates

If (U, φ) is a local coordinate chart around p with $\varphi : U \to \mathbb{R}^n$ and $\varphi(p) = 0$, then the coordinate functions $(x^1, x^2, \ldots, x^n)$ induce a basis for T_pM. The basis vectors are the partial derivatives at p:

$$\left\{\left.\frac{\partial}{\partial x^i}\right|_p\right\}_{i=1}^n.$$

Any tangent vector $v \in T_pM$ can be written as a linear combination of these basis vectors:

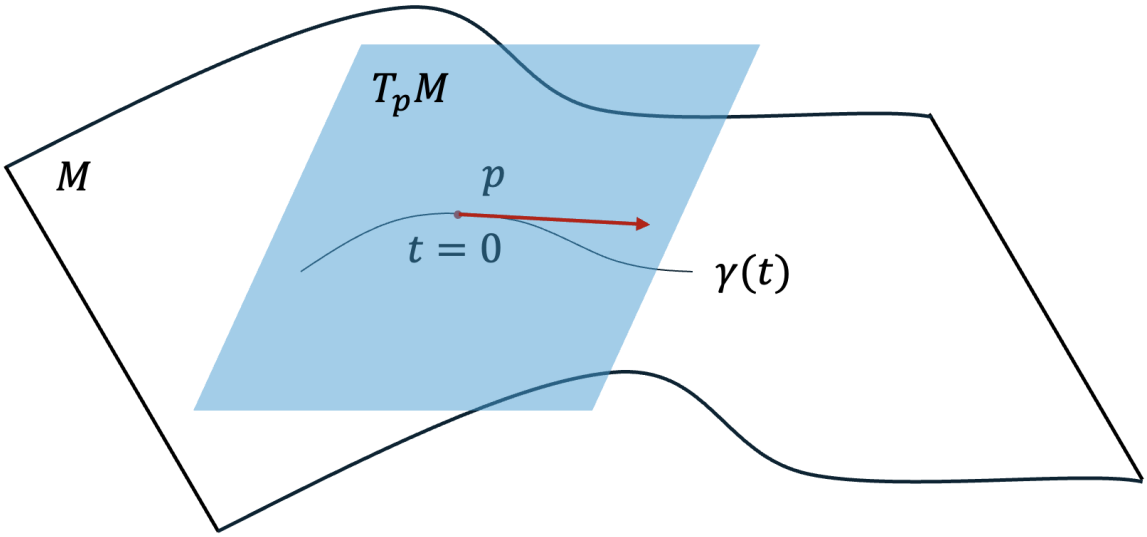

Figure 2.2: Figure showing tangent space T_pM at a point p on the Manifold M. T_pM is the collection of all the vectors that are tangents at p to the curves $\gamma(t)$.

$$v = \sum_{i=1}^{n} v^i \frac{\partial}{\partial x^i}\bigg|_p,$$

where $v^i \in \mathbb{R}$ are the components of v in the coordinate basis.

2.4 Lie Group and Lie Algebra

A **Lie group** is a group that is also a smooth manifold, such that the group operations (multiplication and inversion) are smooth maps. In other words, a Lie group combines the algebraic structure of a group with the geometric structure of a smooth manifold.

Lie Group

A Lie group $\mathcal{G}$ is a group that is also a smooth manifold such that:

- The multiplication map $\mu : \mathcal{G} \times \mathcal{G} \to \mathcal{G}$ defined by $\mu(g, h) = gh$ is smooth.
- The inversion map $\iota : \mathcal{G} \to \mathcal{G}$ defined by $\iota(g) = g^{-1}$ is smooth.

Formally, for $g, h \in \mathcal{G}$:

$$\mu(g, h) = gh \quad \text{and} \quad \iota(g) = g^{-1}.$$

Lie Algebra

A **Lie algebra** $\mathfrak{g}$ is a vector space over a field (typically $\mathbb{R}$ or $\mathbb{C}$) equipped with a binary operation called the Lie bracket $[\cdot,\cdot] : \mathfrak{g} \times \mathfrak{g} \to \mathfrak{g}$ that satisfies the following properties:

1. **Bilinearity**: For all $x, y, z \in \mathfrak{g}$ and scalars $a, b \in \mathbb{R}$:

$$[ax + by, z] = a[x, z] + b[y, z] \quad \text{and} \quad [z, ax + by] = a[z, x] + b[z, y].$$

2. **Alternativity**: For all $x \in \mathfrak{g}$:

$$[x, x] = 0.$$

3. **Jacobi Identity**: For all $x, y, z \in \mathfrak{g}$:

$$[x, [y, z]] + [y, [z, x]] + [z, [x, y]] = 0.$$

The Logarithmic and the Exponential Map between the Lie Group and Lie Algebra

In the context of Lie groups and their corresponding Lie algebras, the logarithmic map (or logarithm map) and its inverse, the exponential map, provide a way to relate elements of the Lie group to elements of the Lie algebra. These are particularly useful for understanding the local structure of the Lie group near the identity element.

Exponential Map

The exponential map $\exp : \mathfrak{g} \to \mathcal{G}$ is a map from the Lie algebra $\mathfrak{g}$ to the Lie group $\mathcal{G}$. For a matrix Lie group, the exponential map is given by the matrix exponential:

$$\exp(A) = \sum_{n=0}^{\infty} \frac{A^n}{n!}, \tag{2.1}$$

where A is an element of the Lie algebra $\mathfrak{g}$ and

$$A^n = \begin{cases} I & \text{if } n = 0 \\ \underbrace{A \cdot A \cdot \ldots \cdot A}_{n} & \text{otherwise.} \end{cases}$$

Logarithmic Map

The logarithmic map (or logarithm map) $\log : \mathcal{G} \to \mathfrak{g}$ is the inverse of the exponential map, defined locally around the identity element of the Lie group. It maps an element of the Lie group $g \in \mathcal{G}$ back to an element of the Lie algebra $\mathfrak{g}$.

For a matrix Lie group, the logarithmic map can be defined using the matrix logarithm. For $R \in \mathcal{G}$ close to the identity element, the logarithm map $\log : \mathcal{G} \to \mathfrak{g}$ is given by:

$$\log(R) = \sum_{n=1}^{\infty} \frac{(-1)^{n+1}}{n}(R - I)^n, \tag{2.2}$$

where I is the identity matrix of appropriate dimension.

2.5 Common Manifolds and Kinematics

In this section, a brief introduction to the commonly used manifolds in the robotics community (unit two sphere $\mathcal{S}^2$, the Special Orthogonal Group (SO(3)) and the Special Euclidean Group (SE(3))) will be presented. Next, kinematics of particles constrained to move on these manifolds will be presented. And finally, given desired trajectories on these manifolds, we will define the notions of errors in configurations and velocities, by defining variations of configurations, that can be used in the design of control laws for mechanical systems evolving on these manifolds with inspiration from [9, 44].

Special Orthogonal Group ($SO(3)$):

Dynamical systems evolving on SO(3), the group of all rotations in three-dimensional space, encapsulate the essence of rotational motion. Examples include the attitude dynamics of spacecraft, where controlling the orientation relative to an inertial frame is crucial for mission success. Other applications include the study of spinning tops and gyroscopes as well as molecular geometries in chemistry. The Special Orthogonal group (SO(3)) represents the set of all possible rotation matrices R without any reflections. The following is a representation of the group as an embedding in $\mathbb{R}^{3\times3}$

$$SO(3) = \{R \in \mathbb{R}^{3\times3} \ : \ R^T R = RR^T = I \ , det(R) = 1\}. \tag{2.3}$$

The Lie algebra of SO(3), denoted $\mathfrak{so}(3)$, is the tangent space at the identity matrix of the Lie group SO(3). It consists of all 3×3 skew-symmetric matrices. A matrix $S \in \mathfrak{so}(3)$ satisfies:

$$S^T = -S.$$

A general element of $\mathfrak{so}(3)$ can be written as:

$$\begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \in \mathbb{R}^{3\times3},$$

where $\omega_1, \omega_2, \omega_3 \in \mathbb{R}$. Given $\omega = [\omega_1, \omega_2, \omega_3]^T \in \mathbb{R}^3$, the hat $\hat{(\cdot)}$ and vee $(\cdot)^\vee$ maps are defined as follows to conveniently move between vector and matrix versions as follows

$$\hat{\omega} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}, \quad \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}^\vee = \omega. \tag{2.4}$$

The exponential map defined in (2.1) applied to the $SO(3)$ manifold takes an element $\hat{\omega}$ from the Lie algebra $\mathfrak{so}(3)$ to the Lie group $SO(3)$. It can be calculated as follows [51]

$$\exp_{\mathfrak{so}(3)}(\hat{\omega}) = I + \sin(\|\omega\|_2)\frac{\hat{\omega}}{\|\omega\|_2} + (1 - \cos(\|\omega\|_2)\frac{\hat{\omega}^2}{\|\omega\|_2^2}, \tag{2.5}$$

where $\|\omega\|_2$ is the 2-norm of the vector ω and is defined as $\|\omega\|_2 = \sqrt{\omega_1^2 + \omega_2^2 + \omega_3^2}$. This formula is called the Rodrigues formula. A detailed derivation of the formula is provided in lemma 5 in Appendix A.

The logarithmic map defined in (2.2) applied to the $SO(3)$ manifold takes an element from the Lie group $SO(3)$ to the Lie algebra $\mathfrak{so}(3)$. It be calculated as follows [51]

$$\log_{SO(3)}(R) = \frac{\phi}{2\sin\phi}(R - R^T) \in \mathfrak{so}(3), \tag{2.6}$$

with $\phi = \cos^{-1}\left(\frac{1}{2}(\mathrm{tr}(R) - 1)\right)$ and $|\phi| < \pi$.

The kinematics of a body that is just restricted to rotate without any translations can be written as

$$\dot{R} = R\hat{\Omega}, \tag{2.7}$$

where $\Omega \in \mathbb{R}^3$ is the angular velocity expressed in the body-fixed frame and $\hat{\Omega} \in \mathfrak{so}(3)$.

A variation on $SO(3)$ consistent with the kinematics in (2.7) can be defined for $\epsilon > 0$ and $\hat{\eta} \in \mathfrak{so}(3)$ as

$$R_{\epsilon\eta} \triangleq R\exp_{\mathfrak{so}(3)}(\epsilon\hat{\eta}). \tag{2.8}$$

The infinitesimal variation $\delta R_\eta \in T_R SO(3)$ (Tangent space of $SO(3)$ at configuration R) can be defined as

$$\delta R_\eta \triangleq \frac{d}{d\epsilon}R_{\epsilon\eta}\Big|_{\epsilon=0} = R\hat{\eta}. \tag{2.9}$$

The configuration error between a desired configuration R_d and the current configuration R can be defined as

$$R_e \triangleq R_d^T R. \tag{2.10}$$

This error is called the right error according to [9]. Note that the error R_e is also an element of $SO(3)$, and the error becomes I when $R = R_d$. The body fixed angular velocity error according to [9] can be defined using the body fixed angular velocity $\hat{\Omega} = R^T\dot{R}$ and the desired angular velocity $\hat{\Omega}_d = R_d^T\dot{R}_d$ via

$$\begin{aligned} \dot{R}_e &= \tfrac{d}{dt}(R_d^T R) = \dot{R}_d^T R + R_d^T \dot{R} \\ &= \hat{\Omega}_d^T R_d^T R + R_d^T R\hat{\Omega} = -\hat{\Omega}_d R_e + R_e\hat{\Omega} \\ &= R_e(\hat{\Omega} - R_e^T\hat{\Omega}_d R_e) = R_e\hat{\Omega}_e. \end{aligned} \tag{2.11}$$

Using $(R_e^T \hat{\Omega}_d R_e)^\vee = R_e^T \Omega_d$ (lemma 1 in A), we define the body fixed angular velocity error as

$$\Omega_e \triangleq \Omega - R_e^T \Omega_d. \tag{2.12}$$

Special Euclidean Group ($SE(3)$):

Dynamical systems evolving on $SE(3)$, the Special Euclidean group that encompasses all rotations and translations in three-dimensional space, is fundamental in understanding and controlling motion in the physical world. This group is crucial for modeling systems like robotic arms, which require precise articulation in 3D space, and autonomous vehicles, including drones and underwater robots, that navigate by adjusting their position and orientation within their environment.
The Special Euclidean group ($SE(3)$) describes the pose of a rigid body in 3D space via a rotation matrix R and a position p. The following is a representation of the group:

$$SE(3) = \{(R, p) \in SO(3) \times \mathbb{R}^3 \ : \ R^T R = I \ , det(R) = 1\}.$$

The group operation is given by:

$$(R_1, p_1) \cdot (R_2, p_2) = (R_1 R_2, R_1 p_2 + p_1).$$

In matrix form, an element of SE(3) can be represented as a 4×4 matrix:

$$g = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix},$$

with the regular matrix multiplication as the group operation. This matrix representation is called the homogeneous representation of the $SE(3)$ group. The Lie algebra of $SE(3)$, denoted $\mathfrak{se}(3)$, consists of all 4×4 matrices of the form:

$$\begin{bmatrix} \hat{\omega} & b \\ 0 & 0 \end{bmatrix},$$

where $\hat{\omega} \in \mathfrak{so}(3)$ (with $\omega \in \mathbb{R}^3$) and $b \in \mathbb{R}^3$. This can be represented in terms of a map $\Gamma(\cdot)$ similar to the hat map defined as follows. Given $\xi = [\omega^T, b^T]^T \in \mathbb{R}^6$,

$$\Gamma(\xi) = \begin{bmatrix} \hat{\omega} & b \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{4 \times 4}. \tag{2.13}$$

The exponential map defined in (2.1) takes an element $\Gamma(\xi)$ from the Lie algebra $\mathfrak{se}(3)$ to the Lie group $SE(3)$ and can be obtained as (lemma 6 in A) ([51])

$$\exp_{\mathfrak{se}(3)}(\Gamma(\xi)) = \begin{bmatrix} \exp_{\mathfrak{so}(3)}(\hat{\omega}) & A(\omega) b \\ 0 & 1 \end{bmatrix}, \tag{2.14}$$

with

$$A(\omega)=I+\left(\frac{1-\cos(\|\omega\|_2)}{\|\omega\|_2}\right)\frac{\hat{\omega}}{\|\omega\|_2}+\left(1-\frac{\sin(\|\omega\|_2)}{\|\psi\|}\right)\frac{\hat{\omega}^2}{\|\omega\|_2^2}. \tag{2.15}$$

$\Gamma(\xi) = \log(g) \in \mathfrak{g}$ can now be defined as the exponential coordinates of the group element g, and the logarithmic map is regarded as a local chart of the manifold G according to [8]. The logarithm map on $g = (R, p) \in SE(3)$ such that $\mathrm{tr}(R) \neq -1$ is defined in the following way, given $g = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix} \in SE(3)$

$$\begin{aligned} \log_{SE(3)}(g) &= \begin{bmatrix} \hat{\omega} & A^{-1}(\omega)p \\ 0 & 0 \end{bmatrix} \in \mathfrak{se}(3), \quad \text{with} \\ A^{-1}(\omega) &= I - \frac{1}{2}\hat{\omega} + (1 - \alpha(\|\omega\|))\,\frac{\hat{\omega}^2}{\|\omega\|^2}, \end{aligned} \tag{2.16}$$

where $\hat{\omega} = \log_{SO(3)}(R)$, $\alpha(y) = (y/2)\cot(y/2)$.
The kinematics of a body evolving on $SE(3)$ can be written as

$$\dot{g} = g\Gamma(V^b), \tag{2.17}$$

where

$$g = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix}, V^b = \begin{bmatrix} \Omega \\ v \end{bmatrix}, \Gamma(V^b) = \begin{bmatrix} \hat{\Omega} & v \\ 0 & 0 \end{bmatrix}. \tag{2.18}$$

Here g is called the homogeneous representation of the $SE(3)$ group and $v \in \mathbb{R}^3$ is the translational velocity with $\dot{p} = v$ represented in the body coordinates.
For $\epsilon > 0$, $\eta = [\eta_1^T, \eta_2^T]^T \in \mathbb{R}^6$ with $\eta_1, \eta_2 \in \mathbb{R}^3$, a variation on $SE(3)$ can be defined using the exponential map (2.14) as

$$g_{\epsilon\eta} \triangleq g \exp_{\mathfrak{se}(3)}(\epsilon\Gamma(\eta)) \tag{2.19}$$

Note that we regard η_1 as a translational element and η_2 as a rotational element following the convention of [51]. The infinitesimal variation can now be defined as

$$\delta g_\eta \triangleq \tfrac{d}{d\epsilon} g_{\epsilon\eta}\big|_{\epsilon=0} = g\Gamma(\eta). \tag{2.20}$$

The configuration error between a desired configuration g_d and the current configuration g can be defined as follows utilizing R_e from (2.10) and $p_e = p - p_d$ denoting the translational error

$$g_e \triangleq g_d^{-1} g = \begin{bmatrix} R_e & R_d^T p_e \\ 0 & 1 \end{bmatrix}, g_d = \begin{bmatrix} R_d & p_d \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{\eta}_2 & \eta_1 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} R\hat{\eta}_2 & R\eta_1 \\ 0 & 1 \end{bmatrix}. \tag{2.21}$$

Note that the configuration error g_e is also an element of $SE(3)$ and the error becomes I when $g = g_d$.
By taking the time derivative of g_e and following the same steps as (2.11), we can obtain the following

$$\dot{g}_e = g_e\Gamma(e_V), \tag{2.22}$$

where e_V is the velocity error defined by the following utilizing the desired quantities with subscript d

$$e_V \triangleq \underbrace{\begin{bmatrix} v \\ \Omega \end{bmatrix}}_{V^b} - \underbrace{\begin{bmatrix} R_e^T v_d + R_e^T \hat{\Omega}_d R_d^T (p - p_d) \\ R_e^T \Omega_d \end{bmatrix}}_{V_d^*} = \begin{bmatrix} e_v \\ e_\Omega \end{bmatrix}. \tag{2.23}$$

Two Sphere ($\mathcal{S}^2$):

Dynamical systems on the two sphere, such as the spherical pendulum and spin axis stabilization of satellites, explore the complex behaviors of objects constrained to move on a spherical surface.
The spherical pendulum shows interesting motion patterns due to its unrestricted swinging directions, while spin axis stabilization is essential for maintaining a satellite's orientation in orbit. These systems highlight the rich dynamics and control challenges inherent in spherical geometries. A unit two sphere can be represented as follows

$$\mathcal{S}^2 = \{r \in \mathbb{R}^3 \ : \ \|r\|_2 = 1\}, \tag{2.24}$$

where r is the coordinate of a point on the sphere with respect to a spatial frame attached to the origin (center of the sphere). The kinematics of a point whose configuration evolves on $\mathcal{S}^2$ can be written as

$$\dot{r} = \omega \times r = \hat{\omega} r, \tag{2.25}$$

where $\omega \in \mathbb{R}^3$ is an angular velocity vector and $\times$ represent the cross product. The time derivative $\dot{r}$ is an element on the tangent space ($T_r\mathcal{S}^2$) of $\mathcal{S}^2$ at r. It is easy to visualize that any vector tangent to the sphere will have to be perpendicular to r as r is always normal to the sphere. And, $\omega \times r$ is always perpendicular to both ω and r. For a desired configuration $r_d \in \mathcal{S}^2$, the configuration error $r_e \in \mathbb{R}$ can be defined as

$$r_e \triangleq r_d^T r. \tag{2.26}$$

Note that, when the configuration r reaches r_d, r_e becomes 1 and not 0. Though $1 - r_d^T r$ is also a valid representations of the error, which can make the error zero at the desired configuration, we will use the definition of r_e in (2.26) to be consistent with the definitions of errors on the SO(3) and SE(3) manifolds in this dissertations which are also elements on the manifold. Notice that, r_e is not an element on the manifold but a scalar. With the current definition of error, the control law should be designed to drive the error to 1 instead of 0.
A variation on the sphere can be defined as a configuration on the sphere obtained by flowing with an angular velocity $\eta \in \mathbb{R}^3$ for a time $\epsilon \in \mathbb{R}$ from the configuration r. Using the exponential map (see [51]), the variation $r_{\epsilon\eta}$ can be defined as follows

$$r_{\epsilon\eta} \triangleq \exp_{\mathfrak{so}(3)}(\epsilon\hat{\eta}) r. \tag{2.27}$$

(a) (b)

Figure 2.3: (a) Variation of an element on $\mathcal{S}^2$ (b) Configurations and their Tangent spaces from [66]

The infinitesimal variation $\delta r_\eta \in T_r\mathcal{S}^2$, can now be defined as

$$\delta r_\eta \triangleq \tfrac{d}{d\epsilon} r_{\epsilon\eta}\big|_{\epsilon=0} = \hat{\eta} r = \eta \times r. \tag{2.28}$$

The velocity error on $\mathcal{S}^2$ can now be defined using a transport map from [9] $\mathcal{T}_{(r,r_d)} : T_{r_d}\mathcal{S}^2 \mapsto T_r\mathcal{S}^2$ as

$$e_v \triangleq \dot{r} - \mathcal{T}_{(r,r_d)}\dot{r}_d, \tag{2.29}$$

where the transport map $\mathcal{T}_{(r,r_d)}$ on $\mathcal{S}^2$ is given by

$$\mathcal{T}_{(r,r_d)} = (r_d^T r)I + (\widehat{r_d \times r}). \tag{2.30}$$

Since the current velocity $\dot{r}$ and the desired velocity $\dot{r}_d$ lie on different tangent spaces, i.e., $T_r\mathcal{S}^2$ and $T_{r_d}\mathcal{S}^2$ respectively, they cannot be directly compared like in the case of Euclidean space. Therefore, it is necessary to transport one vector to the tangent space of the other and compare them. This comparison can be made with the help of this transport map, and the potential functions must be designed such that they are compatible with the transport map.

Left Invariance:

It can be seen from the following equations that by transforming the current and desired configurations both from the left arbitrarily by the same translation, we do not get a change in the error.

$$\begin{aligned}
(R_l r_d)^T (R_l r) &= r_d^T (R_l^T R_l) r = r_d^T r = r_e, \\
(R_l R_d)^T (R_l R) &= R_d^T (R_l^T R_l) R = R_d^T R = R_e, \\
(g_l g_d)^{-1} (g_l g) &= g_d^{-1} (g_l^{-1} g_l) g = g_d^{-1} g = g_e,
\end{aligned} \tag{2.31}$$

This is an essential property as incorporating this property allows us to transfer trained skills from one scene to another scene without any new training. We will use these errors to construct potentials in chapter 3. Since the potentials depend solely on the errors, the potential functions are also left invariant. It is also easy to see that for left error representations, we have right invariance. [76] presents a more elaborate explanation of invariance.

Variation of a function:

For any scalar function $u : M \mapsto \mathbb{R}$ mapping from a manifold M to the the reals, the derivative $D_m u$ at $m \in M$ can be defined with the help of its infinitesimal variation δu described as

$$\delta u(m) = \frac{d}{d\epsilon} u_{\epsilon\eta}\big|_{\epsilon=0} = \frac{d}{d\epsilon} u(m_{\epsilon\eta})\big|_{\epsilon=0} = D_m u(m) \cdot \eta, \tag{2.32}$$

with $m_{\varepsilon\eta}$ as the variation of m on M obtained by flowing from m with η for a period of ϵ. Furthermore,

$$D_m u \triangleq \frac{\partial}{\partial \eta} \delta u. \tag{2.33}$$

This derivative $D_m u$ will be used to define the control law in the later sections.

2.6 Linear Time Invariant Systems Norms

In this section, a few mathematical preliminaries from robust control theory for Linear Time Invariant (LTI) systems will be provided.

For a continuous time transfer function $G_{m\times n}(s)$ or a discrete time transfer function $G_{m\times n}(z)$ with sampling time T_s with m inputs and n outputs, two types of system norms are defined namely H_∞ norm and H_2 norm. Let the realization of a continuous time system G in terms of system matrices A, B, C, and D be denoted as

$$G \sim \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]. \tag{2.34}$$

H_∞ Norm:

The H_∞ norm is defined as the supremum of the largest singular value of the transfer function evaluated across the entire frequency range.
For continuous time systems, the H_∞ norm is defined as

$$||G_{m\times n}||_\infty = \sup_{\omega\in\Omega} \bar{\sigma}(G_{m\times n}(j\omega)), \tag{2.35}$$

where $\Omega = (-\infty, \infty)$ is the entire frequency range and $\bar{\sigma}(\cdot)$ is the largest singular value.
By utilizing the definition of matrix norms, the H_∞ norm can also be equivalently written as

$$||G_{m\times n}||_\infty = \sup_{\omega\in\Omega} ||G_{m\times n}(j\omega)||_2. \tag{2.36}$$

For Single Input Single Output (SISO) systems, the definition boils down to

$$||G_{1\times1}||_\infty = \sup_{\omega\in\Omega} |G_{m\times n}(j\omega)|. \tag{2.37}$$

Similarly, for discrete time systems, the H_∞ norm is defined as

$$||G_{m\times n}||_\infty = \sup_{\omega\in\Omega} \bar{\sigma}(G_{m\times n}(e^{j\omega T_s})) = \sup_{\omega\in\Omega} ||G_{m\times n}(e^{j\omega T_s})||_2, \tag{2.38}$$

where $\Omega = (-\frac{\pi}{T_s}, \frac{\pi}{T_s}]$ denotes the entire frequency range upto the Nyquist frequency. For SISO systems, the definition boils down to

$$||G_{1\times1}||_\infty = \sup_{\omega\in\Omega} |G_{m\times n}(e^{j\omega T_s})|. \tag{2.39}$$

For Single Input Single Output (SISO) systems, it can be seen that the H_∞ norm is the peak of the bode magnitude plot which is also the same as the maximum amplification of an input sinusoidal wave for stable systems. Therefore

$$||G_{m\times n}||_\infty = \sup_{u\in\mathcal{L}_2, ||u||_2\geq1} \frac{||y||_2}{||u||_2}, \tag{2.40}$$

where the space $\mathcal{L}_2$ is the space of all square integrable functions.
Instead of sweeping the entire frequency range, the H_∞ norm can be obtained from the state matrices by utilizing the bounded real lemma. For this, a Hamiltonian matrix ($\mathcal{H}$) is constructed as follows ([40])

$$\mathcal{H} = \begin{bmatrix} A & \frac{BB^T}{\gamma^2} \\ -C^TC & -A^T \end{bmatrix}. \tag{2.41}$$

The smallest γ that keeps the Hamiltonian matrix asymptotically stable is the H_∞ norm of the system. This value is usually found using an iterative scheme like the bisection technique. An equivalent result exists for discrete time systems as well.

H_2 Norm:

H_2 norm of a system can be viewed as the energy of the dynamical system. For SISO systems, it is a representation of the area under the bode plot. The H_2 norm of a system is finite only when the system is stable and the feedthrough matrix D is zero i.e., the system must be strictly proper. The H_2 norm for a continuous time Multiple Input Multiple Output (MIMO) system is defined as

$$||G_{m\times n}||_2^2 = \frac{1}{2\pi}\int_{-\infty}^{\infty} tr[G(j\omega)^*G(j\omega)]d\omega \tag{2.42a}$$

$$= \frac{1}{2\pi}\int_{-\infty}^{\infty} tr[G(-j\omega)^TG(j\omega)]d\omega, \tag{2.42b}$$

where $(\cdot)^*$ represents the complex conjugate. For systems whose frequency responses are available instead of the model, the approximate H_2 norm is given as

$$||G_{m\times n}||_2^2 \approx \frac{1}{2\pi}\sum_{k=1}^{N} tr[G(j\omega_k)^* G(j\omega_k)]\Delta\omega_k, \tag{2.43}$$

where ω_k is the k^{th} frequency at which the response is available.
Utilizing Parseval's theorem, the H_2 norm in (2.42) can be written in terms of the system matrices as

$$||G_{m\times n}||_2^2 = tr[CPC^T], \tag{2.44}$$

where $tr[\cdot]$ denotes the trace and the matrix P is the controllability grammian given by

$$P = \int_{-\infty}^{\infty} e^{At} BB^T e^{A^T t} dt, \tag{2.45}$$

which is also the solution of the Lyapunov equation given by

$$AP + PA^T = -BB^T. \tag{2.46}$$

We can also express (2.44) in terms of the input matrix B, and in that case, the grammian would be the observability grammian. H_2 norms can also be calculated for discrete time systems in a similar fashion.

Part I

Geometric Control of Fully Actuated Mechanical Systems

Chapter 3

Tracking and Variable Impedance Control using Deep Geometric Potential Fields

3.1 Introduction

The development of learning-based control laws for systems operating on smooth manifolds remains an interesting problem in the realm of control systems engineering. Such systems are crucial in a variety of applications, such as aerial robots [43, 42, 45, 35, 37, 38], and robotic manipulators with impedance and admittance control systems [23, 76, 77, 75]. These applications require advanced control strategies capable of managing the inherent nonlinearities and complexities of the systems. In this framework, employing potential functions whose elastic forces determine the control laws presents itself as an effective approach for controller synthesis [8, 9], offering a reliable method to direct system behavior toward desired states or trajectories.

It has been noted that potential functions designed in [9, 43] on the Special Orthogonal group ($SO(3)$) exhibit vanishing gradients when the discrepancy between the desired and current orientation is π radians. Consequently, due to the configuration of the potential functions, at specific errors, the forces are minimal and have slow responses even when the errors are significant. Research in [8, 56, 55] uses a metric on the Lie algebras of $SO(3)$ and the Special Euclidean group ($SE(3)$) that is uniformly quadratic, but the logarithmic map is not defined when the desired and current configurations are π radians apart since a rotation about any axis by π radians would yield the same orientation. These potential functions can integrate matrix gains for positions and orientations, which can be utilized to shape the system's response, but tuning these gains easily based on the required response is not as straightforward as in the case of linear systems. These limitations inspire us to use neural networks, for their representational power, to construct potential functions whose gradients produce elastic and damping forces which are nonlinear generalizations of proportional and

derivative terms respectively. Such a method would fall under the umbrella of Physics Informed Neural Networks (PINNs) [68, 69, 16, 18, 70, 50, 15, 73, 74, 74, 94] as instead of using a plain feedforward neural network, the knowledge of physics of springs and dampers are incorporated into the framework. These neural network potential functions might still possess only almost global properties instead of global properties since it is impossible to create a continuous control law that provides a continuous vector field on a compact manifold with a globally asymptotic equilibrium point [6]. Since this cannot be overcome, instead of having potential functions with fixed properties, we aim to create potential functions whose "problematic" points are positioned in the regions on the manifold that are not of interest to us. The way we move the problematic points away is by defining an optimization problem and using gradient descent to encourage the problematic points to not fall around the desired trajectories. Though this may seem a little vague, this will become clear when the cost function and the optimization problem are defined.

Another significant application of creating potential functions using the representational power of neural networks is in learning potential fields through demonstrations by an expert in an invariant manner. In our previous work in [76], we present a neural network architecture for learning gains as a function of the state from expert demonstrations using potential fields. The work in this chapter can serve as a generalized framework for learning these state-dependent gains through potential functions, which provide a provably stabilizing control law directly during inference.

One of the challenges in constructing these potential functions is ensuring that their structure satisfies specific Lyapunov function properties, such as being zero at the equilibrium and positive everywhere else (or equivalently being lower bounded and attaining the minimum at its equilibrium). These properties will be achieved through the use of Input Convex Neural Networks (ICNNs) [2] and their application to the construction of Lyapunov functions for learning stable dynamical systems in [47]. Although these Lyapunov functions were initially constructed for Euclidean spaces, the subsequent sections will demonstrate how this approach is also advantageous for systems evolving on smooth manifolds. The primary application of this method will be for the orientation control of a spacecraft on $SO(3)$ and for the impedance control of a robotic manipulator whose end effector pose evolves on $SE(3)$. The main contributions of this chapter can be summarized as follows:

1. A Neural Network architecture to construct deep invariant geometric potential functions on smooth manifolds is presented. These potential functions are designed to produce stabilizing elastic and dissipative wrench pairs from any random initial network setup.

2. Kinematic control laws utilizing conservative potential functions and their stability are presented.

3. An impedance control law for robotic manipulators and an orientation control law for satellites based on these potential functions, including a training procedure aimed at refining the potential function to enhance convergence of trajectories is presented.

3.2 Standard Potential Functions

In this section, we will present two types of conservative geometric potential functions that are common in the literature for $SO(3)$ manifolds. Though their simplicity makes them an easy choice for use in geometric control laws, a short discussion about their shortcomings will be presented.

Potential Function 1: Frobenius Norm Potential

First, we will consider the potential function from [9] utilizing the trace operator $tr[\cdot]$ which was later used in [43, 42, 45, 35, 36, 37, 96] etc. It was also shown that this potential function can be expressed using the Frobenius norm in [77].

$$\Psi_1(R_e) = tr[I - R_e] = tr[I - R_d^T R]. \tag{3.1}$$

$R, R_d, R_e = R_d^T R \in SO(3)$ and $\Psi_1 : SO(3) \mapsto \mathbb{R}^+$. The elastic force f_{c1} generated by this potential function can be obtained from its variation.

$$\begin{aligned}
\delta\Psi_1{}_\eta &= \tfrac{d}{d\epsilon}\Psi_{1,\epsilon\eta}\big|_{\epsilon=0} = \tfrac{d}{d\epsilon} tr[I - R_d^T R_{\epsilon\eta}]\big|_{\epsilon=0} \\
&= \tfrac{d}{d\epsilon} tr[I - R_d^T R \exp(\epsilon\hat{\eta})]\big|_{\epsilon=0} \\
&= tr[-R_d^T R \exp(\epsilon\hat{\eta})\hat{\eta}]\big|_{\epsilon=0} \\
&= tr[-R_d^T R\hat{\eta}] \\
&\text{We can use the property of trace that } tr[A\hat{b}] = (A^T - A)^\vee \cdot b \text{ (lemma 4) and get} \\
&= (R_d^T R - R^T R_d)^\vee \cdot \eta = (R_e - R_e^T)^\vee \cdot \eta = -f_{c1} \cdot \eta.
\end{aligned} \tag{3.2}$$

Potential Function 2:

We now consider the potentials from [8, 56, 55] (Lie algebra potentials) using the logarithmic map from the Lie group to its Lie algebra (see [51]) as follows

$$\Psi_2(R_e) = \tfrac{1}{2}\|\log(R_e)\|_F^2 = \tfrac{1}{2}\|\log(R_d^T R)\|_F^2, \tag{3.3}$$

where $\|A\|_F = tr[A^T A]^{1/2}$ represents the Frobenius norm of the matrix A. The elastic force f_{c2} generated from this potential can be obtained from its variation as follows

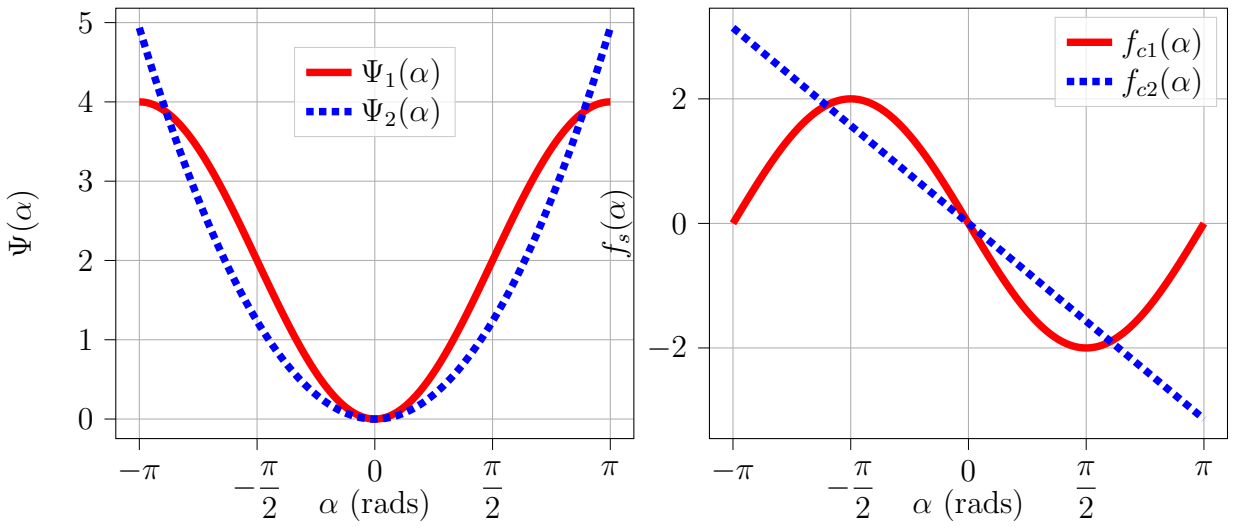

Figure 3.1: Comparison of Potential Functions (left) and corresponding elastic forces (right) for Frobenius Norm potential (Lie group based) and Logarithm map potential (Lie algebra based).

$$
\begin{aligned}
\delta\Psi_{2,\eta} &= \tfrac{d}{d\epsilon}\Psi_{2,\epsilon\eta}\big|_{\epsilon=0} = \tfrac{1}{2}\,\tfrac{d}{d\epsilon}\|\log(R_d^T R_{\epsilon\eta})\|_F^2\big|_{\epsilon=0} \\
&= \tfrac{1}{2}\,\tfrac{d}{d\epsilon}\|\log(R_d^T R\exp(\epsilon\hat{\eta}))\|_F^2\big|_{\epsilon=0} \\
&\text{since } R_d^T R \in SO(3), \text{ we can express it in terms of the exponential map in (2.5)} \\
&\text{using an element } \hat{v} \in \mathfrak{so}(3), \text{ such that } R_d^T R = \exp(\hat{v}), \text{ and obtain} \\
&= \tfrac{1}{2}\,\tfrac{d}{d\epsilon}\|\log(\exp(\hat{v})\exp(\epsilon\hat{\eta}))\|_F^2\big|_{\epsilon=0} \\
&\text{since } \hat{v} \text{ and } \epsilon\hat{\eta} \text{ are skew symmetric matrices, their matrix multiplication commutes} \\
&\text{and using the fact that } \exp(A+B) = \exp(A)\cdot\exp(B) \text{ when } A\cdot B = B\cdot A, \\
&\text{we can write} \\
&= \tfrac{1}{2}\,\tfrac{d}{d\epsilon}\|\log(\exp(\hat{v}+\epsilon\hat{\eta}))\|_F^2\big|_{\epsilon=0} \\
&= \tfrac{1}{2}\,\tfrac{d}{d\epsilon}\|\hat{v}+\epsilon\hat{\eta}\|_F^2\big|_{\epsilon=0} \\
&= \tfrac{1}{2}\,\tfrac{d}{d\epsilon}\mathrm{tr}\left((\hat{v}+\epsilon\hat{\eta})^T(\hat{v}+\epsilon\hat{\eta})\right)\big|_{\epsilon=0} \\
&= \tfrac{1}{2}\left(2\mathrm{tr}\left(\hat{v}^T\hat{\eta}\right) + 2\epsilon\mathrm{tr}\left(\hat{\eta}^T\hat{\eta}\right)\right)\big|_{\epsilon=0} \\
&= \mathrm{tr}\left(\hat{v}^T\hat{\eta}\right) \\
&= \mathrm{tr}\left(\log(R_e)^T\hat{\eta}\right) \\
&\text{we can use the property of trace that } tr[A\hat{b}] = (A^T - A)^\vee \cdot b \text{ (lemma4) and get} \\
&= -\log(R_e)^\vee \cdot \eta = -f_{c2}\cdot\eta. \qquad (3.4)
\end{aligned}
$$

Fig. 3.1 shows a comparison between the potential function and their corresponding

elastic forces for $SO(2)$ manifold (planar rotations with α as the angle between desired and current configurations). Here we can see that for smaller errors, Ψ_1 is slightly better, and for larger errors, Ψ_2 outperforms as f_{c1} tends to become zero even when the error is π radians. Since the rotations are planar, they can be viewed as rotations about a fixed axis and there is no ambiguity in the axis. But when it comes to 3D rotations, a rotation about any axis by π radians causes the same orientation, which causes a discontinuity in Ψ_2, and therefore, the logarithmic map is defined only when the argument is not the identity. We wish to overcome these issues by using more expressive potential functions that can be built through neural networks.

3.3 Potential Functions

In this section, a methodology for designing positive definite conservative potential functions and positive semi definite dissipative potential functions utilizing Fully Input Convex Neural Networks (FICNNs) and Partially Input Convex Neural Networks (PICNNs) from [2] and their use in the construction of Lyapunov functions [47] will be presented.

Input Convex Neural Networks

Input Convex Neural Networks (ICNNs) ([2]) are a class of neural networks designed to ensure convexity with respect to their inputs.

Fully Input Convex Neural Networks (FICNNs)

In an FICNN, every layer of the network is designed to preserve convexity with respect to the input. This is accomplished by using convex, non-decreasing activation functions, such as ReLU, a smoothed ReLU, softplus etc, and ensuring that the weights applied to the inputs are non-negative entries. Constraints on weights and biases are applied to guarantee that the entire network remains convex with respect to its inputs. An FICNN can be fully represented with the help of the following recursive equations representing the outputs of each layer of the neural network.

$$\begin{aligned} z_1 &= \sigma_0(W_0 x + b_0) \\ z_{i+1} &= \sigma_i(W_i z_i + U_i x + b_i), \quad \text{for } i = 1, \ldots, k-1 \end{aligned} \tag{3.5}$$

Explanation of Terms:

- x: The input vector.
- z_i: The output of layer i after the activations.
- W_i: Weight matrices applied to the activations z_i. The number of columns of W_i decides the size of layer i and the number of rows of W_i decides the size of layer $i+1$.

- U_i: Non-negative weight matrices applied directly to the input x, ensuring convexity. The number of columns of U_i must be the same as the input dimension and the number of rows of U_i must match the size of the layer $i+1$.
- b_i: Bias vectors added at each layer i. The size of the bias vector must be the same as the size of layer i.
- σ_i: Activation functions, which must be convex and non-decreasing (e.g., ReLU).
- z_k: The final output of the network with k layers.

Partially Input Convex Neural Networks (PICNNs)

PICNNs relax the requirement for convexity by allowing certain layers or components of the network to be non-convex. This approach enables the network to retain convexity with respect to a subset of inputs while allowing for greater expressiveness and flexibility in modeling complex, non-convex relationships. The structure of a PICNN consists of two paths, a non-convex path in input x and a convex path in input y. The network can be represented in a recursive way in terms of the outputs of layers along each path as follows

$$\begin{aligned} u_0 &= x \\ u_{i+1} &= \tilde{\sigma}_i(\tilde{W}_i u_i + \tilde{b}_i) \\ z_{i+1} &= \sigma_i \left(U_i^{(z)} \left(z_i \circ \left[W_i^{(zu)} u_i + b_i^{(z)} \right]_+ \right) + \right. \\ &\qquad \left. W_i^{(y)} \left(y \circ \left(W_i^{(yu)} u_i + b_i^{(y)} \right) \right) + W_i^{(u)} u_i + b_i \right). \end{aligned} \tag{3.6}$$

Explanation of Terms

- x : The input vector over which the function need not be convex.
- y : The input vector over which the function needs to be convex.
- u_i: Output of layer i in the non-convex path.
- $\tilde{W}_i$: Weight matrix applied to u_i.
- $\tilde{b}_i$: Bias vector along non-convex path added at each layer i.
- $\tilde{\sigma}_i$: Activation function applied to the linear combination along the non-convex path. The activations need not be convex.
- z_i: Output of layer i in the convex path after the activations.
- $U_i^{(z)}$: Non-negative weight matrix applied to z_i to ensure convexity.
- $W_i^{(zu)}$: Weight matrix applied to u_i in interaction with z_i.

- $b_i^{(z)}$: Bias vector for z at layer i.
- $\circ$: Element-wise product (also known as the Hadamard product).
- σ_i: Convex, non-decreasing activation function applied to the entire expression.
- $W_i^{(y)}$: Weight matrix applied to the y-input interaction at layer i.
- $W_i^{(yu)}$: Weight matrix applied to u_i in interaction with y.
- $b_i^{(y)}$: Bias vector for y at layer i.
- $W_i^{(u)}$: Weights applied to u_i along the convex path.
- b_i: Bias added to the final sum along the convex path at layer i.
- z_k: Final output of the network.
- $u_0 = x$: Initial state of the non-convex u-path.

For the remainder of the section, quantities defined by W are real matrices, quantities defined by U are non-negative matrices (non-negative entries), and quantities defined by b are real bias vectors, all of appropriate dimensions. The nonlinear activation function σ must be a convex and non-decreasing like ReLU but $\tilde{\sigma}$ could be arbitrary nonlinear activation.

Positive Definiteness

The built convex functions using ICNNs need not be positive semi-definite. To ensure positive definiteness, we first shift the output of ICNN to be zero at the locations we desire and pass the shifted output through a non negative activation function like ReLU. To ensure smoothness of functions for easier gradient flows, we can use a smoothed ReLU as follows

$$\sigma(x) = \begin{cases} 0 & \text{if } x \leq 0, \\ x^2/2d & \text{if } 0 < x < d, \\ x - d/2 & \text{otherwise.} \end{cases} \tag{3.7}$$

Here the parameter d is used to control the smoothness around the origin. Fig. 3.2 shows a smoothed ReLU activation function and its derivative.
A small positive definite term is added at the end using $\epsilon_1, \epsilon_2 > 0$ to ensure the positive definiteness of the potential functions.

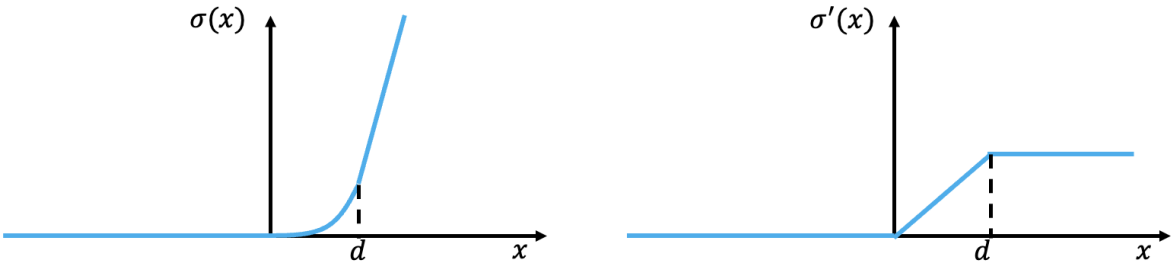

Figure 3.2: Figure showing smoothed ReLU (left) and its derivative (right).

Properties of Conservative Potentials

We consider the following properties to build conservative potential functions Ψ that generate elastic forces f_c which are the nonlinear generalizations of proportional terms in PD control. The potential function Ψ

1. is a function of only the configuration error.
2. is positive definite.
3. obtains its unique global minimum when the current configuration and desired configuration are the same.

The aim is to make the conservative potential function zero when the current configuration and desired configuration coincide and positive everywhere else. Though convexity is a restriction, we can relax this by adding another layer of an invertible residual network [5] layer before the ICNN to improve representative power. A caveat here is that though we are using an FICNN to build the conservative potential functions, the manifold is not convex, and hence, the conservative potential function and domain pair together are not convex, but the conservative potential function with a convex domain remains convex.

Properties of Dissipative Potentials

We consider the following properties to build dissipative potential functions $\mathcal{R}$ (Rayleigh dissipation potentials) that generate damping forces which are the nonlinear generalizations of derivative terms in PD control. The potential function $\mathcal{R}$

1. is a non negative function of the configuration error (first argument) and the velocity error (second argument).
2. is positive definite in its second argument.

3. is zero when the velocity error is zero.

$$0 = \mathcal{R}(\cdot, 0) \leq \mathcal{R}(\cdot, \cdot) \tag{3.8}$$

4. is convex in its second argument i.e., velocity error. This property is necessary for proving asymptotic stability.

Potential Functions on $\mathcal{S}^2$:

Conservative Potentials:

We can define the conservative potential function $\Psi : [-1, 1] \mapsto \mathbb{R}^+$ such that $\Psi(1) = 0$ as follows:

$$\begin{aligned} z_1 &= \sigma_0(W_0 r_e + b_0) \\ z_{i+1} &= \sigma_i(U_i z_i + W_i r_e + b_i), i = 1, \ldots, k-1 \\ \Phi(r_e) &\equiv z_k \\ \Psi(r_e) &= \sigma_{k+1}(\Phi(r_e) - \Phi(1)) + \epsilon_1 \|1 - r_e\|_2^2. \end{aligned} \tag{3.9}$$

Here

- r_e is the configuration error on $\mathcal{S}^2$ defined in (2.26).
- z_i are the outputs of the i^{th} layer.
- σ_i are the convex non-decreasing activation functions.
- W_i are the weights applied to z_i.
- b_i are the biases added at layer i.
- $\Phi(r_e) = z_k$ is the output of the final layer of FICNN.

Dissipative Potentials:

We can define the dissipative potential function $\mathcal{R} : [-1, 1] \times \mathbb{R}^3 \mapsto \mathbb{R}^+$ such that $\mathcal{R}(r_e, 0_{3\times 1}) = 0$ as follows:

$$\begin{aligned} z_1 &= \sigma_0(W_0 r_e + b_0) \\ u_{i+1} &= \tilde{\sigma}_i(\tilde{W}_i u_i + \tilde{b}_i) \\ z_{i+1} &= \sigma_i(U_i^z(z_i \circ [W_i^{zu} u_i + b_i^z]_+) + W_i^y(y \circ (W_i^{yu} u_i + b_i^y)) + W_i^u u_i + b_i) \\ \xi(r_e, e_v) &\equiv z_k \\ \mathcal{R}(r_e, e_v) &= \sigma_{k+1}(\xi(r_e, e_v) - \xi(r_e, 0)) + \epsilon_2 \|e_v\|_2^2. \end{aligned} \tag{3.10}$$

- $y = z_0 = e_v$ is the input over which the function needs to be convex. e_v is the velocity error on $\mathcal{S}^2$ from (2.29)

- z_i is the output of layer i along the convex path.
- u_i is the output of layer i along the non-convex path.
- σ_i are convex activation functions
- $\tilde{\sigma}_i$ are the activation functions along the non-convex path. These need not be convex.
- Subscript + denotes softplus activation.
- U_i^z are the non-negative weights applied to z_i along the convex path.
- $\xi(r_e, e_v)$ is the output of the final layer of PICNN.
- The remaining quantities are the same as (3.6).

Potential Functions on $SE(3)$

Conservative Potentials:

We can define $\Psi : SE(3) \mapsto \mathbb{R}^+$ such that $\Psi(I) = 0$ as follows

$$\begin{aligned} z_1 &= \sigma_0(W_0 \bar{g}_e + b_0) \\ z_{i+1} &= \sigma_i(U_i z_i + W_i \bar{g}_e + b_i), \\ \Phi(g_e) &\equiv z_k \\ \Psi(g_e) &= \sigma_{k+1}(\Phi(g_e) - \Phi(I)) + \epsilon_1 \|I - g_e\|_F^2. \end{aligned} \tag{3.11}$$

Here

- g_e is the configuration error on $SE(3)$ defined in (2.21).
- $\bar{g}_e = \begin{bmatrix} \bar{R}_e \\ R_d^T p_e \end{bmatrix}$ (The idea is to use the non trivial information of g_e and converting it into a vector for ease of calculations).
- $\bar{R}_e$ is the 9×1 vector reshaping of R_e, i.e., $\bar{R}_e = \text{vec}(R_e)$.
- R_e is the rotation error from (2.10).
- p_e is the translation error.
- R_d is the desired rotation configuration.
- z_i is the output of layer i.
- σ_i are the convex non decreasing activation functions.
- W_i are the weights applied to z_i.

- b_i are the biases added at layer i.
- $\Phi(g_e) = z_k$ is the output of the final layer of FICNN.

Dissipative Potentials:

We can define the dissipative potential function $\mathcal{R} : SE(3)\times\mathbb{R}^6 \mapsto \mathbb{R}^+$ such that $\mathcal{R}(g_e, 0_{6\times1}) = 0$ as follows:

$$\begin{aligned}
&u_0 = \bar{g}_e \\
&u_{i+1} = \tilde{\sigma}_i(\tilde{W}_i u_i + \tilde{b}_i) \\
&z_{i+1} = \sigma_i(U_i^z(z_i \circ [W_i^{zu} u_i + b_i^z]_+) + W_i^y(y \circ (W_i^{yu} u_i + b_i^y)) + W_i^u u_i + b_i) \\
&\xi(g_e, e_V) \equiv z_k \\
&\mathcal{R}(g_e, e_V) = \sigma_{k+1}(\xi(g_e, e_V) - \xi(g_e, 0_{6\times1})) + \epsilon_2 \|e_V\|_2^2,
\end{aligned} \tag{3.12}$$

Here

- $y = z_0 = e_V$ is the input over which the function needs to be convex. e_V is the velocity error on $SE(3)$ from (3.36)
- z_i is the output of layer i along the convex path.
- u_i is the output of layer i along the non-convex path.
- σ_i are convex non decreasing activation functions
- $\tilde{\sigma}_i$ are the activation functions along the non-convex path. These need not be convex.
- U_i^z are the non-negative weights applied to z_i along the convex path.
- $\xi(g_e, e_V)$ is the output of the final layer of PICNN.
- The remaining quantities are the same as (3.6).

By making $p_e = 0$, the potential function in (3.11) reduces to the potential on SO(3). Similarly, making $p_e = 0, v_e = 0$ (v_e is the translational velocity error) and just using R_e, Ω_e in (3.12) also reduces the dissipative potential to that of $SO(3)\times\mathbb{R}^3$. It is worth noting here that the potential functions defined in (3.11) and (3.12) always produce a stabilizing elastic force and damping force pair (likewise the potentials in (3.9) and (3.10)) irrespective of the initialization of the weights as long as U matrices are non-negative. This constraint on U can be enforced by first initializing them randomly and then using the softplus activation function [97] to make them all positive. The subscript $+$ in dissipation potentials follows the convention in [2] and makes its corresponding arguments positive using softplus activation function.
In all of the potential functions described (both conservative and dissipative), we take the

final layer of the ICNN and shift it to make it equal to zero at the locations we desire (conservative potentials need to be zero at identity and dissipative potentials need to be zero when error velocities are zero). The shifted outputs are passed through smoothed ReLU which makes them positive semi-definite. Finally small positive definite terms are added at the end that makes the conservative potential functions positive definite and dissipative potential functions positive semi definite. These positive definite terms are typically already existing and well known potential functions.

3.4 Kinematic Control

In this section, a detailed description of how to obtain the elastic forces from the potential functions will be presented for $\mathcal{S}^2$, $SO(3)$ and $SE(3)$ manifolds. Stability proofs for $\mathcal{S}^2$ and $SO(3)$ will be presented as the extension to $SE(3)$ from SO(3) is straightforward. Although we are more interested in applications to dynamics on manifolds, we will emphasize the derivations for kinematic systems as the control laws straightforwardly extend to dynamical systems by adding a damping term to the elastic force. The stability proofs will become clear once we construct Lyapunov candidates using the designed potential functions.

Unit two sphere $\mathcal{S}^2$:

The variation of the potential function defined in (3.9) will be used to obtain the elastic force as follows

$$\begin{aligned} \Psi = \Psi(r_d^T r) \implies \Psi_{\epsilon\eta} &= \Psi(r_d^T e^{\epsilon\hat{\eta}} r) \\ \implies \delta\Psi_\eta = \tfrac{d}{d\epsilon}\Psi_{\epsilon\eta}\big|_{\epsilon=0} &= \Psi'(r_d^T r)(r_d^T e^{\epsilon\hat{\eta}}\hat{\eta} r)\big|_{\epsilon=0} \\ &= \Psi'(r_d^T r)(r_d^T \hat{\eta} r) \end{aligned} \tag{3.13}$$

where $\Psi' \triangleq \frac{\partial}{\partial z}\Psi(z)\big|_{z=r_d^T r}$. Utilizing the cyclic property of the box product $[a\ b\ c] = a\cdot(b\times c) = a^T\hat{b}c$ that $[a\ b\ c] = [b\ c\ a] = [c\ a\ b]$, we can write

$$\delta\Psi = \Psi'(r_d^T r)(\hat{r}r_d)^T\eta = \Psi'(r_d^T r)(\hat{r}r_d)\cdot\eta = D_r\Psi\cdot\eta. \tag{3.14}$$

The corresponding elastic force is now given by

$$f_c = -D_r\Psi = -\Psi'(r_d^T r)(\hat{r}r_d). \tag{3.15}$$

Since Ψ has tunable parameters (weights and biases), its partials or gradients can be found using automatic differentiation packages like PyTorch's *autograd*.

Theorem 1. *The kinematic tracking control law $\omega = \omega_d + f_c$, where the elastic force f_c is given by $f_c = -\Psi'(r_d^T r)(\hat{r}r_d)$, almost globally asymptotically tracks $r_d(t)$ for the system (2.25).*

Proof. We will consider the potential function Ψ as the Lyapunov function candidate as it is positive definite, and the only point where it is 0 is when r_e is 1. The time derivative of the Lyapunov function yields

$$\begin{aligned}\dot{\Psi} &= \Psi'(r_d^T r)(\dot{r}_d^T r + r_d^T \dot{r}) = \Psi'(r_d^T r)(r^T \hat{\omega}_d r_d + r_d^T \hat{\omega} r) \\ &= \Psi'(r_d^T r) r_d^T (\hat{\omega} - \hat{\omega}_d) r \\ &= \Psi'(r_d^T r) r_d^T (\hat{\omega}_d - \hat{\omega}_d - \Psi'(r_d^T r)\widehat{(r \times r_d)}) r.\end{aligned}$$

Utilizing the property that $a \times (b \times c) = b(a \cdot c) - c(a \cdot b)$

$$\begin{aligned}\dot{\Psi} &= -\Psi'(r_d^T r)^2 r_d^T (r_d \|r\|_2^2 - r(r_d \cdot r)) \\ &= -\Psi'(r_d^T r)^2 (\|r_d\|_2^2 \|r\|_2^2 - (r_d \cdot r)^2) \leq 0.\end{aligned} \tag{3.16}$$

□

Special Orthogonal Group SO(3):

A potential on $SO(3)$ can be represented as $\Psi(R_e)$ by letting $p_e = 0$ in (3.11). The variation of this potential function defined will be used to obtain the elastic force as follows

$$\begin{aligned}\Psi = \Psi(R_d^T R) \implies \Psi_{\epsilon\eta} &= \Psi(R_d^T R e^{\epsilon\hat{\eta}}) \\ \implies \delta\Psi_\eta = \tfrac{d}{d\epsilon}\Psi_{\epsilon\eta}\big|_{\epsilon=0} &= tr[\Psi'(R_d^T R)^T R_d^T R e^{\epsilon\hat{\eta}}\hat{\eta}] \\ &= tr[\Psi'(R_d^T R)^T R_d^T R \hat{\eta}],\end{aligned} \tag{3.17}$$

where $\Psi' = \frac{\partial}{\partial M}\Psi(M)\big|_{M=R_e}$ ($\Psi'_{ij} = \frac{\partial}{\partial M_{ij}}\Psi(M)\Big|_{M=R_e}$ where $[\cdot]_{ij}$ denotes the element in the i^{th} row and j^{th} column of the matrix $[\cdot]$). Utilizing the property of trace from lemma 4, we can write

$$\begin{aligned}\delta\Psi_\eta &= (R_e^T \Psi'(R_e) - \Psi'(R_e)^T R_e)^\vee \cdot \eta \\ \implies D_R\Psi = D_{R_e}\Psi &= (R_e^T \Psi'(R_e) - \Psi'(R_e)^T R_e)^\vee.\end{aligned} \tag{3.18}$$

The corresponding elastic force is now given by

$$f_c = -D_R\Psi = -(R_e^T \Psi'(R_e) - \Psi'(R_e)^T R_e)^\vee. \tag{3.19}$$

Theorem 2. *The kinematic tracking control law $\Omega = R_e^T \Omega_d + f_c$, where the elastic force f_c is given by (3.19), almost globally asymptotically tracks $R_d(t)$ for the system (2.7).*

Proof. We will consider the potential function Ψ as the Lyapunov function candidate as it is positive definite and the only point where it is 0 is when R_e is I. The time derivative of the Lyapunov function yields

$$\begin{aligned}\dot{\Psi} &= tr[\Psi'(R_e)^T \dot{R}_e] = tr[\Psi'(R_e)^T R_e \hat{\Omega}_e] \\ &= [(\Psi'^T R_e)^T - (\Psi'^T R_e)]^\vee \cdot \Omega_e.\end{aligned}$$

Utilizing the angular velocity error from (2.12), we get

$$\dot{\Psi} = -\|(R_e^T\Psi'(R_e) - \Psi'(R_e)^T R_e)^\vee\|_2^2 \leq 0. \tag{3.20}$$

□

Special Euclidean Group SE(3):

The variation of the potential function on $SE(3)$ defined in (3.11) can be used to obtain the elastic force.

$$\begin{aligned} \Psi &= \Psi(R_d^T R, R_d^T p_e) \\ \implies \Psi_{\epsilon\eta} &= \Psi(R_d^T R e^{\epsilon\eta_2}, R_d^T(p + \epsilon R\eta_1 - p_d)) \\ \implies \delta\Psi_\eta &= \tfrac{d}{d\epsilon}\Psi_{\epsilon\hat{\eta}}\big|_{\epsilon=0} = tr[\partial_1\Psi^T R_d^T R\hat{\eta}_2] + \partial_2\Psi^T R_d^T R\eta_1, \end{aligned}$$

where $\partial_1\Psi = \frac{\partial}{\partial M}\Psi(M,z)\big|_{M=R_e, z=R_d^T p_e}$ and $\partial_2\Psi = \frac{\partial}{\partial z}\Psi(M,z)\big|_{M=R_e, z=R_d^T p_e}$. This yields the elastic force as

$$f_c = \begin{bmatrix} -R_e^T\partial_2\Psi \\ -(R_e^T\partial_1\Psi - \partial_1\Psi^T R_e)^\vee \end{bmatrix}. \tag{3.21}$$

Theorem 3. *The control law given by the following equation almost globally asymptotically tracks $g_d(t)$*

$$\begin{bmatrix} \Omega \\ v \end{bmatrix} = \begin{bmatrix} R_e^T\Omega_d \\ v_d \end{bmatrix} + f_c. \tag{3.22}$$

Proof. For brevity, we skip the stability proof, as the process is similar to that of $SO(3)$ for rotations with a trivial extension to translations. □

In every case, the time derivative of the Lyapunov function is zero only at a finite number of points when $f_c = 0$, but negative everywhere else, and therefore, the equilibrium is almost globally asymptotically stable. The remaining points are unstable equilibria.

3.5 Elastic and Damping Wrenches

In this section, we define the damping forces as gradients of dissipative potentials.

Special Orthogonal Group SO(3):

A conservative potential on $SO(3)$ can be represented as $\Psi(R_e)$ by letting $p_e = 0$ in (3.11) and a dissipative potential on $SO(3) \times T_R SO(3)$ can be represented as $\mathcal{R}(R_e, \Omega_e)$ by letting $p_e = 0, v_e = 0$. The elastic (f_c) and damping (f_d) forces can be defined respectively as

$$\begin{aligned} f_c &= -D_{R_e}\Psi = -(R_e^T\Psi'(R_e) - \Psi'(R_e)^T R_e)^\vee, \\ f_d &= -\partial_{\Omega_e}\mathcal{R}(R_e, \Omega_e), \end{aligned} \tag{3.23}$$

where $\Psi' = \frac{\partial}{\partial M}\Psi(M)\big|_{M=R_e}$ and $\partial_{\Omega_e}\mathcal{R}(R_e,\Omega_e) = \frac{\partial\mathcal{R}(R_e,\Omega_e)}{\partial\Omega_e}$.

Special Euclidean Group SE(3):

The variation of the conservative potential function on $SE(3)$ defined in (3.11) and the gradient of dissipative potential function defined in (3.12) generate the following elastic and damping forces as follows

$$\begin{aligned} f_c &= \begin{bmatrix} -R_e^T \partial_2 \Psi \\ -(R_e^T \partial_1 \Psi - \partial_1 \Psi^T R_e)^\vee \end{bmatrix}, \\ f_d &= -\partial_{e_V} \mathcal{R}, \end{aligned} \tag{3.24}$$

where $\partial_1 \Psi = \frac{\partial}{\partial M} \Psi(M, z)\big|_{M=R_e, z=R_d^T p_e}$, $\partial_2 \Psi = \frac{\partial}{\partial z} \Psi(M, z)\big|_{M=R_e, z=R_d^T p_e}$ and $\partial_{e_V} \mathcal{R} = \frac{\partial \mathcal{R}(g_e, e_V)}{\partial e_V}$.

Theorem 4. *The dissipative potential functions defined in* (3.12) *satisfy* $\Omega_e^T \partial_{\Omega_e} \mathcal{R}(R_e, \Omega_e) \geq 0$ *and* $e_V^T \partial_{e_V} \mathcal{R}(g_e, e_V) \geq 0$.

Proof. For a smooth convex function $h(\cdot)$, we have the following first-order necessary and sufficient condition in terms of its gradient $\forall\, x, y \in \mathrm{dom}(h)$

$$h(y) \geq h(x) + \partial_x h(x)^T (y - x) \tag{3.25}$$

By interchanging x and y, we get

$$\begin{aligned} & h(x) \geq h(y) + \partial_y h(y)^T (x - y) \\ \Longrightarrow\ & \partial_x h(x)^T (y - x) \leq h(y) - h(x) \leq \partial_y h(y)^T (y - x) \\ \Longrightarrow\ & (\partial_y h(y) - \partial_y h(x))^T (y - x) \geq 0 \end{aligned} \tag{3.26}$$

Since $\mathcal{R}(\cdot, \cdot)$ is convex in its second argument, we have

$$(\partial_y \mathcal{R}(\cdot, y) - \partial_x \mathcal{R}(\cdot, x))^T (y - x) \geq 0, \tag{3.27}$$

By letting $x = 0$, we get

$$\partial_y \mathcal{R}(\cdot, y)^T y = y^T \partial_y \mathcal{R}(\cdot, y) \geq 0. \tag{3.28}$$

This is equivalent to

$$f_d(\cdot, y)^T y \leq 0. \tag{3.29}$$

By letting $y = e_v, \Omega_e$ or e_V, the theorem can be proved for all manifolds. □

3.6 Dynamic Control

In this section, we consider two interesting and practical problems on SO(3) and SE(3) manifolds namely orientation control of a satellite and Impedance control of a robotic manipulator respectively. The dynamics of both of the systems, and their respective stabilizing control laws using the constructed potential functions will be presented.

Control of a Satellite on SO(3):

A simple model of a rigid body rotating without translating can be used to describe the orientation control problem of a satellite. The orientation of the satellite is described through rotation matrices $R \in SO(3)$. The control is achieved through momentum wheels attached to three perpendicular axes of the satellite. Again for simplicity, we will ignore the dynamics of the reaction wheels and assume the availability of three independent torque components along its three perpendicular axes. The dynamics can be written as follows with $\mathbb{J} \in \mathbb{S}^{++} \subset \mathbb{R}^{3\times3}$ as the symmetric positive definite inertia matrix, $\Omega \in \mathbb{R}^3$ as the angular velocity represented in the body-fixed frame and $\tau \in \mathbb{R}^3$ as the torque.

$$\begin{aligned} \dot{R} &= R\hat{\Omega} \\ \mathbb{J}\dot{\Omega} + \hat{\Omega}\mathbb{J}\Omega &= \tau \end{aligned} \tag{3.30}$$

Theorem 5. *The following control almost globally asymptotically tracks $R_d(t)$ for a dynamical system described by (3.30) with the elastic force f_s and damping force f_d described by (3.23)*

$$\tau = \hat{\Omega}\mathbb{J}\Omega - \mathbb{J}\hat{\Omega}_e R_e^T \Omega_d + \mathbb{J}R_e^T\dot{\Omega}_d + \mathbb{J}(f_c + f_d). \tag{3.31}$$

Proof. This control law achieves the following autonomous error dynamics

$$\begin{aligned} \dot{\Omega}_e + \partial_{\Omega_e}\mathcal{R}(R_e, \Omega_e) + D_{R_e}\Psi(R_e) &= 0 \\ \text{or } \dot{\Omega}_e - f_d(R_e, \Omega_e) - f_c(R_e) &= 0 \end{aligned} \tag{3.32}$$

We will consider the following positive definite Lyapunov candidate function

$$\begin{aligned} W &= \Psi(R_e) + \tfrac{1}{2}\Omega_e^T\Omega_e \\ \implies \dot{W} &= \dot{\Psi}(R_e) + \Omega_e^T\dot{\Omega}_e = \Omega_e^T(\dot{\Omega}_e - f_c(R_e)) \\ &= \Omega_e^T f_d(R_e, \Omega_e) \ \leq\ 0 \text{ from (3.29)}. \end{aligned}$$

□

Using Lasalle's Invariance principle, we can also conclude that the equilibrium $R_e = I$ of the error dynamics in (3.32) is almost globally asymptotically stable as the largest invariant set where $\dot{W} = 0$ only when $R_e = I$ (removing the other unstable equilibria). It may firstly be a bit non-intuitive, but it should be noted that Ω_e can be expressed in terms of R_e and $\dot{R}_e$ which makes the entire error equation a function of just R_e and its time derivatives. The expression is omitted for compactness.

Control of a Robotic Manipulator on SE(3):

Another problem where potential functions play an important role is in the control of robotic manipulators. We will demonstrate an application to impedance control of a robotic manipulator [77]. The manipulator equations in joint space can be written as follows with $q \in \mathbb{R}^n$

as the vector of generalized coordinates of the manipulator.

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau + J_b(q)^T T_e, \tag{3.33}$$

where $M(q) \in \mathbb{R}^{n\times n}$ is the symmetric positive definite inertia matrix, $C(q,\dot{q}) \in \mathbb{R}^{n\times n}$ is a Coriolis matrix, $G(q) \in \mathbb{R}^n$ represents the gravitational terms, $\tau \in \mathbb{R}^n$ is the joint torque, and $T_e \in \mathbb{R}^6$ is an external wrench at the end-effector from contacts, human inputs etc. The Coriolis matrix satisfies the property that $\dot{M} - 2C$ is skew-symmetric.

In the field of impedance control combined with operational space formulation, it is well known from [32] that the robot dynamics (3.33) can be rewritten as

$$\tilde{M}\dot{V}^b + \tilde{C}V^b + \tilde{G} = \tilde{\tau} + \tilde{\tau}_e, \tag{3.34}$$

where the definitions of $\tilde{M}$, $\tilde{C}$, $\tilde{G}$, $\tilde{\tau}$, and $\tilde{\tau}_e$ can be written as (from [77]).

$$\begin{aligned}
\tilde{M}(q) &= J_b(q)^{-T} M(q) J_b(q)^{-1}, \\
\tilde{C}(q,\dot{q}) &= J_b(q)^{-T}(C(q,\dot{q}) - M(q)J_b(q)^{-1}\dot{J})J_b(q)^{-1}, \\
\tilde{G}(q) &= J_b(q)^{-T} G(q),\ \tilde{\tau} = J_b(q)^{-T}\tau,\ \tilde{\tau}_e = J_b(q)^{-T} T_e.
\end{aligned}$$

where J_b is a body-frame Jacobian matrix which relates body velocity and joint velocity by $V^b = J_b\dot{q}$.

Theorem 6. *The following control law almost globally asymptotically tracks $g_d(t)$ for a dynamical system described by (3.34) with the elastic and damping forces described by (3.24) when $T_e = 0$*

$$\tilde{\tau} = \tilde{M}\dot{V}_d^* + \tilde{C}V_d^* + \tilde{G} + \tilde{M}(f_d + f_c). \tag{3.35}$$

Proof. When $T_e = 0$, this control law achieves the following error dynamics

$$\begin{aligned}
\dot{e}_V + \partial_{e_V}\mathcal{R}(g_e, e_V) + D_{g_e}\Psi(g_e) = 0 \\
\text{or } \dot{e}_V - f_d(g_e, e_V) - f_c(g_e) = 0
\end{aligned} \tag{3.36}$$

We will consider the following positive definite Lyapunov candidate function

$$\begin{aligned}
W &= \Psi + \tfrac{1}{2} e_V^T e_V \\
\implies \dot{W} &= \dot{\Psi} + e_V^T \dot{e}_V = e_V^T(\dot{e}_V - f_c(g_e)) \\
&= e_V^T f_d(g_e, e_V) \le 0 \text{ from (3.29)}.
\end{aligned} \tag{3.37}$$

Using Lassalle's, we can conclude again that the desired equilibrium is almost globally asymptotically stable. □

The elastic and dissipative forces in (3.31) and (3.35) are invariant to left translations i.e., when both the scene and the end effector's pose are left-translated by the same, the elastic force doesn't change as the elastic force is only a function of error configuration and from the way it was defined in (2.10) and (2.21), error configuration was shown to be left invariant in (2.31). It can also be seen that the the error dynamics in (3.32) and (3.36) only depend on the error configuration and the error velocities and have no dependence on the physical properties of the robot. Therefore, control laws designed for robots with the same kinematic structure can be transferred from one robot to the other without the hassle of tuning the potentials afresh. This becomes extremely handy when mimicking expert demonstrations as smaller robots can be trained with human demonstrations and can be seamlessly transferred to larger robots. A more detailed discussion about invariance can be found in [76].

3.7 Training the Neural Network

Once the structure of the potential function is finalized (by fixing the number of layers and their sizes), an objective according to the needs of the user can be specified which can be posed as an optimization problem of minimizing a loss function by gradient descent. For improving the performance, we can consider an LQR-style problem where we have a running cost along the trajectory that needs to be minimized. A sample loss function for an error trajectory is shown in (3.38) with a positive weight λ. We could also add weighting matrices like in the LQR problem instead of the scalar λ.

$$L_1 = \int_0^T (\|\mathcal{I}_3 - R_e(t)\|_F^2 + \lambda \|\Omega_e(t)\|_2^2) dt \tag{3.38}$$

The procedure to shape the potential function to minimize this loss is shown in Fig. 3.3. We first start by forming a set of initial conditions around which the system is expected to start. The set of parameters defining the neural network potential functions will be denoted by $\theta = [\theta_c,\ \theta_d]$ with θ_c denoting the parameters of the conservative potential function and θ_d denoting the parameters of the dissipative potential function. Since any potential with random initialization (of course with some non-negative weights which can be taken care of by softplus function in PyTorch) becomes a valid potential function, we can obtain the corresponding stabilizing elastic and damping forces and integrate the system forward to obtain the error trajectories for each of the initial conditions for a user-defined fixed time T without the fear of trajectories blowing up. Here we have shown it for SO(3) error dynamics in (3.32), but any of the error dynamics can be used here depending on the system and situation of interest. A mean loss is computed by taking the average of individual losses corresponding to the trajectory for each initial condition. Next, standard back-propagation algorithms with the choice of gradient descent, such as Stochastic Gradient Descent (SGD), RMSprop, and ADAM, can be used to update the parameters of the potential functions. The function $\alpha(\cdot)$ is used to represent the choice of our optimizer. These updated potential

functions generate the updated elastic and damping forces and the system is integrated forward for all the initial conditions again. This cycle is repeated till convergence or any other user-specified termination criterion. Note that since the dynamics evolve on manifolds, a variational integrator like [34, 33, 65] could be a better choice to integrate the system forward as they preserve the geometry of the manifold. In Fig.3.3, the dynamical equations

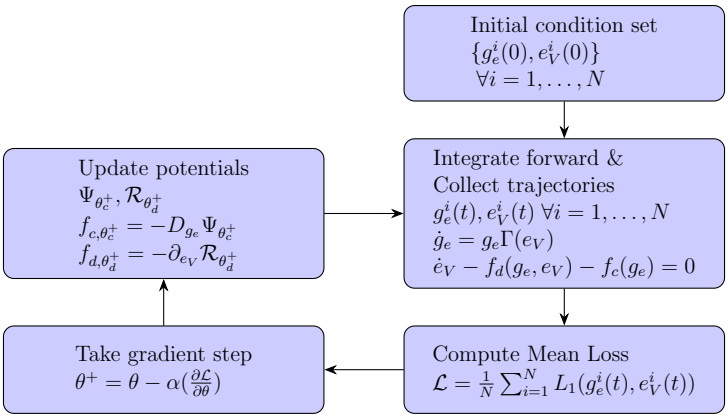

Figure 3.3: Flow chart showing the training process for potential functions for a satellite orientation control problem.

in the integration block can be replaced by (3.36) and the loss function can be updated appropriately to account for SE(3) trajectories.
For mimicking expert potential fields, we can sample the combined error configuration and error velocity space to form a set of N points and minimize the distance between the expert potential field and the neural network potential field at these points through the loss function (3.39). The loss can be minimized by updating the parameters of the potential functions using a gradient descent technique similar to the previous method.

$$\begin{aligned}\mathcal{L}_2(\theta) &= \sum_{i=1}^{N} \|\dot{\Omega}_{e,des}^i - \dot{\Omega}_e^i\|_2^2 \\ &= \sum_{i=1}^{N} \|\dot{\Omega}_{e,des}^i - f_{d,\theta_d}(R_e^i, \Omega_e^i) - f_{c,\theta_c}(R_e^i)\|_2^2\end{aligned} \tag{3.39}$$

Here $\dot{\Omega}_{e,des}^i$ is the i^{th} sample of the desired potential field from the expert and $\dot{\Omega}_e^i = f_{d,\theta_d}(R_e^i, \Omega_e^i) + f_{c,\theta_c}(R_e^i)$ is the i^{th} sample of the potential field from the neural network and the error dynamics in (3.32) at samples R_e^i and Ω_e^i.

3.8 Simulation Results

Implementation on Satellite with linear damping

We consider the problem of regulating the rotation error R_e to $\mathcal{I}_3$ for dynamics (3.32). A neural network with 3 hidden layers of 14 neurons each, using Kaiming initialization within bounds $(-\sqrt{3}, \sqrt{3})$, was trained on 50 initial conditions. Initial conditions were set with rotation errors normally distributed around π radians from the identity matrix according to (3.40) and angular velocity errors with zero mean and unit variance. The training utilized the Adam optimizer at a learning rate of 0.01 to minimize the loss function in (3.38). Results in Fig.3.4 illustrate the evolution of rotation error trajectories for neural network, logarithmic, and Frobenius norm potential functions. Despite being designed for large initial errors around π radians, performance for more minor errors is comparable across potential functions, as shown in Fig.3.4(c).

Implementation on a Manipulator

In this section, we consider the problem of regulating the error g_e to I of a rigid body evolving on SE(3) according to the error dynamics (3.36). A neural network with 3 hidden layers of 7 neurons each, using Kaiming initialization within bounds $(-\sqrt{3}, \sqrt{3})$, was trained on 50 initial conditions. Initial conditions were set with rotation errors normally distributed around π radians from the identity matrix, position errors normally distributed around $[1, 1, 1]$ with unit variance and velocity errors with zero mean and unit variance. The training utilized the Adam optimizer at a learning rate of 0.01 to minimize the loss function in (3.38) with $\lambda = 1$. Results in Fig.3.5 illustrates the evolution of configuration error trajectories for neural network potential functions for a random initialization without any training. Fig. 3.6 shows improved trajectories after the training process. We can see that the trajectories remain stable despite random initializations even in this case. In all the plots, we will use $\Psi_F(g_e) = \frac{1}{2}\|I - g_e\|_F^2$ to evaluate the configuration error.

Fig. 3.7 shows the trajectories with the trained potential functions but with large initial errors. The benchmark potential functions used are $\Psi_b = \frac{1}{2}\|I - g_e\|_F^2$ and $\mathcal{R}_b = \frac{1}{2}\|e_V\|_2^2$.

Implementation on Mujoco environment

We have implemented our proposed approach to a UR5e robotic manipulator in the Mujoco environment (see Fig. 3.8) to show its feasibility with the same parameters as that of the satellite but with a different first layer to account for input dimensions of SE(3).

To show the advantages of the proposed approach, we will compare the regulation performance under large initial error conditions for the proposed approach and the benchmark approach, the geometric impedance control [77] with Frobenius norm-based potential func-

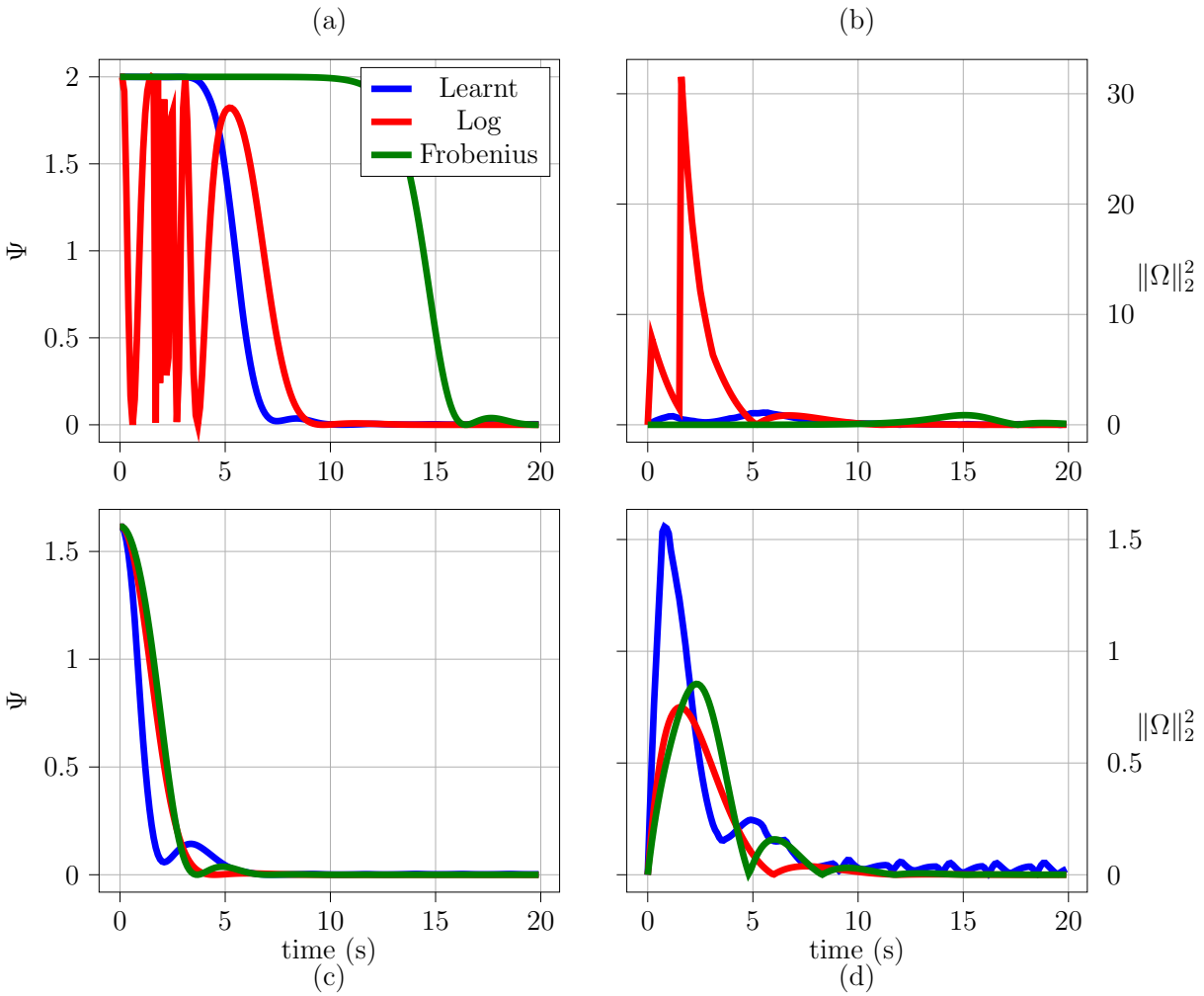

Figure 3.4: Comparison of responses with various potential functions (a) R_e trajectories for 0.99π radians initial error(b) and corresponding angular velocities errors. (c) R_e trajectories for 0.1π radians initial error (d) and corresponding angular velocity errors.

tion. The goal pose $g_d = (R_d, p_d)$ of the end-effector is given as

$$p_d = \begin{bmatrix} 0.7 & 0.0 & 0.4 \end{bmatrix}^T, \quad R_d = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

The initial condition $g_i = (R_i, p_i)$ is given as

$$p_i = p_d, \quad R_i = \text{Rot}(w, \theta) R_d, \tag{3.40}$$

where $w = [-1, 1, 1]^T$, $\theta = \pi - \varepsilon$ with $0 < \varepsilon \ll 1$, and $\text{Rot}(w, \theta)$ is an axis-angle representation of the rotation matrix, e.g., one can use Rodrigues formula to convert it into the rotation

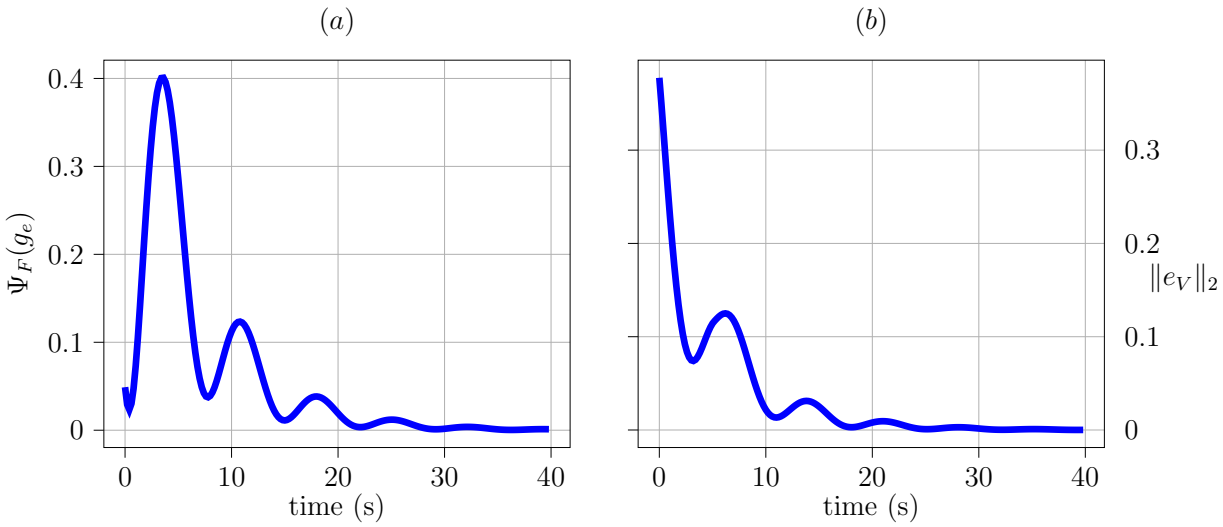

Figure 3.5: Randomly Initialized Network: (a) Configuration error trajectory and (b) Corresponding velocity error trajectory for a rigid body with initial rotation error $R_e(0) = 0.1\pi$ rotated along the z-axis, $p_e = [1, 1, 1]^T$ and zero initial velocity error.

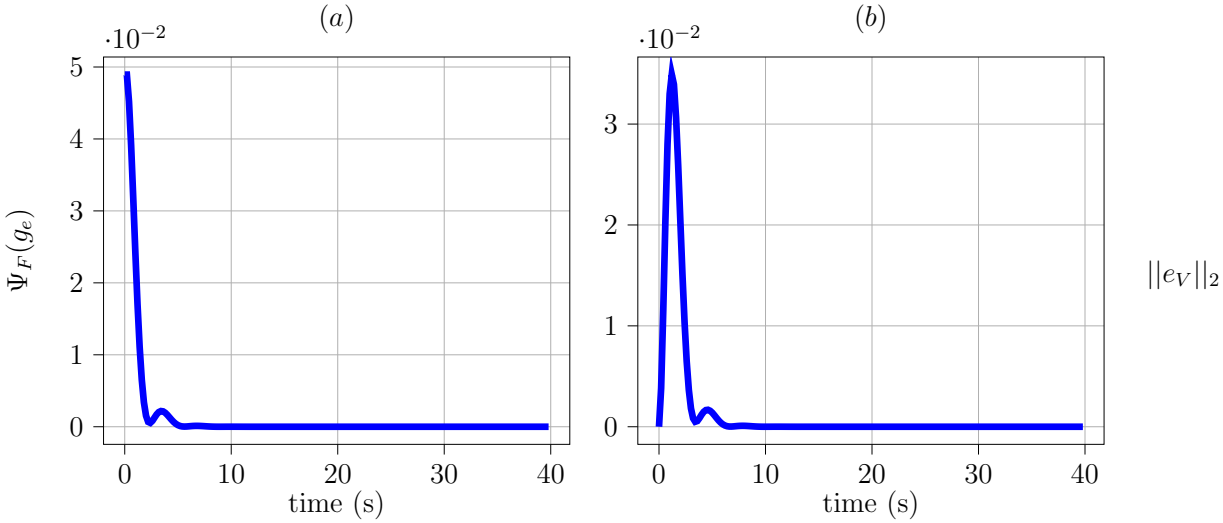

Figure 3.6: Trained Network: (a) Configuration error trajectory and (b) Corresponding velocity error trajectory for a rigid body with initial rotation error $R_e(0) = 0.1\pi$ rotated along the z-axis, $p_e = [1, 1, 1]^T$ and zero initial velocity error.

matrix. The results of this large initial rotational angle scenario are shown in Fig. 3.9. The proposed approach showed faster convergence compared to the Frobenius-norm-based ones but showed more spikes in the norm of velocity error because of faster convergence.

One caveat for the proposed approach in application to the robotic manipulator system is that the closed-loop system may not be exactly feedback linearized to (3.36) via control

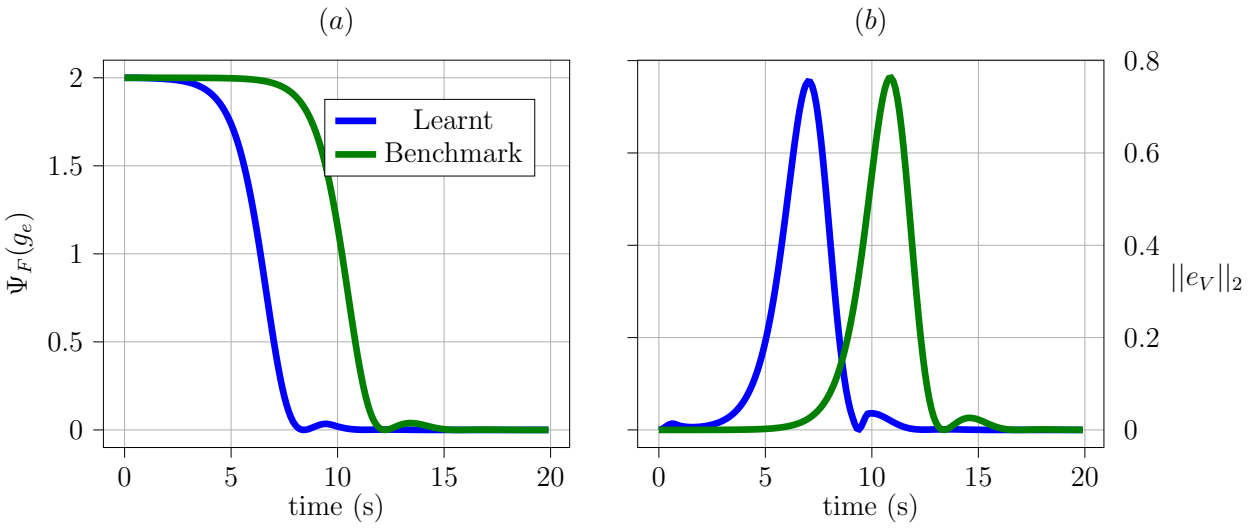

Figure 3.7: Trained Network: (a) Configuration error trajectory and (b) Corresponding velocity error trajectory for a rigid body with initial rotation error $R_e(0) = 0.999\pi$ rotated along the z-axis, $p_e = [1, 1, 1]^T$ and zero initial velocity error.

law (3.35) due to inherent uncertainties. For example, one of the assumptions from original geometric impedance control from [77] is that the Jacobian matrix $J_b(q)$ needs to be non-singular. However, the near singular point can be encountered as the proposed approach is executed in the robotic manipulator via a feedback linearization scheme. This singularity and non-precise feedback linearization is the main reason for performance differences between single rigid-body systems, like satellites, and more complex robotic manipulator systems.

3.9 Summary

In this chapter, a generic design methodology for designing trainable conservative and dissipative potential functions for fully actuated dynamical systems evolving on manifolds has been presented. The corresponding elastic and damping wrenches/forces obtained from the potential functions were shown to be stabilizing irrespective of the initialization of the network. These wrenches/forces were used to formulate the loss functions which can be minimized using gradient descent algorithms to achieve user's requirements. Analysis of the potential functions and stability for various manifolds of interest has also been presented. Finally, the methodology was demonstrated on two problems - a satellite orientation control and a robotic manipulator impedance control.

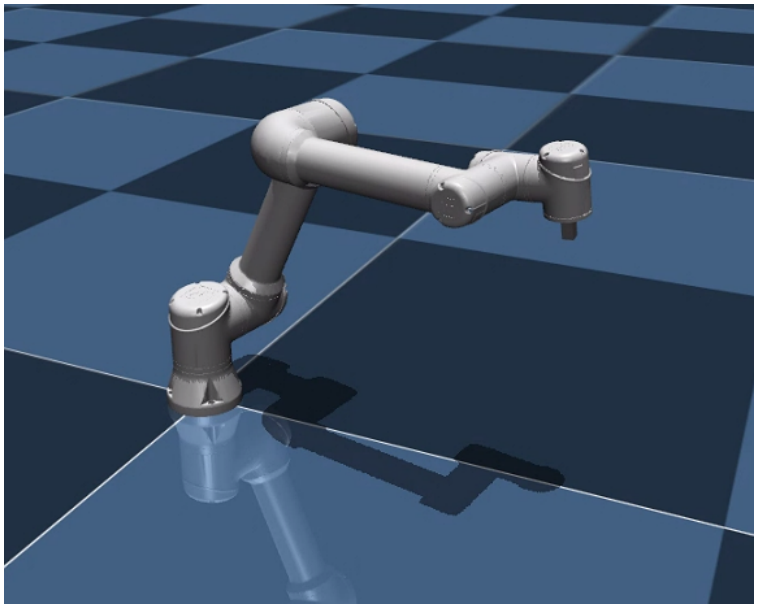

Figure 3.8: UR5e robot manipulator implemented in Mujoco environment.

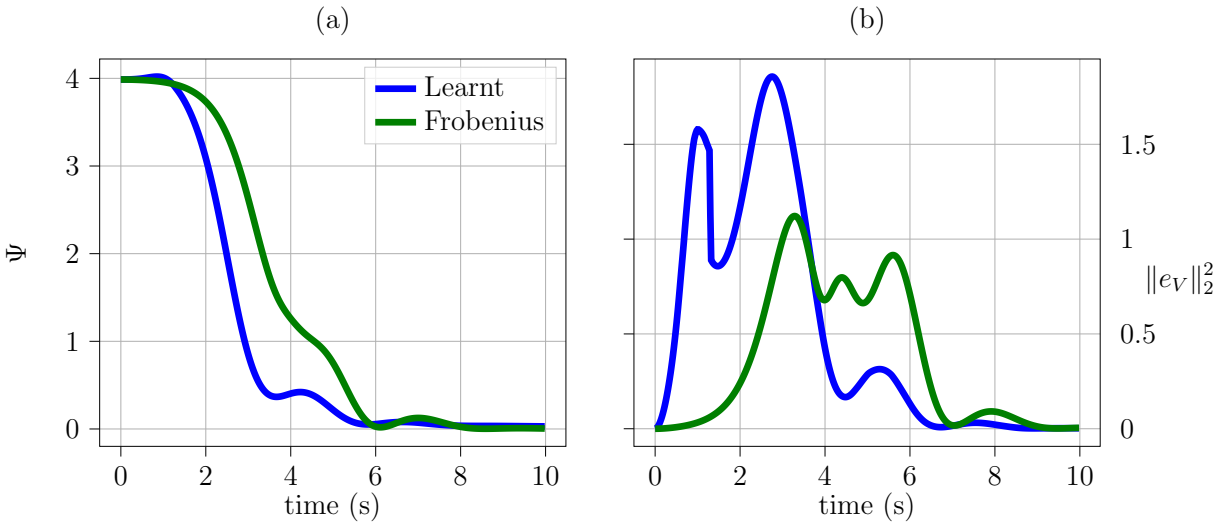

Figure 3.9: (a) Potential function Ψ with respect to time, (b) 2-norm of velocity errors, for the learning-based controller and the Frobenius-norm-based controllers are presented.

Part II

Identification and Control of Hard Disk Drives

Chapter 4

Clustering Techniques for Stable Linear Dynamical Systems with applications to Hard Disk Drives

4.1 Introduction

Transfer function clustering is a valuable technique in control systems engineering because it enables us to identify groups of transfer functions that share similar characteristics. This can help us gain a better understanding of a system's behavior and design appropriate control strategies. Some sub-fields of dynamical systems where clustering transfer functions can be useful are as follows:

1. System identification: Clustering transfer functions can assist in identifying different modes of a system, which can facilitate model identification and system identification. By clustering transfer functions with similar behaviors, distinct system modes and their associated dynamics can be identified.

2. Control design: Clustering transfer functions can aid in designing control strategies tailored to specific modes of a system. For instance, in a system with multiple modes, a control strategy might be designed to switch between different controllers depending on the current mode.

Overall, clustering transfer functions can provide valuable insights into a system's behavior and help design effective control strategies and diagnose faults. In this chapter, the focus will be on clustering stable linear dynamical systems common for HDD plants. Once the clusters have been finalized, controllers can be designed for each of the plants using existing control design techniques so that the controllers are both optimal and robust within each cluster. Although the focus will be on stable LTI systems, some preliminary ideas on extending this technique to unstable LTI systems will also be presented.

Recently, in most personal computers, Solid State Drives (SSD) have replaced HDDs due to their higher data transfer rate. Despite this, HDDs continue to dominate the market in data centers due to their lower cost and higher reliability compared to SSDs. The cost of memory per square inch of an HDD can be reduced by decreasing the spacing between successive tracks on which the data is stored. One factor dictating the spacing between tracks in an HDD is the sensitivity of the controller to disturbances. To improve disturbance rejection characteristics, additional actuators are added. The Voice Coil Motor (VCM) actuator compensates for low-frequency disturbances and has higher strokes, while piezoelectric micro actuators (PZT) compensate for high-frequency disturbances and have lower strokes, typically around one to three tracks. A dual-stage actuator (DSA) system has a VCM and a PZT actuator, while a triple-stage actuator (TSA) system has a VCM and two PZTs.

Recently, data-driven feedback control design approaches from frequency response measurements ([19, 28, 29]) for multi-stage systems have been developed in [3, 4, 60, 63] to suppress disturbances. These data-driven controllers are robust to variations in the plant models. Although robustness is ensured when a solution to the optimization problem exists, a common controller might not be optimal for each individual HDD due to variations in the plants and disturbances. To address these issues, data-driven feedforward controllers have been developed in [79, 78, 80] based on the frequency response measurement of disturbance processes on top of the robust feedforward controllers. Another add-on adaptive feedforward controller has been developed in [11] on top of the robust feedback controller to account for plant variations. Robust controllers for Linear Time Invariant (LTI) plants with parametric uncertainties with applications to HDDs have also been developed in [13, 12], utilizing plant models instead of a data-driven approach. Performance can be improved while still keeping the controllers robust to plant variations by clustering the plants into different clusters and designing a controller for each cluster separately so that the controller is optimal within the cluster. Since every cluster contains other plants, robustness is also imposed. Once the controllers are designed, controllers for each system coming out of the manufacturing plant can be selected using time responses/frequency responses of the system to determine which cluster the system belongs to.

The flow of this chapter is organized as follows: In section 4.2, we present some of the distance metrics used for clustering in the literature and list their shortcomings. In section 5.2, we present some of the preliminaries for $\mathcal{H}$ norms of systems and how they can be used to define distances between two linear dynamical systems. In section 4.3, we present the k-medoids algorithm for hard clustering LTI systems, and in section 4.4, we present a soft clustering algorithm utilizing Gaussian Mixture Models (GMM) clustering for a class of plants common for HDDs. Finally, in section 4.5, we show the effectiveness of the algorithm in clustering HDD plants.

4.2 Previous Work

The most important part of clustering, such as k-means or k-medoids, is the concept of distance between two data points. Each data point in this chapter represents an LTI system. It could either be a transfer function or a set of frequency responses. One of the first notions of distances between two linear dynamical systems was introduced in [49], with the distance defined between two ARMA models. Let $H_1(z^{-1})$ and $H_2(z^{-1})$ be the z-transforms of two discrete time stable LTI systems of the same order defined as follow

$$H_1(z^{-1}) = \frac{b_0^1 + b_1^1 z^{-1} + b_2^1 z^{-2} + \cdots + b_m^1 z^{-m}}{1 + a_1^1 z^{-1} + a_2^1 z^{-2} + \cdots + a_n^1 z^{-n}} \tag{4.1a}$$

$$H_2(z^{-1}) = \frac{b_0^2 + b_1^2 z^{-1} + b_2^2 z^{-2} + \cdots + b_m^2 z^{-m}}{1 + a_1^2 z^{-1} + a_2^2 z^{-2} + \cdots + a_n^2 z^{-n}}, \tag{4.1b}$$

then the distance $d(H_1, H_2)$ according to [49] is defined as

$$d(H_1, H_2)^2 = \sum_{i=0}^{m} \lambda_b (b_i^1 - b_i^2)^2 + \sum_{j=1}^{n} \lambda_a (a_j^1 - a_j^2)^2, \tag{4.2}$$

where λ_a and λ_b are some positive weights.
One of the major drawbacks of this distance metric is that in certain cases, two LTI dynamical systems which produce the same output due to pole-zero cancellations might have a non-zero distance between them according to the above distance metric. A simple example that demonstrates the issue is as follows

$$\begin{aligned} H_1(z^{-1}) &= \frac{1 + z^{-1}}{1 + 0.5z^{-1}} \\ &= \frac{1 + z^{-1} + 0z^{-2}}{1 + 0.5z^{-1} + 0z^{-2}} \end{aligned} \tag{4.3a}$$

$$\begin{aligned} H_2(z^{-1}) &= \frac{1 + z^{-1}}{1 + 0.5z^{-1}} \times \frac{1 + 0.5z^{-1}}{1 + 0.5z^{-1}} \\ &= \frac{1 + 1.5z^{-1} + 0.5z^{-2}}{1 + z^{-1} + 0.25z^{-2}}. \end{aligned} \tag{4.3b}$$

Though H_1 and H_2 have the same outputs for any given input, the distance between them is non-zero.
In [67], another distance metric was proposed in terms of the system matrices. If the realizations of strictly proper stable LTI systems H_1 and H_2 are (A_1, B_1, C_1) and (A_2, B_2, C_2) respectively, then the distance $d(H_1, H_2)$ between them with positive weights λ_A, λ_B, and λ_C is defined as

$$d(H_1, H_2)^2 = ||C_1 - C_2||_F^2 + \lambda_A ||A_1 - A_2||_F^2 + \lambda_B ||B_1 - B_2||_F^2, \tag{4.4}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. Such a distance is defined only when both systems have the same order. Since multiple realizations are possible for the same system, such a

distance metric would produce non-zero distances between two different realizations of the same system.
To overcome the shortcomings of these distance metrics, we propose a distance metric in terms of the $\mathcal{H}$ norms of the system, the details of which are presented in the upcoming sections.

4.3 Hard Clustering

H_2 and H_∞ Distances

The H_2 and H_∞ norms summarized in 2 can be used to define the notion of another distance between transfer functions which can then be used to cluster transfer functions. Let us consider two stable continuous time MIMO systems G_1 and G_2 with state space realizations as follows

$$G_1 \sim \left[\begin{array}{c|c} A_1 & B_1 \\ \hline C_1 & D_1 \end{array}\right], G_2 \sim \left[\begin{array}{c|c} A_2 & B_2 \\ \hline C_2 & D_2 \end{array}\right]. \tag{4.5}$$

The difference between these two transfer functions has the following state space realization

$$G_1 - G_2 \sim \left[\begin{array}{cc|c} A_1 & 0 & B_1 \\ 0 & A_2 & B_2 \\ \hline C_1 & -C_2 & D_1 - D_2 \end{array}\right] \tag{4.6a}$$

$$= \left[\begin{array}{c|c} A_{12} & B_{12} \\ \hline C_{12} & D_{12} \end{array}\right]. \tag{4.6b}$$

The idea behind using the difference is to see how much the outputs of the systems differ under the same input.
If $D_{12} = 0$, the H_2 distance between the transfer functions G_1 and G_2 exists and can be found as follows

$$||G_1 - G_2||_2^2 = tr[C_{12} P_{12} C_{12}^T], \tag{4.7}$$

where

$$P_{12} = \int_{-\infty}^{\infty} e^{A_{12}t} B_{12} B_{12}^T e^{A_{12}^T t} dt, \tag{4.8}$$

which is the solution of the Lyapunov equation given by

$$A_{12} P_{12} + P_{12} A_{12}^T = -B_{12} B_{12}^T. \tag{4.9}$$

The H_∞ distance between the transfer functions G_1 and G_2 can be found by constructing the Hamiltonian matrix as

$$\mathcal{H}_{12} = \begin{bmatrix} A_{12} & \frac{B_{12} B_{12}^T}{\gamma^2} \\ -C_{12}^T C_{12} & -A_{12}^T \end{bmatrix}. \tag{4.10}$$

The smallest γ that keeps the Hamiltonian matrix $\mathcal{H}_{12}$ asymptotically stable is the H_∞ distance between the transfer functions G_1 and G_2. For systems whose frequency responses are available instead of the model, the approximate H_2 distance is given as

$$||G_1\text{-}G_2||_2^2 \approx \frac{1}{2\pi}\sum_{k=1}^{N} tr[(G_1\text{-}G_2)(j\omega_k))^*((G_1\text{-}G_2)(j\omega_k)]\Delta\omega_k. \tag{4.11}$$

In addition, the approximate H_∞ distance can be calculated as

$$||G_1 - G_2||_\infty = \sup_{\omega_k \in \{\omega_1,\ldots,\omega_N\}} ||G_1(j\omega_k) - G_2(j\omega_k)||_2. \tag{4.12}$$

Though these norms and distances are defined for stable LTI systems, we can extend the notion of the distance to unstable LTI systems by first designing a common controller for all the systems and using the same notions on the closed loop systems. Note that this distance would not be representative of how much the outputs differ as the outputs diverge for unstable systems in open loop. Nevertheless, when a stabilizing controller exists, we can use the distances for the closed loop systems to cluster the open loop plants. But it is possible that such a stabilizing controller might not exist for all plants that we are interested in clustering.

In this section, utilizing the distance metrics, the transfer functions/frequency responses can be clustered using a k-medoids algorithm with the chosen distance metric. We prefer a k-medoids algorithm to a k-means algorithm for two reasons:

- Robustness to outliers: k-medoids is more robust to outliers than k-means because medoids are selected as the most centrally located point in a cluster, which is less sensitive to outliers than the mean. This means that k-medoids can handle noisy data better than k-means.

- Defining a mean frequency response is straightforward for data as a simple arithmetic mean serves as the center of mass. However, for transfer functions, the mean transfer function obtained by summing all the given transfer functions and dividing by the number of data points would produce a very high-dimensional transfer function. Computing the distances from this mean to other data points might become computationally very expensive. Nonetheless, it is still theoretically possible to have such a mean.

The mean system obtained by taking the average of N systems with (A_i, B_i, C_i, D_i) representing the system matrices of the i^{th} system is

$$\frac{1}{N}\sum_{i=1}^{N} G_i \sim \left[\begin{array}{cccc|c} A_1 & 0 & \ldots & 0 & B_1 \\ 0 & A_2 & \ldots & 0 & B_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & A_N & B_N \\ \hline \frac{C_1}{N} & \frac{C_2}{N} & \ldots & \frac{C_N}{N} & \frac{1}{N}\sum_{i=1}^{N} D_i \end{array}\right]. \tag{4.13}$$

It can easily be seen that the order of the mean system scales linearly with the number of data points. Therefore, to improve robustness and, more importantly, to keep the computational effort reasonable, instead of finding the mean, we find the point in the group that minimizes the total sum of distances to the other points.

4.4 Soft Clustering

Soft clustering allows a data point to belong to multiple clusters to varying degrees, while, in contrast, hard clustering assigns a data point to only one cluster. In soft clustering, each data point is assigned a probability of belonging to each cluster, with the sum of probabilities across all clusters equaling one. These probabilities are based on the distance between the data point and the cluster centers and the degree of overlap between the clusters. This approach is useful in situations where there is ambiguity or uncertainty in the data, or when a data point may belong to multiple groups simultaneously. The advantage of soft clustering is that it can be used to design probabilistic controllers. In this section, we utilize a Gaussian Mixtures Model (GMM) for soft clustering on a feature vector that summarizes the plant. A feature vector is used as an alias for the entire plant, and a covariance matrix can be defined for the feature vectors, whereas a notion of covariance does not make sense for the frequency response data. Instead of GMMs, one could also use a technique like fuzzy C-means clustering ([41]) directly on the data points, without the intermediate step of finding the feature vector, which just requires the notion of a distance. A fuzzy C-means algorithm can be used with the $\mathcal{H}$ distances defined in section 4.2. The frequency response data for a

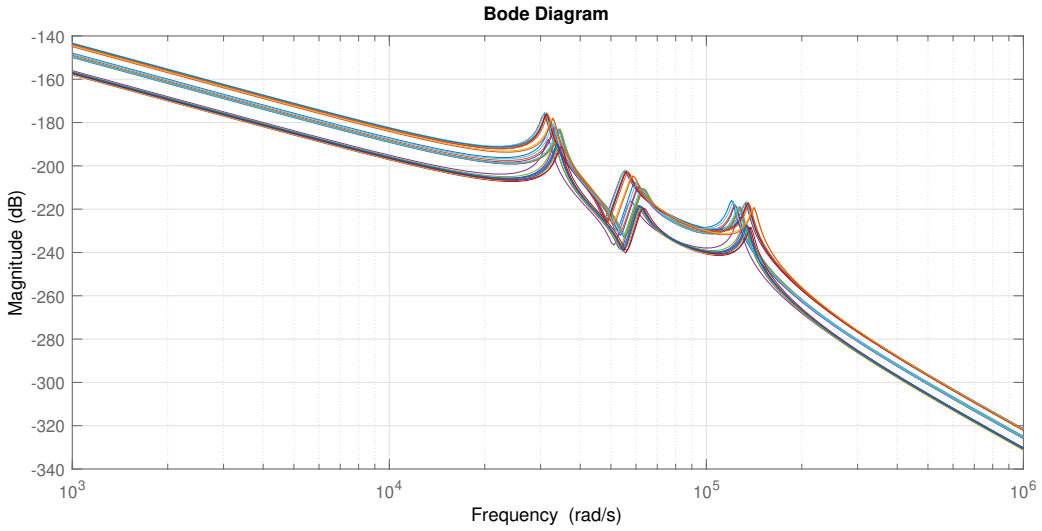

Figure 4.1: Frequency responses of 30 VCM plants

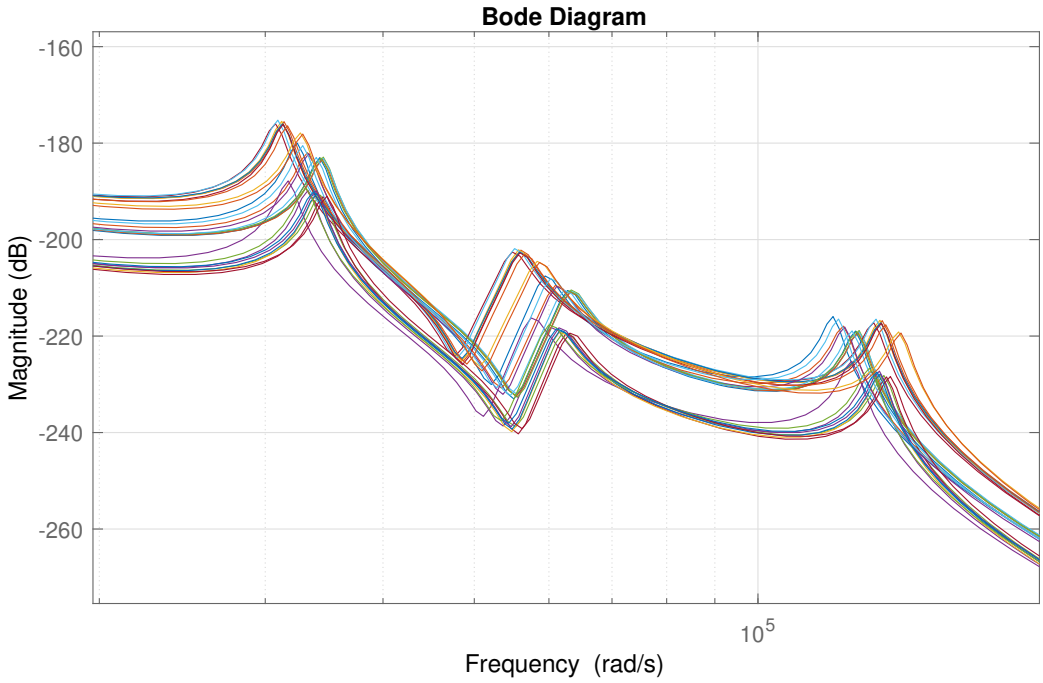

Figure 4.2: Zoomed frequency responses of the 30 VCM plants in Fig.4.1 around the resonant peaks.

batch of 30 VCM plants is depicted in Fig. 4.1. The data shows several discernible resonance modes. The data has been generated from HDD plant models representative of real plants. The various modes in increasing order of natural frequency are Butterfly mode at around 1.9 KHz, Torsion mode around 2.3 KHz, and Sway mode around 16.5 KHz. Hence, it is suitable to represent the transfer function between the input and output as a sum of modes, which can be expressed as follows:

$$G(s) = \frac{b_0}{s^2} + \sum_{k=1}^{n} \frac{b_k}{s^2 + 2\zeta_k \omega_k s + w_k^2} \tag{4.14}$$

In this section, we stick to plants of this form, but soft clustering can also be done for other stable LTI plants using the fuzzy C-means clustering algorithm. Fuzzy C-means clustering can be applied to an extended class of transfer functions where the systems' orders are different, which is not possible with GMMs as the structure restricts the systems being compared to be of the same order.

To represent the transfer function as a sum of modes, only three parameters per mode are needed: natural frequency, damping coefficient, and modal constant. The parameters for each mode in the summation form can be easily determined through a single degree

of freedom (SDOF) modal analysis. The SDOF model is based on the assumption that the dynamics of a system near a resonance frequency are mainly dominated by the corresponding resonance mode, and other modes contribute to a lesser extent. This assumption is generally valid for disk drive structures ([14]). The circular shape of the data points in the Nyquist

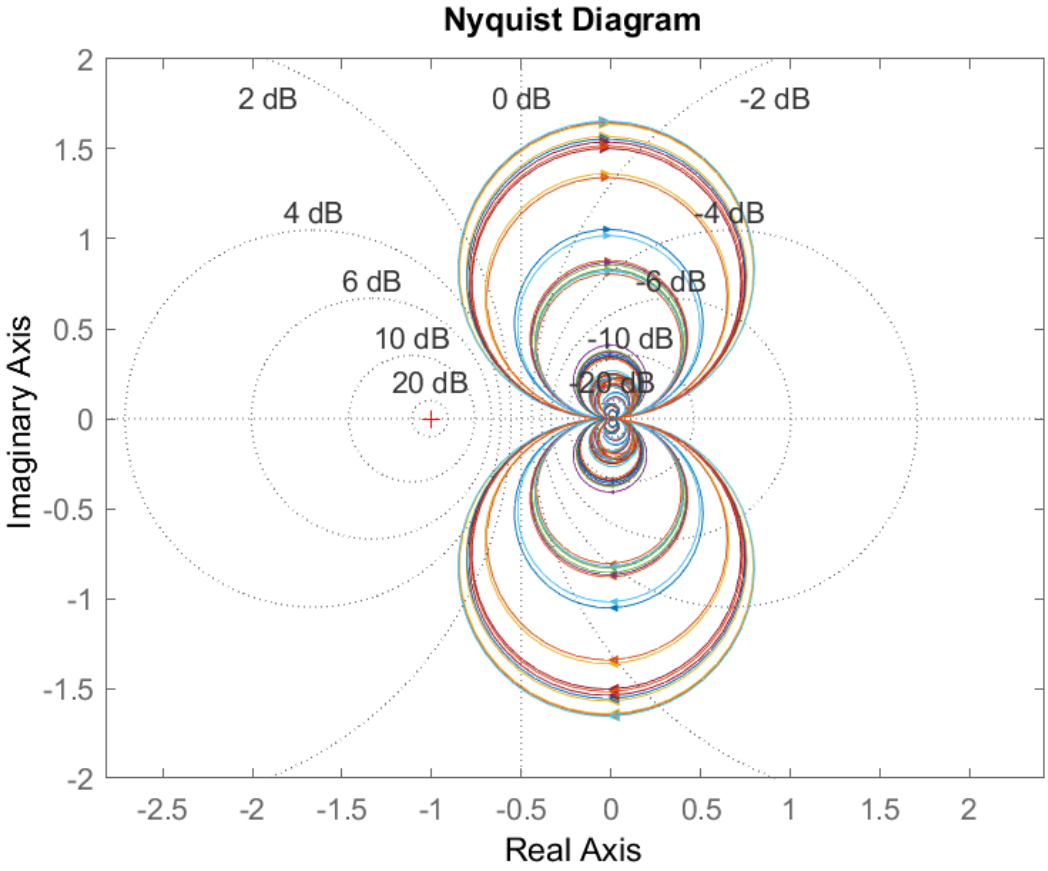

Figure 4.3: Nyquist plots of the 30 VCM plants in Fig.4.1 showing the circles in the complex plane near the resonant frequencies.

plane in Fig. 4.4 is utilized to identify the parameters, i.e., natural frequency, damping coefficient, and modal constant, as these can be extracted from the circle's geometry. A least-square algorithm can be used to fit the circle, making the process easier. We will present the equations that are used to find the modal parameters from the frequency responses of the plant, and the reader is referred to [14, 17] for more details about the method.
The frequency at which the maximum sweep rate of data points around the circle is attained

is known as the natural frequency, and it is defined by the expression

$$\gamma(\omega^2) = \frac{d\theta}{d(\omega^2)} \approx \frac{\Delta\theta}{\Delta(\omega^2)}. \tag{4.15}$$

The other modal parameters can be determined using the following equations

$$\eta = \frac{2}{\omega_n^2 \gamma_{max}}, \tag{4.16a}$$

$$b_0 = 2R\omega_n^2\eta. \tag{4.16b}$$

This process is repeated for all the modes by first picking all the local peaks and using the circle fit method around these local peaks. All the modal parameters can be collected to form a feature vector ϕ for each plant as follows

$$\phi = [b_0, b_1, \ldots, b_n, \zeta_1, \ldots, \zeta_n, \omega_1, \ldots, \omega_n]^T \tag{4.17}$$

Now, a GMM can be used to cluster the plants based on these feature vectors.

Gaussian Mixtures Model Clustering

Gaussian Mixtures Model (GMM) clustering is a statistical method for clustering data points into multiple groups or clusters based on their similarity. In this method, each cluster is modeled as a probability distribution with a Gaussian (normal) distribution. A GMM assumes that the data points in each cluster are generated from a mixture of multiple Gaussian distributions with different means and variances. The GMM clustering algorithm estimates the parameters of these Gaussian distributions (such as means and variances) and assigns each data point to the cluster with the highest probability of generating that point.

$$P_\Phi(\phi) = \sum_{k=1}^{K} \pi_k P_{\Phi|Z=k}(\phi; \mu_k, \Sigma_k), \tag{4.18}$$

where

ϕ :	is the observed quantity
Z :	is one of K classes
$\Phi \vert Z = k$:	is Gaussian
μ_k :	is the mean of cluster K
Σ_k :	is the covariance of cluster K
π_k :	is the probability of belonging to cluster k
P_Φ :	is the probability density of ϕ

with

$$P_{\Phi|Z=k}(\phi, \mu_k, \Sigma_k) = \frac{1}{(2\pi)^{D/2}|\Sigma_k|^{1/2}} \exp(-\frac{1}{2}(\phi - \mu_k)\Sigma_k^{-1}(\phi - \mu_k)). \tag{4.20}$$

Maximum Likelihood Estimate (MLE) for the Guassian Mixtures is given by

$$\theta = (\pi_1, \mu_1, \Sigma_1, \pi_2, \mu_2, \Sigma_2, \ldots, \pi_K, \mu_K, \Sigma_K), \tag{4.21}$$

which is the solution of the following optimization problem

$$\begin{aligned} \underset{\theta}{\text{maximize}} \quad & \sum_{i=1}^{N} \ln(\sum_{k=1}^{K} \pi_k P_{\Phi|Z=k}(\phi_i, \mu_k, \Sigma_k)) \\ \text{subject to} \quad & \sum_{k=1}^{K} \pi_k = 1 \\ & \pi_k \geq 0 \\ & \Sigma_k \succ 0. \end{aligned} \tag{4.22}$$

The above optimization problem can be solved by the Expectation Maximization (EM) algorithm ([95]).

4.5 Results

In this section, we present the results of hard and soft clustering algorithms applied to the frequency response data of VCM plants shown in Fig. 4.1. It can be observed that the responses can be categorized into three different clusters. However, for the algorithm, we used the elbow method to determine the optimal number of clusters for hard clustering. Both algorithms were successfully able to cluster the plants into three clusters.

For the GMM approach, feature vectors of size ten (one feature corresponding to the rigid body mode, three features corresponding to each resonant mode) were constructed from the frequency response data for each plant. A total of 300 data points were used for soft clustering.

4.6 Summary

In this chapter, we presented a novel way of clustering transfer functions and frequency response data of stable LTI plants by defining distance metrics in terms of H_∞ and H_2 norms. The distance metrics were used to cluster the systems using k-means and k-medoids algorithm. We also presented a novel way of soft clustering transfer functions and frequency response data utilizing Gaussian Mixture Models for a class of systems which are common

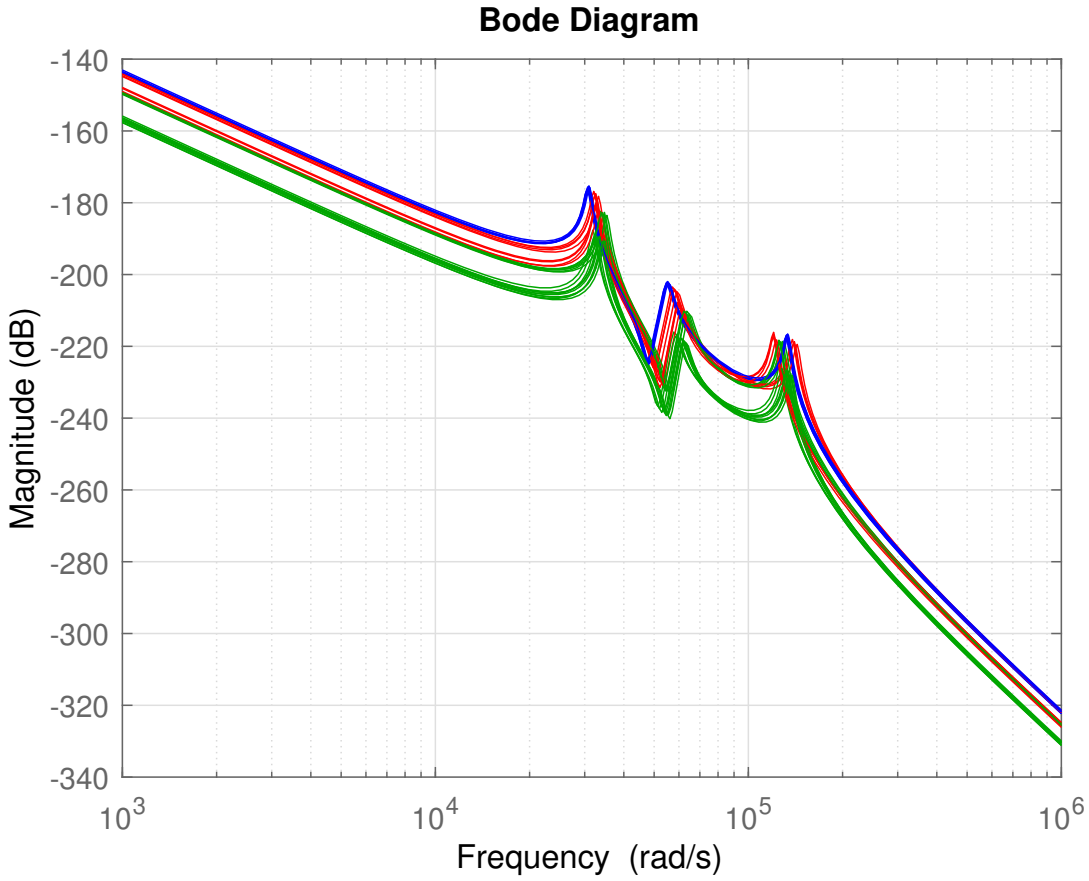

Figure 4.4: Clustering results for 30 VCMs using k-medoids algorithm with H_2 distance respectively.

in mechatronic systems like HDDs by extracting features from the frequency response data and applying using these features to cluster the frequency responses. The techniques have been demonstrated on various plants used in HDDs and the results are presented.

Chapter 5

Data-Driven Robust Feedback Control Design for Multi-Actuator Hard Disk Drives

5.1 Introduction

One factor that determines the spacing between tracks in an HDD is the controller's sensitivity to disturbances. To enhance the disturbance rejection characteristics of HDDs, additional actuators are incorporated. The VCM actuator compensates for low-frequency disturbances and has a higher range of motion, while PZTs address high-frequency disturbances with a smaller range of motion, typically covering one to three tracks.

An important source of disturbances in data centers is the acoustic noise from adjacent HDDs and other machinery. These disturbances have high-frequency components (5-10 kHz). To enhance disturbance rejection in these scenarios, robust and precise controllers need to be designed.

Recently, data-driven feedback control design approaches from frequency response measurements [19, 28, 29] for DSA HDDs have been developed in [3, 4] to suppress disturbances. These data-driven controllers are robust to variations in plant models. Sometimes designing a common controller for all the plants might not be optimal for each plant and in some cases a stabilizing controller might not even exist. In such cases to maintain robustness and optimality, we cluster the plants based on the distance between each other using techniques presented in chapter 4 and design common controllers within each cluster.
To further improve the disturbance rejection capabilities which cannot be handled by feedback controllers alone, data-driven feedforward controllers have been developed in [79, 78, 80] based on frequency response measurements of disturbance processes. Additionally, an adaptive feedforward controller has been developed in [11], layered on top of the robust feedback controller to account for plant variations.

In this chapter, we will focus on developing robust feedback controllers using multiple

plant measurements for a TSA HDD equipped with a VCM actuator and two PZT actuators (namely milli and micro). The controller design will be formulated as a convex optimization problem with the objective of minimizing the H_∞ norm or H_2 norm of a function of closed-loop transfer functions, subject to constraints on the H_∞ and H_2 norms. The H_∞ constraints will shape the closed-loop transfer functions using numerical weights in the frequency domain and ensure the stability of the closed-loop system [3]. The H_2 constraints will be used to limit the variances of various signals.

5.2 Preliminaries

In this section, we present some preliminaries and nomenclature related to the structure of a TSA system. The discussion will be confined to the discrete-time case (z-domain) since the design process in the s-domain follows straightforwardly. Consider the control block diagram shown in Fig. 5.1. The HDD plant $G(z)$ is a multi-input single-output (MISO) system that takes the control signals generated by the controller and produces the displacement of the head. The controller $K(z)$ is a single-input multi-output (SIMO) system that takes the measured position error signal and generates the control inputs.

In this diagram:

- r represents the runout that needs to be followed,
- y is the total head position,
- e is the actual position error signal,
- n represents the measurement noise,
- u is the vector control input signal generated by the controller,
- $D(z)$ is the disturbance process, and
- $\bar{d}$ is zero-mean white noise of unit power, assumed to produce the actual disturbance d through $D(z)$.

In general, we only have access to the measured position error signal, which is $e+n$. For brevity, the dependence on z will be omitted in the remainder of the chapter.

Though the plant is denoted by $G(z)$, it must be noted that the actual plant is not available to us. Instead, only a set of frequency responses representative of the plant are available, i.e., the frequency responses $G(e^{j\omega T_s})$, with T_s as the sampling time, are available for a finite set of $\omega \in \Omega = \left(-\frac{\pi}{T_s}, \frac{\pi}{T_s}\right)$.

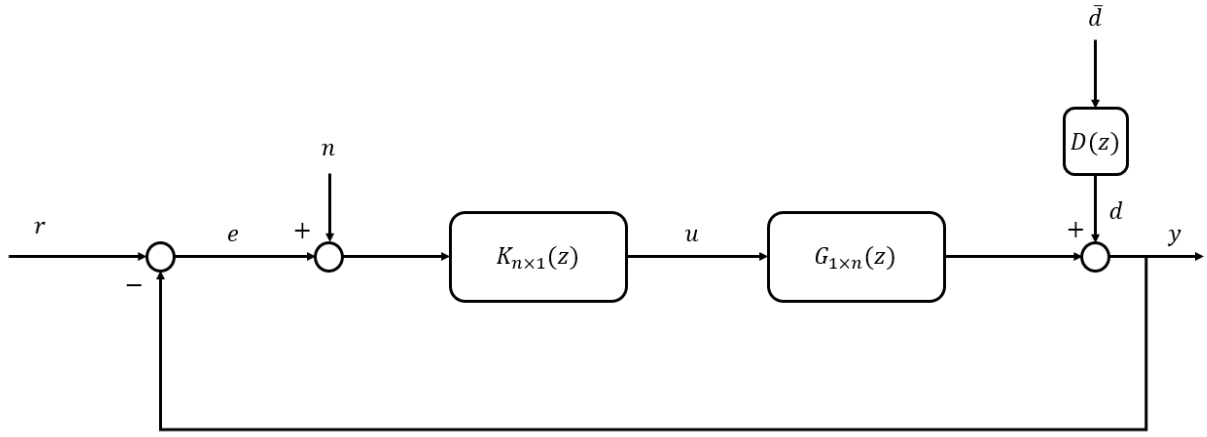

Figure 5.1: Control block diagram for a MSA HDD.

Triple Stage Actuation System

Consider the block diagram shown in Fig. 5.2. The plant includes three actuators: a VCM actuator and two PZT actuators (milli and micro). The plant G_{VCM} represents the VCM actuator, G_{milli} represents the milli PZT actuator, and G_{micro} represents the micro PZT actuator. The control inputs to these actuators are u_{VCM}, u_{milli}, and u_{micro} respectively. Let K_1, K_2, and K_3 represent the Single Input Single Output (SISO) controller transfer functions from the measured position error signal to the control inputs u_{VCM}, u_{milli}, and u_{micro}, respectively.

The magnitude frequency responses of the benchmark model for the VCM and PZT actuators are shown in Fig. 5.3. Additionally, some typical variations in resonance frequencies are illustrated. These variations usually limit the achievable bandwidth of the servo system. To address this, milli and micro actuators, which have higher resonance modes than the VCM but can only achieve a stroke of a few tracks, are used to enhance overall disturbance attenuation in the servo system. In our case study model, the milli PZT actuator has a nominal resonance frequency of 9.5 kHz, while the micro PZT actuator has a higher nominal resonance frequency of 15 kHz.

In an actual HDD product, multiple frequency response measurements for various TSA systems can be collected, as the dynamics of actuator assemblies differ slightly between disk platters and among different drives in the same product line. Additionally, even for a specific actuator, the dynamics may vary due to environmental factors such as temperature. Benchmark models are utilized in model-based control design, while multiple measurements with slight variations are used in data-driven control design.

In large data centers, millions of enterprise HDDs are stacked in server boxes, operating in

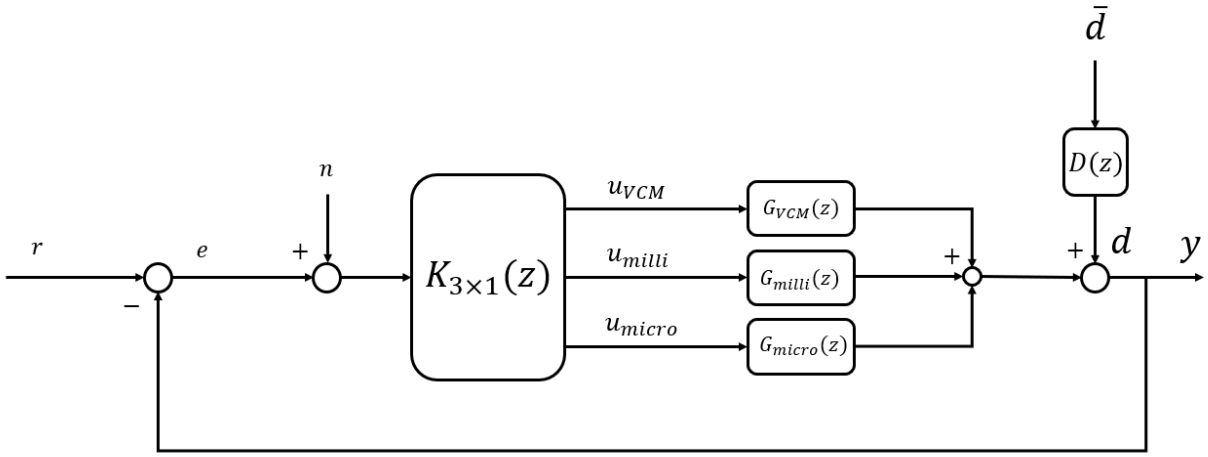

Figure 5.2: Control block diagram for a TSA HDD.

a harsh environment with significant vibrations generated by neighboring drives and cooling fans. Following conventional industry practices, the effects of track run-out, windage, and internal and external disturbances of an HDD are incorporated into the definition of the disturbance filter $D(z)$. A typical magnitude frequency response of the disturbance filter $D(z)$ is shown in Fig. 5.4. The impact of $D(z)$ on the position error signal is significantly greater than that of the measurement noise n and the runout r. Therefore, in the design presented here, the effects of n and r are not considered, although they can easily be included in our design framework.

Sensitivity Decoupling Structure

The feedback control block diagram for a TSA system with the sensitivity decoupling structure ([52]) is shown in Fig. 5.5. $\hat{G}_{milli}$ and $\hat{G}_{micro}$ are the plant estimates of milli and micro PZT actuators respectively. K_{VCM}, K_{milli} and K_{micro} are single-input-single-output (SISO) controllers for VCM, milli and micro PZT actuators respectively (not to be confused with K_1, K_2 and K_3 defined earlier).

The sensitivity decoupling structure is necessary during the implementation of the controller in real HDDs because it can stabilize just the VCM loop in case the other actuators fail. The plant estimates are usually close to the actual plant models at low frequencies and there are only high frequency uncertainties. The overall closed loop sensitivity from the disturbance d to the position error signal e approximated by Eq. (5.1). Eq. (5.1) clearly shows how the use of double-stage and triple stage systems increase the overall attenuation of the servo system, as the overall error rejection closed loop sensitivity transfer function can

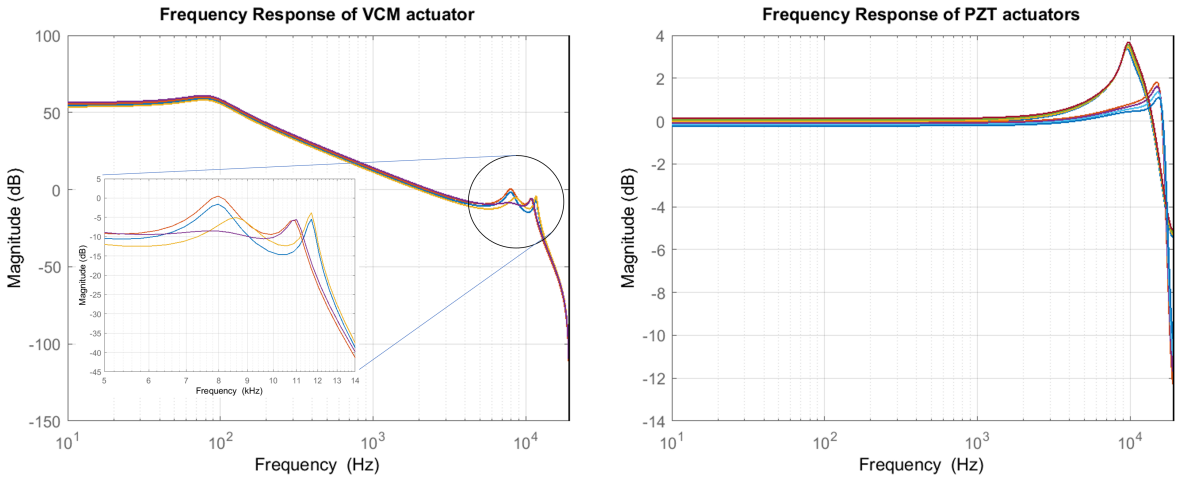

Figure 5.3: VCM actuator frequency responses (left) and PZT actutor frequency responses (right)

be approximated by the series combination of all three single-stage sensitivities.

$$S_{d\to e} \approx S_{d\to e}^{VCM} S_{d\to e}^{milli} S_{d\to e}^{micro}, \tag{5.1}$$

where

$$\begin{aligned} S_{d\to e}^{VCM} &= \frac{1}{(1+K_{VCM}G_{VCM})}, \\ S_{d\to e}^{milli} &= \frac{1}{(1+K_{milli}G_{milli})}, \\ S_{d\to e}^{micro} &= \frac{1}{(1+K_{micro}P_{micro})}. \end{aligned} \tag{5.2}$$

A more general control diagram for a TSA system with a SIMO controller K_{TSA} is shown in Fig. 5.2. In order to take full advantage of the TSA system, we first design the optimal SIMO controller K_{TSA} and transform the SIMO controller into three SISO controllers K_{VCM}, K_{milli} and K_{micro}. The SISO controllers can be computed as

$$\begin{aligned} K_{VCM} &= \frac{K_1}{1+K_2\hat{G}_{milli}+K_3\hat{G}_{micro}}, \\ K_{milli} &= \frac{K_2}{1+K_3\hat{G}_{micro}}, \quad K_{micro}=K_3, \end{aligned} \tag{5.3}$$

where $\hat{G}_{milli}$ and $\hat{G}_{micro}$ are estimates of the dynamics of milli and micro PZT actuators respectively and are used in the sensitivity decoupling controller implementation shown in Fig. 5.5.

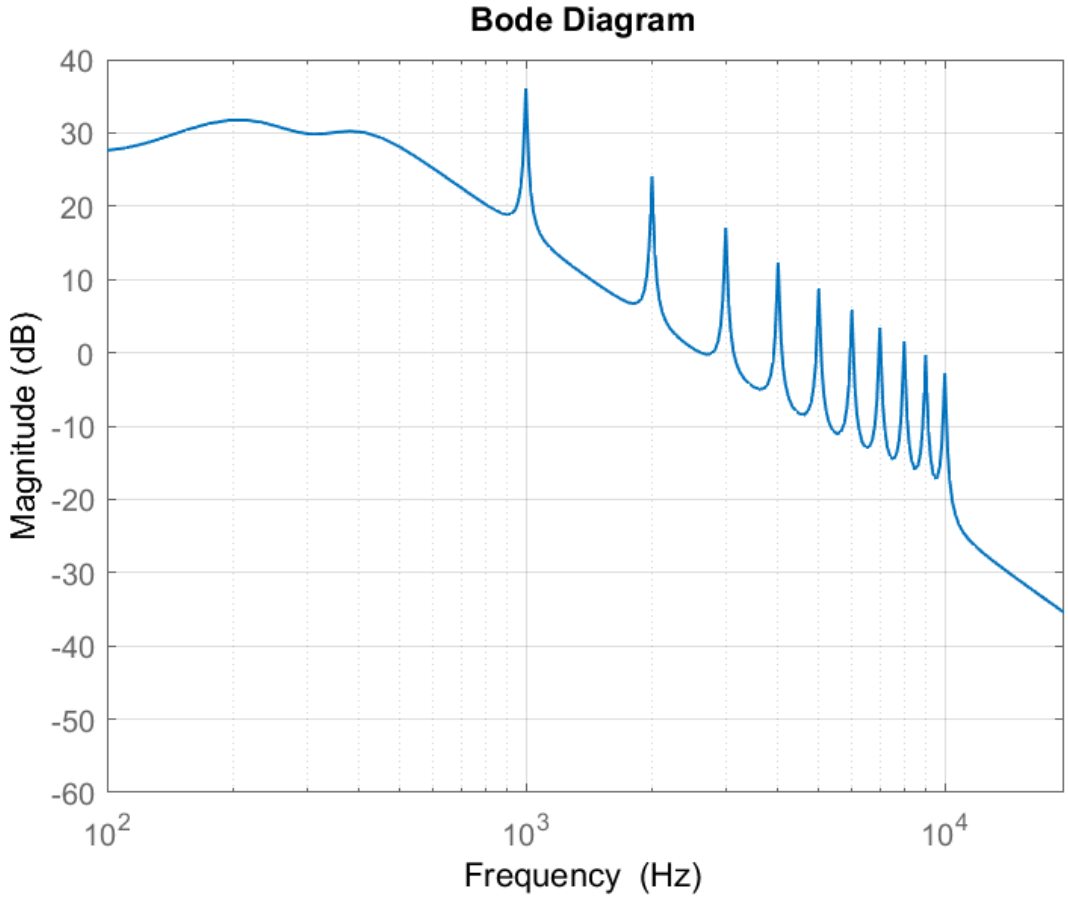

Figure 5.4: Typical frequency response of the disturbance filter $D(z)$ in Figs. 5.1 and 5.2.

Controller Factorization

The vector controller transfer function can be factorized as $K = XY^{-1}$ following the convention from [28] as

$$X(z) = \mathcal{X}_p z^p + \mathcal{X}_{p-1} z^{p-1} + \cdots + \mathcal{X}_0,$$
$$Y(z) = z^p + \mathcal{Y}_{p-1} z^{p-1} + \cdots + \mathcal{Y}_0,$$

where p is the controller order and the controller parameters are $\{\mathcal{Y}_{p-1}, \ldots \mathcal{Y}_0\} \in \mathcal{R}$ and $\{\mathcal{X}_p, \mathcal{X}_{p-1}, \ldots \mathcal{X}_0\} \in \mathcal{R}^{n\times 1}$. For the remainder of the paper, the convention $X_i = \mathcal{X}_p(i)z^p + \mathcal{X}_{p-1}(i)z^{p-1} + \cdots + \mathcal{X}_0(i)$ will be used to denote the i^{th} row of $X(z)$.

Closed loop transfer functions

With this controller factorization, various essential closed loop transfer functions can be easily calculated as follows

$$\begin{aligned} S_{d\to e} &= \frac{Y}{Y+GX}, U_{d\to u_{VCM}} &= \frac{X_1}{Y+GX}, \\ U_{d\to u_{milli}} &= \frac{X_2}{Y+GX}, U_{d\to u_{micro}} &= \frac{X_3}{Y+GX}, \\ Y_{d\to y_{milli}} &= \frac{G_{milli}X_2}{Y+GX}, Y_{d\to y_{micro}} &= \frac{G_{micro}X_3}{Y+GX}. \end{aligned} \tag{5.4}$$

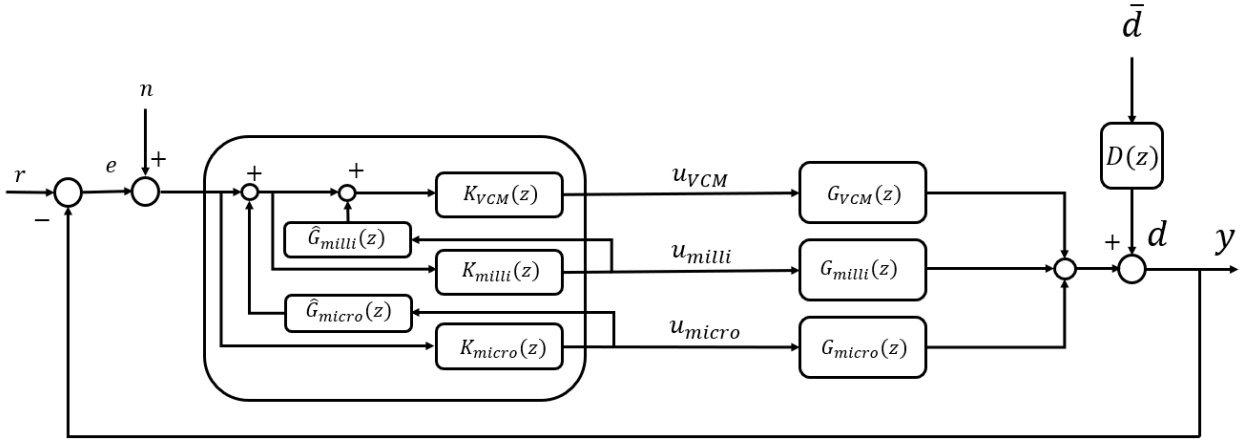

Figure 5.5: Control block diagram for a TSA HDD with sensitivity decoupling structure.

5.3 Data-Driven Control Design

In this section, various objectives and constraints that can be used to design the controller will be formulated in such a way that a convex optimization solver can obtain a solution. The formulation will be demonstrated for one of the closed loop transfer functions, but the same procedure can be followed for any of the closed loop transfer functions. The H_∞ constraints are used to shape the closed loop transfer functions and H_2 constraints are used to constrain the variance of various signals. In the data-driven design, the constraints will only be enforced at the frequencies at which the responses are available. It is assumed that the data set is rich enough i.e., the frequencies at which the data is collected represent the key characteristics of the plants.

Constraints and Objectives

H_∞ Constraints:

H_∞ constraints are used to shape the closed loop transfer functions. A feasible controller that satisfies the H_∞ constraints is guaranteed to stabilize the closed loop system ([3]). A typical H_∞ constraint on the weighted sensitivity of the TSA can be formulated as $||W(\omega)S_{d\to e}(e^{j\omega})||_\infty \leq \gamma$, where $W(\omega)$ is a numerically shaped weight for all the frequencies $\omega \in \Omega = (-\frac{\pi}{T_s}, \frac{\pi}{T_s})$. Using the definition for sensitivity with the controller factorization, this constraint can be equivalently written as (from [3])

$$\gamma^{-1}|W(\omega)Y(e^{j\omega})| < Re(Y(e^{j\omega}) + G(\omega)X(e^{j\omega})). \tag{5.5}$$

Note that for a MISO system, this constraint is a convex constraint as it can be expressed as a Second Order Cone constraint. Any of the available Second Order Conic Program (SOCP) solvers can handle these constraints.

H_2 Constraints:

H_2 constraints can be used to minimize the variances of various signals. For a normally distributed random variable, the confidence of finding the random variable within three standard deviations from the mean is 99.7%. Though hard limits on the magnitudes of the signals cannot be enforced in the frequency domain design unlike in state space techniques like Model Predictive Control (MPC), we can constrain the signal's variance to increase the confidence that the signal does not go beyond the limit 99.7% of the time. This can be done by restricting the L_2 norm of the zero mean signal to be always less than a third of the limit.

The constrains can be applied on various signals like the inputs to the actuators, the stroke limits of the actuators, the overall position error signal etc. In case of multiple plants, we can either constrain the L_2 norms of the signals for each of the plants or constrain the average L_2 norm of the signals for all the plants. For example, if the average variance of the VCM actuator input due to the disturbance needs to be constrained to be less than μ for l plants, using Parseval's theorem, it can be be formulated as

$$\begin{aligned} &||u_{VCM,i}||_2 = ||U_{d\to u_{VCM,i}}D||_2 \\ \implies &\frac{1}{l}\sum_{i=1}^{l}||u_{VCM,i}||_2^2 = \frac{1}{l}\sum_{i=1}^{l}||U_{d\to u_{VCM,i}}D||_2^2 \leq \mu, \end{aligned} \tag{5.6}$$

where the subscript i corresponds to the i^{th} plant and D is the disturbance filter shaping transfer function. The H_2 norm for this input transfer function is given by

$$||U_{d\to u_{VCM,i}}D||_2^2 = \frac{T_s}{2\pi}\int_\Omega tr(D^*U_{d\to u_{VCM,i}}^* U_{d\to u_{VCM,i}}D)d\omega. \tag{5.7}$$

Using slack variables, the H_2 constraint in Eq. (5.6) can be reformulated into two constraints as

$$\frac{1}{l}\sum_{i=1}^{l}\frac{T_s}{2\pi}\int_{\Omega} tr(\Gamma_i^{u_{VCM}}(\omega))d\omega \leq \mu, \tag{5.8}$$

$$D^* U^*_{d\to u_{VCM},i} U_{d\to u_{VCM},i} D \preceq \Gamma_i^{u_{VCM}}. \tag{5.9}$$

Since the responses are available only at a finite set of frequencies, the integral in Eq. (5.8) will be evaluated approximately using a trapezoidal rule by replacing the function $\Gamma_i^{u_{VCM}}(\omega)$ with a finite set of $\Gamma_{k,i}^{u_{VCM}}$ for $k = 1$ to N where N is the number of frequency responses available per plant. The matrix inequality in Eq. (5.9) can be reformulated using controller factorization and the Schur complement lemma as

$$\begin{bmatrix} \Gamma_{k,i}^{u_{VCM}} & X_1 D \\ D^* X_1^* & (Y + G_i X)^*(Y + G_i X). \end{bmatrix} (\omega_k) \succeq 0 \tag{5.10}$$

Notice that this is a non convex constraint, as $P^*P = (Y + G_iX)^*(Y + G_iX)$ is quadratic. However, this quadratic part can be linearized using a first order Taylor series expansion around a nominal controller $K_c = X_c/Y_c$, as shown in [28], using

$$P^*P \approx P_c^* P_c + (P - P_c)^* P_c + P_c^*(P - P_c), \tag{5.11}$$

where $P_c = Y_c + G_i X_c$ as

$$\begin{bmatrix} \Gamma_{k,i}^{u_{VCM}} & X_1 D \\ D^* X_1^* & P_c^* P_c + (P - P_c)^* P_c + P_c^*(P - P_c) \end{bmatrix} (\omega_k) \succeq 0. \tag{5.12}$$

H_∞ Objective:

Consider the case in which the worst case weighted sensitivity of the TSA system needs to be minimized. The optimization problem in this case can be formulated as

$$\begin{aligned} \min_{X,Y} \quad & \max_{i=\{1,\ldots,l\}} ||W S^i_{d\to e}||_\infty \\ \text{s.t.} \quad & H_\infty \text{ and } H_2 \text{ constraints}, \end{aligned} \tag{5.13}$$

where W is a numerically shaped weighing filter as a function of frequency ω and the superscript i corresponds to the i^{th} set of actuators. The same optimization problem can be reformulated using a slack variable γ as follows

$$\begin{aligned} \min_{X,Y} \quad & \gamma \\ \text{s.t.} \quad & ||W S^i_{d\to e}||_\infty \leq \gamma \;\forall\; i = \{1, \ldots, l\} \\ & H_\infty \text{ and } H_2 \text{ constraints} \end{aligned} \tag{5.14}$$

or equivalently using Eq. (5.5) as

$$\begin{aligned} \min_{X,Y} \quad & \gamma \\ \text{s.t.} \quad & \gamma^{-1}|W(\omega)Y(e^{j\omega})| \leq Re(Y(e^{j\omega}) + G(\omega)X(e^{j\omega})) \\ & H_\infty \text{ and } H_2 \text{ constraints.} \end{aligned} \tag{5.15}$$

Note that the first constraint in Eq. (5.15) is a non-convex constraint but it turns into a second order conic constraint when γ is fixed. In the next section, this structure will be exploited to develop a bisection technique to find the globally minimum γ.

H_2 Objective:

H_2 objectives can be used in cases when the variances of signals need to be minimized. A very obvious choice would be to minimize the variance of the tracking error i.e., the position error signal e. The position error signal due to the disturbance $\bar{d}$ is given by $S_{d\to e}D\bar{d}$.
Since $\bar{d}$ is is zero mean white noise of unit variance, its Fourier transform is just 1. Now the variance of the position error signal e due to the disturbance $\bar{d}$ can be equivalently written in the frequency domain as $||S_{d\to e}D||_2^2$.
The optimization problem in it's raw form can be written as

$$\begin{aligned} \min_{X,Y} \quad & ||S_{d\to e}D||_2^2 \\ \text{s.t.} \quad & H_\infty \text{ and } H_2 \text{ constraints.} \end{aligned} \tag{5.16}$$

Using a similar approach shown for the H_2 constraints, we can formulate the optimization problem using slack variables as follows

$$\begin{aligned} \min_{X,Y} \quad & \int_\Omega tr(\Gamma_{S_{d\to e}}(\omega))d\omega \\ \text{s.t.} \quad & \begin{bmatrix} \Gamma_{S_{d\to e}}(\omega) & YD \\ D^*Y^* & (Y+G_iX)^*(Y+G_iX) \end{bmatrix} \succeq 0 \\ & H_\infty \text{ and } H_2 \text{ constraints.} \end{aligned} \tag{5.17}$$

Minimum H_2 norm controller

As was shown previously, the H_2 norm minimization is a non-convex problem. The quadratic part can be linearized around a nominal controller to make the constraint convex or more precisely a Linear matrix Inequality (LMI). The final controller can be designed by first obtaining the nominal controller using just H_∞ constraints without any objective and then linearizing around it to obtain a better controller with the H_2 constraints and H_2 objective. The detailed algorithm is shown algorithm 1.

Algorithm 1 Iterative algorithm for minimum H_2 norm controller

Require: $N_{iter} \in \mathcal{Z}^+$ ▷ number of iterations required for satisfactory convergence

1: Define H_∞ Constraints
2: Optimize(H_∞ Constraints)
3: Obtain initial controller $K_0 = X_0/Y_0$
4: Define linearized H_2 Constraints with X_0, Y_0
5: **while** $k \leq N_{iter}$ **do**
6: Optimize(H_2 Objective, H_∞, H_2 Constraints)
7: Obtain $K_k = X_k/Y_k$
8: Define linearized H_2 Constraints with X_k, Y_k
9: **end while**

Minimum H_∞ norm controller

The optimization problem formulated in Eq. (5.15) is non-convex and hence finding a global minimum is non-trivial. But in the first set of constraints in Eq.(5.15), if γ is fixed, the constraints turn out to be Second Order Conic constraints. This structure can be exploited and a bisection technique can be used to find the globally minimum γ. In the bisection technique, the optimization problem will be converted to a feasibility problem by fixing the γ and the value of γ will be decreased till the problem is no longer feasible. The bisection algorithm is summarized in algorithm 2. Here γ_{max} and γ_{min} are chosen such that the problem is feasible for $\gamma = \gamma_{max}$ but infeasible for $\gamma = \gamma_{min}$. In cases where the knowledge of the system is limited, a very large value for γ_{max} and a very small value for γ_{min} can be chosen. The stopping criterion for the bisection algorithm is determined by the user defined tolerance ϵ.

5.4 Case Study: Application to a triple Stage Actuation HDD

In this section, the data-driven control techniques will be applied to a TSA HDD with a VCM actuator and two PZT actuators to minimize i) H_2 norm of $S_{d\to e}D$, which is equivalent to the standard deviation of the position error signal due to the external disturbance d, and ii) H_∞ norm of $W_{S_{d\to e}}S_{d\to e}$. The frequency response measurements for the VCM and two PZT actuators used in the design are shown in Fig. 5.3. Firstly for the H_2 norm minimization problem, the design and analysis will be conducted on one plant for each of the actuators and for the H_∞ norm minimization problem, four plant measurements are used in the design for each of the actuators. In both these designs, H_∞ constraints on the closed loop sensitivities will be enforced to guarantee stability of the system for each stage, as well as some additional performance attenuation and robustness requirements, as described subsequently. The optimization problem for all the cases was set up using MOSEK and

Algorithm 2 Bisection Technique for finding the minimum H_∞ norm controller

Require: $\gamma_{min} > 0, \gamma_{max} > 0, \epsilon > 0, N_{iter} \in \mathcal{Z}^+$

$\gamma \leftarrow \gamma_{max}$
while $|\gamma_{max} - \gamma_{min}| \geq \epsilon$ **do**
    $\gamma \leftarrow (\gamma_{max} + \gamma_{min})/2$
    Define H_∞ constraints as a function of γ
    Optimize(H_∞ Constraints)
    Obtain initial controller $K_0 = X_0/Y_0$
    Define linearized H_2 Constraints with X_0, Y_0
    while $k \leq N_{iter}$ **do**
        Optimize(H_∞, H_2 Constraints)
        Obtain $K_k = X_k/Y_k$
        Define linearized H_2 Constraints with X_k, Y_k
    end while
    if feasible **then**
        $\gamma_{max} \leftarrow \gamma$
    else
        $\gamma_{min} \leftarrow \gamma$
    end if
end while

YALMIP in MATLAB R2021a.

H_∞ Constraints

Shown below are the H_∞ constraints that have been used on various closed loop transfer functions. W with a corresponding subscript represents a numerically shaped weight that is applied for each of the closed loop transfer functions H_∞ constraints. For example, $W_{S_{d\rightarrow e}^{DSA}}$ is the weight that is applied to the error rejection closed loop sensitivity transfer function $S_{d\rightarrow e}^{VCM}$ for the single-stage H_∞ minimization constraints in Eq. (5.18). The superscript i in the closed loop transfer functions (e.g. $S_{d\rightarrow e}^{VCM,i}$ indicates that the i^{th} frequency response plant data is being used in a computation, as multiple plants' frequency responses can be evaluated. The inverses of the frequency weights W's are shown as dotted lines along with the magnitude bode plots of their respective resulting closed loop transfer functions in all the figures. These weight inverse plots provide an upper bound at each frequency to the magnitude frequency response of the closed loop transfer function that they are being used with in the H_∞ constraint. Thus, they are used to impose minimum performance requirements, perform frequency response shaping and to satisfy robustness requirements. In the case of the error rejection closed loop sensitivity transfer functions, the weights W's are chosen to specify desired minimum levels of attenuation, bandwidths and maximum levels of amplification (see for example Figs. 5.6, 5.7, 5.9 and 5.10). In the case of closed loop

control action transfer functions (e.g. $U^{VCM}_{d\to u_v}$ in Eq. (5.18)), the weights are being used to prevent the excitation of uncertain resonance modes and to constrain the frequency range of operation of each of the three actuators. In our case study, the VCM is desired to operate in the low frequency range [0 4kHz] range, the milli PZT actuator in [1khZ 3kHz] and the micro PZT actuator in [3kHz 10kHz] range. Since it is undesirable to excite the uncertain resonant modes of the actuators, the weights are shaped to have a 0 dB or greater magnitude at the nominal resonant frequencies (i.e., 8kHz for the VCM, 9.5kHz for the milli PZT actuator and 15kHz for the micro PZT actuator) of the actuator that they are being used with in the H_∞ constraints (see for example Figs. 5.6, 5.8 and 5.10). Furthermore, uncertainties in the high frequency regions can arise from measurements and the data might not be representative of the actual dynamics. To avoid operating in higher frequencies than the regions in which the data is reliable, the weights on the closed loop control actions are increased to large values at high frequencies. In this case study, the weights are designed such that their inverses have high roll offs at high frequency regions with 0 dB at the nominal resonant frequencies.

$$\begin{aligned}&\text{Single Stage (VCM):}\\ &||W_{S^{VCM}_{d\to e}} S^{VCM,i}_{d\to e}||_\infty < 1,\ ||W_{U^{VCM}_{d\to u_v}} U^{VCM,i}_{d\to u_v}||_\infty < 1.\end{aligned} \tag{5.18}$$

$$\begin{aligned}&\text{Dual Stage (DSA): } ||W_{S^{DSA}_{d\to e}} S^{DSA,i}_{d\to e}||_\infty < 1,\\ &||W_{U^{DSA}_{d\to u_v}} U^{DSA,i}_{d\to u_v}||_\infty < 1,\ ||W_{U^{DSA}_{d\to u_{milli}}} U^{DSA,i}_{d\to u_{milli}}||_\infty < 1.\end{aligned} \tag{5.19}$$

$$\begin{aligned}&\text{Triple Stage (TSA):}\\ &||W_{S^{TSA}_{d\to e}} S^{TSA,i}_{d\to e}||_\infty < 1,\ ||W_{U^{TSA}_{d\to u_{micro}}} U^{TSA,i}_{d\to u_{micro}}||_\infty < 1,\\ &||W_{U^{TSA}_{d\to u_{milli}}} U^{TSA,i}_{d\to u_{milli}}||_\infty < 1,\ ||W_{U^{TSA}_{d\to u_v}} U^{TSA,i}_{d\to u_v}||_\infty < 1.\end{aligned} \tag{5.20}$$

The superscripts VCM refers to the case when only the VCM actuator is turned on i.e., when $K_2 = K_3 = 0$, DSA refers to the case when the VCM and milli PZT actuators are turned on i.e., when $K_3 = 0$, TSA refers to the case when all the actuators are operating. Though we are interested in the TSA system, it is necessary to impose H_∞ constraints on the the cases when only the VCM is working i.e., when the PZT actuators fail and for the case when only the tertiary actuator fails i.e., the DSA system as H_∞ constrains ensure the stability of the closed loop system. With these constraints, the system will be guaranteed to remain stable despite actuator failures. As the tertiary micro actuator is mounted on the secondary milli actuator, the tertiary actuator gets automatically disconnected when the secondary actuator fails, so, the case when the VCM and micro PZT actuators are working need not be considered.

H_2 Constraints

In this case study we assume that the maximum available input to the VCM is 6 volts and hence the H_2 norm of the control input to VCM is constrained to be less than 2 volts. The

maximum stroke limit of the PZT actuators needs to be constrained to keep them within their respective linear regimes and prevent saturation. The H_2 norm of the outputs of the PZT actuators were constrained to be 1 track each. The track width considered for the design is 30 nano-meters. The means of variances will be constrained in this formulation, but the variances for individual plants can also be constrained separately.

$$\text{Single Stage (VCM):} \frac{1}{l}\sum_{i=1}^{l} ||U^{VCM,i}_{d\to u_{VCM}}||_2^2 < (2 \text{ volts})^2. \tag{5.21}$$

$$\begin{aligned} \text{Dual Stage (DSA):} \frac{1}{l}\sum_{i=1}^{l} ||U^{DSA,i}_{d\to u_{DSA}}||_2^2 &< (2 \text{ volts})^2. \\ \frac{1}{l}\sum_{i=1}^{l} ||Y^{DSA,i}_{d\to y_{milli}}||_2^2 &< (30 \text{ nm})^2. \end{aligned} \tag{5.22}$$

$$\begin{aligned} \text{Triple Stage (TSA):} \frac{1}{l}\sum_{i=1}^{l} ||U^{TSA,i}_{d\to u_{TSA}}||_2^2 &< (2 \text{ volts})^2. \\ \frac{1}{l}\sum_{i=1}^{l} ||Y^{TSA,i}_{d\to y_{milli}}||_2^2 &< (30 \text{ nm})^2. \\ \frac{1}{l}\sum_{i=1}^{l} ||Y^{TSA,i}_{d\to y_{micro}}||_2^2 &< (30 \text{ nm})^2. \end{aligned} \tag{5.23}$$

Tracking error minimization:

We will pose the optimization problem with the objective to minimize the variance of the position error signal due to the disturbance process D. Controllers will be designed for all the three stages i.e., single stage VCM, DSA and TSA, and the performance will be evaluated by using the residual error as a metric. We will just consider one set of plant frequency response measurements ($l = 1$) for the design in this case to unambiguously evaluate the performance in each stage as additional actuators are added. A 20^{th} order controller is designed for all the cases.

Single Stage (VCM):

The position error minimization problem with just the VCM actuator can be formulated as

$$\begin{aligned} \min_{X,Y} \quad & ||S^{VCM}_{d\to e} D||_2^2 \\ \text{s.t.} \quad & \text{Constraints in (5.18) and (5.21).} \end{aligned} \tag{5.24}$$

The standard deviation (σ) of residual position error i.e., the square root of minimum attained value of the objective in this case was 41.326 nano-meters. This implies that the 3σ

value is around 4 tracks. For good performance of the HDD, this value needs to be less than 0.15 tracks (15%) for writing and 0.30 tracks (30%) for reading processes. Magnitude frequency response plots of the closed loop transfer functions, along with the inverses of their corresponding numerically shaped weights (dotted lines), which provide an upper bound of the transfer function magnitude frequency response at each frequency, are shown in Fig. 5.6.

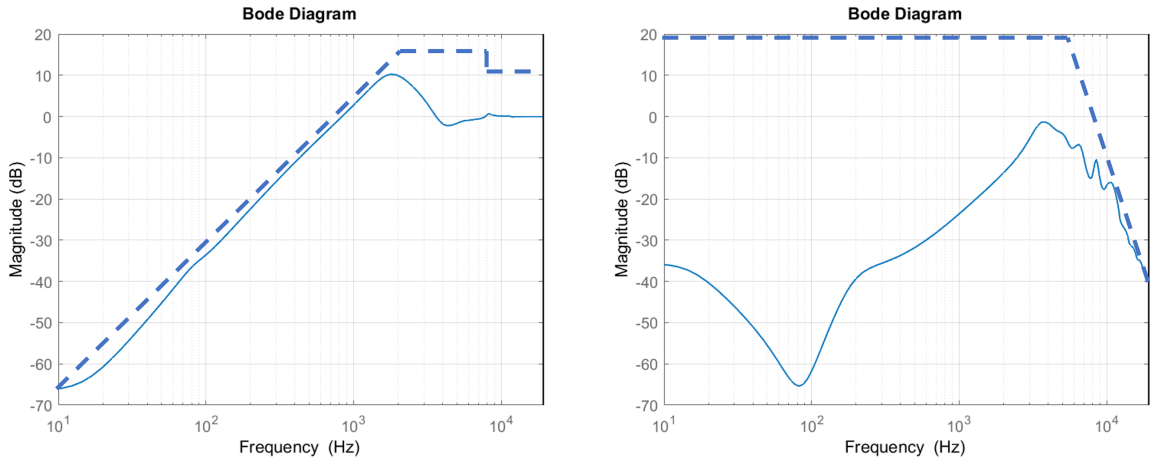

Figure 5.6: The closed loop disturbance to error sensitivity function $S_{d\to e}^{VCM}$ (left) and disturbance to control transfer functions $U_{d\to u_{VCM}}^{VCM}$ for the single stage H_2 norm minimization case.

Dual Stage (DSA):

Position error minimization problem for the DSA system with both VCM actuator and milli PZT actuator can be formulated as

$$\begin{aligned} \min_{X,Y} \quad & ||S_{d\to e}^{DSA} D||_2^2 \\ \text{s.t.} \quad & \text{Constraints in (5.18), (5.19) and (5.22).} \end{aligned} \tag{5.25}$$

The standard deviation (σ) of residual position error in this case was 14.4 nano-meters. This implies that the 3σ value is around 1.5 tracks. With the addition of one PZT actuator (milli), the 3σ value of position error signal reduced by 2.5 tracks when compared to the previous case when only the VCM actuator was operating. Frequency response plots of the closed loop transfer functions along with the inverses of their corresponding numerically shaped weights (dotted lines) for this design are shown in Fig. 5.7 and 5.8.

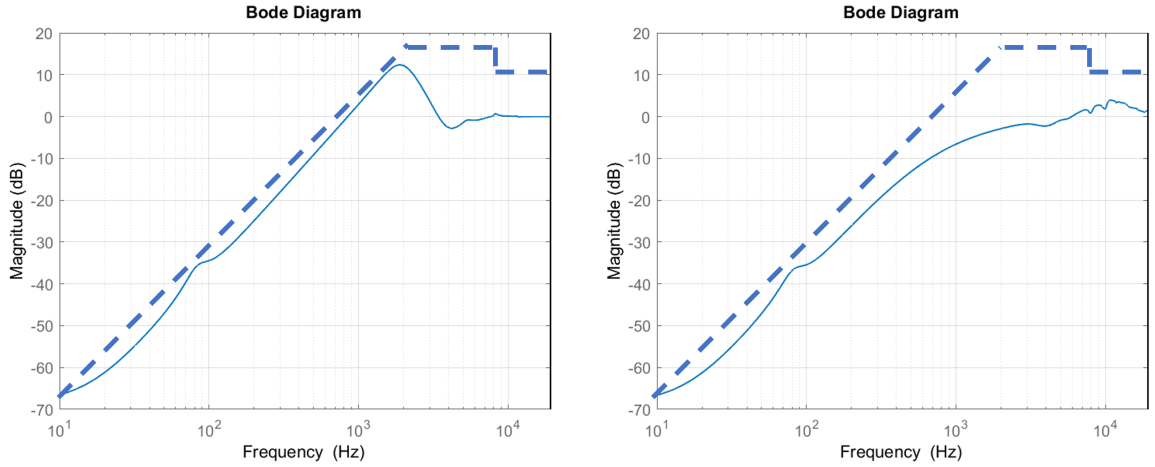

Figure 5.7: The closed loop disturbance to error sensitivity function $S_{d\to e}^{VCM}$ (left) and $S_{d\to e}^{DSA}$ (right) for the DSA H_2 norm minimization case.

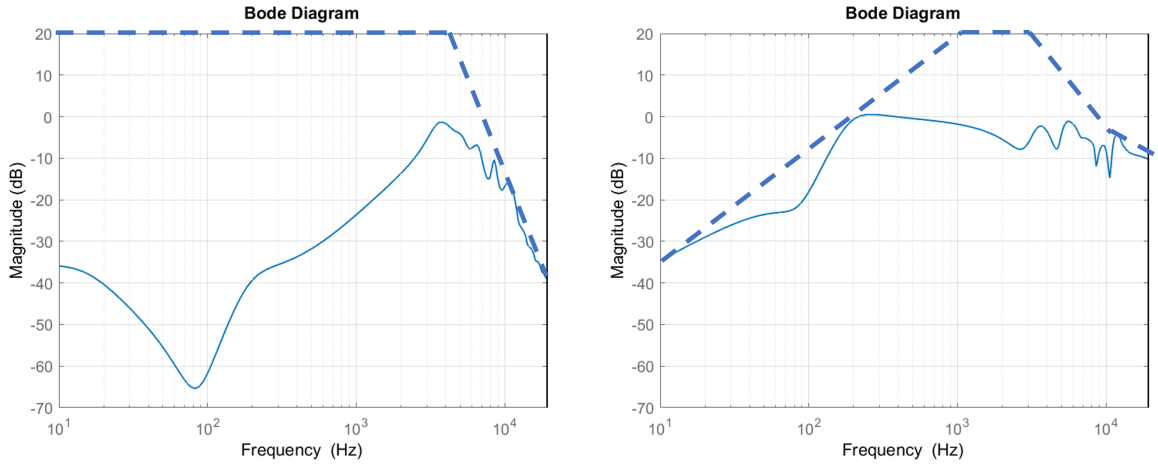

Figure 5.8: The closed loop disturbance to control transfer functions $U_{d\to u_{VCM}}^{DSA}$ (left) and $U_{d\to u_{milli}}^{DSA}$ (right) for the double stage H_2 norm minimization case.

Triple Stage (TSA):

Position error minimization problem for the TSA system with VCM actuator, milli and micro PZT actuators can be formulated as

$$\begin{aligned} \min_{X,Y} \quad & ||S_{d\to e}^{TSA} D||_2^2 \\ \text{s.t.} \quad & \text{Constraints in (5.18), (5.19), (5.20) and (5.23)} \end{aligned} \tag{5.26}$$

The standard deviation (σ) of residual position error in this case was 8.985 nano-meters. This implies that the 3σ value is around 1 track. With the addition of another PZT actuator (micro, the 3σ value of position error signal reduced by 0.5 tracks when compared to the DSA system case.

Frequency response plots of the closed loop transfer functions along with the inverses of their corresponding numerically shaped weights (dotted lines) for this design are shown in Fig. 5.9 and 5.10.

The design was repeated by removing all the H_2 constraints for the TSA system and a residual tracking error of 8.524 nm was obtained. This is the lower limit on the residual error that can be achieved with the three actuators for the considered H_∞ constraints and further reduction of the residual will only be possible by either adding additional actuators, using feedforward controllers or relaxing the H_∞ constraints to make the controller more aggressive.

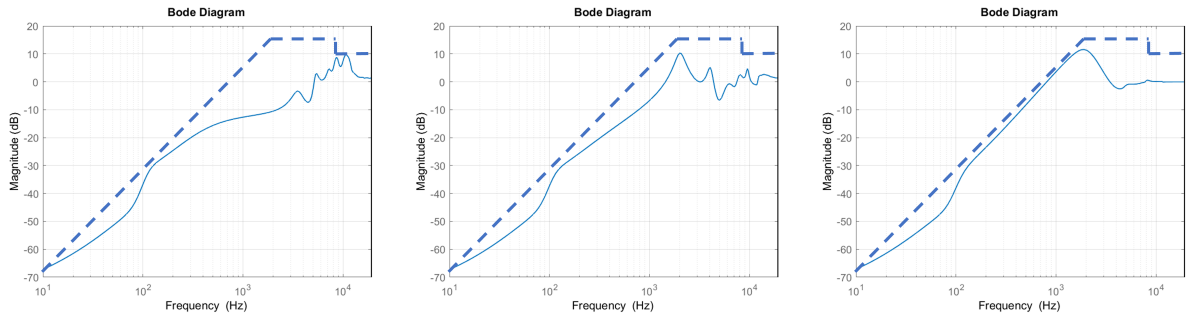

Figure 5.9: The closed loop disturbance to error sensitivity function $S_{d\to e}^{VCM}$ (left), $S_{d\to e}^{DSA}$ (center) and $S_{d\to e}^{TSA}$ (right) for the TSA H_2 norm minimization case.

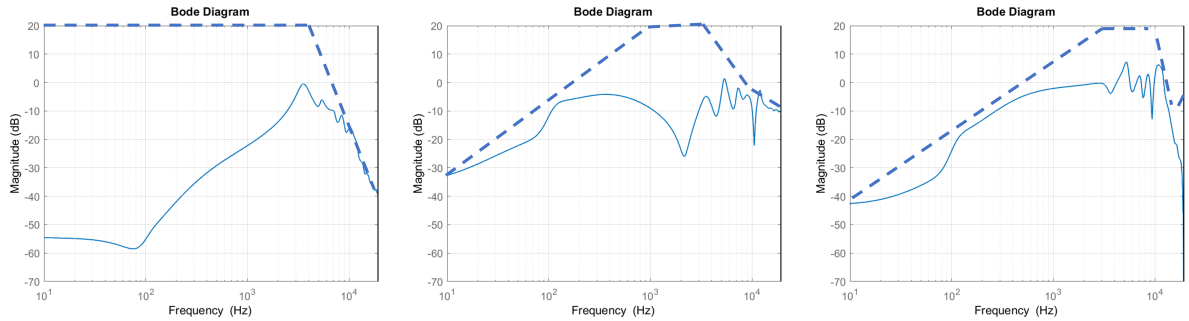

Figure 5.10: The closed loop disturbance to control transfer functions $U_{d\to u_{VCM}}^{TSA}$ (left), $U_{d\to u_{milli}}^{TSA}$ (centre) and $U_{d\to u_{micro}}^{TSA}$ (right) for the TSA H_2 norm minimization case.

Minimum H_∞ norm controller design:

The H_∞ norm minimization problem for the weighted sensitivity of the TSA system can be posed as

$$\begin{aligned} \min_{X,Y} \quad & \max_{i\in\{1,\dots l\}} ||W_{S^{TSA}_{d\to e}} S^{TSA,i}_{d\to e}||_\infty \\ \text{s.t.} \quad & \text{Constraints in (5.18), (5.19), (5.20) and (5.23).} \end{aligned} \tag{5.27}$$

The problem was solved using the bisection technique shown in algorithm 2 for 4 frequency response measurement sets for each of the actuators shown in Fig. 5.3. A 20^{th} order controller was used for the design. γ_{max} (feasible) was set at 10 and γ_{min} (infeasible) was set at 0.1. The controller obtained was able to stabilize all the plants. The global minimum value for the H_∞ norm was found to be 0.3906 when the tolerance for stopping criterion was set at $\epsilon = 0.1$. The frequency responses of the closed loop transfer functions and inverses of their corresponding weights (dotted lines) are shown in Fig. 5.11, 5.12 and 5.13. The plots show that all the H_∞ constraints have been respected thus providing good low frequency disturbance rejection characteristics.

The residual error with the disturbance process D i.e., $||S^{TSA}_{d\to e} D||_2$ was found to be 20.556 nm while the residual error using H_2 norm minimization control was 8.985 nm. Therefore, if the knowledge of the disturbance process is available, H_2 norm minimization controller provides the best possible tracking but when the knowledge is unavailable H_∞ norm minimizing controller tries to get the best sensitivity shape to unknown disturbances with respect to the weights provided.

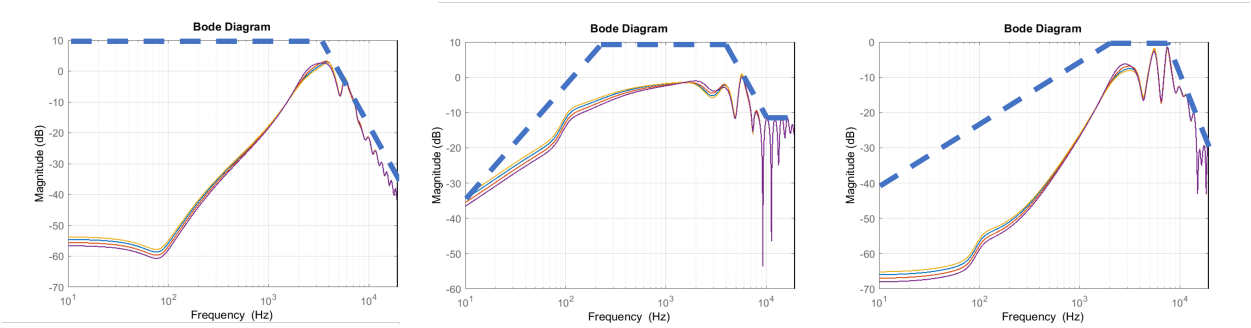

Figure 5.11: The closed loop disturbance to control transfer functions $U^{TSA}_{d\to u_{VCM}}$ (left), $U^{TSA}_{d\to u_{milli}}$ (centre) and $U^{TSA}_{d\to u_{micro}}$ (right) for the TSA H_∞ norm minimization case.

The problem is then solved for controller orders p from 5 to 45 for the TSA system with one frequency response measurement set for each actuator to understand the effect of the controller order on the minimum achieved. The value of the obtained norm is shown in Fig. 5.14. It was seen that there wasn't considerable improvement beyond a 30^{th} order

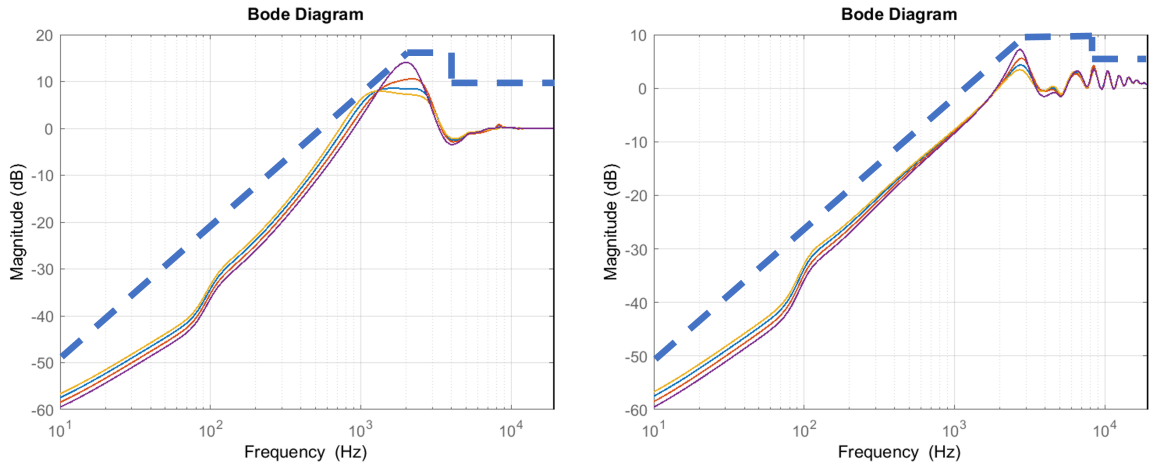

Figure 5.12: Sensitivity plots. $S_{d\to e}^{VCM}$ (left) and $S_{d\to e}^{DSA}$ (right) for the triple stage H_∞ norm minimization case.

controller. It wasn't possible to obtain a stabilizing controller with an order less than 5 for the considered frequency response measurement data.

5.5 Summary

In this chapter, data driven robust feedback control techniques for MISO systems with applications to triple stage hard disk drives have been presented. Controller design has been posed as a convex optimization problem with H_∞ and H_2 objectives and constraints. H_∞ norm constraints on the closed loop transfer functions have been imposed for both H_∞ and H_2 norm minimization problems to ensure stability of the closed loop system. The methodologies were applied on single, dual and triple stage actuation system HDDs and the results were presented. It was only possible to convexify the H_∞ constraints in the case of MISO systems and an approach to convexify them for Multi-Input Multi-Output (MIMO) systems is yet to be developed.

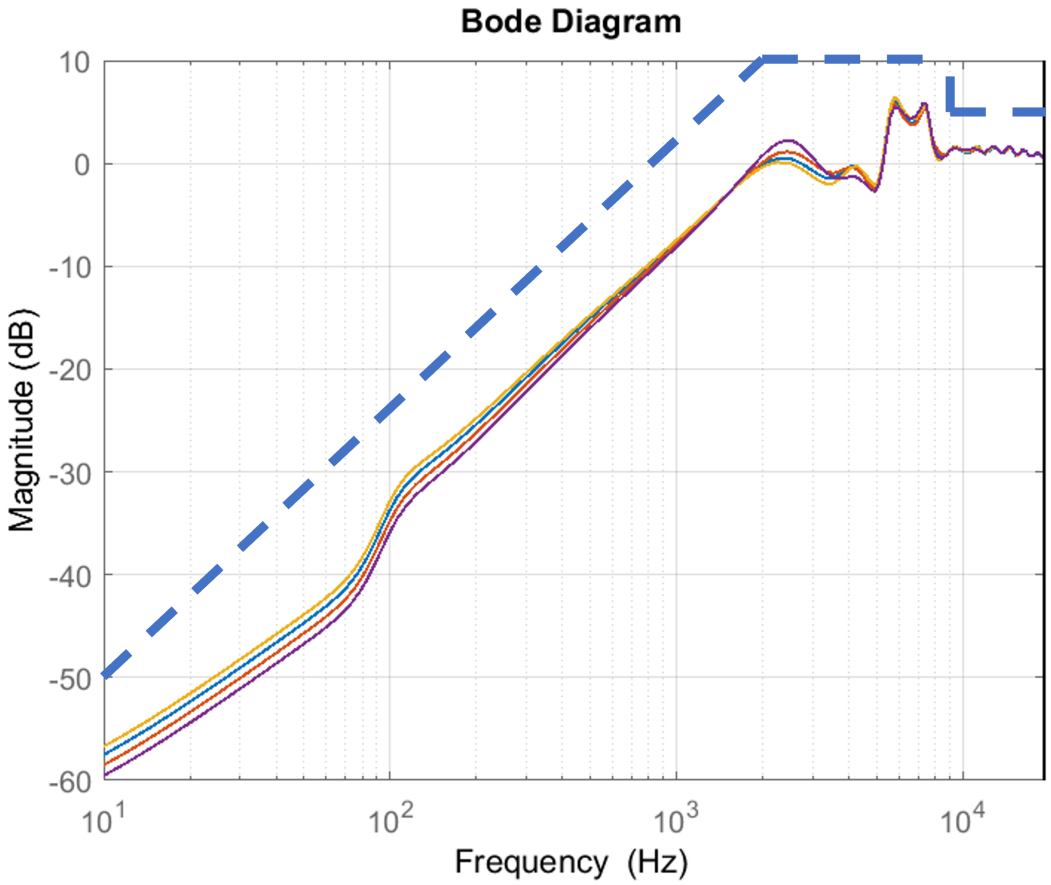

Figure 5.13: Sensitivity plot for $S_{d\to e}^{TSA}$ for the TSA H_∞ norm minimization case.

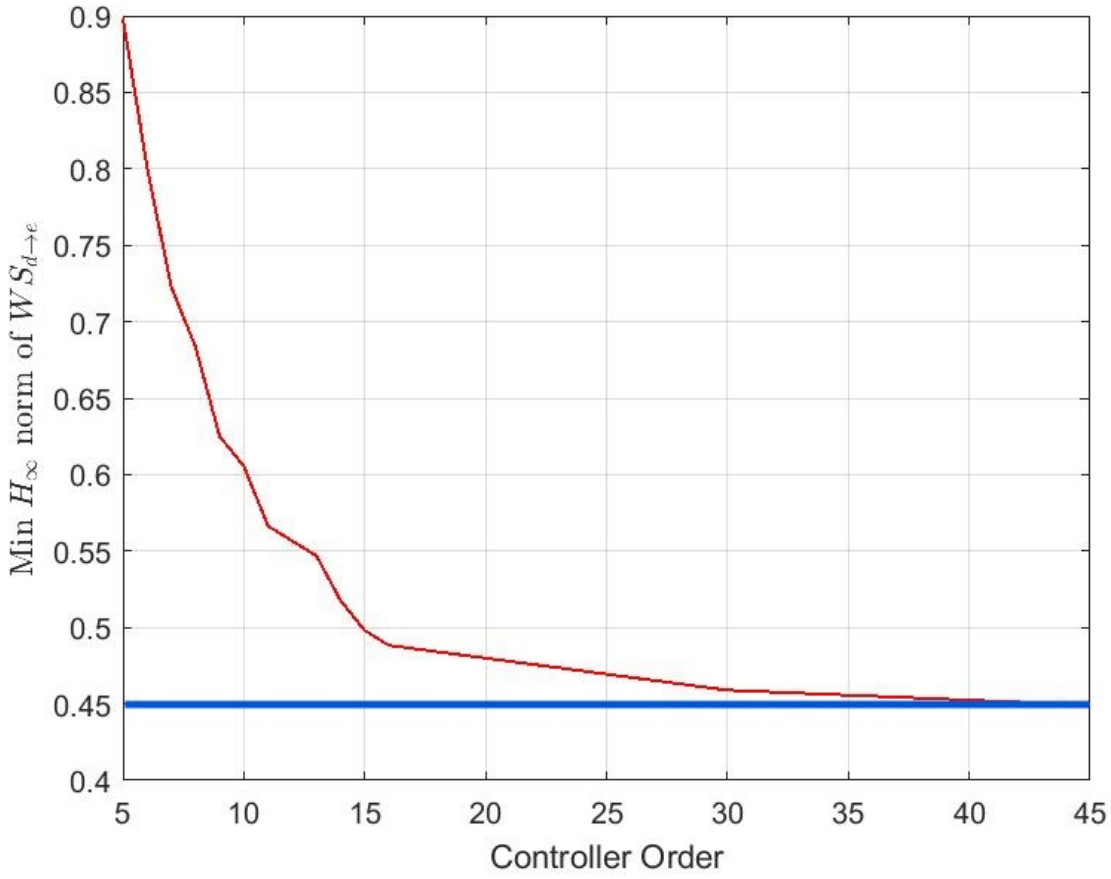

Figure 5.14: Minimum attained H_∞ norm of the disturbance to error sensitivity transfer function vs Controller order. The line in blue indicates the lower limit produced by 'hinfsyn' in MATLAB.

Chapter 6

System Identification in Multi-Actuator Hard Disk Drives with Colored Noises using Observer/Kalman Filter Identification (OKID) Framework

6.1 Introduction

In 2017, Seagate introduced its new multi actuator technology as a significant advancement capable of doubling the data transfer performance of next-generation HDDs designed for hyper-scale data centers. In these drives, the read-write (R/W) heads are divided into two distinct sets—upper and lower halves. This division enables simultaneous operation of the upper and lower platter sets, effectively doubling the data transfer rate.

The implementation of multi actuator technology posed new challenges in the design of HDD controllers. Since both actuator sets operate around the same pivot point, the forces and torques generated by one actuator can impact the operation of the other. The interaction between the two actuators can be classified into three primary scenarios. In the first scenario, both actuators are in track following mode, where it is expected that their interaction is minimal. In the second scenario, both actuators are in seek mode. Here, the vibrational coupling interaction typically remains insignificant compared to the substantial trajectories of both actuators. In the third scenario, one actuator is in seek mode while the other is in track following mode. In this case, the seeking actuator introduces vibrations that disturb the track following actuator, significantly affecting its performance. The feedback controllers may not fully attenuate these transferred disturbances, necessitating the design of feedforward controllers, which rely on an accurate model of the disturbance process.

Modern HDDs utilize their built-in actuators for writing servo tracks instead of employ-

ing a separate servowriter or disk writer. This approach helps to reduce manufacturing costs significantly. A servowriter typically relies on large motors and robust equipment to achieve precise servo track writing, which would be impractically bulky for integration within an HDD. The inherent precision of HDD actuators during the writing process is limited compared to a servowriter, mainly due to their smaller size and less robust construction. Various disturbances such as spindle vibrations and windage affect the writing process, leading to deviations from the ideal circular shape of servo tracks. This deviation is referred to as runout, which can be modeled as white noise colored by a stable disturbance process. The presence of runout introduces colored noise that corrupts the time series signals during operation.

Methods for system identification tend to yield biased estimates in the presence of colored noise corruption. In this chapter, we extend the Observer/Kalman filter Identification (OKID) technique to estimate the disturbance cross transfer function from the Voice Coil Motor (VCM) control input of one actuator to the output of another actuator. This modification utilizes time series data collected under conditions of colored noise corruption and provide unbiased estimates.

OKID [27, 90, 88, 91, 89, 57, 81] is a technique used to identify a linear dynamical system and its associated Kalman filter using input-output measurements that may be affected by noise. Initially developed at NASA under the name OKID/ERA algorithm, OKID stands out from other methods due to its formulation involving state observers, offering a clear interpretation from the perspective of control theory.

6.2 Observer/Kalman filter Identification formulation

In this section, we expand the OKID algorithm to accommodate colored noise. Specifically, we address scenarios where colored noise affects the system's output, such as in HDDs where runout introduces colored noise. This approach can also be adapted for situations where the process noise itself exhibits colored characteristics. Consider the following dynamical system of interest

$$\begin{aligned} x_a(k+1) &= A_a x(k) + B_a u(k) + \bar{w}_p(k) \\ y_a(k) &= C_a x_a(k) + D_a u(k) + w_{cm}(k), \end{aligned} \tag{6.1}$$

where $x_a(k)$ is the state of the system, $\bar{w}_p(k)$ is a white process noise, $u(k)$ is the known control input, $y_a(k)$ is the measured output, w_{cm} is a colored noise (runout in the case of a HDD) and (A_a, B_a, C_a, D_a) represent the state, control, output and feedthrough matrices respectively of the actual system. The subscript 'a' is used for the actual system of interest. The colored noise can be assumed to be the output of a stable dynamical system with a white noise as its input as follows

$$\begin{aligned} x_c(k+1) &= A_c x(k) + w_{wm}(k) \\ y_c(k) &= C_c x_c(k) + D_c w_{wm}(k), \end{aligned} \tag{6.2}$$

where $x_c(k)$ is the state of the coloring process, $w_{wm}(k)$ is a white noise, $y_c(k)$ is is the colored noise ($y_c(k) = w_{cm}(k)$) and (A_c, B_c, C_c, D_c) represent the state, control, output and feedthrough matrices respectively of the coloring process. The subscript 'c' is used for the coloring process. Both the dynamical systems in (6.1) and (6.2) can be augmented to form

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + w_p(k) \\ y(k) &= Cx(k) + Du(k) + w_m(k), \end{aligned} \tag{6.3}$$

where $x = \begin{bmatrix} x_a \\ x_c \end{bmatrix}, A = \begin{bmatrix} A_a & 0 \\ 0 & A_c \end{bmatrix}, B = \begin{bmatrix} B_a \\ 0 \end{bmatrix}, C = \begin{bmatrix} C_a & C_c \end{bmatrix}, D = D_a, w_p = \begin{bmatrix} \bar{w}_p \\ w_m \end{bmatrix}$ & $w_m = D_c w_{wm}$. It can also be seen that the transfer function from the output to the input of the augmented system is the same as that of the actual system from

$$\frac{Y_a(s)}{U(s)} = C_a(sI - A_a)^{-1}B_a + D_a, \tag{6.4}$$

and

$$\begin{aligned} \frac{Y(s)}{U(s)} &= C(sI - A)^{-1}B + D \\ &= [C_a\ C_c](sI - \begin{bmatrix} A_a & 0 \\ 0 & A_c \end{bmatrix})^{-1} \begin{bmatrix} B_a \\ 0 \end{bmatrix} + D \\ &= [C_a\ C_c] \begin{bmatrix} (sI_a - A_a)^{-1} & 0 \\ 0 & (sI_c - A_c)^{-1} \end{bmatrix} \begin{bmatrix} B_a \\ 0 \end{bmatrix} + D \\ &= C_a(sI_a - A_a)^{-1}B_a + D_a, \end{aligned} \tag{6.5}$$

since $D = D_a$. Here I, I_a and I_c represent identity matrices with same sizes as A, A_a and A_c respectively. Here neither the matrices, the noises nor their variances and covariances are assumed to be known. The only assumption made is that the noises in (6.3) are zero mean gaussian.

Since the pair (A_a, C_a) is detectable and A_c is Schur, the pair (A, C) will also be detectable if $C_c \neq 0$.Now, as (A, C) is detectable, a steady state Kalman filter can be designed for the system if we exactly know the matrices (A, B, C, D). Hence a Kalman filter gain K exists such that $A - KC$ is Schur. With the filter gains defined (as unknowns), the observer dynamics can be expressed as

$$\begin{aligned} \hat{x}(k+1) &= (A - KC)\hat{x}(k) + (B - KD)u(k) + Ky(k) \\ \hat{y}(k) &= C\hat{x}(k) + Du(k). \end{aligned} \tag{6.6}$$

Let $F = A - KC,\ H = B - KD$ and $G = K$. Defining $L = [H\ G]$ and $\nu_x(k) = [u^T(k)\ y^T(k)]^T$ for brevity, we get the observer in predictor form as

$$\begin{aligned} \hat{x}(k+1) &= F\hat{x}(k) + Lv_x(k) \\ \hat{y}(k) &= C\hat{x}(k) + Du(k). \end{aligned} \tag{6.7}$$

Using (6.6), the observer state at k^{th} time instance can be derived as

$$\hat{x}(k) = F^p \hat{x}(k-p) + Tz(k), \tag{6.8}$$

where

$$T = \begin{bmatrix} I & F & ... & F^{p-2} & F^{p-1} \end{bmatrix} L,$$
$$z(k) = \begin{bmatrix} \nu_x^T(k-1) & \nu_x^T(k-2) & ... & \nu_x^T(k-p) \end{bmatrix}.$$

The stability of the observer ensures that F^p becomes negligible for sufficiently large values of p ($p >> n$) and hence the observer state in (6.8) becomes

$$\hat{x}(k) \approx Tz(k). \tag{6.9}$$

Substituting (6.9) in (6.7), we get the estimated output as

$$\hat{y}(k) = CTz(k) + Du(k). \tag{6.10}$$

The estimated output in (6.10) is related to the measured output as

$$y(k) = \Phi\nu(k) + \epsilon(k), \tag{6.11}$$

where $\nu(k) = [u^T(k) z^T(k)]^T$, $\Phi = [D\ CL\ CFL\ ...\ CF^{p-2}L\ CF^{p-1}L]$ and $\epsilon(k)$ is the error between the measured output and the estimated output.
The outputs at different time instances can be collected and stacked to obtain the following equation

$$Y = \Phi V + E, \tag{6.12}$$

where

$$Y = [y(p)\ y(p+1)\ ...\ y(l-1)], \tag{6.13a}$$
$$V = [\nu(p)\ \nu(p+1)\ ...\ \nu(l-1)], \tag{6.13b}$$
$$E = [\epsilon(p)\ \epsilon(p+1)\ ...\ \epsilon(l-1)] \tag{6.13c}$$

for l measurements. The best estimate of Φ is obtained using least squares formulation as $\hat{\Phi} = YV^\dagger$ ($\dagger$ denotes the pseudo inverse). Various system matrices and the Kalman filter gain can be extracted from $\hat{\Phi}$ using the Eigensystem Realization Algorithm (ERA).

6.3 Eigensystem Realization Algorithm (ERA)

Eigensystem Realization Algorithm, first developed in [26] and later applied and extended in [22, 54, 72, 10, 7], is a system identification technique used most popularly for aerospace and civil structures from the input and output time domain data. Though ERA uses impulses as inputs to excite the system, it can be intertwined with OKID framework to estimate the system matrices using non-impulsive inputs. There are many variants to the ERA depending

on the application and the type of data collected. In this section we will summarize one of the variants to extract estimates of the system matrices and the Kalman filter gain from $\hat{\Phi}$.

It can be easily seen that D is the first column in $\hat{\Phi}$. From the Markov parameters in $\hat{\Phi}$, the following Hankel matrices can be defined

$$\mathcal{H}(0) = \begin{bmatrix} CL & CFL & CF^2L & \dots \\ CFL & CF^2L & CF^3L & \dots \\ \vdots & \vdots & \ddots & \end{bmatrix} = \mathcal{OC}, \tag{6.14}$$

$$\mathcal{H}(1) = \begin{bmatrix} CFL & CF^2L & CF^3L & \dots \\ CF^2L & CF32L & CF^4L & \dots \\ \vdots & \vdots & \ddots & \end{bmatrix} = \mathcal{O}F\mathcal{C}, \tag{6.15}$$

where $\mathcal{O}$ is the observability matrix and $\mathcal{C}$ is the controllability matrix of the dynamical system in (6.7) with ν_x as its input.
Using singular value decomposition, $\mathcal{H}(0)$ can be decomposed as

$$\mathcal{H}(0) = U\Sigma V^T = \begin{bmatrix} U_+ & U_- \end{bmatrix} \begin{bmatrix} \Sigma_+ & 0 \\ 0 & \Sigma_- \end{bmatrix} \begin{bmatrix} V_+^T \\ V_-^T \end{bmatrix}. \tag{6.16}$$

The subscript '+' denotes the singular values above a specified threshold and '−' for the singular values below the threshold. The desired degree of the estimated system can be decided from these dominant singular values.
The Observability and Controllability matrices can be split as follows ([20] presents other ways in which the matrices can be split)

$$\mathcal{O} = U_+\Sigma_+^{\frac{1}{2}}, \tag{6.17}$$

$$\mathcal{C} = \Sigma_+^{\frac{1}{2}}V_+^T. \tag{6.18}$$

Now using the Hankel matrix $\mathcal{H}(1)$, the matrix F can be obtained as

$$\begin{aligned} \mathcal{H}(1) = \mathcal{O}F\mathcal{C} &= U_+\Sigma_+^{\frac{1}{2}}F\Sigma_+^{\frac{1}{2}}V_+^T \\ \implies F &= \Sigma_+^{-\frac{1}{2}}U_+^T\mathcal{H}(1)V_+\Sigma_+^{-\frac{1}{2}}. \end{aligned} \tag{6.19}$$

Since $\mathcal{C}$ is the controllability matrix of (6.7), L can be obtained from the first few columns of $\mathcal{C}$ and similarly C can be obtained from the first few rows of $\mathcal{O}$ based on the dimensions. Further H and G can be obtained from L as $L = [H\ G]$. The Kalman filter gain $K = G$. Now, the state matrix A and the input matrix B can be estimated from F and H matrices respectively as

$$F = A - KC \implies A = F + GC, \tag{6.20}$$

$$H = B - KD \implies B = H + GD. \tag{6.21}$$

6.4 Results

The algorithm extended to include colored noises in OKID/ERA framework has been used to estimate the disturbance cross transfer function between the VCM voltage input of the track seeking actuator and the PES of the track following actuator in a multi actuator HDD.

Fig.6.1 shows three seeking wavelet inputs applied one after the other in the top plot. This series forms the control input to the DSA in the track seeking mode. As both the DSAs in the multi actuator drive are mounted on the same pivot timber, the large movements of the track seeking actuator cause vibrations to transfer through the pivot timber and affect the track following DSA. For this estimation, the track following DSA is turned off so that its free response can be used to estimate the cross transfer function. The PES is collected during the excitation and used for the estimation. The sampling frequency in this case was 38520 Hz. The bottom plot of fig.6.1 shows the PES as a function of time. Fig.6.2 shows the frequency response plot of the coloring process which was used as the filter to generate the colored noise. A Gaussian Noise signal with zero mean and unit variance is passed through this filter to generate the colored noise.

A total of 1600 measurements were used and p was chosen to be 800. Ten most dominant singular values of the Hankel matrix were used for the estimation which gave a tenth order plant estimate. Fig.6.4 shows frequency response plots of the actual and estimated plots. It is known from experiments that the low frequency components of the seeking signal have minimal effect on the PES of the track following actuator but the high frequency components, typically more than 2000 Hz have a large impact on the PES and as a result, the Signal to Noise Ratio (SNR) is high in the high frequency region. Due to high SNR at high frequencies, the estimate of the disturbance transfer function at higher frequencies is more accurate than the low frequency region. Despite a bad fit in the low frequency region, the estimated models are still acceptable as frequency response magnitudes in this region are very small and their contribution towards PES predictions remains very small. This can also be seen from Fig.6.5 that the actual and the estimated PES match well.

6.5 Summary

In this chapter, OKID/ERA algorithm was extended to include colored noises. The algorithm was used to estimate the disturbance cross transfer function between the voice coil motor input of the track seeking actuator and the output of the track following actuator in a multi actuator HDD. The input voltage to the VCM actuator and the measured PES, which is the combination of the actual error corrupted by colored noise, were used to estimate the disturbance process.

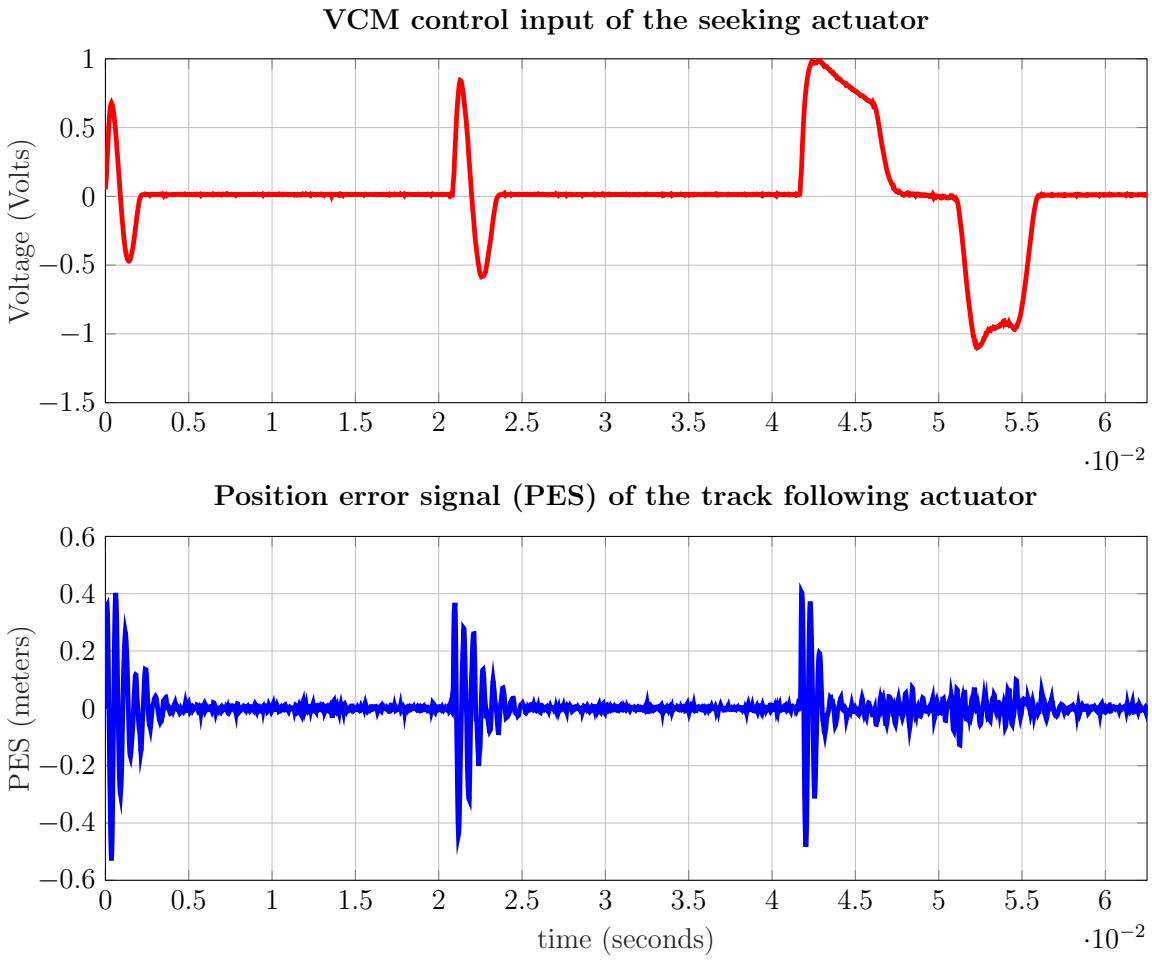

Figure 6.1: Three seeking input voltage pulses applied one after the other to the Voice Coil Motor actuator of the seeking actuator (top) and the corresponding PES measured at the head of the track following actuator with no input voltage (bottom) as a function of time.

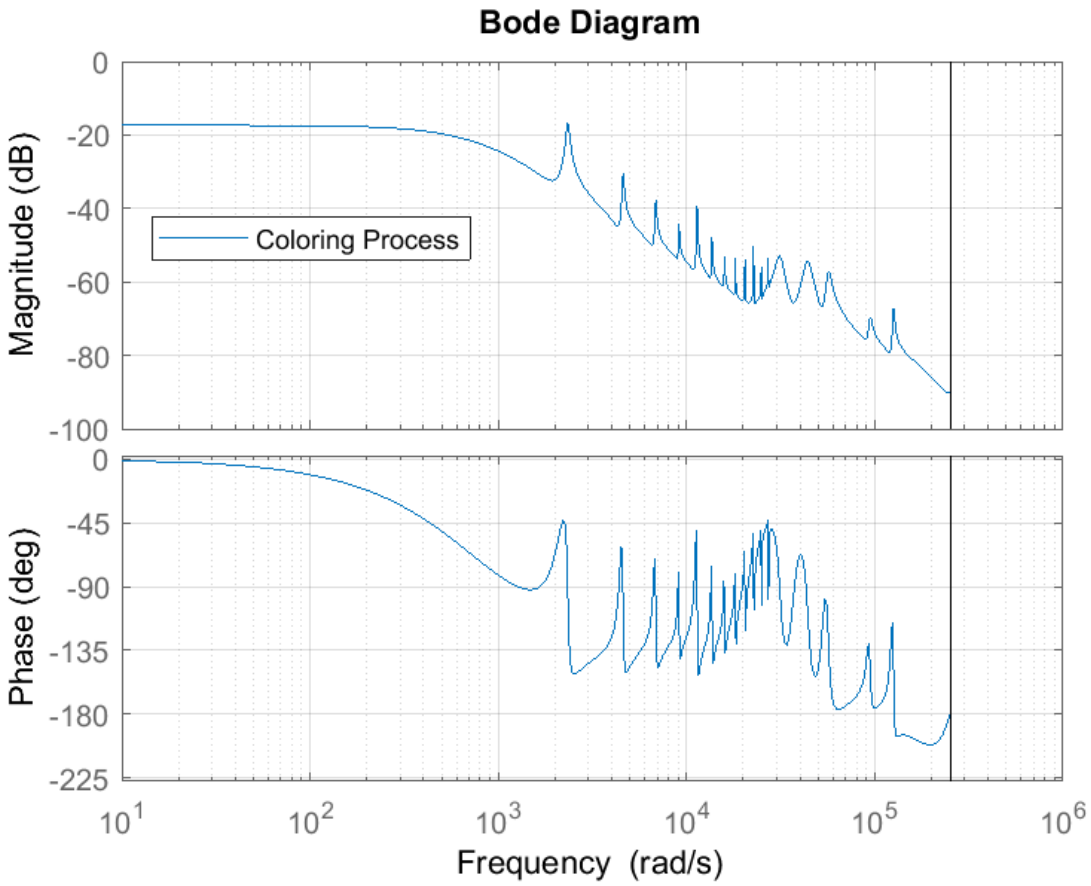

Figure 6.2: Frequency response plot of the coloring dynamics used as filter to generate the colored noise.

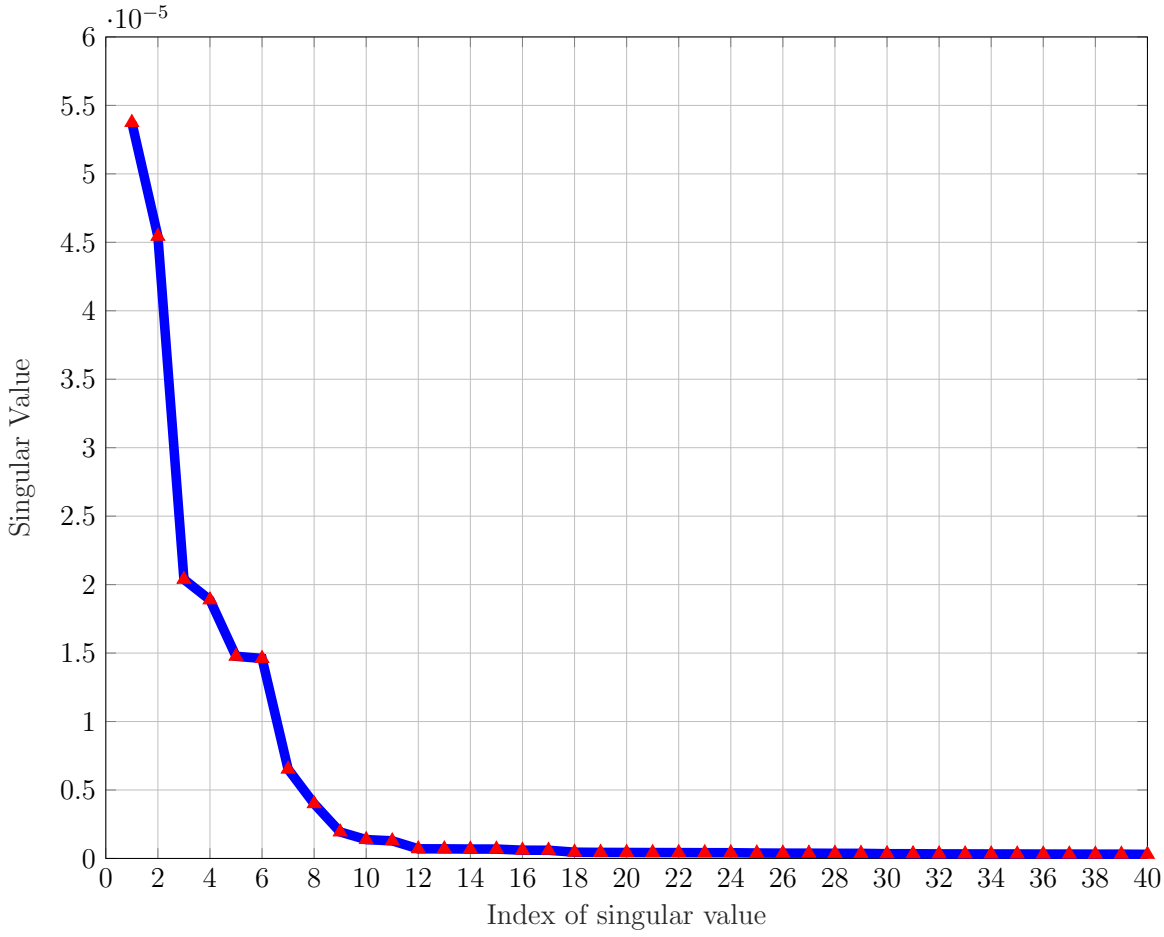

Figure 6.3: Singular values of the Hankel matrix in (6.14) with the highest magnitudes (40 most prominent).

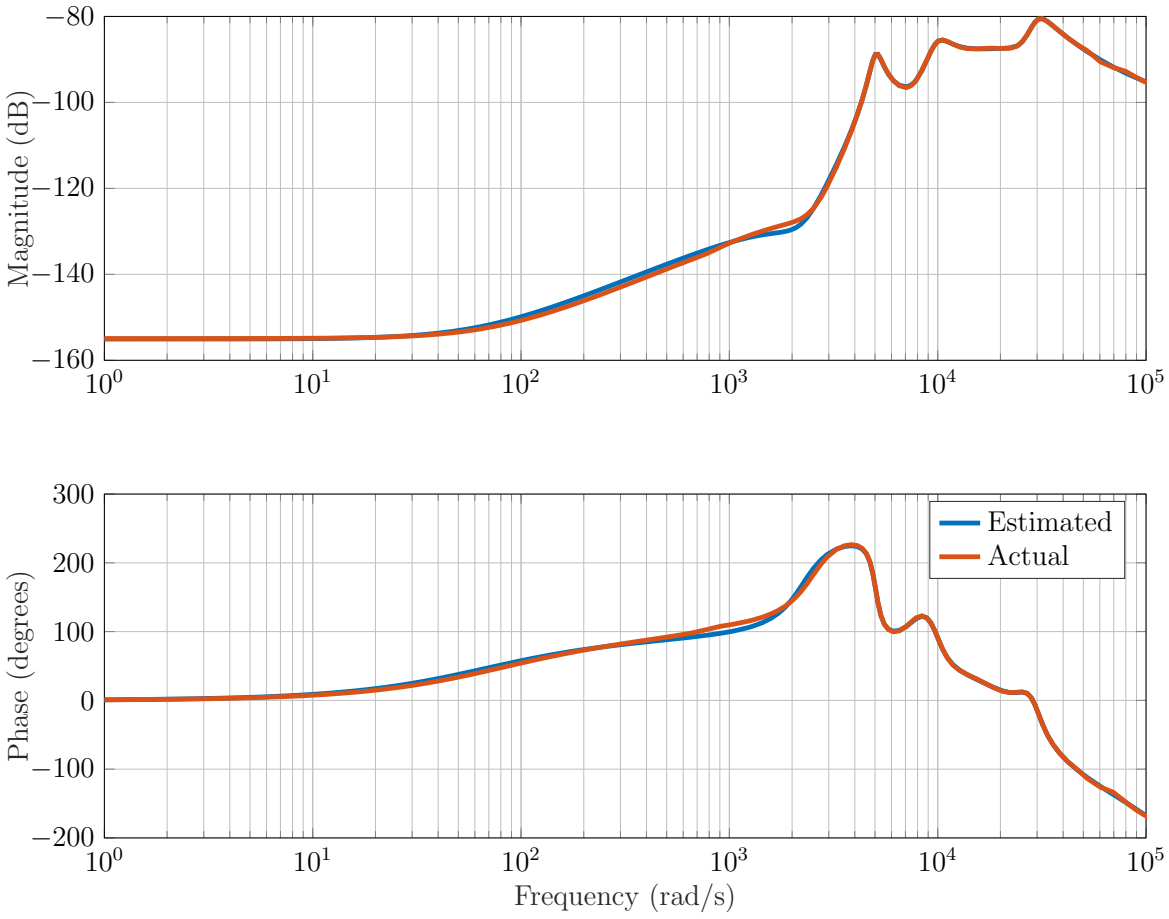

Figure 6.4: Frequency responses of the actual (red) and the estimated plants (blue) of the cross disturbance process.

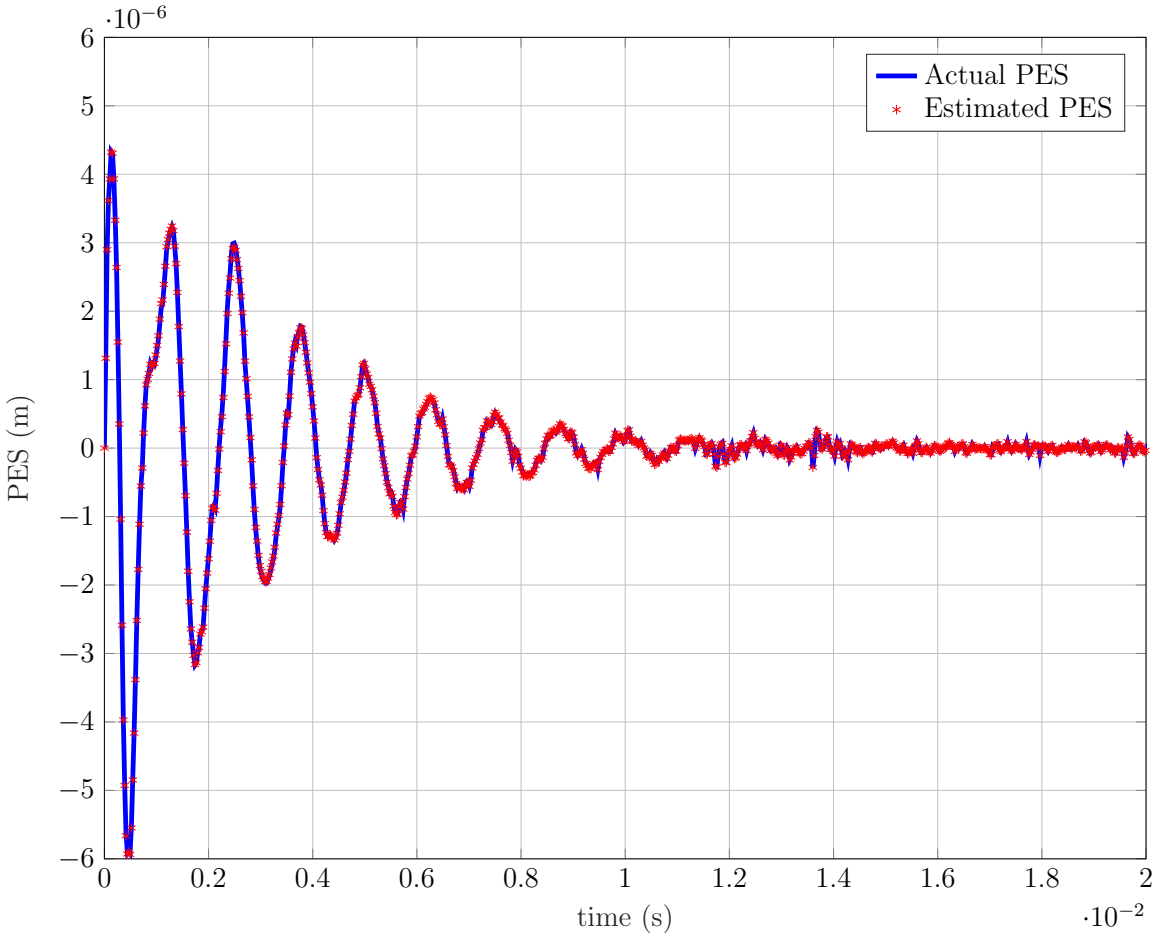

Figure 6.5: Plot showing the actual PES used to estimate and the estimated PES obtained by filtering the VCM voltage in Fig.6.1 with the estimated plant in Fig.6.4.

Chapter 7

Data-Driven Strictly Positive Real System Identification with prior System Knowledge

7.1 Introduction

Strictly Positive Real (SPR) transfer functions are important in engineering, especially in passivity theory for circuit analysis and adaptive control. In passivity theory, storage functions and Lyapunov functions are connected by the positive real lemma [71]. Passivity theory and hyper-stability help analyze the stability of parameter adaptation algorithms, which need an SPR condition [25].

Series-parallel adaptation algorithms (equation error methods) naturally meet the SPR condition but can produce biased plant estimates with measurement noise. Parallel predictors (output error methods) handle measurement noise better but can have stability issues. Stability in this case requires the transfer function with only the system's poles to be SPR, which isn't always the case. To solve this, a pre-filter is designed to make sure the system, formed by the ratio of the pre-filter and the system's poles, meets the SPR condition. Designing this filter is complex and often relies on prior system knowledge. This complexity drives the need for an algorithm to identify SPR systems.

In many cases, frequency response measurements of plants can be obtained, particularly when the plants are stable, and these measurements can be used to build plant models, which can serve as initial conditions for adaptation algorithms. However, even with accurate frequency domain data, obtaining good plant estimates is difficult. Sometimes, even when the plant is known to be stable, least squares fits [21] might yield an unstable estimate. Time domain system identification algorithms, such as the series-parallel adaptation algorithm, generally avoid this issue as they compare error signals at each time instant, typically converging to a stable model if the plant being identified is stable. Therefore, frequency domain system identification techniques and time domain techniques like OKID/ERA can

be combined to achieve SPR estimates.

This chapter presents a systematic methodology to obtain SPR estimates by integrating time domain response data and frequency response measurements of the system of interest. Initially, time domain data can be used to estimate the system using OKID/ERA or the series-parallel adaptation algorithm. This initial estimate may not be SPR, but its poles and zeros would be close to the actual values. A time domain method is chosen over a frequency domain method as they typically produce stable estimates from input-output time series for stable plants while least squares fit from frequency responses might produce unstable estimates as well. In the next step, the poles of the estimated system are used to construct Generalized Orthonormal Basis Functions (GOBFs). The actual system is assumed to be a linear combination of these GOBFs, and their coefficients are determined to minimize the $\mathcal{L}_2$ norm of the error between the actual frequency response measurements and the estimated system's frequency response. The SPR condition is converted to a finite set of linear inequality constraints evaluated at various frequencies, which are incorporated into the optimization problem. As this is a data-driven technique, the SPR constraint is enforced at a finite set of frequencies, and the algorithm's confidence level can be enhanced by using a large set of frequency response data. A simple second order example will be used to demonstrate the method and explain its shortcomings. Next, a non-convex method utilizing only the frequency domain data with less stricter assumptions about the poles of the system will be presented so that an estimate can be produced only from the frequency response data without having to obtain time domain response data. We will demonstrate this method with a practical example from HDD adaptive control problem [11].

Motivating Example

To further motivate the necessity of SPR identification, example of an adaptive control algorithm from Multi Actuator DSA HDD from [11] is used. In a Multi Actuator DSA HDD the most prominent disturbance is the the cross disturbance between from the track seeking actuator to the track following actuator as both the actuators operate on the same pivot timber. In this application, an adaptive controller C as shown in Fig. 7.1 is used to adapt to the uncertainties in the estimates of the closed loop disturbance and compensate for them.

In fig. 7.1, u_{v2} represents the VCM voltage input of the track seeking actuator, Q represents the closed loop disturbance transfer function which is equal to the open loop disturbance transfer function multiplied by the sensitivity transfer function S, d_s represents the residual disturbance signal after the feedback controller compensates for the disturbance, y represents the actual output of the track following DSA, r_o represents the runout, r_{os} represents the filtered runout through the sensitivity transfer function S, e represents the PES from the track, K_m represents the SISO feedback controller of micro PZT, $\hat{G}_m$ represents the estimate of micro PZT, $R_{r_v \to y}$ represents the transfer function from the reference signal of the VCM r_v to the output y and $R_{r \to y}$ represents the the transfer function from the reference to the DSA r to the output y. Here $\hat{Q}$ is the estimate of the closed loop disturbance transfer function obtained designed from the frequency response measurements available from all the

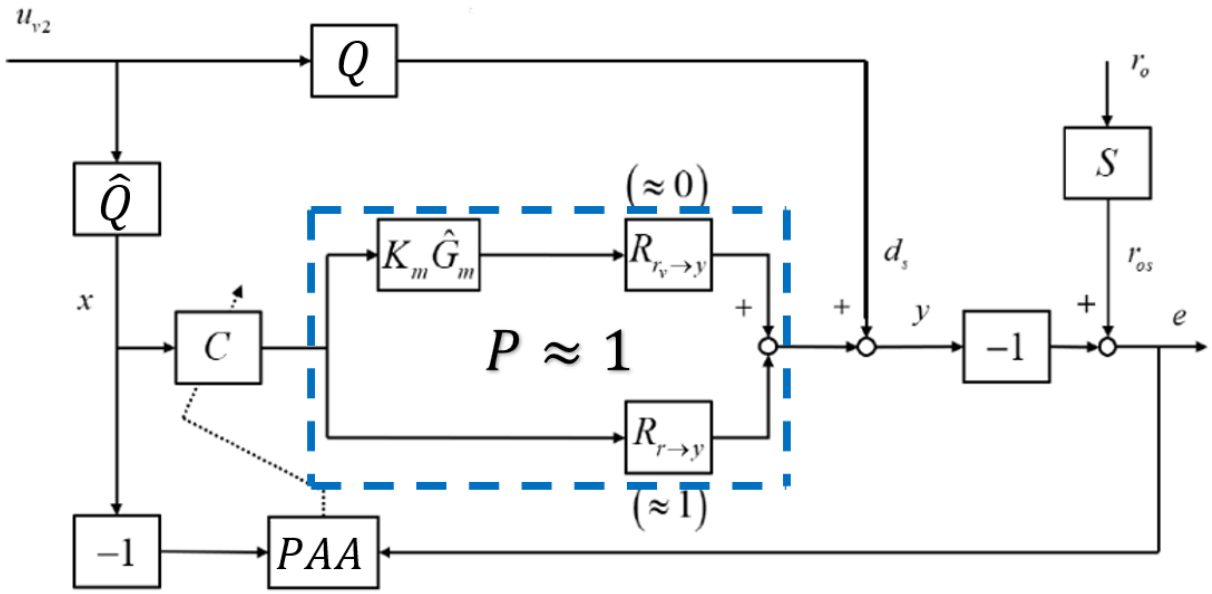

Figure 7.1: Simplified block diagram of the adaptive control structure for disturbance rejection in multi-actuator hard disk drives

plants in the system. In this control structure, block shown in blue is always close to 1 at all frequencies (see [11]). Therefore the block diagram can be further simplified to the structure shown in fig. 7.2.

The block diagram shown in fig.7.2 can be recognised to be using the parallel predictor for the parameter adaptation algorithm (PAA). This adaptive control scheme has Q in the primary path and $\hat{Q}$ and C in the secondary path. For the stability of the adaptation algorithm, it is essential that the ratio $\hat{Q}/Q$ is strictly positive real (SPR). The design of $\hat{Q}$ has mostly been an iterative process and heuristic. The algorithms presented in this chapter could be used as a handy tool to design filters such as $\hat{Q}$ from the frequency response of Q if we know a priori the structure of Q. The a priori knowledge of Q can be obtained from the time series data of its inputs and output using any of the standard time domain system identification techniques like OKID/ERA. Other techniques such as the equation error method, ARMA or ARMAX models can also be used for the time series identification method. Once the poles are obtained, we will only pick the stable poles of the system and project the remaining into the unit circle and use them for constructing the filters. Next, a complete system identification technique in frequency domain will be posed as a non-convex optimization problem.

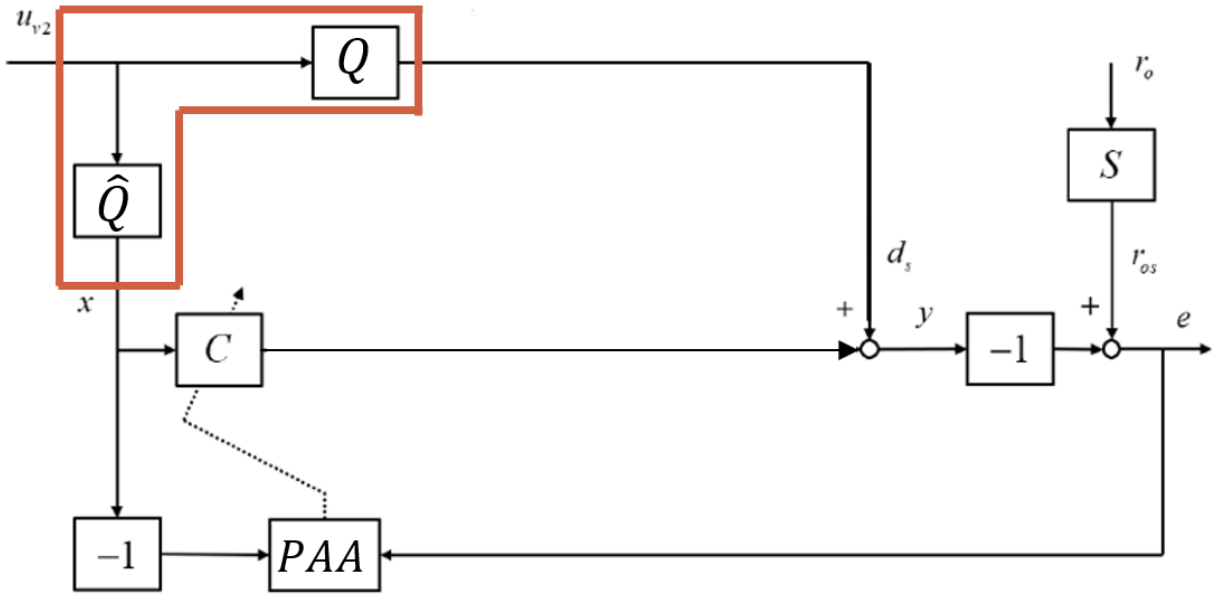

Figure 7.2: Further simplified block diagram of the structure in fig.7.1

7.2 Generalized Orthonormal Basis Functions (GOBF)

In this section a brief description of Generalized Orthonormal Basis Functions (GOBFs) will be provided. Orthogonal functions were used in [46, 92] for the identification of a finite sequence of expansion coefficients. The idea is based on the fact that every stable system can be represented as a unique series expansion in terms of a pre-chosen basis. A representation with finite number of terms can serve as an approximate model. Historically, the coefficients of the series expansion have been estimated from the input-output time domain data, but we will show that they can also be estimated from the frequency domain data.

Consider a stable discrete time system $G(q)$ where q^{-1} is the delay operator (similar to z^{-1} in z-domain but the term q is used in cases where we want to use it as an operator). Then there exists a unique series expansion in terms of orthonormal basis function $\{f_k(q)\}_{k=1,2,...}$ according to [46, 92] such that

$$G(q) = \sum_{k=1}^{\infty} a_k f_k(q), \tag{7.1}$$

where $\{a_k\}_{k=1,2,...}$ are real coefficients. Since $\{f_k(q)\}_{k=1,2,...}$ are orthonormal, they satisfy the

following property

$$\frac{1}{2\pi}\int_{-\frac{\pi}{T_s}}^{\frac{\pi}{T_s}} f_l(e^{j\omega T_s})f_m(e^{-j\omega T_s})d\omega = \begin{cases}1 & if \quad l=m\\ 0 & if \quad l\neq m.\end{cases} \tag{7.2}$$

If we are choosing a finite number of basis functions, it is profitable to use filters that reflect the dominant dynamics of the system i.e., the poles. The poles of the estimated system from OKID/ERA will be used to build these filters. The most well known filters are Laguerre filters for well damped poles and Kautz filters for underdamped poles.
Laguerre filters:

$$L_k(q,a) = \frac{\sqrt{1-a^2}}{q-a}\left[\frac{1-aq}{q-a}\right]^{k-1},\ a\in\mathbb{R}, |a|<1, k\geq 1.$$

Kautz filters:
for odd k

$$\Psi_k(q,a) = \frac{\sqrt{1-c^2}(q-b)}{q^2+b(c-1)q-c}\left[\frac{-cq^2+b(c-1)q+1}{q^2+b(c-1)q-c}\right]^{\frac{k-1}{2}}, \tag{7.3}$$

for even k

$$\Psi_k(q,a) = \frac{\sqrt{(1-c^2)(1-b^2)}}{q^2+b(c-1)q-c}\left[\frac{-cq^2+b(c-1)q+1}{q^2+b(c-1)q-c}\right]^{\frac{k-2}{2}}. \tag{7.4}$$

$$b,c\in\mathbb{R}, |b|<1, |c|<1, k\geq 1.$$

Here a, b and c are the parameters that decide the location of the poles.

7.3 SPR System Identification as Convex Problem with Prior Knowledge

In this section, utilizing the GOBFs, identification of the SPR transfer function will be posed as an optimization problem. The problem that we are trying to solve here is to find an estimate $\hat{G}$ of given frequency response of a transfer function G such that $W\hat{G}$ is SPR for a given filter W. Note that if $W=1$, then the estimate $\hat{G}$ would be an SPR estimate of G and if $W=\frac{1}{G}$, then the estimate $\hat{G}$ would be such that $\frac{\hat{G}}{G}$ is SPR, which is the necessary condition in the motivating example of adaptive control.
Various texts provide various definitions of SPR transfer functions, so, to be consistent, we present the definition from [30].
Definition: A discrete time transfer function $G(q)$ is said to be SPR iff

1. $G(q)$ is Schur,
2. $Re[G(e^{j\omega T_s})] > 0\ \forall\ \omega\in[-\frac{\pi}{T_s},\frac{\pi}{T_s}]$.

Note that in discrete time, a transfer function can be SPR only if its relative degree is zero. The first condition is automatically satisfied when we choose the GOBFs to have stable poles. After choosing the GOBFs, we proceed with formulating the identification problem as a convex optimization problem. The transfer function estimate can be approximated as a linear combination of the GOBFs as

$$\hat{G}(q) = \sum_{i=1}^{p} a_i f_i(q) = \theta^T \Phi(q), \tag{7.5}$$

where $f_i(q)$ is either a Laguerre or a Kautz filter and the constants a_i are to be determined. Let $G(e^{j\omega_t T_s})$ denote the frequency response measurements of the system at N different frequencies ω_n for $t = 1, ..., N$. T_s denotes the sampling time of the system. The frequency responses of the estimate can be found by evaluating at the available frequencies as

$$\hat{G}(e^{j\omega_t T_s}) = \sum_{i=1}^{p} a_i f_i(e^{j\omega_t T_s}) = \theta^T \Phi(e^{j\omega_t T_s}), \tag{7.6}$$

where $\theta = [a_1 \ a_2 \ \dots \ a_p]^T$ and $\Phi(q) = [f_1(q) \ f_2(q) \ \dots \ f_p(q)]^T$.
The SPR constraint on $W\hat{G}$ can be written as $Re(W\hat{G}) \geq \epsilon$ for $\epsilon > 0$ (user defined tolerance) at every available frequency. This can be transformed into a set of linear constraints in θ as follows

$$\begin{aligned} & W(e^{j\omega_t T_s})\hat{G}(e^{j\omega_t T_s}) + W^*(e^{j\omega_t T_s})\hat{G}(e^{-j\omega_t T_s}) \geq \epsilon \\ \implies & \sum_{i=1}^{p} a_i Re(W(e^{j\omega_t T_s}) f_i(e^{j\omega_t T_s})) \geq \epsilon \\ \implies & \theta^T Re(W(e^{j\omega_t T_s})\Phi(e^{j\omega_t T_s})) \geq \epsilon, \end{aligned} \tag{7.7}$$

which forms a set of linear inequality constraints.
The problem of finding θ can now be formulated as a convex quadratic program by minimizing the squared distance between the estimate's frequency response and the frequency response of the actual plant with the SPR constraints formulated as linear constraints as follows

$$\begin{aligned} \hat{G}^* = \underset{\hat{G}}{\text{argmin}} \quad & \sum_{t=1}^{N} ||G(e^{j\omega_t T_s}) - \hat{G}(e^{j\omega_t T_s})||_2^2 \\ \text{subject to} \quad & Re(W(e^{j\omega_t T_s})\hat{G}(e^{j\omega_t T_s})) \geq \epsilon, \ t = 1, \dots, N, \end{aligned} \tag{7.8}$$

which from (7.5) is equivalent to

$$\begin{aligned} \theta^* = \underset{\theta}{\text{argmin}} \quad & \sum_{t=1}^{p} ||G(e^{j\omega_t T_s}) - \theta^T \Phi(e^{j\omega_t T_s})||_2^2 \\ \text{subject to} \quad & \theta^T Re(-W(e^{j\omega_t T_s})\Phi(e^{j\omega_t T_s})) \leq -\epsilon, \ t = 1, \dots, N. \end{aligned} \tag{7.9}$$

7.4 SPR System Identification as a Non-Convex Problem

In this section, identification of continuous time SPR transfer functions with relative degree zero from the frequency response measurements of the system will be posed as a non-convex optimization problem. To be consistent with the definitions of SPR transfer functions, we present the definition from [30].
Definition: A continuous time transfer function $G(s)$ is said to be positive real iff

1. $G(s)$ is Hurwitz
2. $Re[G(j\omega)] > 0 \; \forall \; \omega$

Note that in continuous time, a proper transfer function can be SPR only if its relative degree is at most one. But, in this chapter, we will restrict the identification problem to systems with relative degree zero.
Given frequency response data of a continuous time plant G (we are using the same notation as before to be consistent, but the discrete time plant is replaced by a continuous time plant), we want to find an estimate $\hat{G}$ of G such that $W\hat{G}$ is SPR. Here W is a user defined transfer function or numerically shaped weight. It can be seen that if $W = 1$, $\hat{G}$ becomes the SPR estimate of G and if $W = \frac{1}{G}$, $\hat{G}$ becomes the estimate such that $\frac{\hat{G}}{G}$ is SPR.
The optimization problem in it's raw form can be expressed as follows

$$\begin{aligned} J^* = \min_{b,a} \quad & \sum_{t=1}^{N} ||b(j\omega_t) - a(j\omega_t)G(j\omega_t)||_2^2 \\ \text{s.t} \quad & a(s) \text{ Hurwitz} \\ & Re(W(j\omega_t)\frac{b(j\omega_t)}{a(j\omega_t)}) > 0 \; \forall t = 1, \ldots, N, \end{aligned} \tag{7.10}$$

where $\hat{G}(s) = \frac{b(s)}{a(s)}$.
In the above optimization problem, the cost function J is motivated from the regular least squares formulation of system identification from frequency domain data. For equally spaced frequencies, this can be viewed as the equivalent of an H_2 norm. Also the SPR constraints have been evaluated only at a finite number of points instead of infinite points. This approximation is motivated by the fact that as the number of frequency response measurements increase, the confidence level of the fit increases.
Estimates can also be obtained from plant measurements obtained from multiple similar plants (n plants) by slightly modifying the cost function as follows

$$\begin{aligned} J^* = \min_{b,a} \quad & \sum_{t=1}^{N}\sum_{i=1}^{n} ||b(j\omega_t) - a(j\omega_t)G_i(j\omega_t)||_2^2 \\ \text{s.t} \quad & a(s) \text{ Hurwitz} \\ & Re(W_i(j\omega_t)\frac{b(j\omega_t)}{a(j\omega_t)}) > 0 \; \forall t = 1, \ldots, N, \forall \; i. \end{aligned} \tag{7.11}$$

In the remainder of the section, we will recast the constraints to obtain a more suitable form to be given to an optimization solver. Let us first consider the Hurwitz constraint. The following theorem will be used to derive a sufficient condition for ensuring that $a(s)$ is Hurwitz.

Theorem 7. *Let $h(s) = \frac{a(s)}{p(s)}$ for any chosen Hurwitz polynomial $p(s)$. If $\forall\omega$, $Re(h(j\omega)) > 0$, then $a(s)$ is Hurwitz.*

Proof. As $p(s)$ is Hurwitz, the number of poles of $h(s)$ on the open right half plane is 0. And since $Re(h(j\omega)) > 0$ there will be no encirclements of the origin, which ensures that number of zeros of $h(s)$ on the open right half plane is 0 from Nyquist theorem. Therefore since $a(s)$ does not have any unstable zeros, it is Hurwitz. □

This condition is a set of linear convex constraints when evaluated at a finite set of frequencies. Any prior knowledge of $a(s)$ can be used to construct $p(s)$. But the knowledge need not be as precise as in the previous convex formulation using GOBFs where the poles are to be known somewhat precisely.
The second set of constraints will be recast into a set of linear inequality constraints and nonlinear equality constraints.

$$\hat{G}(s) = \frac{b(s)}{a(s)} = \frac{b(s)a(-s)}{a(s)(-s)} = \frac{k(s)}{a(s)a(-s)}, \tag{7.12}$$

where $k(s) = b(s)a(-s)$. Here, the denominator is always a positive quantity at all frequencies. Hence the second set of constraints can be recast as

$$W(j\omega_t)k(j\omega_t) \geq 0, \tag{7.13a}$$
$$b(s)a(-s) = k(s). \tag{7.13b}$$

The second equation in (7.13) is a set of nonlinear equality constraints obtained by comparing the coefficients of polynomials $b(s)$,$a(s)$ and $k(s)$.
Now the optimization problem in (7.10) with the recast constraints becomes

$$\begin{aligned} J^* = \min_{b,a} \quad & \sum_{t=1}^{N} ||b(j\omega_t) - a(j\omega_t)G(j\omega_t)||_2^2 \\ \text{s.t} \quad & Re(h(j\omega_t)) > 0, \forall \; t = 1, ..., N, \\ & b(s)a(-s) = k(s). \end{aligned} \tag{7.14}$$

The decision variables can be reduced in the case when $W = 1$, i.e., for the problem of identifying SPR transfer functions. The following theorem from [48] can be used for the reduction.

Theorem 8. *Let $G(s) = \frac{b(s)}{a(s)}$. Given a Hurwitz polynomial $a(s) = s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0 = \rho_a(s)\hat{a}$ of degree $n \geq 2$, the set of polynomials $b(s) = b_n s^n + b_{n-1}s^{n-1} + \cdots + b_1 s + b_0 = \rho_b(s)\hat{b}$ that make $G(s)$ SPR can be parametrized using $n+1$ real numbers $k_1,\ k_2, \ldots\ k_{n+1}$, which are chosen such that*

$$k(s) = \sum_{i=1}^{n+1} (-1)^{i-1} s^{2(i-1)} k_i = \rho_k(s)\hat{(k)}$$
$$k(j\omega) > 0, \forall \omega \tag{7.15}$$

. Moreover $\hat{b} = X^{-1}\hat{k}$ where

$$X = \begin{bmatrix} 1 & 0 & 0 & 0 & \ldots & 0 \\ -a_{n-2} & a_{n-1} & -1 & 0 & \ldots & 0 \\ -a_{n-4} & a_{n-3} & -a_{n-2} & a_{n-1} & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ldots & \vdots \\ a_0 & -a_1 & a_2 & -a_3 & \ldots & \vdots \\ 0 & 0 & -a_0 & a_1 & \ldots & -a_2 \\ 0 & 0 & 0 & 0 & \ldots & a_0 \end{bmatrix} \tag{7.16}$$

The matrix X is obtained by comparing the powers on both sides of $b_e(s)a_e(s) - b_o(s)a_o(s) = k(s)$. Here the subscript $'e'$ denotes the even part and $'o'$ denotes the odd part of the polynomials.
The problem of identifying an SPR transfer function $\hat{G}$ translates to the problem of finding a hurwitz $a(s)$ and an even $k(s)$ such that $k(j\omega) > 0\ \forall\ \omega$, which can be expressed as follows

$$\begin{aligned} J^* = \min_{\hat{a},\hat{k}} \quad & \sum_{t=1}^{N} ||\rho_b(j\omega_t)\hat{b}(\hat{a},\hat{k}) - \rho_a(j\omega_t)\hat{a}G(j\omega_t)||_2^2 \\ \text{subject to} \quad & Re(h(j\omega_t)) > 0, \\ & k(j\omega_t) > 0\ t = 1, \ldots, N, \end{aligned} \tag{7.17}$$

which is equivalent to

$$\begin{aligned} J^* = \min_{\hat{a},\hat{k}} \quad & \sum_{t=1}^{N} ||\rho_b(j\omega_t)X(\hat{a})^{-1}\hat{k} - \rho_a(j\omega_t)\hat{a}G(j\omega_t)||_2^2 \\ \text{subject to} \quad & Re(h(j\omega_t)) > 0, \\ & k(j\omega_t) > 0\ t = 1, \ldots, N. \end{aligned} \tag{7.18}$$

This is an optimization problem with a nonconvex objective and convex constraints. Similarly, the optimization problem for the ratio to be SPR can also be posed.

7.5 Simulations

In this section, the SPR identification algorithms presented in the previous section will be demonstrated on a simple transfer function to demonstrate the and a second example to design filters for adaptive control that satisfy the SPR condition. The first example considers the case when the actual transfer function is SPR and the second example considers the case in which the actual transfer function is not SPR.

Example 1 using GOBFs: In this example, a simple second order discrete time transfer function with a sampling time of 1 second will be considered which is SPR. The transfer function corresponding to the frequency response considered is

$$G(q) = \frac{q^2 + 0.2q + 0.3}{q^2 + 0.4q + 0.5}. \tag{7.19}$$

100 equally spaces frequencies from 0 to 3 rad/s were considered and the frequency responses for simulation were obtained by evaluating the transfer function in (7.19) at those frequencies. Kautz filters were built with $c = -0.2$ and $b = -0.33$. It can be seen that the values used for b and c do not correspond to the actual poles of the system and yet the simulation results will show that a good fit can be obtained. N was chosen to be 8 for the algorithm (this would make the order of the estimated plant equal to 16). The Bode plot in Fig.7.3 and Nyquist plot in Fig.7.4 show a close fit and that the SPR conditions are also being satisfied.

Despite a good fit, we can notice that for a second order transfer function, when the poles are not accurate, we required a 16^{th} order transfer function to get a good approximation. This might not be acceptable in some cases where there is a limit on the computational capabilities.

Example 2 using Non-Convex formulation: In this example, the filter $\hat{Q}$ will be designed for the adaptive control algorithm introduced in subsection 7.1 such that the SPR condition on the ratio $\hat{Q}/Q$ is satisfied. We will consider 4 different plant frequency responses for the closed loop disturbance cross transfer functions and try to obtain a filter that satisfies the SPR condition for all of the frequency responses. The problem will be set up in the style of Eq. (7.11) with the reformulated constraints following Eq. (7.11). $p(s) = (s+1)^7$ and a 7^{th} order transfer function is estimated from the frequency response. The results are plotted in Fig. 7.5 and Fig. 7.6. It can be seen that the estimate is close to the frequency responses from Fig. 7.5 and that the estimate satisfies the SPR constrain i.e., the nyquist plot of the ration $\frac{\hat{Q}}{Q}$ is always on the right of the imaginary axis.

7.6 Conclusions

In this chapter, a data-driven technique for estimating strictly positive real transfer functions from its frequency response with prior knowledge of the system is presented. The problem is first posed as a convex optimization problem with a quadratic objective function and

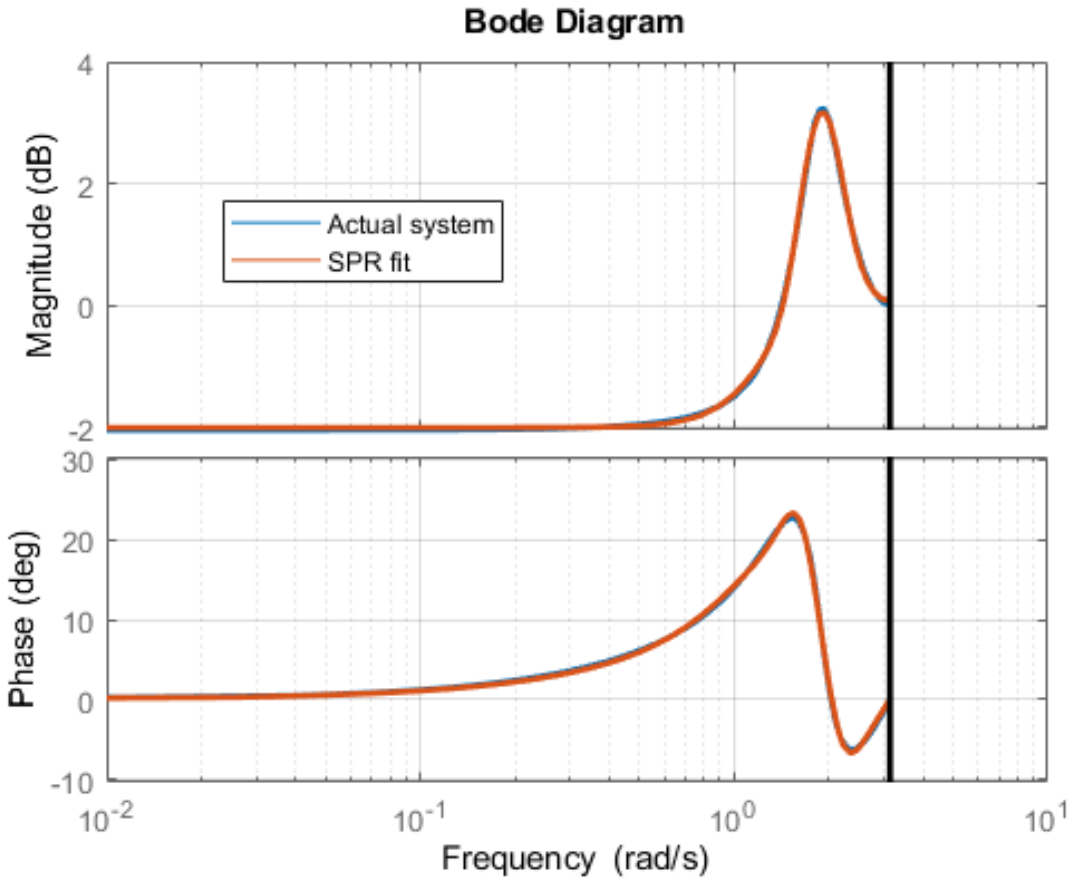

Figure 7.3: Frequency responses of the actual plant shown in Eq. (7.19) and the estimated 16^{th} order plant.

linear constraints in which prior knowledge of the poles was assumed. A simple example was provided on which simulations were done. Next, a non-convex algorithm was presented in which the knowledge of the locations of the poles was relaxed to a large extent. This algorithm was used to build SPR condition satisfying filters for adaptive feedorward control of a multi actuator DSA HDD.

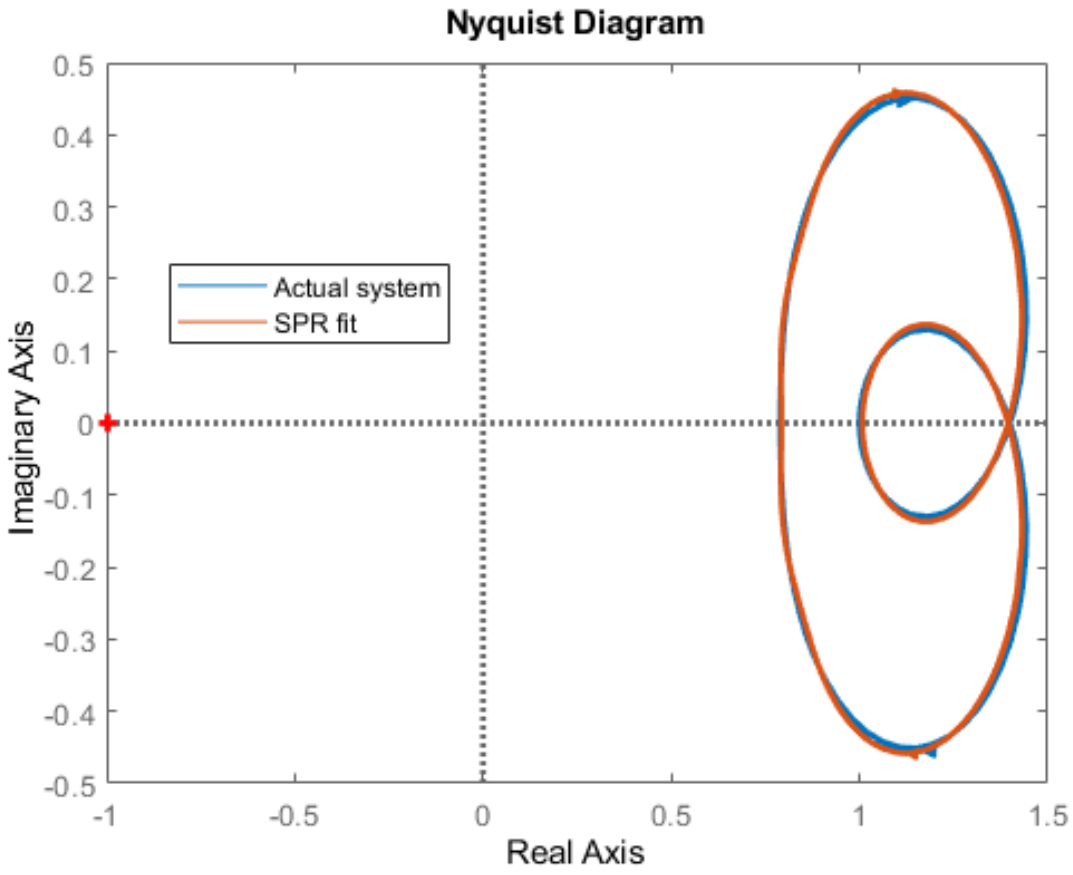

Figure 7.4: Nyquist plot of the actual plant shown in (7.19) and the estimated 16^{th} order plant

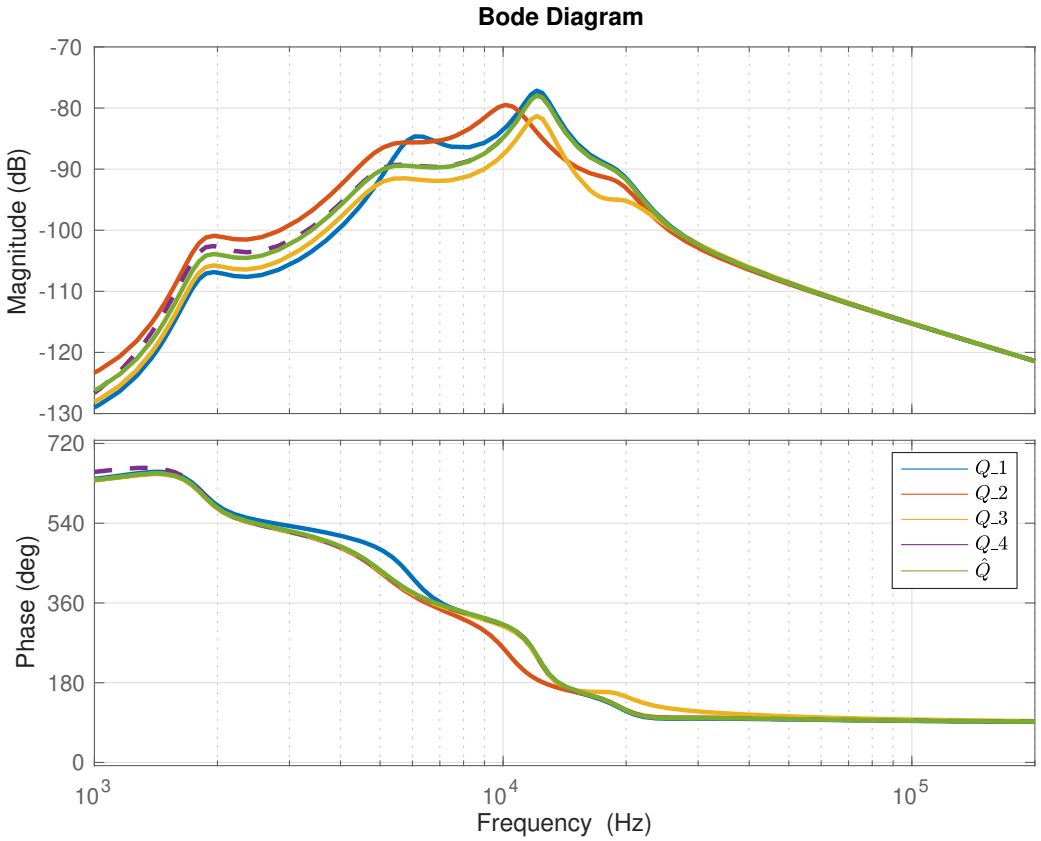

Figure 7.5: Frequency response plots of four transfer functions (shown by solid lines) and their estimate (shown by the dotted line).

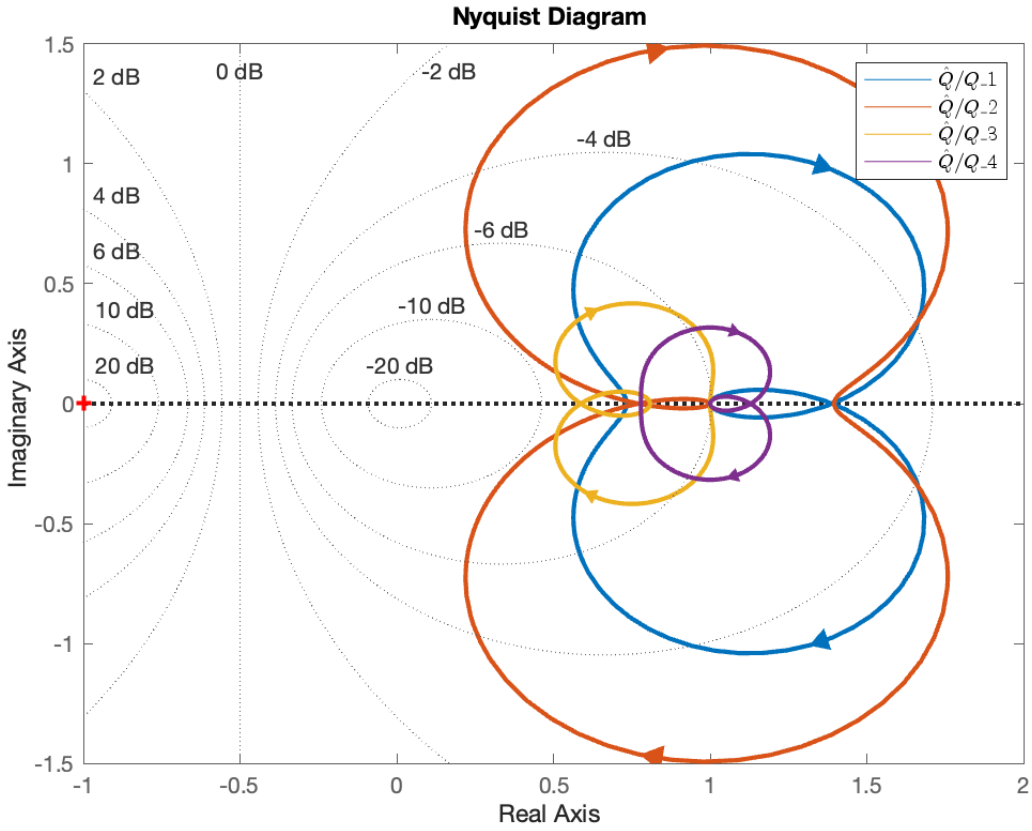

Figure 7.6: Nyquist plots of ratio of the estimate with four transfer functions showing that the estimate satisfies the SPR condition i.e., all the plots are on the right hand side of the imaginary axis.

Chapter 8

Conclusion and Future Work

8.1 Conclusion

In this dissertation, we utilized ideas from machine learning, estimation theory, control theory and optimization to design controllers for two kinds of mechanical systems (a) fully actuated mechanical systems with full state measurement evolving on smooth manifolds (b) flexible manipulators inside HDDs with output measurements. The initial part of the dissertation focused on building potential functions using neural networks and tuning the weights of the neural network to shape them according to the needs of the task using gradient descent. Neural networks were used as they are universal approximators and increasing the size of the network can increase the optimality of potential functions for the task. Two kind of potential functions, a conservative potential function and a dissipative potential function were built using Input Convex Neural Networks (ICNNs). A methodology to obtain spring/elastic forces/wrenches from the conservative potential functions and damping forces/wrenches from dissipative potential functions was presented using ideas from differential geometry. The elastic forces were nonlinear generalizations of proportional control terms and the damping forces were nonlinear generalizations of the derivative terms in PD control. The method was applied to systems evolving on some common manifolds like a sphere, the special orthogonal group and the special euclidean group such as a satellite, an underwater vehicle and a robotic manipulator.

The latter part of the dissertation focused on identification and robust control of a flexible manipulator inside a HDD. Even though, these dynamical systems are LTI systems, control design is extremely challenging due to the extreme amount of precision that is required inside a HDD. To achieve high precision, the control design was split into designing a feedback controller to reject unknown disturbances and then a feedforward controller to reject any expected but unwanted motions. Due to the variability of the plants inside HDDs, the designed controller needs to be robust to plant variations. Since, designing a common controller might be sub-optimal, a methodology to cluster LTI plants by defining the distance between two stable LTI plants using H-norms was presented. Next, a data driven controller

design methodology for designing a common robust controller for the available frequency responses of plants within each cluster was presented. It was demonstrated that any user defined shape can be used as a limit for shaping various important transfer functions like the sensitivity transfer function, error to input transfer function etc. The robustness and disturbance rejection capabilities of the controller were demonstrated.

Next, an introduction to feedforward design methodologies to further attenuate disturbances, by compensating for any known parts of the disturbance was presented. Feedforward design schemes require a good model of the disturbance processes to predict the disturbances and apply control to compensate for the expected motions. One of the challenges with HDDs is that the measurements are corrupted by coloured noises and hence regular system identification might not estimate the systems well without introducing any bias. An extended OKID/ERA technique was developed to learn stable LTI systems in the presence of colored noises with applications to learning disturbance processes in HDDs. A more generalized form of feedforward control is the adaptive feedforward control scheme in which we relax the assumption that we know the parameters of the transfer function but we still attribute a structure to the transfer function and learn the parameters of the system from the data. This control scheme required an SPR condition on some filters used in the control loop for stability. For ensuring stability of the adaptive feedforward control scheme, a system identification technique from the frequency response measurements to design filters that satisfy the SPR condition was developed.

8.2 Future Work

In this section, we present directions in which the presented ideas in this dissertation can be extended.

Output feedback control for systems on manifold: In this dissertation, the developed feedback control for mechanical systems evolving on manifolds utilized the full configuration and the velocities of the systems, but, in general the full configuration and velocities of the system are not available and a linear/nonlinear function of these and the input is available to us as the measurement. In such cases, a geometric estimator (typically nonlinear) must be added to the loop to first estimate the configuration and the velocity of the system before feeding them to the potential functions. Design of geometric estimators along with ensuring stability with cascaded estimators is left for work in the future.

Contact Rich applications: The most interesting and important applications of Impedance control of robotic manipulators are in the area of contact rich manipulation where the manipulators can interact smartly with the environment without damaging them. The design of control laws utilizing the designed potential functions for contact rich manipulation is an easy extension and is left for work in the near future.

Integration with barrier functions: The control law for the robotic manipulator was designed such that the end effector's configuration error follows a certain trajectory either for trajectory tracking or set point tracking. Since the designed potential functions are unaware

of the kinematic structure of the manipulator, the control law might command a certain configuration that's unreachable for the robot which means that the robot would get into singular configurations i.e., were the jacobian becomes rank deficient. In such cases, it would be a good idea to integrate the learning process with barrier functions, like [39], to ensure that the robot does not get into singular configurations which would help in shaping the potential functions such that the singular configurations are avoided.

Neural Lyapunov functions: The control laws designed for systems evolving on manifolds were restricted to fully actuated systems and hence the lyapunov functions could be built utilizing the designed potential functions which could prove the stability of the system. This is not the case for underactuated systems and instead of designing these potential functions, one could directly build a class of candidate lyapunov functions parameterized by neural networks and then design a method to pick the parameters in such a way that the candidate function becomes a lyapunov function. This of course is not very straightforward and needs further study.

Clustering nonlinear and unstable LTI systems: A method for clustering stable LTI systems was presented but these are only a small class of dynamical systems. An idea for clustering unstable LTI systems is to design a common stabilizing controller for all of the systems and then cluster the closed loop stable LTI plants. But, a common stabilizing controller might not exist and one might have to resort to another method. The clustering of Linear Parameter Varying (LPV) and nonlinear systems are even more challenging and the study is left for future.

MIMO data drive robust control: HDDs have Multiple Input Single Output (MISO) plants in which the control inputs from three different actuators (in case of a TSA system) attached in parallel are used to move one read/write head and hence generates one PES signal. It was shown that utilizing a H_∞ constraint on at least one of the closed loop transfer function generated a stabilizing data driven controller. This was restricted to MISO systems and extending it to generic Multiple Input Multiple Output (MIMO) systems is left for the future.

In conclusion, this dissertation has demonstrated the potential of integrating learning-based methods with traditional control techniques to address complex control challenges in mechanical systems. The proposed approaches pave the way for future research aimed at developing more intelligent, adaptable, and robust control systems, ultimately contributing to advancements in the field of control engineering.

Bibliography

[1] A. Al Mamun et al. "Digital processing of position error signal in hard disk drives". In: *IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*. Como, Italy: IEEE, May 2004. ISBN: 0-7803-8248-X. DOI: 10.1109/IMTC.2004.1351237.

[2] Brandon Amos, Lei Xu, and J. Zico Kolter. "Input Convex Neural Networks". In: *International Conference on Machine Learning (ICML)*. New York, NY, USA, June 2016. URL: https://api.semanticscholar.org/CorpusID:15898758.

[3] O. Bagherieh. "Estimation, identification and data-driven control design for hard disk drives". In: *PhD thesis, University of California, Berkeley* (2017).

[4] Omid Bagherieh, Prateek Shah, and Roberto Horowitz. "Application of Mixed Data Driven Control Design to Dual Stage Hard Disk Drives". In: *ASME Conference on Dynamics Systems and Control* (2018).

[5] Jens Behrmann et al. *Invertible Residual Networks*. 2019. arXiv: 1811.00995 [cs.LG].

[6] Sanjay P. Bhat and Dennis S. Bernstein. "A topological obstruction to continuous global stabilization of rotational motion and the unwinding phenomenon". In: *Systems & Control Letters* (2000).

[7] James Mark William Brownjohn et al. "Assessment of Highway Bridge Upgrading by Dynamic Testing and Finite-Element Model Updating". In: *Journal of Bridge Engineering* 8.3 (2003). CiteSeerX 10.1.1.194.8245, pp. 162–172. ISSN: 1084-0702. DOI: 10.1061/(ASCE)1084-0702(2003)8:3(162).

[8] F. Bullo and R. M. Murray. *Proportional Derivative (PD) Control on the Euclidean Group*. Technical Report Caltech/CDS 95–010. Available at http://resolver.caltech.edu/CaltechCDSTR:1995.CIT-CDS-95-010. California Institute of Technology, May 1995.

[9] Francesco Bullo and Richard M. Murray. "Tracking for fully actuated mechanical systems: a geometric framework". In: *Automatica* 35.1 (1999), pp. 17–34.

[10] Juan Martin Caicedo, Shirley J. Dyke, and Erik A. Johnson. "Natural Excitation Technique and Eigensystem Realization Algorithm for Phase I of the IASC-ASCE Benchmark Problem: Simulated Data". In: *Journal of Engineering Mechanics* 130.1 (2004), pp. 49–60. DOI: 10.1061/(ASCE)0733-9399(2004)130:1(49).

[11] Zhi Chen, Nikhil Potu Surya Prakash, and Roberto Horowitz. "Adaptive Feedforward Reference Design for Active Vibration Rejection in Multi-Actuator Hard Disk Drives". In: *2022 American Control Conference (ACC)*. Atlanta, GA, USA: IEEE, June 2022, pp. 1788–1793. ISBN: 978-1-6654-5196-3. DOI: 10.23919/ACC53348.2022.9867313.

[12] Jongeun Choi, Ryozo Nagamune, and Roberto Horowitz. "Multiple robust track-following controller design in hard disk drives". In: *International Journal of Adaptive Control and Signal Processing* (2008).

[13] Jongeun Choi, Ryozo Nagamune, and Roberto Horowitz. "Synthesis of multiple robust controllers for parametric uncertain LTI systems". In: *American Control Conference* (2006).

[14] Richard Conway, Sarah Felix, and Roberto Horowitz. "Model Reduction and Parametric Uncertainty Identification for Robust Control Synthesis for Dual-Stage Hard Disk Drives". In: *IEEE Transactions on Magnetics* (2007).

[15] Tim De Ryck, Ameya D. Jagtap, and Siddhartha Mishra. "Error estimates for physics informed neural networks approximating the Navier–Stokes equations". In: *arXiv preprint arXiv:2203.09346* (2022). arXiv: 2203.09346 [math.NA]. URL: https://arxiv.org/abs/2203.09346.

[16] Nathan Doumèche, Gérard Biau, and Claire Boyer. "Convergence and error analysis of PINNs". In: *arXiv preprint arXiv:2305.01240* (May 2023). arXiv: 2305.01240 [math.ST]. URL: https://arxiv.org/abs/2305.01240.

[17] D. J. Ewins. "Modal Testing: Theory and Practice." In: *New York: Wiley* (1986).

[18] Jinlong Fu et al. "Physics-data combined machine learning for parametric reduced-order modelling of nonlinear dynamical systems in small-data regimes". In: *Computer Methods in Applied Mechanics and Engineering* 404 (Feb. 2023), p. 115771. DOI: 10.1016/j.cma.2022.115771.

[19] Gorka Galdos, Alireza Karimi, and Roland Longchamp. "H infinity controller design for spectral MIMO models by convex optimization". In: *Journal of Process Control* (2010).

[20] Henri P. Gavin. "Eigensystem Realization". In: *Course Notes, Duke University* (2020).

[21] Per Hägg, Håkan Hjalmarsson, and Bo Wahlberg. "A least squares approach to direct frequency response estimation". In: *2011 50th IEEE Conference on Decision and Control and European Control Conference*. Orlando, FL, USA: IEEE, Dec. 2011, pp. 6261–6266. ISBN: 978-1-61284-801-3. DOI: 10.1109/CDC.2011.6160615.

[22] Marlon D. Hill. "An Experimental Verification of the Eigensystem Realization Algorithm for Vibration Parameter Identification". In: 2005. URL: https://api.semanticscholar.org/CorpusID:40301260.

[23] Neville Hogan. "Impedance control: An approach to manipulation: Part II—Implementation". In: *Journal of Dynamic Systems, Measurement, and Control* 107.1 (1985), pp. 8–16.

[24] P.A. Ioannou, E.B. Kosmatopoulos, and A.M. Despain. "Position error signal estimation at high sampling rates using data and servo sector measurements". In: *Proceedings of the 41st IEEE Conference on Decision and Control*. Las Vegas, NV, USA: IEEE, Dec. 2002. ISBN: 0-7803-7516-5. DOI: 10.1109/CDC.2002.1184469.

[25] Rolf Isermann. "Parameter adaptive control algorithms—A tutorial". In: *Automatica* 18.5 (1982), pp. 513–528. ISSN: 0005-1098. DOI: 10.1016/0005-1098(82)90002-4. URL: https://www.sciencedirect.com/science/article/pii/0005109882900024.

[26] Jer-Nan Juang and Richard S. Pappa. "An Eigensystem Realization Algorithm for Modal Parameter Identification and Model Reduction". In: *Journal of Guidance, Control, and Dynamics* 8 (1985).

[27] Jer-Nan Juang et al. "Identification of Observer/Kalman Filter Markov Parameters: Theory and Experiments". In: *Journal of Guidance Control and Dynamics* 16 (1993).

[28] Alireza Karimi and Christoph Kammer. "A data driven approach to robust control of multivariable systems by convex optimization". In: *Automatica* (2017).

[29] Alireza Karimi, Achille Nicoletti, and Yuanming Zhu. "Robust hinfinity controller design using frequency domain data via convex optimization". In: *International Journal of Robust and Non Linear Control* (2016).

[30] Hassan K. Khalil. *Nonlinear Control*. Pearson Education, 2015. ISBN: 978-0-13-349926-1.

[31] Pramod P. Khargonekar and Mario A. Rotea. "Mixed H2/H infinity control: a convex optimization approach." In: *IEEE Transactions on Automatic Control* (1991).

[32] Oussama Khatib. "A unified approach for motion and force control of robot manipulators: The operational space formulation". In: *IEEE Journal on Robotics and Automation* 3.1 (1987).

[33] Marin Kobilarov, Keenan Crane, and Mathieu Desbrun. "Lie group integrators for animation and control of vehicles". In: *ACM Trans. Graph.* 28.2 (May 2009). ISSN: 0730-0301.

[34] Marin B. Kobilarov and Jerrold E. Marsden. "Discrete Geometric Optimal Control on Lie Groups". In: *IEEE Transactions on Robotics* 27.4 (2011), pp. 641–655.

[35] Prasanth Kotaru, Ryan Edmonson, and Koushil Sreenath. *Geometric L1 Adaptive Attitude Control for a Quadrotor Unmanned Aerial Vehicle*. 2020. arXiv: 1910.07730 [math.OC].

[36] Prasanth Kotaru, Guofan Wu, and Koushil Sreenath. "Differential-flatness and control of quadrotor(s) with a payload suspended through flexible cable(s)". In: *2018 Indian Control Conference (ICC)*. 2018, pp. 352–357. DOI: 10.1109/INDIANCC.2018.8308004.

[37] Prasanth Kotaru, Guofan Wu, and Koushil Sreenath. "Dynamics and control of a quadrotor with a payload suspended through an elastic cable". In: *2017 American Control Conference (ACC)*. 2017, pp. 3906–3913. DOI: 10.23919/ACC.2017.7963553.

[38] Klas Kronander and Aude Billard. "Stability considerations for variable impedance control". In: *IEEE Transactions on Robotics* 32.5 (2016), pp. 1298–1305.

[39] Vince Kurtz, Patrick M. Wensing, and Hai Lin. "Control Barrier Functions for Singularity Avoidance in Passivity-Based Manipulator Control". In: *2021 60th IEEE Conference on Decision and Control (CDC)*. Funding Agency: 10.13039/100000001-National Science Foundation. Austin, TX, USA: IEEE, Dec. 2021, pp. 4126–4132. ISBN: 978-1-6654-3290-5. DOI: 10.1109/CDC45484.2021.9683597.

[40] George E Kuster. "H-infinity Norm Calculation via a State Space Formulation". In: *M.S Thesis, Virginia Polytechnic Institute and State University* (2013).

[41] Nicolas Labroche. "Online fuzzy medoid based clustering algorithms". In: *Neurocomputing* (2014).

[42] Taeyoung Lee. "Exponential stability of an attitude tracking control system on SO (3) for large-angle rotational maneuvers". In: *Systems & Control Letters* 61.1 (2012), pp. 231–237.

[43] Taeyoung Lee et al. "Geometric tracking control of a quadrotor UAV on SE(3)". In: *49th IEEE Conference on Decision and Control (CDC)*. IEEE. 2010, pp. 5420–5425.

[44] Taeyoung Lee, Melvin Leok, and Harris N. McClamroch. "Global Formulations of Lagrangian and Hamiltonian Dynamics on Manifolds". In: 13 (2017).

[45] Taeyoung Lee, Melvin Leok, and N Harris McClamroch. "Nonlinear robust tracking control of a quadrotor UAV on SE (3)". In: *Asian journal of control* 15.2 (2013), pp. 391–408.

[46] Peter Lindskog. "Algorithms and Tools for System Identification Using Prior Knowledge". PhD thesis. Linköping University, 1994. ISBN: 91-7871-422-2.

[47] Gaurav Manek and J. Zico Kolter. "Learning Stable Deep Dynamics Models". In: *CoRR* abs/2001.06116 (2020). arXiv: 2001.06116.

[48] H.J. Marquez and C.J. Damaren. "Analysis and synthesis of strictly positive real transfer functions". In: *Journal of the Franklin Institute* 333.2 (1996), pp. 245–256. ISSN: 0016-0032. DOI: 10.1016/0016-0032(96)00008-7. URL: https://www.sciencedirect.com/science/article/pii/0016003296000087.

[49] Richard J. Martin. "A Metric for ARMA Processes". In: *IEEE Transactions on signal processing* (2000).

[50] Siddhartha Mishra and Roberto Molinaro. "Estimates on the generalization error of Physics Informed Neural Networks (PINNs) for approximating a class of inverse problems for PDEs". In: *arXiv preprint arXiv:2007.01138* (Apr. 2021). arXiv: 2007.01138 [math.NA]. URL: https://arxiv.org/abs/2007.01138.

[51] Richard M. Murray, Zexiang Li, and S. Shankar Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.

[52] J. Pan et al. "Triple-Stage Track-Following Servo Design for Hard Disk Drives". In: *ASME Dynamic Systems and Control Conference, Minneapolis, MN*. 2016.

[53] Liang Pan and David Bogy. "Heat-assisted magnetic recording". In: *Nature Photon 3* (2009). DOI: https://doi.org/10.1038/nphoton.2009.40.

[54] Richard S. Pappa and J-N. Juang. "Galileo spacecraft modal identification using an eigensystem realization algorithm". In: *Structures, Structural Dynamics and Materials Conference, 25th, Palm Springs, CA*. 1984.

[55] Frank C Park, James E Bobrow, and Scott R Ploen. "A Lie group formulation of robot dynamics". In: *The International journal of robotics research* 14.6 (1995), pp. 609–618.

[56] Frank C. Park. "Distance metrics on the rigid-body motions with applications to mechanism design". In: *Journal of Mechanical Design* 117.1 (1995), pp. 48–54.

[57] Minh Q. Phan et al. "State-Space Model and Kalman Filter Gain Identification by a Kalman Filter of a Kalman Filter". In: *ASME. Journal of Dynamical Systems, Measurement, and Control* (2018). DOI: https://doi.org/10.1115/1.4037778.

[58] Nikhil Potu Surya Prakash, Zhi Chen, and Roberto Horowitz. "Data-Driven Strictly Positive Real System Identification with Prior System Knowledge". In: *2022 American Control Conference (ACC)*. 2022, pp. 3949–3954. DOI: 10.23919/ACC53348.2022.9867618.

[59] Nikhil Potu Surya Prakash, Zhi Chen, and Roberto Horowitz. *System Identification in Multi-Actuator Hard Disk Drives with Colored Noises using Observer/Kalman Filter Identification (OKID) Framework*. 2021. arXiv: 2109.12460 [eess.SY]. URL: https://arxiv.org/abs/2109.12460.

[60] Nikhil Potu Surya Prakash and Roberto Horowitz. "Data-Driven Robust Feedback Control Design for Multi-Actuator Hard Disk Drives". In: *IFAC-PapersOnLine* 55.37 (2022), pp. 131–138. ISSN: 2405-8963. DOI: 10.1016/j.ifacol.2022.11.173. URL: https://www.sciencedirect.com/science/article/pii/S2405896322028166.

[61] Nikhil Potu Surya Prakash et al. "Clustering Techniques for Stable Linear Dynamical Systems with applications to Hard Disk Drives". In: *IFAC-PapersOnLine* 56.3 (2023), pp. 529–534. ISSN: 2405-8963. DOI: 10.1016/j.ifacol.2023.12.078. URL: https://www.sciencedirect.com/science/article/pii/S2405896323024114.

[62] Nikhil Potu Surya Prakash et al. *Data-Driven Track Following Control for Dual Stage-Actuator Hard Disk Drives*. IFAC World Congress, Yokohama, Japan, 2023. Apr. 2023. DOI: 10.48550/arXiv.2304.00720. arXiv: 2304.00720 [eess.SY]. URL: https://arxiv.org/abs/2304.00720.

[63] Nikhil Potu Surya Prakash et al. "Data-driven track following control for dual stage-actuator hard disk drives". In: *arXiv preprint arXiv:2304.00720* (Apr. 2023). URL: https://arxiv.org/abs/2304.00720.

[64] Nikhil Potu Surya Prakash et al. "Variable Impedance Control using Deep Geometric Potential Fields". In: *Modeling Estimation and Controls Conference (MECC)*. Chicago, 2024.

[65] Nikhil Potu Surya Prakash. *A Computational Approach for Variational Integration of Attitude Dynamics on SO(3)*. 2022. arXiv: 2201.00713 [math.NA].

[66] Nikhil Potu Surya Prakash et al. "Deep Geometric Potential Functions for Tracking on Manifolds". In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2024). September 2024, Abu Dhabi.

[67] Goktug T. Cinar; Carlos A. Loza; Jose C Principe. "Hierarchical Linear Dynamical Systems: A new model for clustering of time series." In: *International Joint Conference on Neural Networks (IJCNN)* (2014).

[68] Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. "Physics Informed Deep Learning (Part I): Data-driven Solutions of Nonlinear Partial Differential Equations". In: *arXiv preprint arXiv:1711.10561* (Nov. 2017). arXiv: 1711.10561 [cs.AI]. URL: https://arxiv.org/abs/1711.10561.

[69] Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations". In: *Journal of Computational Physics* 378 (Feb. 2019), pp. 686–707. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2018.10.045.

[70] Maziar Raissi, Alireza Yazdani, and George Em Karniadakis. "Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations". In: *Science* 367.6481 (Feb. 2020), pp. 1026–1030. ISSN: 0036-8075. DOI: 10.1126/science.aaw4741.

[71] Anders Rantzer. "On the Kalman—Yakubovich—Popov lemma". In: *Systems & Control Letters* 28.1 (1996), pp. 7–10. ISSN: 0167-6911. DOI: 10.1016/0167-6911(95)00063-1.

[72] J.J. Sanchez-Gasca. "Computation of turbine-generator subsynchronous torsional modes from measured data using the eigensystem realization algorithm". In: *Power Engineering Society Winter Meeting, 2001. IEEE*. Vol. 3. IEEE. 2001.

[73] Enrico Schiassi et al. "Extreme theory of functional connections: A fast physics-informed neural network method for solving ordinary and partial differential equations". In: *Neurocomputing* 457 (Oct. 2021), pp. 334–356. DOI: 10.1016/j.neucom.2021.06.015.

[74] Enrico Schiassi et al. "Physics-informed neural networks for the point kinetics equations for nuclear reactor dynamics". In: *Annals of Nuclear Energy* 167 (Mar. 2022), p. 108833. DOI: 10.1016/j.anucene.2021.108833.

[75] Joohwan Seo et al. *A Comparison Between Lie Group- and Lie Algebra- Based Potential Functions for Geometric Impedance Control*. 2024. arXiv: 2401.13190 [cs.RO].

[76] Joohwan Seo et al. "Contact-rich SE(3)-Equivariant Robot Manipulation Task Learning via Geometric Impedance Control". In: *IEEE Robotics and Automation Letters* (2023).

[77] Joohwan Seo et al. "Geometric Impedance Control on SE(3) for Robotic Manipulators". In: *IFAC World Congress 2023, Yokohama, Japan* (2023).

[78] Prateek Shah and Roberto Horowitz. "Active Vibration Rejection in Multi Actuator Drives – Data Driven Approach". In: *ASME Conference on Dynamics Systems and Control* (2019).

[79] Prateek Shah and Roberto Horowitz. "Data Driven Feedforward Control Design and Input Shaping Techniques for Multi Actuator Drives". In: *ASME 2020 Conference on Dynamics Systems and Control* (2020).

[80] Prateek Shah and Roberto Horowitz. "Data Driven Feedforward Control Design and Input Shaping Techniques for Multi Actuator Drives". In: *ASME 2021 Letters in Dynamics Systems and Control* (2021).

[81] Guojun Sun et al. "Observer Kalman filter identification of suspen-dome". In: *Journal of Mathematical Problems in Engineering* (2017).

[82] Masayuki Takagishi et al. "Microwave assisted magnetic Recording: Physics and application to hard disk drives". In: *Journal of Magnetism and Magnetic Materials* (2022). DOI: https://doi.org/10.1016/j.jmmm.2022.169859.

[83] Yoichiro Tanaka, Shunichi Iwasaki, and Hiroaki Muraoka. "Perpendicular Magnetic Recording". In: *2017 IEEE History of ELectrotechnolgy CONference (HISTELCON)*. 2017, pp. 13–18. DOI: 10.1109/HISTELCON.2017.8535815.

[84] Roshan Mathew Tom and David B. Bogy. "Ballistic material transport in HAMR head-disk interface". In: *2023 IEEE 34th Magnetic Recording Conference (TMRC)*. 2023, pp. 1–2. DOI: 10.1109/TMRC59626.2023.10264026.

[85] Roshan Mathew Tom, Qilong Cheng, and David Bogy. "A Hybrid Simulation For Smear Growth On HAMR Heads". In: *2023 IEEE International Magnetic Conference-Short Papers (INTERMAG Short Papers)* (2023).

[86] Roshan Mathew Tom, Qilong Cheng, and David Bogy. "A Numerical Simulation of PFPE Lubricant Kinetics in HAMR Air Bearing". In: *Tribology Letters* (2024).

[87] Roshan Mathew Tom et al. "Optical forces in heat-assisted magnetic recording head-disk interface". In: *Scientific Reports* 13 (2023). DOI: https://doi.org/10.1038/s41598-023-35126-3.

[88] Francesco Vicario et al. "Extension of OKID to Output-Only System Identification". In: (2014).

[89] Francesco Vicario et al. "Generalized framework of OKID for linear state-space model identification". In: *Modeling, Simulation and Optimization of Complex Processes HPSC* (2017).

[90] Francesco Vicario et al. "OKID via Output Residuals: A Converter from Stochastic to Deterministic System Identification". In: *Journal of Guidance, Control, and Dynamics* 40 (12 2007).

[91] Francesco Vicario et al. "Output-only observer/Kalman filter identification O^3KID". In: *Journal of Structural Control and Health Monitoring* (2015).

[92] Bo Wahlberg. "System Identification Using Kautz Filters". In: *IEEE Transactions on Automatic Control* 39.6 (June 1994), pp. 1276–1282. DOI: 10.1109/9.293174.

[93] Roger Wood. "Shingled Magnetic Recording (SMR) and Two-Dimensional Magnetic Recording (TDMR)". In: *Journal of Magnetism and Magnetic Materials* (2022). DOI: https://doi.org/10.1016/j.jmmm.2022.169670.

[94] Liu Yang, Xuhui Meng, and George Em Karniadakis. "B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data". In: *Journal of Computational Physics* 425 (Jan. 2021), p. 109913. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2020.109913. arXiv: 2003.06097 [math.NA].

[95] Miin-Shen Yang, Chien-Yo Lai, and Chih-Ying Lin. "A robust EM clustering algorithm for Gaussian mixture models". In: *Pattern Recognition* (2012).

[96] Jun Zeng, Prasanth Kotaru, and Koushil Sreenath. "Geometric Control and Differential Flatness of a Quadrotor UAV with Load Suspended from a Pulley". In: *2019 American Control Conference (ACC)*. 2019, pp. 2420–2427. DOI: 10.23919/ACC.2019.8815173.

[97] Hao Zheng et al. "Improving deep neural networks using softplus units". In: *2015 International Joint Conference on Neural Networks (IJCNN)*. Killarney: IEEE, July 2015, pp. 1–8. ISBN: 978-1-4799-1960-4. DOI: 10.1109/IJCNN.2015.7280459.

Appendix A

Derivations

A.1 Powers of Skew-Symmetric Matrices

Given a vector $\omega = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \in \mathbb{R}^3$, the corresponding skew-symmetric matrix $\hat{\omega}$ is defined as

$$\hat{\omega} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}.$$

Its square can be found using simple matrix multiplication as

$$\begin{aligned} \hat{\omega}^2 &= \hat{\omega} \cdot \hat{\omega} \\ &= \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \\ &= \begin{bmatrix} -\omega_3^2 - \omega_2^2 & \omega_1\omega_2 & \omega_1\omega_3 \\ \omega_1\omega_2 & -\omega_3^2 - \omega_1^2 & \omega_2\omega_3 \\ \omega_1\omega_3 & \omega_2\omega_3 & -\omega_2^2 - \omega_1^2 \end{bmatrix}. \end{aligned}$$

Notice that

$$\omega\omega^T = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \begin{bmatrix} \omega_1 & \omega_2 & \omega_3 \end{bmatrix} = \begin{bmatrix} \omega_1^2 & \omega_1\omega_2 & \omega_1\omega_3 \\ \omega_1\omega_2 & \omega_2^2 & \omega_2\omega_3 \\ \omega_1\omega_3 & \omega_2\omega_3 & \omega_3^2 \end{bmatrix}.$$

We can now compactly express $\hat{\omega}^2$ in terms of the identity matrix I, the vector ω, and its transpose ω^T as

$$\hat{\omega}^2 = -(\omega_1^2 + \omega_2^2 + \omega_3^2)I + \omega\omega^T.$$

Since $\omega_1^2 + \omega_2^2 + \omega_3^2 = \|\omega\|_2^2$, we can write it even more compactly as

$$\hat{\omega}^2 = -\|\omega\|_2^2 I + \omega\omega^T. \tag{A.1}$$

Higher powers can be easily computed using the square of $\hat{\omega}$ with some basic algebra. We can compute the cube as

$$\begin{aligned}\hat{\omega}^3 &= \hat{\omega} \cdot \hat{\omega}^2 \\ &= \hat{\omega} \cdot (-\|\omega\|_2^2 I + \omega\omega^T) \\ &= -\|\omega\|_2^2 \hat{\omega} + \hat{\omega}\omega\omega^T \end{aligned} \tag{A.2}$$

Next, consider $\hat{\omega}\omega$:

$$\hat{\omega}\omega = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

This is because $\hat{\omega}\omega = \omega \times \omega = 0$ for an three dimensional vector ω. Thus, the final expression for $\hat{\omega}^3$ can be obtained as

$$\hat{\omega}^3 = -\|\omega\|_2^2 \hat{\omega}. \tag{A.3}$$

Similarly, we can express the higher powers of $\hat{\omega}$ either in terms of $\hat{\omega}$ or $\hat{\omega}^2$ as follows

$$\hat{\omega}^4 = \hat{\omega}\hat{\omega}^3 = -\|\omega\|_2^2 \hat{\omega}^2 \tag{A.4}$$

$$\hat{\omega}^5 = \|\omega\|_2^2 \hat{\omega}^3 = \|\omega\|_2^4 \hat{\omega} \tag{A.5}$$

$$\hat{\omega}^6 = \|\omega\|_2^4 \hat{\omega}^2 \text{ and so on.} \tag{A.6}$$

A.2 Necessary Lemmata

***Lemma* 1.** *For a rotation matrix $R \in SO(3)$ and a vector $\omega \in \mathbb{R}^3$,*

$$(R^T \hat{\omega} R)^\vee = R^T \omega. \tag{A.7}$$

Proof. Given a vector $v \in \mathbb{R}^3$, the cross product of ω and v can be expressed using the hat operator as:

$$\omega \times v = \hat{\omega} v$$

Now consider the following rotated versions of these vectors, $R^T\omega$ and $R^T v$. The cross product of these rotated vectors is

$$(R^T\omega) \times (R^T v).$$

By the definition of the hat operator, we can write this cross product as:

$$(R^T\omega) \times (R^T v) = \widehat{R^T\omega}(R^T v).$$

On the other hand, we can use the original vectors and transform the result of their cross product as

$$R^T(\omega \times v) = R^T(\hat{\omega} v) = R^T \hat{\omega} v = R^T \hat{\omega} R R^T v = R^T \hat{\omega} R (R^T v).$$

Thus, we have

$$\widehat{R^T\omega}(R^T v) = R^T\hat{\omega}R(R^T v).$$

Since this equation holds for any vector $R^T v$, we can conclude:

$$\widehat{R^T\omega} = R^T\hat{\omega}R \implies (R^T\hat{\omega}R)^\vee = R^T\omega.$$

□

***Lemma* 2.** *For a symmetric matrix $A \in \mathbb{R}^{3\times 3}$ and a skew-symmetric matrix $\hat{b} \in \mathbb{R}^{3\times 3}$,*

$$tr[A\hat{b}] = 0. \tag{A.8}$$

Proof. The trace of a product of two matrices can be written as

$$tr[A\hat{b}] = tr[\hat{b}A].$$

Since $\hat{b}^T = -\hat{b}$, we have

$$tr[\hat{b}A] = tr[A^T\hat{b}^T].$$

Since A is symmetric, $A^T = A$, and hence

$$tr[A^T\hat{b}^T] = tr[A(-\hat{b})] = -tr[A\hat{b}]$$

Combining the results, we get

$$tr[A\hat{b}] = -tr[A\hat{b}] \implies tr[A\hat{b}] = 0.$$

□

***Lemma* 3.** *For two vectors $a, b \in \mathbb{R}^3$ and their skew-symmetric counterparts $\hat{a}, \hat{b}$,*

$$tr[\hat{a}\hat{b}] = -2(a \cdot b). \tag{A.9}$$

Proof. Given two vectors $a = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}$ and $b = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$, the corresponding skew-symmetric matrices $\hat{a}$ and $\hat{b}$ are defined as

$$\hat{a} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}, \hat{b} = \begin{bmatrix} 0 & -b_3 & b_2 \\ b_3 & 0 & -b_1 \\ -b_2 & b_1 & 0 \end{bmatrix}.$$

Their product $\hat{a}\hat{b}$ can be computed as

$$\hat{a}\hat{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} 0 & -b_3 & b_2 \\ b_3 & 0 & -b_1 \\ -b_2 & b_1 & 0 \end{bmatrix} = \begin{bmatrix} -a_2b_2 - a_3b_3 & a_2b_1 & a_3b_1 \\ a_3b_2 & -a_1b_1 - a_3b_3 & a_1b_3 \\ a_2b_3 & a_1b_3 & -a_1b_1 - a_2b_2 \end{bmatrix}.$$

The trace of this matrix is the sum of its diagonal elements. Therefore,

$$tr[\hat{a}\hat{b}] = (-a_2b_2 - a_3b_3) + (-a_1b_1 - a_3b_3) + (-a_1b_1 - a_2b_2) = -2(a_1b_1 + a_2b_2 + a_3b_3) = -2a \cdot b.$$

□

***Lemma* 4.** *For a matrix $A \in \mathbb{R}^{3\times3}$ and a vector $b \in \mathbb{R}^3$,*

$$tr[A\hat{b}] = (A^T - A)^\vee \cdot b. \tag{A.10}$$

Proof. Given a matrix A and a vector $b = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$, the corresponding skew-symmetric matrix $\hat{b}$ is defined as

$$\hat{b} = \begin{bmatrix} 0 & -b_3 & b_2 \\ b_3 & 0 & -b_1 \\ -b_2 & b_1 & 0 \end{bmatrix}.$$

Any matrix A can be decomposed into its symmetric and skew-symmetric parts:

$$A = A^{skew} + A^{sym},$$

where $A^{sym} = \frac{1}{2}(A + A^T)$ is the symmetric part and $A^{skew} = \frac{1}{2}(A - A^T)$ is the skew-symmetric part.
With the matrix A decomposed into its symmetric and skew-symmetric parts, we have

$$tr[A\hat{b}] = tr[(A^{sym} + A^{skew})\hat{b}] = tr[A^{sym}\hat{b}] + tr[A^{skew}\hat{b}].$$

From lemmata 2 and 3, we have

$$tr[A\hat{b}] = tr[A^{skew}\hat{b}] = (A^T - A)^\vee \cdot b.$$

□

***Lemma* 5.** *Rodrigues' formula: For $\hat{\omega} \in \mathfrak{so}(3)$, the exponential map is given by*

$$\exp_{\mathfrak{so}(3)}(\hat{\omega}) = I + \sin(\|\omega\|_2)\frac{\hat{\omega}}{\|\omega\|_2} + (1 - \cos(\|\omega\|_2)\frac{\hat{\omega}^2}{\|\omega\|_2^2} \tag{A.11}$$

Proof. Given a skew-symmetric matrix $\hat{\omega} \in \mathfrak{so}(3)$, the matrix exponential of $\hat{\omega}$ is defined using (2.1) as

$$\exp_{\mathfrak{so}(3)}(\hat{\omega}) = I + \hat{\omega} + \frac{\hat{\omega}^2}{2!} + \frac{\hat{\omega}^3}{3!} + \frac{\hat{\omega}^4}{4!} + \cdots$$

We can group the terms involving $\hat{\omega}$ and $\hat{\omega}^2$ as

$$\exp_{\mathfrak{so}(3)}(\hat{\omega}) = I + \left(\hat{\omega} - \frac{\hat{\omega}^3}{3!} + \frac{\hat{\omega}^5}{5!} - \cdots\right) + \left(\frac{\hat{\omega}^2}{2!} - \frac{\hat{\omega}^4}{4!} + \cdots\right)$$

and using the powers of skew-symmetric matrices defined in section A.1, we can obtain the following

$$\exp_{\mathfrak{so}(3)}(\hat{\omega}) = I + \hat{\omega}\left(1 - \frac{\|\omega\|_2^2}{3!} + \frac{\|\omega\|_2^4}{5!} - \cdots\right) + \hat{\omega}^2\left(\frac{1}{2!} - \frac{\|\omega\|_2^2}{4!} + \cdots\right).$$

Using the Taylor series expansions of $\sin(\|\omega\|_2)$ and $\cos(\|\omega\|_2)$ that

$$\sin(\|\omega\|_2) = \|\omega\|_2 - \frac{\|\omega\|_2^3}{3!} + \frac{\|\omega\|_2^5}{5!} - \cdots$$

and

$$\cos(\|\omega\|_2) = 1 - \frac{\|\omega\|_2^2}{2!} + \frac{\|\omega\|_2^4}{4!} - \cdots,$$

we can obtain the exponential of $\hat{\omega}$ as

$$\exp_{\mathfrak{so}(3)}(\hat{\omega}) = I + \sin(\|\omega\|_2)\frac{\hat{\omega}}{\|\omega\|_2} + (1 - \cos(\|\omega\|_2))\frac{\hat{\omega}^2}{\|\omega\|_2^2}.$$

□

***Lemma* 6.** *Exponential of an element in $\mathfrak{se}(3)$: For an element $\Gamma(\xi) = \begin{bmatrix} \hat{\omega} & v \\ 0 & 0 \end{bmatrix} \in \mathfrak{se}(3)$, where $\hat{\omega} \in \mathfrak{so}(3)$ and $v \in \mathbb{R}^3$, the exponential map is given by*

$$\exp_{\mathfrak{se}(3)}(\Gamma(\xi)) = \begin{bmatrix} \exp_{\mathfrak{so}(3)}(\hat{\omega}) & A(\omega)v \\ 0 & 1 \end{bmatrix} \tag{A.12}$$

with

$$A(\omega) = I + \frac{1 - \cos(\|\omega\|_2)}{\|\omega\|_2^2}\hat{\omega} + \frac{\|\omega\|_2 - \sin(\|\omega\|_2)}{\|\omega\|_2^3}\hat{\omega}^2 \tag{A.13}$$

Proof. The exponential map of $\Gamma(\xi)$ using the matrix exponential definition can be written as

$$\exp_{\mathfrak{se}(3)}(\Gamma(\xi)) = I + \Gamma(\xi) + \frac{\Gamma(\xi)^2}{2!} + \frac{\Gamma(\xi)^3}{3!} + \cdots.$$

Substituting $\Gamma(\xi)$, we have

$$\begin{aligned}\exp_{\mathfrak{se}(3)}(\Gamma(\xi)) &= \begin{bmatrix} I & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} \hat{\omega} & v \\ 0 & 0 \end{bmatrix} + \frac{1}{2!}\begin{bmatrix} \hat{\omega}^2 & \hat{\omega}v \\ 0 & 0 \end{bmatrix} + \frac{1}{3!}\begin{bmatrix} \hat{\omega}^3 & \hat{\omega}^2 v \\ 0 & 0 \end{bmatrix} + \cdots \\ &= \begin{bmatrix} \exp_{\mathfrak{so}(3)}(\hat{\omega}) & \sum_{n=0}^{\infty} \frac{\hat{\omega}^n}{(n+1)!}v \\ 0 & 1 \end{bmatrix}.\end{aligned}$$

Using the powers of skew-symmetric matrices defined in A.1 and the Taylor series expansions of $\sin(\|\omega\|_2)$, $\cos(\|\omega\|_2)$, it can be seen that

$$A(\omega) = I + \frac{1 - \cos(\|\omega\|_2)}{\|\omega\|_2^2}\hat{\omega} + \frac{\|\omega\|_2 - \sin(\|\omega\|_2)}{\|\omega\|_2^3}\hat{\omega}^2 = \sum_{n=0}^{\infty} \frac{\hat{\omega}^n}{(n+1)!}.$$

Therefore

$$\exp_{\mathfrak{se}(3)}(\Gamma(\xi)) = \begin{bmatrix} \exp_{\mathfrak{so}(3)}(\hat{\omega}) & A(\omega)v \\ 0 & 1 \end{bmatrix}.$$

□