MODELING, DESIGN, TESTING AND CONTROL OF A TWO-STAGE ACTUATION MECHANISM USING PIEZOELECTRIC ACTUATORS FOR AUTOMOTIVE APPLICATIONS

DISSERTATION

Presented in Partial Fulfillment of the Requirements for the Degree Doctor of Philosophy in the Graduate School of The Ohio State University

By

Vijay Anand Neelakantan, B.Tech., M.S.

The Ohio State University

2005

Approved by

Dissertation Committee:

Dr. Gregory Washington, Adviser
Dr. Giorgio Rizzoni
Dr. Krishnaswamy Srinivasan
Dr. Vadim Utkin

Adviser
Mechanical Engineering Graduate Program
Abstract

High bandwidth actuation systems capable of simultaneously producing relatively large forces and displacements are required for use in automobiles and other industrial applications. Conventional hydraulic actuation mechanisms used in automotive brakes and clutches are complex, inefficient and have poor control robustness. For instance, the hydraulic clutch actuation mechanism used in automatic transmissions requires pumping hardware that derives power from the engine. Along with inefficient torque converters these systems lead to reduced fuel economy, controllability issues and other disadvantages. Therefore, using advanced technologies to develop and implement novel devices, as replacements for the conventional hydraulic actuation mechanisms will improve the vehicle fuel economy significantly. This thesis presents the concept, design, development, modeling, testing and control of a novel two-stage hybrid actuation mechanism by combining classical actuators like DC motors and advanced smart material actuators like piezoelectric stack actuators. This two-stage mechanism takes advantage of the unique stiffness (force-stroke) characteristic of a typical clutch or a brake engagement process. This two-stage mechanism is modeled and designed by splitting the system into two operating regimes, namely the stroke phase and the force phase. Importance is placed on modeling the nonlinearities like the hysteresis property of piezoelectric actuators and techniques to overcome it using appropriate analysis and control methodologies. Also a
technique to estimate force based on the charge stored in the piezoactuator is discussed, which leads to the elimination of the mechanical force sensor. A simple laboratory prototype experimental setup is built to demonstrate the system functioning and to test the different control strategies. A major part of this research includes the development of robust control methodologies using advanced concepts like Internal Model Control (IMC), Model Predictive Control (MPC) and a new strategy called Model Predictive Sliding Mode Control (MPSMC). The different control strategies are used to guide the two-stage actuation system to track time-varying reference force inputs. The IMC concept is used to develop a robust controller based on the uncertainty-bound on the system model. MPC is used to produce a sub-optimal controller that uses a receding-horizon window for future prediction of system behavior. The new concept MPSMC is developed to overcome the limitations of the conventional discrete-time sliding mode control. In this method, the system is forced to reach the sliding mode in a smooth sub-optimal trajectory. This optimization is carried out using MPC. Experimental results are highlighted in each case comparing the effectiveness of the different methods.
Dedicated to

my beloved grandfather Shri V. Natarajan

for his love, encouragement and support
ACKNOWLEDGMENTS

I express my sincere thanks to my advisor Prof. Gregory Washington, for all his guidance, support and encouragement throughout my graduate studies. His expertise, knowledge, advice and comments were priceless. I admire his vision and ability to look at things in a bigger perspective.

I express my sincere gratitude to the personnel in the Powertrain Systems Research Lab at General Motors R&D for offering me a great opportunity to work with them. This research would not have been possible without their financial and technical support. I especially would like to thank Dr. Patrick Usoro, Dr. Norman Bucknor, Dr. Chi-Kuan Kao and Dr. Hazem Ezzat for all the confidence they had in me to carry out the research for this thesis. I express my gratitude to Prof Giorgio Rizzoni, Prof Krishnaswamy Srinivasan and Prof Vadim Utkin for participating in my doctoral dissertation committee in spite of their busy schedules.

In addition, I thank Amitesh, Farzad, Jun-Kyu, LeAnn, Leon, Dr. Yoon and all other past members of the Intelligent Structures and Systems Lab for their invaluable help and comradeship during graduate school.

I thank my family in India for their wonderful support throughout my life, and I thank the Almighty for his blessings.
VITA

May 06, 1979................................. Born – Nagapattinam, Tamil Nadu, India

July 2000................................. B.Tech, Mechanical Engineering

Indian Institute of Technology, Madras

Chennai, India

March 2003................................. M.S., Mechanical Engineering

The Ohio State University, Columbus

Ohio, USA

Sep 2000 – Present........................ Graduate Research Associate

Department of Mechanical Engineering

The Ohio State University

FIELDS OF STUDY

Major Field: Mechanical Engineering

Major Areas of Specialization: Dynamic Systems, Smart Materials and Control Systems
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>ii</td>
</tr>
<tr>
<td>Dedication</td>
<td>iv</td>
</tr>
<tr>
<td>Acknowledgments</td>
<td>v</td>
</tr>
<tr>
<td>Vita</td>
<td>vi</td>
</tr>
<tr>
<td>List of Figures</td>
<td>xi</td>
</tr>
<tr>
<td>List of Tables</td>
<td>xvii</td>
</tr>
<tr>
<td>Chapters:</td>
<td></td>
</tr>
<tr>
<td>1. INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Motivation</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Introduction to Smart Material Systems and Structures</td>
<td>6</td>
</tr>
<tr>
<td>1.3 Literature Review</td>
<td>9</td>
</tr>
<tr>
<td>1.3.1 Existing Controllable Clutch and Brake Actuation Technologies</td>
<td>9</td>
</tr>
<tr>
<td>1.3.2 Piezoelectric Actuator System Modeling</td>
<td>12</td>
</tr>
<tr>
<td>1.3.3 Advanced Control Strategies</td>
<td>16</td>
</tr>
<tr>
<td>1.4 Research Objectives</td>
<td>23</td>
</tr>
<tr>
<td>1.5 Outline of Subsequent Chapters</td>
<td>24</td>
</tr>
<tr>
<td>2. TWO-STAGE CLUTCH AND BRAKE ACTUATION MECHANISM</td>
<td>25</td>
</tr>
<tr>
<td>2.1 Clutch/Brake Actuation Requirements</td>
<td>25</td>
</tr>
<tr>
<td>2.2 Alternative Clutch/Brake Actuation Mechanisms</td>
<td>29</td>
</tr>
<tr>
<td>2.2.1 MRF (Magneto-Rheological Fluid) Clutch</td>
<td>30</td>
</tr>
<tr>
<td>2.2.2 Magnetic Particle Clutch</td>
<td>31</td>
</tr>
<tr>
<td>2.2.3 Electromagnetic Clutch</td>
<td>31</td>
</tr>
</tbody>
</table>
4.5.1 IMC-based PID tuning Example 1................................. 109
4.6 Variable Delay and Gain Model for the Piezoactuator System ....... 110
4.7 Internal Model Control of Force Phase of the System................. 113
  4.7.1 PID Gains................................................................. 116
4.8 Experimental results and Discussion ......................................... 118

5. MODEL PREDICTIVE CONTROL OF SYSTEM ...................... 125
  5.1 Introduction................................................................. 125
  5.2 Math Background and Notations for MPC ......................... 129
    5.2.1 Discrete Time Systems ........................................... 129
      5.2.1.1 Z-Transform.................................................... 130
      5.2.1.2 Transfer Function Model .................................... 130
      5.2.1.3 State Space Models .......................................... 132
    5.2.2 State Vectors of Past, Present and Future Values ........... 133
    5.2.3 Toeplitz and Hankel Matrices .................................. 134
      5.2.3.1 Multiplication of polynomials.............................. 135
      5.2.3.2 Inversion of Toeplitz Matrix............................... 137
  5.3 Model Predictive Control Formulation............................... 138
    5.3.1 Transfer Function Model ........................................ 138
      5.3.1.1 MPC of Systems with Transfer Function Model ........ 140
    5.3.2 State Space Model................................................ 141
      5.3.2.1 MPC Algorithm for Systems with State Space Model ... 142
  5.4 Stability Analysis of MPC .............................................. 144
  5.5 Model Predictive Control of Dual Stage Actuation System ....... 145
  5.6 Experimental Results.................................................. 145
    5.6.1 Control Law Implementation and Testing .................... 152
  5.7 Concluding Remarks.................................................... 154

6. MODEL PREDICTIVE SLIDING MODE CONTROL OF SYSTEM ...... 159
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1 Introduction</td>
<td>160</td>
</tr>
<tr>
<td>6.2 Conventional Discrete-Time Sliding Mode Control</td>
<td>165</td>
</tr>
<tr>
<td>6.2.1 Limitations of Conventional DSMC</td>
<td>168</td>
</tr>
<tr>
<td>6.3 Model Predictive Sliding Mode Control – Method 1</td>
<td>169</td>
</tr>
<tr>
<td>6.3.1 Simulation Example 1</td>
<td>174</td>
</tr>
<tr>
<td>6.3.2 Experimental Result</td>
<td>178</td>
</tr>
<tr>
<td>6.4 Model Predictive Sliding Mode Control – Method 2</td>
<td>185</td>
</tr>
<tr>
<td>6.4.1 Equivalent Control Augmentation</td>
<td>187</td>
</tr>
<tr>
<td>6.4.2 Simulation Example 2</td>
<td>189</td>
</tr>
<tr>
<td>6.4.3 Experimental Result</td>
<td>192</td>
</tr>
<tr>
<td>6.5 Concluding Remarks</td>
<td>197</td>
</tr>
<tr>
<td>7. CONCLUSIONS</td>
<td>199</td>
</tr>
<tr>
<td>7.1 Summary</td>
<td>199</td>
</tr>
<tr>
<td>7.2 Scope for Future Work</td>
<td>204</td>
</tr>
</tbody>
</table>

REFERENCES ................................................................................................................206
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Schematic of a Torque Converter</td>
<td>3</td>
</tr>
<tr>
<td>1.2</td>
<td>Typical characteristics of a torque converter in automatic transmissions</td>
<td>3</td>
</tr>
<tr>
<td>1.3</td>
<td>Schematic of a hydraulic Clutch Actuation Mechanism</td>
<td>4</td>
</tr>
<tr>
<td>2.1</td>
<td>Schematic showing Hydraulic Clutch before and after engagement</td>
<td>26</td>
</tr>
<tr>
<td>2.2</td>
<td>Clutch Force Requirement Calculation</td>
<td>27</td>
</tr>
<tr>
<td>2.3</td>
<td>Clutch actuation concept based on piezoelectric actuator [Keeney-2001]</td>
<td>34</td>
</tr>
<tr>
<td>2.4</td>
<td>A brake actuation concept based on magnetostrictive actuator [Murata-1998]</td>
<td>35</td>
</tr>
<tr>
<td>2.5</td>
<td>Typical Stiffness Curve of a Clutch Actuation Process</td>
<td>37</td>
</tr>
<tr>
<td>2.6</td>
<td>Serial and Parallel Actuation System Concepts</td>
<td>39</td>
</tr>
<tr>
<td>2.7</td>
<td>Two-stage clutch actuation - embodiment 1</td>
<td>42</td>
</tr>
<tr>
<td>2.8</td>
<td>Two-stage clutch actuation - embodiment 2</td>
<td>43</td>
</tr>
<tr>
<td>2.9</td>
<td>Two-stage actuation mechanism for braking applications</td>
<td>44</td>
</tr>
<tr>
<td>3.1</td>
<td>DC Motor and piezoactuator based two-stage high force/high stroke actuator system</td>
<td>47</td>
</tr>
<tr>
<td>3.2</td>
<td>Reference Speed Profile for DC motor</td>
<td>49</td>
</tr>
</tbody>
</table>
3.3  Force-Displacement characteristic of piezoelectric actuator loaded against a spring................................................................. 57
3.4  Simple model for Force Phase dynamics.................................................................................................................................. 57
3.5  Hysteresis Model Output for PZT – 5A with a = 4.2e5 V/m................................................................. 64
3.6  (a) Classical hysteresis operator (b) Modified hysteresis operator................................................................. 65
3.7  Schematic interpretation of the Preisach model........................................................................................................... 66
3.8  Shaded region $S^+(t)$ which contains the hysteresis operators of +1 value ....... 67
3.9  (a) input voltage ($\alpha$ decreases and $\beta$ is zero) (b) input voltage ($\alpha$ and $\beta$ both decrease) ........................................................................................................ 67
3.10  (a) Hysteresis loop having the common switching value for $\beta$ and different switching values for $\alpha$ (b) Hysteresis loop showing several switching values for both $\alpha$ and $\beta$........................................................................................................ 68
3.11  Comparison of Preisach Model Results with Experimental Data for PZT stack PSt 1000/16/80 VS25 type, Piezomechanik GmbH................................................................. 68
3.12  Schematic of the Variable delay phenomenon........................................................................................................... 70
3.13  Hysteresis Curves from Experimental Measured Data ........................................................................................................ 72
3.14  Hysteresis Curves based on Variable Delay and Variable Gain Model ................. 72
3.15  Experimental setup.......................................................................................................................................................... 74
3.16  Stroke Phase Test Results ................................................................................................................................................ 76
3.17  Step voltage input ......................................................................................................................................................... 77
3.18  Response of piezoeactuator-based force phase........................................................................................................... 78
4.11 Control Voltages for Sinusoid Input (1 Hz) Tracking for various ‘$\lambda$’ values..... 120

4.12 Sinusoid Input (10 Hz) Tracking for various ‘$\lambda$’ values using IMC................. 121

4.13 Control Voltages for Sinusoid Input (10 Hz) Tracking for various ‘$\lambda$’ values using IMC..................................................................................................................... 121

4.14 Step Input Tracking using a PI controller tuned using IMC for ‘$\lambda$’ = 0.005 ...... 122

4.15 Control Voltage for PI controller tuned using IMC for ‘$\lambda$’ = 0.005 ....................122

5.1 Schematic of Model Predictive Controller ................................................................. 127

5.2 Illustration of the working of MPC......................................................................... 127

5.3 Equivalent Discrete-Time Model of a Continuous Time System ......................... 130

5.4 Experimental Setup................................................................................................. 151

5.5 Step Reference Tracking – Force vs Time for various values of ‘$\lambda$’ ............... 155

5.6 Step Reference Tracking – Control Voltage vs Time for various values of ‘$\lambda$’. 155

5.7 Sinusoidal Reference (1 Hz) Tracking – Generated Force vs Time for various values of ‘$\lambda$’ ........................................................................................................ 156

5.8 Sinusoidal Reference (1 Hz) Tracking – Control Voltage vs Time for various values of ‘$\lambda$’ ........................................................................................................ 156

5.9 Sinusoidal Reference (10 Hz) Tracking – Generated Force vs Time for various values of ‘$\lambda$’ ........................................................................................................ 157

5.10 Sinusoidal Reference (10 Hz) Tracking – Control Voltage vs Time for various values of ‘$\lambda$’ ........................................................................................................ 157

5.11 Closed loop Relation between reference and actual force.................................158
6.1 Simulation result showing divergence and convergence of system to and away from the desired sliding mode using DSMC and 1st MPSMC method respectively ................................................................. 176

6.2 Simulation result showing control input ($u$) for DSMC (unstable) and 1st MPSMC method (stable) ................................................................................................................................. 177

6.3 Simple Schematic of the 1st MPSMC method .............................................................. 179

6.4 Generated force vs time for DSMC and MPSMC’s 1st method ................................ 179

6.5 Generated force vs time (Zoomed in) ........................................................................ 180

6.6 Sliding mode value vs time for DSMC and MPSMC’s 1st method ............................ 180

6.7 Phase plane plot showing elimination of chattering by using MPSMC ............... 181

6.8 Applied control voltage to piezo vs. time for DSMC and MPSMC’s 1st method ................................................................................................................................. 181

6.9 Applied control voltage vs time (Zoomed in) .......................................................... 182

6.10 Control Moves ($\Delta U$) for DSMC and MPSMC’s 1st method .............................. 182

6.11 Experimental result showing response deterioration when faster settling time (< 50 ms) is desired ................................................................................................................................. 184

6.12 Simulation result showing divergence and convergence of system to and away from the desired sliding mode using DSMC and 2nd MPSMC method respectively ................................................................................................................................. 190

6.13 Simulation result showing control input ($u$) for DSMC (unstable) and 1st MPSMC method (stable) ................................................................................................................................. 191

6.14 Simulation Result showing the partial control input ($U^*$) that drives the system to the sliding mode using 2nd MPSMC method for two $\lambda$ values ............................ 191
6.15   Simple Schematic of the 2\textsuperscript{nd} MPSMC method ................................................... 192
6.16   Generated force vs time for DSMC and MPSMC’s 2\textsuperscript{nd} method ...................... 193
6.17   Generated force vs time (Zoomed in) ............................................................................ 193
6.18   Sliding mode value vs time for DSMC and MPSMC’s 2\textsuperscript{nd} method ................. 194
6.19   Phase Plane Plot showing elimination of chattering by using MPSMC’s 2\textsuperscript{nd} method ................................................................................................................. 194
6.20   Applied control voltage to piezo vs. time for DSMC and MPSMC’s 2\textsuperscript{nd} method ........................................................................................................................................ 195
6.21   Applied control voltage vs time (Zoomed in) ................................................................. 195
6.22   Partial control input (\(U^p\)) vs time for DSMC and 2\textsuperscript{nd} MPSMC method .......... 196
7.1    Proposed Design for Size-Reduction .............................................................................. 205
**LIST OF TABLES**

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Comparison of various Smart Material Actuators</td>
</tr>
<tr>
<td>3.1</td>
<td>Parameter Values for Stroke Phase Design</td>
</tr>
<tr>
<td>3.2</td>
<td>Parameter Values in Force Phase</td>
</tr>
<tr>
<td>4.1</td>
<td>Examples of Proper, Strictly Proper and Improper Transfer Functions</td>
</tr>
<tr>
<td>5.1</td>
<td>Notations for Past and Present and Future State Values</td>
</tr>
<tr>
<td>5.2</td>
<td>Values of parameters used in the MPC system</td>
</tr>
</tbody>
</table>
CHAPTER 1

INTRODUCTION

The purpose of this chapter is to provide the reader with an introduction to the research conducted for this thesis. First, the motivation for the research conducted for this thesis is discussed. Then a brief introduction to the area of smart materials and their potential uses in devising intelligent systems in various applications is provided. This is followed by a comprehensive literature review conducted for the various fields associated with this research. The topics include currently used and available controllable clutching and braking technologies, piezoelectric actuator system modeling and development of various advanced control system methodologies. The chapter ends with the objectives for the research conducted, followed by an outline of the subsequent chapters of this thesis.

1.1 Motivation

Automatic transmissions have been in use in commercial automobiles as early as the 1930s. Unlike most of the world, the majority of automobiles (cars and passenger trucks) sold in the USA are equipped with automatic transmissions [Ross, 2004]. An automatic transmission is essentially an automatic gearbox that can provide different
gear-ratios between the input and the output shafts without human intervention. They are preferred more than the manual transmissions because of two main reasons. One obvious reason is that it frees the driver from having to manually shift the gears at different stages. The other reason is that it provides the user with the comfort of a smooth vehicle launch. However, these advantages come at a big cost on vehicle fuel economy. Almost all powertrains using automatic transmissions have three main components, namely i) a Torque Converter that couples the engine crank shaft to the input shaft of the automatic transmission, ii) a set of planetary gears that can allow different gear ratios between the input and output shafts of the automatic gearbox depending on which part of the planetary gear elements are held fixed and iii) a set of hydraulically actuated clutches that enable the system to control the power-flow direction in the gearbox.

The torque converter works on the principle of a turbine operated by a pump with a fluid coupling. During vehicle launch, the torque converter multiplies the torque from the engine by a factor greater than 2 and this factor gradually converges to 1 as the vehicle speed increases. However the efficiency of this mechanism is very poor at low turbine speeds where the worst fuel economy is encountered. Also a significant amount of power is wasted in the form of heat, which is dissipated quickly by the circulating fluid. This adds to the decrease in the total efficiency of the powertrain. Figure 1.1 shows a schematic of a typical torque converter and Figure 1.2 shows the typical characteristics of a torque converter. The torque and speed ratios are the ratios of output (turbine) to input (pump/engine) torque and speed.
Figure 1.1 Schematic of a Torque Converter

Figure 1.2 Typical characteristics of a torque converter in automatic transmissions
The hydraulic clutches used inside the transmission operate by using a high-pressure fluid medium to compress a set of clutch-plates connected to the input and output shafts. Figure 1.3 shows the schematic of a typical hydraulic clutch used in the automatic transmission. The high-pressure fluid used to produce the clutching action is generated by an auxiliary pump driven by the engine. The pump again takes a critical amount of power from the engine thereby reducing the overall efficiency of the powertrain further.

Figure 1.3 Schematic of a hydraulic Clutch Actuation Mechanism

It is estimated that the torque converter and the pump account for 50-60% of the total loss in an automatic transmission. Replacing them with more efficient means of power transmission will improve provide significant vehicle fuel economy improvement. It has been estimated that about 7-8% improvement is possible by replacing the torque converter and 4-6% improvement is possible by replacing the high-pressure fluid pump and its accessories. Hence it is estimated that through proper utilization of the power delivered by the engine using efficient torque transmission devices, the fuel
economy can be improved by up to 10-13% [Kluger-1993, Park-1996, Kao-2003]. This is a major improvement considering the volume and utilization of new vehicles produced and the quantity of fuel consumed by each. Since the average daily gasoline consumption in the US is around 400 million gallons and increasing, this means billions of dollars in annual savings [www.bts.gov]. Recently proposed new Corporate Average Fuel Economy (CAFÉ) standards require an annual saving of about 10 billion gallons of fuel [www.nhtsa.dot.gov] Also, assuming similar engine characteristics, as one lower fuel consumption one lowers the resulting pollution. This is very desirable considering the stringent requirements set forth all over the world in order to maintain good environmental conditions [www.epa.gov].

Another motivation behind this research is to improve the overall bandwidth of the clutch and brake actuation system. The current hydraulic actuation mechanism involves a large number of moving parts that is operated by solenoid valves. This puts a limit on the operating bandwidth and response time of the actuation mechanism. Implementation of advanced braking technologies like anti-lock brakes and advanced shifting technologies in automatic transmissions require systems with higher bandwidth and quicker response times. Hence by replacing the conventional hydraulic actuation systems by mechatronic clutch actuation mechanisms with direct interface to electronics one can improve the overall bandwidth and response time of the system. Typical hydraulic clutch actuation system is limited by a response time of around 150 ms [Han, 2003]. By replacing them with mechatronic and electromechanical actuation mechanisms it is estimated that the response time can be reduced to 10’s of milliseconds using advanced control strategies.
1.2 Introduction to Smart Material Systems and Structures

Smart Structures are usually defined in two paradigms, namely the science and technology paradigms. The *science paradigm* states that a smart structure is a structural system with intelligence and life features integrated in the macrostructure and quite possibly the microstructure of the system to meet stated objectives and to provide adaptive functionality. It does not define the type of materials or state that actuators, sensors, or controls are used. It just describes the philosophy of the structure. The technology paradigm states that a smart material is a structure that involves the integration of actuators, sensors, and controls with a material or structural component. This definition describes the components but it does not state any system goals or objectives. A smart structure is usually comprised of at least one active or smart material. Active materials are special materials that have the capability to alter or change their properties depending on some external parameter. These changes in properties vary from shape change and change in rheological properties to a change in electrical properties. The list of new smart materials continues to grow by the day. However a comprehensive list of smart materials includes piezoelectric/piezoceramic materials, shape memory alloys (thermal and ferromagnetic SMA), electrostrictive materials, magnetostrictive materials, electroactive polymers and electro/magneto-rheological (ER/MR) fluids. Each of these materials has their own unique characteristics. For example, a piezoceramic material can change its shape or dimension depending on the application of an electric field across it. Similarly an MR fluid can change its yield stress as a response to any change in the applied magnetic field.

The advantages of smart material systems compared to conventional systems
include

i) No moving parts, High reliability

ii) Low power requirements

iii) High bandwidth

iv) High resolution and accuracy

v) Smart Materials provide new, synergistic capabilities that are presently not possible

Any smart material is fundamentally a transducer. A transducer changes one form of energy into another. These energy form changes can be utilized to devise systems that can be controlled easily and that respond to changes in their environment and adjust their properties so as to perform their ultimate function appropriately.

Of the various smart materials available, this thesis concentrates on the use of piezoelectric actuators. Hence a brief introduction is presented here for the piezoelectric/piezoceramic materials. When all of the so-called smart materials are considered it becomes apparent that piezoceramic actuators are one of the most popular.

Pierre and Jacques Curie discovered piezoelectricity in the 1880's. They discovered that these materials are able to generate electricity when a stress or a surface strain is applied. Since this “sensing effect” was discovered first, it is termed as the direct effect. Another specialty of these materials is the converse effect, where they generate stress or strain when an external electric field is applied. This bi-directional application makes these materials ideal for use as both actuators and sensors. Under an applied field, these materials however generate low strains (0.075-0.150%) but cover a wide range of actuation frequencies.
The most widely used piezoceramics (such as lead zirconate titanate or PZT) and piezo polymers like Poly-Vinylidene Fluoride (PVDF) are in the form of thin sheets, which can be readily embedded or attached to various structures. Bi-morphs or bending actuators are also available commercially where two layers of these materials are stacked with a thin shim (typically of brass) between them. If an opposite polarity is applied to the two plates, a bending action is created. These materials also exist in the form of stacks. In a stack configuration multiple layers of identical piezoceramic wafers are stacked on top of one another. The top and bottom of each layer is alternatively connected to the positive and negative terminals of a high voltage source. This type of connection causes the material to behave electrically like a capacitor. Commercially available stack actuators are able to generate very high forces up to 80 kN and displacements up to 0.3 mm. Higher displacements can be achieved by using a plurality of piezoceramic stacks in series.

Certain electrostrictive materials like Lead Manganese Niobate (PMN) ceramics possess similar characteristics as the piezoelectric actuators but possess more nonlinear characteristics.

The research conducted in thesis mainly utilizes smart materials in the actuator form. The following Table 1.1 compares the effective capabilities of different smart material actuators. It is noted that though SMA actuators are capable of producing very high displacements and high forces, they are very slow in responding due to their thermal actuation mechanism.
<table>
<thead>
<tr>
<th>Actuators</th>
<th>PZT G-1195</th>
<th>PVDF</th>
<th>PMN</th>
<th>Terfenol DZ</th>
<th>Nitinol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actuation Mechanism</td>
<td>Piezoceramic</td>
<td>Piezofilm</td>
<td>Electrostrictive</td>
<td>Magnetostrictive</td>
<td>Shape Memory Alloy</td>
</tr>
<tr>
<td>Free Strain ε max, μ strain</td>
<td>1000</td>
<td>700</td>
<td>1000</td>
<td>2000</td>
<td>20000</td>
</tr>
<tr>
<td>Modulus E 10^6 psi</td>
<td>9</td>
<td>0.3</td>
<td>17</td>
<td>7</td>
<td>4 for (martensite) 13 for (Austenite)</td>
</tr>
<tr>
<td>Band Width</td>
<td>High</td>
<td>High</td>
<td>High</td>
<td>Moderate</td>
<td>Low</td>
</tr>
<tr>
<td>Strain- Voltage Linearity</td>
<td>First order Linear</td>
<td>First order Linear</td>
<td>Non Linear</td>
<td>Non Linear</td>
<td>Non Linear</td>
</tr>
</tbody>
</table>

Table 1.1 Comparison of various Smart Material Actuators

1.3 Literature Review

The literature review conducted for this thesis is split into three major categories, namely i) Available controllable clutch and brake actuation technologies, ii) Piezoelectric actuator system modeling and characterization and iii) Advanced control strategies.

1.3.1 Existing Controllable Clutch and Brake Actuation Technologies

Controllable clutches and brakes are used in many automotive and industrial applications. For example, with the development of Anti-lock Braking Systems (ABS) [Bosch-1995, Wellstead-1997, Hattwig, Maciuca-1995], advanced automatic shifting strategies, and controllable clutching and braking systems play a very important role particularly in the automotive industry. Engineers and researchers have made big strides in developing

However, all these mechanisms have their own advantages and disadvantages. For instance, hydraulically actuated clutches, which are widely used in automobiles equipped with automatic transmissions, have disadvantages like poor efficiency, low robustness due to the varying bulk modulus of pressurized fluid, the requirement of pumping hardware, complicated system design due to intricate fluid passages and valves with moving parts [Haj-Fraj-1999]. Similarly the MR fluid clutches, though simpler compared to the hydraulically actuated clutches, have uncertain operation characteristics at high rotational speeds due to the effect of centrifuging of the micron-sized magnetizable particles [Neelakantan-2002]. Also electromagnetically actuated clutches require large control power due to the presence of an air gap between the rotor and the armature [Bernard-1988, Drew-1992]. Likewise, magnetic hysteresis and eddy current clutches have very low torque to size ratio characteristics [Toukola-1996].

An automotive friction clutch application works by the compression action of two sets of friction discs, one attached to the input shaft and the other attached to the output shaft. For such a typical friction clutch/brake, the required stroke is around 2 to 3 mm and
the required force is around a few kiloNewtons, ranging from about 4 kNs for a typical clutch engagement to about 10 kNs for a brake engagement [Bosch-2000, Morselli-2003, Wang-2001]. Recent developments in the field of smart materials like Piezoelectric and Magnetostrictive materials have led to the development of advanced actuators with low to moderate stroke and high force capabilities. The best piezoelectric actuators that are commercially available are able to produce very high forces of the order of 10 to 80 kN. Their maximum stroke however is limited to around 100 to 300 microns [www.americanpiezo.com, www.physikinstrumente.com].

Previous work involving smart material actuators for clutching applications involves the use of leverages and hydraulic amplification mechanisms for stroke amplification of the actuators [Keeney-2001, Murata-1998]. These methods however lead to difficulties like high mechanical tolerance requirements, lower fatigue life and an equivalent reduction in force capabilities. However, one may utilize the dual stage characteristics of the force stroke relation of a typical clutch/brake actuation process to develop a new actuation mechanism. This concept is discussed in detail in chapter 2. Analogous ideas of dual stage actuation have been used in i) computer disk hard drives to improve bandwidth of the read/write process [Lee-2000] and ii) non-circular cam turning process [Kim-2001]. For the hard disk drives, a voice coil motor is used in conjunction with a Piezo read/write head to maintain the overall range of motion of the Piezo actuator and simultaneously improve overall bandwidth of the system, while for the non-circular cam turning process, a smart material actuator is used in addition to a hydraulic actuator.
1.3.2 Piezoelectric Actuator System Modeling

In general, two different approaches are commonly used in modeling smart material devices and systems. Any model can usually be categorized as being either i) Phenomenological or Macroscopic or ii) Microscopic. The utilization of statistical mechanics provides a bridge between the two approaches. Microscopic models are developed from the analysis of the microstructure of the actual device. These techniques give a microscopic description of macroscopic events and provide a direct relationship between the actual physics and specific parameters explaining the mechanism. This method usually involves the model written in terms of a microstructure and microscopic functions. These models may involve the principles of non-equilibrium thermodynamics or other phenomena. The mean behavior of multiple microscopic elements can then be averaged using the principles of statistical thermodynamics (mechanics) and used to relate the two methodologies. In reality, however, it is difficult to obtain all of the possible parameters in the linear and nonlinear relationships that relate to all possible physics. Additional nonlinearities emanating from the power supply and drive electronics may also be present during operation especially at high drive levels. It is also very difficult to put these physics based models into a framework amenable for real-time control. Although these methodologies are different and there are wide ranging debates on what the best methodology is, a good methodology is defined here as one that matches (to a reasonable degree) a large number of experimental observations while at the same time provides an excellent framework for real-time control. In this particular study we will examine macroscopic models as they tend to provide an excellent balance between
capturing the response of piezoelectric actuators and facilitating real-time control.

The most general form of a macroscopic description is the expression for the free energy or other thermodynamic function. The macroscopic methodology has at its base the reduction of experimental observation into a single phenomenological expression. These expressions are the constitutive relationships that are used in modeling these materials. In other words, a phenomenological model is one where only the input/output relationships are modeled. Phenomenological techniques are subdivided into three sub-categories: thermodynamic phenomenological theories [Hom-1994, Blackwood-1993], semi-atomic phenomenological theories and general (input/output) phenomenological theories [Ge-1995, Ge-1996]. The term "piezoelectricity" was first suggested by W. Hankel in 1881, and the ‘converse effect’ was observed by Lipmann from thermodynamics principles. By 1910, Voigt’s published his work compiling the science of piezoelectricity and it became a standard reference work detailing the complex electromechanical relationships in piezoelectric crystals [Voigt-1910]. However, the complexity of the science of piezoelectricity made it difficult for it to mature to an application until a few years later. Mueller and Devonshire had later developed the one of earliest phenomenological models based on a thermodynamic platform for ferroelectric materials [Devonshire-1949]. Similar analyses were conducted by other researchers that led to phenomenological models relating the input voltage to the output displacement and force of piezoelectric and ferroelectric actuators [Hom-1994, Blackwood-1993]. These free energy based models are also able to account for other anhysteretic nonlinearities by including higher order terms in the expansion of the free energy relations. One of the drawbacks of these models includes the inability of these models to explain or incorporate the
hysteretic behavior of piezoactuators. Recently however, Smith et al have developed models based on Helmholtz and Gibbs free energy to account for the hysteretic behavior of piezoelectric and magnetostrictive materials [Smith-2003, Hatch-2003]. Here they use a piecewise quadratic formulation for the Helmholtz free energy relation with respect to the polarization. This piecewise quadratic relation that comes from a first order approximation of the statistical mechanics-based Helmholtz free energy gives the basic framework to capture the hysteretic behavior. They also have used this model to capture the data of stack actuators and used the model for inversion in position control in an Atomic Force Microscopy application [Hatch-2004]. The difficulty in this method is that it does not account for changes in the system parameters and not just the stack actuators. Also the performance deteriorates when the same model is used for a different actuator. In such cases the model has to be “retuned” considerably to make it work effectively.

Another modeling strategy developed by various researchers is to use the concept of domain walls to explain the hysteretic behavior of piezoelectric actuators [Jiles-1996, Pasquale-1998, Smith-1999, Smith-2000]. The fundamental theory behind this approach to hysteresis modeling in smart materials utilizes the presence of the inherent domain structures in the material. In this model, piezoelectric hysteresis is attributed to the impediment of domain wall movement by material inclusions and stress non-homogeneities inherent to the materials. In other words, hysteresis and other nonlinearities in piezoelectric materials are explained using the theory that energy is required to translate and bend domain walls that separated different domains with different orientation of the polarized ions. This idea is then quantified by splitting the net polarization for a given input electric field into reversible and irreversible
components. This model also uses the anhysteretic polarization relation proposed by Langevin and Ising and accounts for the hysteresis by using the irreversible polarization for any given electric field [Smith-1999]. The disadvantage of this model is again the requirement of retuning the various parameters associated with this model for different actuators and the fact that they do not provide a suitable framework for real-time control. This model also has the inability to accurately capture the hysteresis data at different frequencies using a single set of parameters.

The other popular technique for smart material modeling is the general phenomenological method, because it is accurate, easy to implement, and can be applied to a broad class of nonlinear systems. The Preisach model belongs to this category [Ge-1995]. The Preisach model has received a significant amount of research attention due to the fact that this scheme can accurately account for the hysteretic behavior of piezoelectric actuators. This model utilizes the Preisach function that incorporates a nonlinear hysteresis operator. The method is adapted from magnetic materials to model the behavior of piezoactuators. The other advantage of this method is that this method can accurately capture the higher order hysteresis loops of the piezoelectric actuators. However this model requires a large number of parameters to be stored in the memory to provide accurate results. Recently, Park et al have proposed a hybrid modeling strategy to model piezoelectric hysteresis using neural networks [Park-2004]. This model utilizes the basic framework of Preisach model but the required number of parameters are reduced. Again the effectiveness of these actuators at higher frequencies is yet to be proved.
1.3.3 Advanced Control Strategies

The primary aim for this part of the literature review was to study various advanced feedback control strategies that may be suitable for the control of a piezoelectric actuator system. The main reason behind using feedback control is to ensure desired system performance in the presence of external disturbances, model uncertainties and unmodeled dynamics. The requirement of the feedback control system for the actuation system in this application is to guide the system to track different constant and time-varying reference force signals while being able to maintain robust stability and performance in spite of the nonlinearities like hysteresis and unmodeled dynamics in the system. While PID control is very popular in most industrial applications because of its simplicity in implementation and its ability to eliminate steady state error, it has significant issues in terms of tuning and stability. For instance, in a system with high nonlinearities, unmodeled dynamics and other time-varying parameters it is essential to retune the PID controller frequently to ensure stability of feedback system and consistent performance. Hence advanced robust control strategies need to be developed and implemented for the piezoelectric actuator system to ensure robust stability and consistent performance. The different robust control strategies reviewed for this research include i) Internal Model Control (IMC), ii) Model Predictive Control (MPC), iii) Discrete Time Sliding Mode Control (DSMC) and iv) a new concept called Model Predictive Sliding Mode Control (MPSC).

Internal Model Control or IMC is a popular control strategy used widely by chemical and process control engineers [Morari-1989]. By late 1970s, process control engineers and
researchers were driven towards the conclusion that the classical optimal control theory was ineffective in controlling chemical processes. This was because certain properties of chemical processes rendered direct application of the classical optimal control theory impractical and sometimes impossible. Some of the characteristics of a typical chemical process including unknown disturbances, model uncertainties, strong nonlinearities, time-varying parameters, a lack of complete understanding of the physics behind the process and constraints and limits on various system states and control inputs meant that it was impossible to obtain good models of the plant that will facilitate the implementation of conventional optimal control theory. This lead to the development of a new control structure called the IMC. A clear advantage of IMC as compared to the classical controller structure is the fact that IMC provides a well-structured method to include the plant model along with its process uncertainty and constraints explicitly in the control system [Braatz-1995]. IMC also provides a clear trade-off between system performance and model uncertainty in closed loop system [Morari-1989]. In addition, IMC provides guaranteed closed-loop stability even in the case when the control input is saturated as long as the plant and the controller are internally stable. Frank has discussed the origins of IMC in his dissertation work [Frank-1974]. Morari and Zafiriou have discussed the details of the IMC design and specific PID tuning rules using IMC in their book [Morari-1989]. Henson and Seborg have compiled a survey of various approaches to design IMC for nonlinear processes [Henson-1991]. Braatz has provided a very comprehensive write-up on IMC where he briefly discusses the origin of IMC, its analysis and its advantages over the classical controller structure [Braatz-1995]. Smith has discussed the implementation of dead-time compensation in feedback systems using an unique
control structure that has been later described using IMC [Smith-1957, Palmor-1995]. Rivera have discussed the tuning of PID controller to provide robust stability and performance based on IMC for certain systems including those modeled by first order and second order systems with time delays [Rivera-1986]. Since a piezoelectric actuator system shares some critical and common characteristics with a chemical process like unmodeled dynamics, time varying parameters, control input saturation and system constraints it is envisioned that IMC will be a suitable strategy for controlling smart material systems. An approximate model using a bounded delay and gain has been proposed by Tsai et al for a piezoelectric actuator system [Tsai-2003]. This model is attractive to researchers because it is based on the actual reason for hysteresis in piezoelectric actuators, which is inherent delay due to domain wall shifting. It also has a form that makes it amenable for use in robust control system design using Internal Model Control (IMC)). IMC has however been very scarcely used in mechanical and electromechanical systems. Al-Mamun et al have discussed one such example [Al-Mamun-2002]. This research conducted in this thesis will investigate the utility of IMC in the field of smart material systems and control.

Model Predictive Control or MPC is another popular control strategy that is currently used widely in many chemical and process control industries. As explained before classical optimal control theory had a lot of limitations when applied to process control industry. But the fundamental concept of optimal control, wherein one is able to produce control actions based on minimizing a cost function was still very attractive to researchers and engineers in the chemical process industry. MPC is one such modified optimal control algorithm that produces sub-optimal control actions at each and every
instant of the system response. While PID controllers give very good response in a majority of industrial applications, they fall short as explained before in systems that: are open-loop unstable, have non-minimum phase portions, have time delays, are multivariable in nature and are highly nonlinear. MPC is one of the available concepts that can tackle these situations with relative ease. The working procedure of MPC is to use a discrete time model for the dynamic system that allows a receding window prediction of future outputs based on the current value of the output and future value of inputs and use this model to arrive at optimal value of future inputs to the system that is best with respect to a predefined cost criterion. MPC is currently known to be useful in a wide variety of applications ranging from petrochemical and glass industry to aviation and robotic systems [Richalet-1993]. Pike et al. have developed a comprehensive write-up on the concept of MPC in its various forms including Generalized Predictive Control (GPC) and Dynamic Matrix Control (DMC) [Pike-1995]. Soeterboek has compiled the various ideas using the concept of MPC in his book where he clearly defines the differences in the various forms using the types of model used and the way the cost function defined and optimized [Soeterboek-1992]. Many other researchers have also discussed the implementation of various forms of MPC including Model Algorithmic Control, Extended Horizon Adaptive Control, Extended Predictive Self Adaptive Control, Performance Functional Control and Adaptive Predictive Control System [Richalet-1978, Martin-Sanchez-1976]. Garcia et al. have presented a survey on the theory and practice of MPC in various applications, highlighting its effectiveness in handling multiple constraints and requirements in industries like the petrochemical industry [Garcia-1989]. This is one advantage of MPC compared to the classical LQG regulators where
control and state constraints cannot be included directly [Ogata-1999]. Another excellent source for MPC is the book by Rossiter, where he explains the philosophy behind MPC more clearly and uses well-defined notations that helps the reader grasp analysis and ideas easily [Rossiter-2003]. Rossiter also discusses the computational load of MPC due to the fact that the optimization is carried out at each and every sampling instant. This issue of high computational load will not adversely affect the implementation of MPC in most chemical process industries since they are all relatively slow systems with large time constants on the order of minutes and low sampling frequencies. However, performing nonlinear constrained optimization online in real-time at each and every sampling instant for a much faster system like a piezoelectric actuator system, wherein the settling time is on the order of fraction of a second, is impractical. Hence the MPC concept must be appropriately adapted so that it can be used in comparatively faster plants that employ high bandwidth smart material devices including piezoelectric actuator systems.

Sliding mode control (SMC) is one of the most widely used nonlinear control schemes popular for its robustness and order reduction properties [Utkin-1977; Utkin-1999]. However, it is usually not possible to exactly meet all the requirements of the ideal continuous-time SMC. The application of continuous-time SMC requires perfect switching of the control action on either side of the sliding mode [Utkin-1977; Utkin-1999]. Unmodeled dynamics usually neglected in system modeling include sensor dynamics, minor nonlinearities and disturbances can cause problems in the convergence of the sliding mode. These lead to the phenomenon of chattering, which is the oscillation of the system states about the sliding surface with a finite amplitude and frequency
Chattering often results in poor system performance and actuator degradation with time. Different methods have been proposed in overcoming the phenomenon of chattering. These include the boundary layer method and the observer-based solution [Young-1999]. Another method of overcoming the phenomenon of chattering is the application of discrete-time sliding mode control (DSMC) [Utkin-1999; Su-2000]. DSMC is studied and used more often these days since most of the modern-day controllers are implemented using digital computers via A/D and D/A converters and microcontrollers that function with a finite sampling rate. A different approach is utilized for applying DSMC based on the discrete-time model of a dynamic system to eliminate discretization chatter. In DSMC, usually the control law is designed to force the system to reach the sliding surface at the very next sampling instant [Utkin-1999]. This DSMC method provides a \( O(T^2) \) (an order of \( T^2 \)) convergence to the sliding mode when the system disturbances are known at each time instant as compared to the \( O(T) \) convergence that is obtained by directly digitizing the discontinuous controller of the continuous-time SMC. The problem of unknown disturbances in DSMC is addressed by using a unit-step delayed disturbance estimate that uses the value of the disturbance at the previous time instant \((k-1)\) for the disturbance at the current instant \((k)\) leading to a \( O(T^2) \) convergence to the sliding mode [Su-2000]. However, this method assumes three important requirements, namely i) the bound on the disturbance and its derivative, ii) availability of all system states and iii) independence of the disturbance on control input. The first assumption is valid in most cases, while the second and third are not. In many practical cases, where the assumptions are invalid, the \( O(T^2) \) convergence of the system to the desired sliding mode is not guaranteed using the conventional DSMC approach,
where the sliding mode is enforced at the very next instant ($S_{k+1} = 0$) [Su-2000]. Hence there is a need for modifying the DSMC law to enforce sliding mode in such cases. Moreover, it is neither necessary nor optimal to force the system to reach the sliding mode at the very next instant since this usually results in saturating the controller or over-compensating the system [Utkin-1999].

As discussed before, Model Predictive Control is a robust discrete-time control methodology that explicitly uses the system model. Its abilities to produce optimal control action at each step, to enhance tracking performance and to handle MIMO, unstable and non-minimum phase systems remain the main attractions of this methodology [Pike-1995]. This thesis will discuss the development and implementation of ways to integrate both these popular control methodologies (DSMC and MPSC) to arrive at a new control strategy (MPSMC) that aims to derive the advantages of both the original methods. The idea is to force the system to reach the sliding surface in an optimal manner as compared to the regular DSMC, wherein the system is forced to reach the sliding mode at the very next step, while the optimization is carried out using the MPC strategy. Recently, researchers have worked on developing a ‘predictive sliding mode control methodology’ [Corradini-1997, Wang-2000, Zhou-2001, Partel-2002].

*Corradini et al.* have proposed a predictive variable structure control methodology for systems with discrete-time transfer function models [Corradini-1997]. But the drawbacks of this method include failure to provide robustness in the presence of model uncertainties and unmodeled disturbances and inability to deal with MIMO systems. *Wang et al.* have proposed a predictive control methodology to control a hot strip mill. However, the predictive part of their method is used in observing the states to
overcome time-delays while the discrete sliding mode controller still has the conventional formulation [Wang-2000]. Zhou et al. have worked on developing a dual mode control algorithm using MPC and DSMC. In their algorithm, MPC is used to drive the system to a predefined terminal region and conventional sliding mode control is used while the system states are within the terminal region [Zhou-2001]. However, the disadvantages of the standard DSMC discussed previously still holds for this method. The method proposed by Camacho et al. uses a partial optimization of the cost function with respect to only the continuous part of the controller. However, it is the discontinuous part that drives the system to the sliding mode and hence the effect of MPC is not fully utilized in this method and the results do not demonstrate the effectiveness of this method over conventional DSMC. Also this method fails to account for the effect of model uncertainties and unmodeled disturbances in the system [Partel-2002].

In this thesis, new methods of integrating DSMC and MPC that lead to new concepts of Model Predictive Sliding Mode Control (MPSMC) strategies are proposed that can account for unmodeled disturbances in the system and drive the system to the desired sliding mode through a sub-optimal trajectory. The methods proposed can also handle multiple-input-multiple-output (MIMO) systems since a state-space approach is used.

1.4 Research Objectives

The primary objectives for this research are

1) To develop a new concept of high stroke and high force actuation mechanism using “Smart Material Actuators”.
2) To model and design the new actuation mechanism based a predefined design requirement

3) To build a prototype of the actuation system capable of producing high stroke and force

4) To develop different advanced control algorithms to effectively control the actuation mechanism to track different time varying reference force requirements

5) To implement the different control algorithms in the setup and test the actuation mechanism and compare results

1.5 Outline of Subsequent Chapters

Chapter 2 discusses in the detail the stroke and force requirement of a typical clutch and brake actuation process. The chapter then presents the new concept of a two-stage actuation mechanism that utilizes the unique stiffness curve of a clutch/brake engagement process. Chapter 3 presents the modeling and design and development of this two-stage actuation mechanism and the experimental test setup. The chapter also presents some simple test results used to validate the model and a method to estimated force using charge information. Chapters 4, 5 and 6 discuss the development and implementation of Internal Model Control, Model Predictive Control and a new control strategy called Model Predictive Sliding Mode Control respectively. Chapter 7 details the summary of the work performed in this dissertation and highlights the scope for future work.
CHAPTER 2

TWO-STAGE CLUTCH AND BRAKE ACTUATION MECHANISM

This chapter describes the conceptual development of the two-stage actuation mechanism capable of producing high forces and high stroke suitable for a typical clutch or brake actuation process. First, the requirements of a typical clutch or brake actuation process are described with particular emphasis on the unique stiffness characteristic. Then the chapter highlights the drawbacks and limitations of the work of previous researchers in developing alternative clutch and brake actuation mechanisms including those using smart material actuators and a new two-stage actuation methodology is introduced. Then different possible design embodiments using the two-stage concept are presented. The chapter also extends the concept for both clutch and brake actuation mechanisms.

2.1 Clutch/Brake Actuation Requirements

A typical automatic transmission uses a plurality of hydraulic clutches. These hydraulic clutches are variably engaged and disengaged to transmit power from the engine via different paths using two planetary gear-sets to obtain different gear ratios between the
input and output shafts of the transmission. These hydraulic clutches, as explained in the previous chapter, are engaged by using a high-pressure fluid medium to compress a set of clutch plates connected alternately to input and output elements of the clutch-pack. The process of engagement is explained here. Initially, the clutch-pack and the clutch-plates are a distance-apart.

Figure 2.1 Schematic showing Hydraulic Clutch before and after engagement

Figure 2.1 shows the schematic of the engagement of a hydraulic clutch. In the left, the clutch is in the unengaged position. The valve for the hydraulic medium is closed and the hydraulic piston is in the relaxed position with the return spring in the unperturbed state. In this state, the clutch plates are also a distance apart. Hence in this state there is no power transmitted from the input to the output shaft. The figure on the right side shows the clutch in the engaged position. Here the valve for the hydraulic medium is open and
the high-pressure fluid pushes the hydraulic piston, extending the return spring and compressing the input and output clutch plates together. This transmits the power from the input shaft to the output shaft of the clutch.

The two main requirements of a clutch engagement-process are the required force and the required displacement.

![Figure 2.2 Clutch Force Requirement Calculation](image)

Using Figure 2.2, the equation for the torque transmitted by the clutch is derived as follows. The torque transmitted by the radial element of thickness ‘\(dr\)’ at a radius ‘\(r\)’ is given by

\[
\begin{align*}
  dT &= rdF = r\mu NP(dA) \\
  &= (2.1)
\end{align*}
\]

where \(dF = \mu NP(dA)\) is the shear force produced by the elemental area, and is equal to the product of the friction coefficient (\(\mu\)) of the friction lining between the sliding
surfaces of the input and output clutch plates, the number of clutching surfaces \((N)\), the hydraulic fluid pressure \((P)\) and the elemental area \((dA)\). This gives

\[dT = 2\pi\mu NPr^2 \, dr\]  \hspace{1cm} (2.2)

Hence the expression for the total torque is given by

\[T = \int dT = \int_{R_i}^{R_o} 2\pi\mu NPr^2 \, dr\]  \hspace{1cm} (2.3)

\[T = \frac{2}{3} \pi\mu NP\left(R_o^3 - R_i^3\right)\]  \hspace{1cm} (2.4)

Hence the compressive force required for the clutch pack, to transmit a required torque \(T'\) is given by

\[F = PA = P\pi \left(R_o^2 - R_i^2\right)\]  \hspace{1cm} (2.5)

where \(P\) pressure of the hydraulic medium and \(A\) is the effective area of the clutching surface. Here it is reasonably assumed that the piston dimensions are identical to the clutch surface dimensions. Combining equations (2.4) and (2.5) to eliminate \(P\), we have

\[F = \frac{3T}{2\mu N} \left(\frac{R_o^2 - R_i^2}{R_o^3 - R_i^3}\right)\]  \hspace{1cm} (2.6)

Hence for a design torque of 300 Nm, and using the data from a typical hydraulic clutch, where \(\mu = 0.1, N = 10, R_o = 10\, cm, R_i = 2.5\, cm\), we have \(F = 4.2\, kN\). The stroke required for the engagement of the clutch from the unengaged position usually varies from 3 to 5 mm, depending on the initial position of the piston and the clutch-pack design.

Thus the fundamental requirements of the clutch actuation mechanism are i) **Very high forces** of the order of kNs and ii) **Relatively high stroke/displacement** of the order of a few mm. Similar numbers can be arrived for a typical brake actuation mechanism, where
the high pressure hydraulic fluid pushes the brake pads against the rotor. For example, a typical braking system must be able to produce at least 10 kN of clamping force. The stroke requirement is again around a few mm.

2.2 Alternative Clutch/Brake Actuation Mechanisms

As discussed before, the merits of the hydraulic clutch actuation mechanism include

- Widely used for long time
- Large force and stroke
- Allows Smooth Vehicle Launch
- Simple Concept
- Built-in Cooling System

The demerits of the hydraulic clutch actuation mechanism include

- Continuous pumping action required
- Pump losses
- Uncertainty in Piston Stroking
- Low Robustness due to varying fluid properties
- Moving parts like valves make system bulky and reduce bandwidth
- Difficulty in Control
- Multifunctional Fluid Properties required
- Extraneous pumping hardware

Hence researchers have worked for years to develop various alternatives to the hydraulic clutch actuation mechanism to overcome the limitations of the hydraulically actuated
clutches. This section briefly discusses the different concepts of alternative clutch actuation mechanisms and compares their relative merits and demerits.

2.2.1 MRF (Magneto-Rheological Fluid) Clutch

The MRF clutch contains a suspension of micron-sized magnetizable iron particles in a base fluid like silicone oil, called the Magneto-rheological fluid (MRF) [Neelakantan-2002, Gopalswamy-2001]. The primary property of the fluid is a variation in its yield stress in response to change in an applied magnetic field, which is utilized to devise a controllable MRF clutch. The MR fluid is contained between two surfaces one connected to the input and the other connected to the output. The Magneto-rheological fluid clutch can be either cylindrical or disc shaped. An electromagnetic coil housed inside the clutch is used to generate the necessary magnetic field in the active region containing the MR fluid. By controlling the current in the coil, the yield stress of the MR fluid changes thereby varying the bonding strength between the input and the output. This is realized as a controllable change in the torque capacity of the MRF clutch. The number of surfaces may be increased to increase the torque capacity of the clutch for a given size. A novel MR fluid clutch operating in drag-flow mode is described in the report; this operating mode potentially increases the torque capacity of the clutch. Disadvantages of the MR fluid clutch design include the effect of centrifuging, as the particles tend to move outward at high rotational speeds, settling of the MR fluid, requirement of commutating device to connect rotating coils to the stationary power supply, and the requirement for a separate cooling system.

30
2.2.2 Magnetic Particle Clutch

The magnetic particle clutch is similar in operation to the MR fluid clutch but here the MR fluid is replaced by loose iron particles [Linke-1971]. Off-state clutch drag is nearly zero due to the fact that the particles move freely at zero magnetic field-strength, thereby disconnecting the input and the output. Drawbacks of this magnetic particle clutch include the need for a cooling system, lower torque capacity and very high control power requirement.

2.2.3 Electromagnetic Clutch

Electromagnetic Clutches operate on the principle of electromagnetic attraction [Geldec-1986]. This device consists of three main parts: namely the rotor, the armature and the field-coil assembly. The rotor is coupled to the input shaft, and one side of the rotor faces the armature with a small air gap between them. The sides of the rotor and the armature facing each other are coated with friction material. The armature and the rotor are made with a magnetically conducting material like low-carbon steel. The field-coil assembly is used to produce the necessary magnetic field in the air gap. When the coil is energized, the generated magnetic field attracts the armature towards the rotor, thereby engaging them. Thus, a controllable amount of torque is transmitted. The number of poles in the magnetic circuit may be increased by having more slots in the rotor and armature faces, allowing a smooth flow of magnetic flux in the circuit. Major disadvantages of the system include cooling system requirement and the noisy engagement of the armature and rotor.
2.2.4 Magnetic Hysteresis and Eddy Current Clutch

A Magnetic Hysteresis Clutch works on the principle of hysteresis, which is the property of magnetic materials to retain their magnetization [Toukola-1996]. This device also consists of three parts, namely the rotor, the hysteresis disc and the field-coil assembly. The hysteresis disc is designed to stay within a slot in the rotor body, with a small air gap between the rotor and the hysteresis disc. As the coil is energized, a magnetic flux is created through the rotor body, the air gap and the hysteresis disc. When the rotor rotates, the rotor and the hysteresis disc are magnetically linked, thereby transmitting torque from the input to the output. The linkage is due to the fact that the hysteresis change rate in the disc is not the same as that in the rotor.

An eddy current clutch consists of an input drum, a field coil and an output rotor. When the input drum rotates to cause slip relative to the output, eddy currents are created which produce a magnetic field that interacts with (or opposes) that produced by the field/coil. A major disadvantage of both these clutches is the extremely low torque capacity. Also, the eddy current clutch cannot transmit any torque at zero slip.

2.2.5 Electrical Clutching and Braking System

DC servomotors have been used to develop conceptual devices to engage/disengage friction discs to transmit a controllable amount of torque from the input to the output [Bay]. The device consists of: two planetary gearsets with the same number of corresponding teeth on the ring, sun and the planet gears; a lead screw that also functions as the second sun gear; two sets of friction discs, one splined to the input and the other to the output; and a controllable DC motor. The planet carriers of both the gearsets are
connected to a common link. The DC motor shaft is connected to the ring gear of the 2nd gearset. By activating the DC motor, a relative differential speed is created between the lead screw and the input shaft. This differential rotational speed is converted to an equivalent axial motion by the lead screw and the friction discs are squeezed together to transmit torque from the input to the output. A complex arrangement using two planetary gearsets is a big disadvantage of this device.

In braking, DC motors have been used to provide regenerative braking action in some electric and hybrid vehicles [Wicks-1997, Chuanwei-2004, Cikanek-2002]. During braking, the DC motors act as generators providing resistance to the wheels. Also since the wheels rotate until the kinetic energy of the vehicle is expended through the generator, the vehicle energy that is usually wasted in the form of heat in conventional braking systems is utilized to recharge the battery. Such regenerative braking systems increase the overall efficiency of the vehicle tremendously and therefore are an attractive solution to efficient braking needs. Completely regenerative braking is limited by certain factors, like the critical velocity while decelerating, the state of charge of the battery and its ability to get re-charged quickly [Paterson-1993]. For example, after the critical velocity, the generation of electricity for return to the battery is outweighed by the need to decelerate quickly. At this point, regenerative braking becomes proportional to the vehicle's velocity, therefore allowing the vehicle to coast to a stop rather than coming to a controlled stop. To achieve quick and controlled braking, a mechanical/hydraulic brake must be applied immediately after slowing down below the critical velocity. However, the switch between the two different braking systems, namely the regenerative braking by DC generators and the mechanical braking has been a tricky issue faced by engineers and
researchers. Abrupt switching will induce uncomfortable jerks felt by the driver and the passengers in the vehicle. Various researchers have been working on developing different means to avoid these sudden jerks. Also one other limitation is the ability of the generator-based braking to provide a high bandwidth control for an advanced braking technology like ABS since the stalling of the generator needs to be taken into account.

2.2.6 Smart Material Clutches and Brakes

Smart materials are gaining popularity by the day and are already finding applications in many day-to-day applications. Conceptual clutch and brake designs using smart material actuators have been developed by previous researchers [Keeney-2001, Murata-1998, Chung, Thornley-1991].

![Smart Material Actuators](image)

Figure 2.3 Clutch actuation concept based on piezoelectric actuator [Keeney-2001]
Figure 2.4 A brake actuation concept based on magnetostrictive actuator [Murata-1998]

In these concepts Piezoelectric or Magnetostrictive actuators are used to generate the necessary axial force and motion to squeeze two sets of friction discs to transmit torque from the input to the output. The major hurdle in realizing practical designs based on these concepts using such smart actuators is that the maximum displacement obtained from these actuators is presently in the micron range (hundreds of microns) and the maximum force produces is on the order of 10’s of kiloNewtons. But the motion needed to engage/disengage the friction discs is around a few millimeters while the force required is around the order of at least a few kiloNewtons. Although the axial motion requirement may be met using hydraulic and linkage amplifying mechanisms, one will have to compromise on the force levels. When the displacement is amplified as shown in Figure 2.3 and Figure 2.4 the force capability of the system reduces equivalently and hence more actuators are required to satisfy the net requirement of the clutch/braking
application. Moreover, the extremely low displacements of the smart material actuators put a bottleneck on the tolerance requirements of the linkage and hydraulic amplification mechanisms.

2.2.7 Pneumatic Clutches

Pneumatic Clutches are mainly found in automobiles with manual transmissions [Ellenberger-1991, Richard-1997, Kamio-1984]. These clutches require a compressed air source. In manual transmissions, an electro-pneumatic modulator variably connects the pressurized air source through a duct to the pneumatic cylinder, depending on the position of the clutch pedal. The piston in the pneumatic cylinder pushes the clutch plates away from each other thereby disengaging the clutch. The clutch is engaged in the default position using bias springs. These are also used in pneumatic brakes. The main demerit is the requirement of a high-pressure cylinder in the vehicle.

2.3 Two Stage Clutch and Brake Actuation System

This section develops the new two-stage actuation mechanism concept and describes the basic system construction using different design embodiments.

2.3.1 Two-Stage Actuation Concept

The merits and demerits of various different controllable clutch and brake actuation mechanisms have been compared in the previous section. This section introduces a new two-stage actuation concept for clutch and brake actuation mechanism using smart material actuators. Though the overall force and stroke requirement of a clutch and brake actuation mechanism have been specified to be around a few kiloNewtons and a few
millimeters respectively, a closer look at the actual stiffness curve of the clutch actuation process is necessary to design an appropriate clutch actuation mechanism.

The force-stroke requirement of a typical clutch/brake actuation process is shown below in Figure 2.5. One may see that the force stroke requirement may be split into two regions. In region I, the required stroke is high but the force required to produce the stroke is relatively low. Physically, this region represents the movement of the brake pad or the input clutch pack close enough to the rotating disc or the output clutch pack respectively. In other words, a major part of the stroke is used to move the clutch pack or the brake pad close to the other end of the device while the force required to perform this
action is comparatively lower. The shallow slope of this region I is due to the reason that the piston essentially compresses or extends the return spring in the hydraulic clutch actuation mechanism. In region II, the required force increases to a very high value within a small increment in the stroke. This region represents the stage when the brake pad or the input clutch pack is compressed against the rotor or the output clutch pack respectively to produce the necessary braking or clutching action. The stiffness in this region is comparatively a lot higher that the stiffness in the region I since the friction lining in the clutch-pack and the brake pad are made of very stiff, hard materials.

Thus the entire actuation process can be split into two phases with contrasting force-stroke requirements. Hence two different actuation devices may be utilized to satisfy the requirements in the two phases. For instance, in the first phase, classical DC motors may be employed to provide the required large stroke with relatively lower force. In the second phase, piezoelectric actuators may be used to produce the required high force with the relative smaller stroke and thereby completing the actuation process. Two different actuators can be coupled together using a mechanical coupling device either in a series or a parallel mode as shown in Figure 2.6 below. However, in series mode both the actuators necessarily have to bear the same load and similarly in parallel mode, mode both the actuators have to see the same displacement. However this limitation can be overcome by using appropriate coupling mechanisms with special properties that will allow the system to produce the required force-displacement characteristic as shown in Figure 2.5.
Three different actuation embodiments are proposed here; two for clutch actuation systems and one for braking. Each one uses DC motors for the first phase and piezoelectric actuators for the second phase of actuation. When two actuators are used in series, it is imperative that both see the same load. However, this rule may be broken to our advantage by using a lead screw or a worm gear type of coupling between the motor and piezoactuator. Both these coupling mechanisms are known to possess a *self-locking* property. By virtue of this property, the input and output members of the coupling cannot be reversed. For instance, in the lead screw-nut assembly, the lead screw can drive the nut but the system will lock when the nut tries to drive the lead screw.

Potential advantages of this mechanism include

- More precise control
- Higher bandwidth
- Elimination of hydraulic circuit and pump hence reduces bulkiness
- Elimination of hydraulic losses, for improved fuel economy.
- Lower power consumption (No current or power required to maintain engagement)
- Simpler overall design that the existing hydraulic mechanism
Potential disadvantages/problems of the mechanism include

- Packaging
- Temperature Effects and Cooling
- Cost
- High voltage source for piezoelectric actuators

Analogous ideas of dual stage actuation have been used in i) computer disk hard drives to improve bandwidth of the read/write process [26] and ii) non-circular cam turning process [27]. For the hard disk drives, a voice coil motor is used in conjunction with a Piezo read/write head to maintain the overall range of motion of the Piezo actuator and simultaneously improve overall bandwidth of the system.

2.3.2 Two Stage Clutch/Brake Actuator - System Description

Figure 2.7 and Figure 2.8 show two embodiments of the proposed mechatronic clutch actuation mechanism [Neelakantan-2004]. In both embodiments, the stroke phase is operated by DC motors and the force phase is provided by piezoelectric actuators.

In the first embodiment, shown in Figure 2.7, the input shaft (1) on the left has a clutch plate (2) with a friction lining (3) rigidly attached to it. The output shaft (9) has a clutch plate (22) connected to it through splines (10). Hence the clutch plate (22) can slide axially on the output shaft (9) while rotating with it. A plurality of DC motors (5), (15) are used for the stroke mode while a plurality of piezoelectric actuators (11), (12) are used for the force mode. In the disengaged state, the clutch plates (2), (22) are a small distance apart and the control efforts to the piezoelectric actuator (11), (12) and the DC Motors (5), (15) are null. Though the drawing shows a plurality of the same devices, for
the purpose of explaining the operating principle, let us consider only one set of the devices. To engage the clutch, first the DC motor (5) is activated. The motor shaft rotates through an angle depending upon the control voltage command to the DC motor. As the motor shaft rotates, a worm (7) and pinion gear (6) connection converts the motor shaft rotation into the pinion gear (6) rotation. The pinion (6) also meshes with a rack (8), thereby converting rotary motion of the pinion (6) into an equivalent linear motion of the rack. As it moves to the left, the rack (8) pushes against a thrust bearing (4), which in turn pushes the clutch plate (22) on the output shaft (9) towards the clutch plate (2) on the input shaft (1). Piezoelectric actuator (11) is positioned between the rack body (8) and the thrust bearing (4) such that in the deactivated state, the piezo actuators fit exactly in the gap between the thrust bearing and the body of the rack. When the two clutch plates (2), (22) are brought together so that they are in contact, the DC motor (5) is held in its position and the piezoelectric actuator (11) is activated to a desired voltage. The left end of the piezoelectric actuator (11) pushes on the thrust bearing (4) with a very high force typical of the blocked force values of state-of-the-art piezoelectric actuators. Here the worm (7) and pinion gear (6) connection between the motor shaft and the pinion plays an important role. As the piezoelectric actuator is actuated, the base of the actuator - attached to the body of the rack - applies a high reaction force to the rack body, which in turn tries to back-drive the pinion. However, the worm (7) and pinion gear (6) connection allows one-way drive only. Hence only the motor (5) can drive the pinion. The system locks when the pinion tries to drive the motor shaft. Therefore, the rack does not move and acts as a *stiff* ground element. The axial reaction thrust is absorbed by the thrust bearing (14); provision must be made to allow the load to pass down the motor shaft and

41
into the thrust bearing. The piezoelectric actuation leads to the squeezing of the clutch plates (2), (22) with high force, thereby engaging the clutch. The function of the thrust bearing (4) is to uniformly load the clutch plates (2), (22) and to isolate the rotating clutch plates (2), (22) from the non-rotating elements like the rack (8) and the piezoelectric actuator (11).

Figure 2.7 Two-stage clutch actuation - embodiment 1
Consider now the second embodiment in Figure 2.8. The basic configuration is the same as for the first embodiment. The following description is confined only to the changes in the second embodiment. For explanatory purposes, we shall again consider only one set of the devices shown. The motor (15) is actuated using a voltage command. As the motor shaft rotates, a lead-screw-nut assembly (17) with a lead screw (18) and a nut (16) converts the rotary motion of the motor shaft into an equivalent linear motion of the nut (16). To achieve this, the nut (16) is constrained from rotating by a support (20). The left end of the nut is rigidly attached to the base of a piezoelectric actuator (15). The actuating
end (left end) of the piezoelectric actuator is attached to the right end of the thrust bearing (4). The left end of the thrust bearing (4) is attached to the output clutch plate (3). As the motor rotates, the clutch plate (3) is moved into engagement with clutch plate (2). When the clutch plates are in contact, the piezoelectric actuator is activated to a desired voltage. The piezoelectric actuator consequently pushes on the thrust bearing which in turn squeezes the clutch plates (2) and (3) together, thereby increasing the torque transmitted from the input shaft (1) to the output shaft (10). The self-locking property of the lead screw and nut assembly stops the nut from back driving the lead screw. The axial thrust is transmitted to the motor shaft and then to the thrust bearing (14). Magnetostrictive actuators may also be used in place of the piezoelectric actuators in both designs shown above.

Figure 2.9 Two-stage actuation mechanism for braking applications
The same embodiments described above may be adapted for stationary clutch or brake applications with simpler configurations. In such applications, the output end of the clutch/brake does not rotate and is always stationary. For example, in a brake, the task is to retard the rotation of the input shaft. With the development of state-of-the-art technologies like brake-by-wire and anti-lock braking systems (ABS), there is a need for easily controllable brake actuation mechanisms. The embodiments discussed above can be easily adapted for this task by removing the thrust bearing (4) and grounding the output clutch plate, preventing it from rotating. A schematic for such a device is shown in Figure 2.9. The thrust bearing may prove to be a limitation in the rotating clutch applications but it can be avoided completely in stationary clutch and brake applications.
CHAPTER 3

DYNAMIC MODELING AND DESIGN OF TWO-STAGE ACTUATION SYSTEM

This chapter discusses the detailed modeling and design of the two-stage actuation system using smart material actuators in combination with DC motors. The chapter discusses the modeling in two phases, the stroke phase and the force phase. The stroke phase concerns the classical DC motor model while the force phase discusses in detail the piezoelectric actuator-based model. The chapter discusses in detail the modeling of predominant nonlinearities associated with the system including the hysteresis property of the piezoactuators. The chapter discusses the development of a simple experimental system designed and built to validate the model and test different control algorithms. Simple experimental results are presented at the end substantiating the model for the system. Finally a method to estimate force without the load cell is discussed.

3.1 Dynamic Model Development

The objective of modeling the system is to arrive at a simple model that captures the important characteristics of the system while also providing a framework for easier
robust controller design. Hence the model development is centered on the construction of equations that lend themselves to controller development. The model consists of the integration of the individual models of the following components of the mechanism, DC motor, lead screw and nut assembly, piezoactuator, load (spring element or brake pad) and other coupling elements. Figure 3.1 shows a simple schematic of the two-stage mechanism that is to be modeled and designed.

![Figure 3.1 DC Motor and piezoactuator based two-stage high force/high stroke actuator system](image_url)

The modeling is again divided into two parts namely the stroke phase and the force phase. The task of the stroke phase is to bring the clutch plate or the brake pad close to the rotating end of the device. In this phase, the DC motor serves as the actuator. For a given design the nominal final position to which the motor must be driven is known approximately, but the actual location of the final point may vary depending on
factors like wear and tear on the device elements and the stiffness property of the load. This point where the switch from stroke phase to force phase is made is termed the ‘transition point’. The transition from the stroke phase to the force phase is critical because of the extremely small displacements of the piezoelectric actuators. If the transition is made too early, before the brake pads/clutch plates are brought into contact with each other, the force phase becomes ineffective because the stroke of the piezoelectric actuators will be wasted in achieving the additional displacement needed to reach the actual transition point. If the transition point is detected too late, though the force phase functionality requirements can be achieved, it may lead to the damage of the DC motor due to extended stalling. Also if the brake pad or the clutch plate has a relatively low stiffness curve for the initial part, one may have to use the DC motor to provide a high enough preload to effectively use the piezoactuator. The DC motor actuates the lead screw assembly, which in turn drives the piezoactuator and the brake pad or the clutch plate to the other end of the device. The modeling of the second phase of the actuation mechanism involves the piezoelectric actuator model couple with the dynamic system involving the clutch/brake pad mass and stiffness. The important nonlinearity in the system, namely the hysteresis property of the piezoelectric actuator is to be modeled and designed. The following subdivisions discuss the modeling of the stroke and force phases individually in detail.

3.1.1 Analysis of Stroke Phase Dynamics

The task of the stroke phase is to bring the clutch plates close together. For a given design the nominal final position to which the motor must be driven is known
approximately, but the actual location of the final point may vary depending on factors like wear and tear on the device elements. The transition from the stroke phase to the force phase is critical because of the extremely small displacements of the piezoelectric actuators. If the transition is made too early, before the clutch plates are brought into contact with each other, the force phase becomes ineffective because the stroke of the piezoelectric actuators will be wasted in achieving the additional displacement needed to reach the actual transition point. If the transition point is detected too late, though the force phase functionality requirements can be achieved, it may lead to the damage of the DC motor due to extended stalling.

![Reference Speed Profile for DC Motor](image)

**Figure 3.2 Reference Speed Profile for DC motor**

The ideal case in the stroke phase would be to control the motor to track a smooth speed profile thereby reaching the final 'transition point' smoothly with exactly zero
speed. Since the exact position of the 'transition point' is not accurately known, a simple compromise may be made in the reference speed profile. Instead of making the final speed at the nominal 'transition point' exactly zero, one can make it close to zero so that the 'transition point' is always reached. If the difference between the actual and the nominal 'transition point' is negligible, the effect of the above-mentioned compromise on the stroke-phase duration is also going to be insignificant.

The reference speed profile may be chosen similar to the one shown in Figure 3.2. Given the small final speed, once the actual 'transition point' is reached, the motor will not be able to rotate further since the torque capacity of the motor is not high enough to squeeze the clutch-pack any further. This results in a sudden drop in the motor speed to zero. However, the motor must be able to provide enough preload in the stroke phase to make the force phase effective. Also the actual position of the final transition point is expected to vary a lot as the clutch-pack or the brake pad wears out. Hence the speed profile shown before will not be an effective way to carry out the stroke phase quickly. One other simple way is to run the DC motor at a predefined constant voltage that essentially drives the system quickly to the transition point.

This 'transition point' can be detected through different means as listed below.

- Using a speed sensor on the motor shaft
- Using a force sensor at the end of the piezoelectric actuator end
- Using the "direct effect" property of the piezoelectric actuator
- Using the motor current information

The motor current, which is an electrical quantity, is easier to measure compared to other mechanical quantities like force or displacement. Utilizing motor current
information as a sensing strategy is applied in production vehicles equipped with power windows driven by DC motors, where the polarity of the applied voltage is reversed once the motor current starts to increase above a pre-defined threshold value. Similarly, the current spike is sensed at the transition point when the system DC motor begins to stall.

The function of the motor is to move the clutch pack plates close together. As the motor is driven by commanding a voltage signal, the lead screw-nut assembly converts the rotary motion into linear motion. Hence the load on the motor is due to the following elements of the device:

- Rotor shaft inertia of the motor
- Lead screw shaft inertia
- Nut mass
- Piezoelectric actuator mass
- Thrust bearing mass
- Clutch pack or Brake Pad mass
- Friction

The first two items in the list above are rotary loads and can be included directly in the DC motor equations. The remaining items on the list are linear loads and their rotary equivalents as seen by the motor have to be calculated. The following analysis shows the derivation for the equivalent load inertias. The relationships for the lead screw-nut assembly is given by

\[ x = \left( \frac{L_e}{2\pi} \right) \theta \]  

(3.1)
\[ \eta = \frac{Fx}{T_m \theta} \]  \hspace{1cm} (3.2)

where, \( x \) = displacement of the nut, \( L_e \) = lead of the lead screw-nut assembly, \( \theta \) = rotation of the motor/lead screw shaft, \( \eta \) = efficiency of the lead screw-nut assembly (usually 30-50\%), \( T_m \) = the motor torque, \( F \) = resulting axial thrust on the nut. Equation (3.2) can be rewritten as

\[ F = \eta T_m \left( \frac{\theta}{x} \right) = \eta T_m \left( \frac{2\pi}{Le} \right) \]  \hspace{1cm} (3.3)

As the nut translates axially, it carries the rest of the mass along with it. By denoting the sum of the masses of the nut, the piezoelectric actuator, the thrust bearing and the clutch-pack as \( M_L \), the inertial load can be written as

\[ I_L = M_L \ddot{x} \]  \hspace{1cm} (3.4)

Considering this as a thrust load on the nut, its equivalent rotary load torque as seen by the motor may be written based on equation (3.3) as

\[ T_{eq-L} = I_L \left( \frac{x}{\theta} \right) \frac{1}{\eta} = \left( M_L \ddot{x} \right) \left( \frac{x}{\theta} \right) \frac{1}{\eta} \]  \hspace{1cm} (3.5)

Equation (3.1) can be extended to the derivatives of the displacement \( x \) and the rotation angle \( \theta \). After some algebraic manipulation we can write the equivalent inertial load as seen by the motor as

\[ J_L = \left( \frac{Le}{2\pi} \right)^2 \frac{M_L}{\eta} \]  \hspace{1cm} (3.6)

The net inertial load on the motor may then be equated to the sum of the inertias of the motor shaft, lead screw shaft and the equivalent load \( J_L \) as calculated in equation (3.6).

Therefore
\[ J_{\text{total}} = J_{\text{motor}} + J_{\text{lead-screw}} + J_L \]  \hspace{1cm} (3.7)

Backlash in the lead screw and nut assembly must also be modeled in order to account for the nonlinearities in the system. Backlash is unavoidable in a mechanical system with lead screws or gears. A simple backlash model may be given based on describing functions or other time domain models but this backlash is expected to affect only the stroke phase where relatively low force is seen and the critical feedback control is mostly performed in the stroke phase using the Piezoelectric actuator.

The DC motor model is well known and is governed by two simple equations, one for the motor circuit and the other for the torque balance. The DC motor equations are

\[ J_{\text{total}} \ddot{\theta} + f_r \dot{\theta} + T_L = Ki \]  \hspace{1cm} (3.8)

\[ L \frac{di}{dt} + Ri + K \dot{\theta} = U \]  \hspace{1cm} (3.9)

where \( J_{\text{total}} = \) the total inertial load on the motor and is defined later in equation (3.7), \( R = \) motor armature resistance, \( L = \) motor armature inductance, \( K = \) motor constant, \( f_r = \) damping friction coefficient, \( i = \) motor current, \( U = \) motor control voltage. Assuming that only current measurement is available, an observer/estimator may be designed to observe and thereby track the motor speed. By writing the above equations in State-Space form, we have

\[
\begin{bmatrix}
\frac{d}{dt} (i) \\
\frac{d}{dt} (\dot{\theta})
\end{bmatrix} =
\begin{bmatrix}
-\frac{R}{L} & -\frac{K}{L} \\
\frac{K}{J_{\text{total}}} & -\frac{f_r}{J_{\text{total}}}
\end{bmatrix}
\begin{bmatrix}
i \\
\dot{\theta}
\end{bmatrix} +
\begin{bmatrix}
\frac{1}{L} \\
0
\end{bmatrix} U +
\begin{bmatrix}
\frac{T_L}{J_{\text{total}}}
\end{bmatrix}
\]

\hspace{1cm} (3.10)

\[ y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} i \\ \dot{\theta} \end{bmatrix} = i \]  \hspace{1cm} (3.11)
The term $T_r$ is the additional load-torque term included to model stalling of the motor at the transition point. The controllability and the observability matrix show that the above system is completely controllable and observable which means that complete information about the system states may be found by measuring only the current. The operating strategy of the first phase of the actuation process using the DC motor is to apply a constant voltage to the DC motor until the transition point is reached. The motor hence drives the nut of the lead screw assembly carrying the piezoelectric actuator and the brake pad forward towards the load cell in the setup shown in Figure 3.1.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J_{total}$</td>
<td>Total Inertial Load</td>
<td>1.5e-4 kg.m$^2$</td>
</tr>
<tr>
<td>$L$</td>
<td>Motor Inductance</td>
<td>5.8 mH</td>
</tr>
<tr>
<td>$R$</td>
<td>Motor Winding Resistance</td>
<td>2.3 ohms</td>
</tr>
<tr>
<td>$K$</td>
<td>Motor Constant</td>
<td>0.115 N.m/amp</td>
</tr>
<tr>
<td>$L_e$</td>
<td>Lead of Lead Screw</td>
<td>0.1 inch</td>
</tr>
<tr>
<td>$\eta$</td>
<td>Lead Screw Efficiency</td>
<td>40% normal and 75% stall</td>
</tr>
</tbody>
</table>

Table 3.1 Parameter Values for Stroke Phase Design

The transition point where the first phase of the actuation process is stopped and the second phase is initiated is determined using the characteristics of the DC motor wherein the current drawn by the motor increases rapidly during stalling. Hence when the
point where the DC motor is unable to push the brake pad any further is reached, there will be a current spike and a simple current sensor available with the power supply is used to sense this sudden spike. The controller instantaneously compares the current value to a predefined current limit and when the current drawn crosses this limit, it stops the voltage supply to the DC motor. This provides a preload on the brake pad compressed against the load cell. The preload is necessary to make the piezoelectric actuator effective. This preload value is determined using the predefined current limits and depends to some extent on the applied voltage. Since the DC motor has very reliable torque-current characteristics, the force generated for a given applied voltage and current limit remains constant even if the spring constant of the brake pad changes. The lead screw usually has around 40% efficiency while rotating and this efficiency is close to 75% during stalling, when there is hardly any rotation involved. This is taken into account while specifying the applied voltage and current limit. An appreciable amount of preload is to be provided by the DC motor since the brake pad combined with the other elements in the device possesses lower stiffness during the initial few hundred microns of stroke. Table 3.1 shows the important parameters and their values for the stroke phase elements.

3.1.2 Analysis of Force Phase Dynamics

This section concerns the design and modeling of the second phase of the actuation process and is the most critical since the piezoactuator produces most of the required force in the actuation process. Once the transition point is detected, the force phase is executed using piezoelectric actuators.

Piezoelectric actuators are smart material devices that produce motion and stress
depending upon applied electrical input (electric field). The etymology of piezoelectric materials comes from the Greek word “Piezo” meaning pressure or stress. Piezoelectric actuators come in various forms including unimorphs, bimorphs, stack and tube actuators [www.physikinstrumente.com, www.americanpiezo.com]. Commercially available stack actuators are able to produce forces up to 80 kN and strokes up to 0.3 mm. Hence these type of actuators are suitable for the two-stage actuation system.

The force exerted by the piezoelectric actuator compresses the clutch-pack or the brake pad. The final operating/settling actuator displacement of the force phase is determined by the intersection of the force-displacement curves of the piezoelectric actuator and the clutch-pack/brake pad stiffness. The piezoelectric actuator produces a very high force when the actuator is blocked from producing any stroke. For a given voltage command, the force capability of the actuator decreases to zero linearly as the allowed stroke increases to its maximum value. The maximum force produced is called “blocked force” and the maximum stroke produced is called “free displacement/stroke”.

Figure 3.3 illustrates the properties of piezoelectric actuator acting against a spring load. As shown, the force-displacement output of the piezoelectric actuator increases approximately linearly with the applied voltage. Hysteresis in piezoelectric actuators is estimated to be around 10-15% and can be compensated for by using closed-loop control.

The expressions for the equilibrium operating point location are

\[
F_{eq} = F_{bl} \left( \frac{K_{spring}}{F_{bl} + K_{spring}} \right), \quad x_{eq} = \frac{F_{eq}}{K_{spring}}
\]

(3.12)
Figure 3.3 Force-Displacement characteristic of piezoelectric actuator loaded against a spring

Figure 3.4 Simple model for Force Phase dynamics
The clutch-pack dynamics during the force phase can be modeled using a mass-spring-damper-actuator system. The friction in the system may be assumed to be a small value and can be determined through experiments. Figure 3.4 shows a simple model of the force phase. Based on the model shown in Figure 3.4, the equation of motion for the system may be deduced as follows. In Figure 3.4 and equations below, \( M_b, K_b, b \) and \( x \) represent the mass, clutch-pack stiffness, friction/damping in the system and the displacement of the clutch-pack respectively. Starting from the constitutive relations for Piezoelectric actuators, we have

\[
T_i = c_{ij}^E S_j - d_{kl} c_{jl}^E E_k
\]

where \( T \) represents the applied stress, \( S \) represents the strain in the piezo and \( E \) represents the applied Electric field, \( c \) represents the material modulus at constant \( E \) and \( d \) is the Piezoelectric constant. In this case, all the stresses and strain may be assumed to be acting only in the 3 direction and the applied stress is from the spring and the damping in the system. Here \( T \) must equal the total dynamical stress from the system including the mass, stiffness and damping. Hence one may extend the above formulation and write the following dynamical equation describing the system.

\[
M_b \ddot{x} + b\dot{x} + K_b x = F_{\text{piezo}}
\]  

(3.13)

\[
F_{\text{piezo}} = F_{bl} \left( 1 - \frac{x_b}{x_f} \right)
\]

(3.14)

\[
\Rightarrow M_b \ddot{x} + b\dot{x} + \left( K_b + \frac{F_{bl}}{x_f} \right) x = F_{bl}
\]

(3.15)

The blocked force \( F_{bl} \) and the free displacement \( x_f \) of the piezoelectric actuator are
functions of the applied voltage and their ratio is the apparent stiffness \((K_p)\) of the actuator. Assuming linear relationships, we have,  
\[ F_{bl} = \frac{A d_{33} c_{33} E}{t} V \quad \text{and} \quad K_p = \frac{A c_{33} E}{n t}, \]
where \(A\) is the surface area of the piezoelectric stack actuator, \(d_{33}\) is the piezoelectric constant, \(c_{33}\) is the material modulus of the actuator, \(n\) is the number of layers in the stack actuator and \(t\) is the thickness of each layer in the actuator. As is evident, the relation between the applied voltage and the blocked force of the piezoelectric actuator is of utmost importance here. Though linear relations between the applied voltage and blocked force or free displacement are used above, this is not valid due to the nonlinearities of piezoactuators like hysteresis. Different models have been used to describe the piezoelectric actuator characteristics. As discussed in the first chapter, all piezoelectric models can usually be categorized as phenomenological or microscopic. The following section will focus on developing various advanced models for piezoelectric actuators that account for nonlinearities like hysteresis.

### 3.2 Piezoelectric Actuator Modeling

The different models to be considered here include i) Domain-Wall Model, ii) Preisach Model and iii) Variable Delay Model. Each one of these models are discussed in detail in the following subdivisions.

#### 3.2.1 Domain-Wall Model

Piezoelectric materials contain micron-sized regions, generally referred to as the Weiss domains, which possess a permanent dipole moment. The orientation of such
dipole moments in different Weiss domains in an un-poled piezoelectric material is random and, hence, there is no net dipole moment at the macroscopic length scale. However, the application of a high electric field or a high stress during the process known as ‘poling’ causes the dipoles of different Weiss domains to align themselves with the principal direction of the applied field or stress, which gives rise to the polarization of the piezoelectric material. When the applied field/stress is removed, the domains tend to regain their initial polarization state, but this reversal is generally not complete and, as a result, a remnant polarization is generally found in the poled piezoelectric materials. Once piezoelectric materials are poled, the application of an electric field (generally substantially smaller in magnitude than the one used for poling of the material) can produce mechanical strains.

Due to the ferroelectric nature of piezoelectric materials, they inherently possess hysteretic and other nonlinear characteristics like saturation. Also the hysteresis behavior of piezoelectric actuators is predominant at high drive levels. When the applied electric field is varied, the different Weiss domains in the piezoelectric material move to orient the dipoles along the applied electric field. During this movement the walls of the neighboring Weiss domains slide along each other. According to models proposed by various researchers, piezoelectric hysteresis is attributed to the impediment of the domain wall movement due to material imperfections and nonhomogenities. When there is no applied field, pinning sites are formed at the locations of these material imperfections. At low applied field, it has been experimentally demonstrated that the domain wall movement is reversible and this is associated with the bending of the domain walls. However at high input fields, the energy barrier linked to the pinning sites is
overcome and the domain walls slide for longer distances. This translation leads to irreversible mechanisms contributing to the hysteresis property of piezoelectric actuators. In the domain wall model, models describing the anhysteretic polarizations are utilized. The hysteresis is then modeled by quantifying the energy required to bend and translate the Weiss domains about the pinning sites. By employing Boltzmann statistics to specify the probability of dipoles occupying certain energy states, and assuming the balance of thermal and electrostatic energies while using the assumption that the material is isotropic and the orientation of cells can be in any direction yields the Langevin equation [Smith-1999]

$$P_{an} = P_s \left[ \coth \left( \frac{E_e}{a} \right) - \left( \frac{a}{E_e} \right) \right]$$

for the anhysteretic polarization, where $E_e$ is the effective electric field acting on dipole moments in the material, ‘$a$’ is a constant depending on the operating temperature and the Curie temperature of the actuator and $P_s$ is the saturation polarization of the material.

A second expression derived using this Boltzmann approach is the Ising spin relation [Smith-1999]

$$P_{an} = P_s \tanh \left( \frac{E_e}{a} \right)$$

which results from the assumption that the dipole moments can occupy only two discrete orientations, either in the direction of the applied field or opposite to it. Depending on the material that we are trying to model, appropriate model can be chosen from the above. The total polarization in piezoelectric actuators for any applied Electric field can be written as a sum of reversible and irreversible polarizations. Both these reversible and
irreversible polarization values are dependent on the value of the anhysteretic polarization at any given instant and other parameters like the change and direction of applied Electric field.

For any given field ‘\(E\)’, that causes an irreversible Polarization ‘\(P\)’, the effective field may be written as

\[
E_e = E + \alpha P
\]  
(3.18)

However, since the total polarization at any instant is not known exactly, we can approximate this equation by replacing the total polarization by ‘\(P_{irr}\)’. Hence we have

\[
E_e = E + \alpha P_{irr}
\]  
(3.19)

In order to apply the equation (3.19), we need to assume a starting level of irreversible polarization. Hence for a given anhysteretic polarization, one may obtain the equation relating the corresponding irreversible polarization by subtracting the energy losses at the pinning sites from the energy required to achieve the anhysteretic polarization. After differentiation and reformulation for quantifying the reversal points, this leads to the following equation.

\[
\frac{dP_{irr}}{dE} = \frac{P_{an} - P_{irr}}{k\delta - \alpha(P_{an} - P_{irr})}
\]  
(3.20)

where the terms \(\delta\) and \(\tilde{\delta}\) are defined as follows.

\[
\delta = \text{sign}(dE)
\]  
(3.21)

This enforces the condition that energy required to overcome the pinning sites is always against the changes in polarization.
\[
\tilde{\delta} = \begin{cases} 1, & \text{if } dE(P - P_{an}) < 0 \\ 0, & \text{otherwise} \end{cases}
\]  

(3.22)

Hence equations (3.20) to (3.22) can be used to update the value of irreversible polarization at the next instant that is required to compute the effective field. The reversible polarization models the effect of domain wall bending. This can be mathematically modeled as

\[
P_{rev} = c(P_{an} - P_{irr})
\]  

(3.23)

where ‘c’ is determined based on the material properties to be modeled. Finally, the total polarization is given by

\[
P = P_{rev} + P_{irr}
\]  

(3.24)

\[\Rightarrow P = cP_{an} + (1 - c)P_{irr}\]  

(3.25)

Example Simulation: Consider a PZT material (PZT-5A) with the following values for the parameters \(c = 0.3\), \(k = 1.8e6\) V/m, \(a = 4.2e5\) V/m, \(\alpha = 3.6e6\) Vm/C and \(P_s = 0.49\) C/m². Figure 3.5 shows the result of the simulation of the domain-wall model for the PZT-5A actuator. It is shown to capture the actual experimental data quite closely [Smith-2000]. However, it is not trivial to use this approach to model unipolar actuators like the stack actuators where the electric field is applied only along one direction varying between 0 and a maximum value. Though unipolar actuators can be treated as a regular piezoactuators with biased polarization, it is however not simple to use the domain wall model to capture their hysteresis property accurately. Also the parameters used in the model are not scalable and have to be retuned considerably to accurately capture the properties of different piezoelectric actuators.
3.2.2 Preisach Model

The classical Preisach model can be used to describe the hysteresis behavior of piezoceramic materials [Ge-1995]. The Preisach model consists of a summation of hysteresis relay operators, which have either a +1 or -1 output value depending on an input value and weighing functions, which describe the relative contribution of each relay to the overall hysteresis [Park-2004]. The mathematical form of the Preisach model is given as follows:

\[
 x(t) = \int_{\alpha \geq \beta} \mu(\alpha, \beta)\gamma_{\alpha\beta}[u(t)]d\alpha d\beta 
\]  

(3.26)

where
\( x(t) \): the output response of a piezoceramic actuator (displacement)

\( \mu(\alpha, \beta) \): a weighing function in the Preisach model

\( \alpha \) and \( \beta \): “up” and “down” switching values of the input respectively (\( \alpha \geq \beta \) (volts))

\( \gamma_{\alpha\beta}[u(t)] \): a hysteresis operator

For a piezoceramic actuator, \( x(t) \) is the output representing the displacement generated by the voltage of the input \( u(t) \).

![Figure 3.6 (a) Classical hysteresis operator (b) Modified hysteresis operator](image)

The classical Preisach model has been used successfully to model magnetic materials [Mayergoyz-1991]. In the case of magnetic materials, the value of the hysteresis operator is -1 or 1 according to the value of the input \( u(t) \), as shown in Figure 3.6 (a). The operator output and input values are determined by the position of the system or material response in the planar quadrants [Khan-2002]. Since the hysteresis loop of the piezoceramic actuator is defined in the first quadrant of the u-x plane, the value of the hysteresis operator, \( \gamma_{\alpha\beta}[u(t)] \), used for the piezoceramic actuator is modified to 0 or 1,
as shown in Figure 3.6 (b). In equation (3.26), the double integration can be illustrated as a summation of weighted relays connected in parallel as shown in Figure 3.7.

![Figure 3.7 Schematic interpretation of the Preisach model](image)

The output of the hysteresis operator $\gamma_{\alpha\beta} [u(t)]$ is obtained by applying the input $u(t)$ subject to the “up” and “down” switching values of the input $\alpha$ and $\beta$. When the input, $u(t)$, increases from $\beta_0$ to $u_1(t)$ along the $\alpha$-axis, the shaded region $S^+(t)$ is formed, as shown in Figure 3.8. In this region, all the hysteresis operators $\gamma_{\alpha\beta} [u(t)]$ are switched to the “up” position, because switching values $\alpha$ on the vertical axis in are less than $u_1(t)$. For example, $\alpha_1$, $\alpha_2$, and $\alpha_3$ are less than $u_1(t)$. When the hysteresis operator $\gamma_{\alpha\beta} [u(t)]$ is in the “up” position, the output of the operator is equal to 1 as shown in Figure 3.6 (b). On the other hand, all the hysteresis operators which belong to $S(t)$ are equal to 0. Thus, $S$ region is not considered in calculating $x(t)$.

Now Equation (3.26) can be reduced as follows:
\[ x(t) = \int_{\alpha \geq \beta} \mu(\alpha, \beta) y_{\alpha \beta} [u(t)] d\alpha d\beta \]

\[ = \int_{S^+} \mu(\alpha, \beta) y_{\alpha \beta} [u(t)] d\alpha d\beta + \int_{S^-} \mu(\alpha, \beta) y_{\alpha \beta} [u(t)] d\alpha d\beta \]

\[ = \int_{S^+} \mu(\alpha, \beta) \times (1) d\alpha d\beta + \int_{S^-} \mu(\alpha, \beta) \times (0) d\alpha d\beta \]

\[ = \int_{S^+} \mu(\alpha, \beta) d\alpha d\beta \tag{3.27} \]

Figure 3.8 Shaded region \( S^+ \) which contains the hysteresis operators of +1 value

Figure 3.9 and Figure 3.10 show the hysteresis loop of a piezoceramic actuator in two different configurations. In configuration (a), the \( \alpha \)'s decrease with time and the value of \( \beta \) is zero. In configuration (b), both \( \alpha \) and \( \beta \) change with respect to time.

Figure 3.9 (a) input voltage (\( \alpha \) decreases and \( \beta \) is zero) (b) input voltage (\( \alpha \) and \( \beta \) both decrease)
Figure 3.10 (a) Hysteresis loop having the common switching value for $\beta$ and different switching values for $\alpha$ (b) Hysteresis loop showing several switching values for both $\alpha$ and $\beta$

Figure 3.11 Comparison of Preisach Model Results with Experimental Data for PZT stack PSt 1000/16/80 VS25 type, Piezomechanik GmbH

Notice that the $\alpha_i$'s are defined for increasing values of input voltage $u(t)$ and the $\beta_i$'s are defined for decreasing voltages. For further detailed information on Preisach model and its development, the readers are referred to the following references [Ge-1995, Park-
2004]. However one result of Preisach model comparing the model results to the experimental data is presented here [Park-2004].

The limitations of Preisach model include i) Need for a large number of numerical data to be stored in the computer memory, ii) Non-scalable parameters, meaning the model has be retuned again considerably for a new actuator in order for the model to work effectively.

### 3.2.3 Variable Delay and Gain Model

A simpler but inaccurate model may be used to design and implement robust control strategies. A small number of researchers have recently used a variable time delay model for representing the behavior of piezoelectric actuators [Tsai-2003]. Here the input to output relation is given by

\[
\frac{X(s)}{U(s)} = H(\cdot)G(s)
\]

where the operator \(H(\cdot)\) is defined as a variable time delay and variable gain model. A similar but more detailed approach is followed here that illustrates the ability of this model to capture the hysteretic property of the piezoelectric actuator. Most of the modeling is performed based on the free displacement since it is very difficult to measure the blocked forces. It is again noted that the free displacement and the blocked force are related by a proportional constant for any given applied voltage. This reasonable assumption is valid since the piezoelectric actuator stiffness does not vary much. The following analysis and example shows that this model can capture the actual case fairly
accurately for certain cases and hence may be used to design an appropriate robust controller using the IMC methodology. To arrive at the overall model of the force phase, we look at a typical response of an unloaded piezoactuator to a biased periodic sinusoidal input as shown in Figure 3.12. Based on Figure 3.12, the hysteresis property may be modeled as a proportional system with a variable gain and variable delay. Looking at the major hysteresis curve, we see that the time delay ($\tau$) starts from zero when the input voltage is zero, increases to a maximum value when the input voltage is midway to its maximum value and then decreases back to zero when the input voltage reaches its maximum value. This phenomenon of variable delay is shown in Figure 3.12. Similarly, the average gain decreases as the input voltage amplitude decreases.

![Figure 3.12 Schematic of the Variable delay phenomenon](image)

Mathematically, the relation between the free displacement and the input voltage may be written as

$$x(t) = k\left(U_{max}\right)u\left(t - \tau(t, \omega)\right)$$

$$X(s) = k e^{-\tau s} U(s)$$

where ‘$k$’ and ‘$\tau$’ vary but may be bounded by upper and lower limits. Though
these parameters are time-varying, it is reasonable to treat them as constants with uncertainties to accommodate controller design. This approximation has been proved to be valid by [Lincoln, 2004]. The result in the paper by Lincoln gives a simple result on the stability of a closed loop linear system with time varying delay and the result does not put any constraint on the way the time-delay varies with time and only requires the upper bound on the time-delay. Since this model is essentially used to develop and implement a closed loop control system, the result in the paper by Lincoln is used here. The input voltage follows a sinusoidal law. Thus

\[ u(t) = \frac{U_{\text{max}}}{2} \left(1 - \cos(\omega t)\right) \]  

(3.31)

According to the model, let us assign a time-varying delay

\[ \tau(t, \omega) = \text{abs}(\tau_{\text{max}} \sin(\omega t)) \Rightarrow 0 \leq \tau \leq \tau_{\text{max}} \]  

(3.32)

Hence we see that the time delay as defined in equation (3.32) depends upon time and frequency and satisfies the properties stated above. Also the gain changes as the maximum voltage value \((U_{\text{max}})\) changes. Hence \(k(U_{\text{max}}) = k_o \left(1 + \varepsilon U_{\text{max}}\right)\)  

(3.33)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\frac{KK_pK_b}{M_b})</td>
<td>3.534e6 N/(V.s^2)</td>
</tr>
<tr>
<td>(\zeta)</td>
<td>0.75</td>
</tr>
<tr>
<td>(\omega_n)</td>
<td>1.113e4 rad/s</td>
</tr>
<tr>
<td>(\bar{k})</td>
<td>6.33e5 mm/V</td>
</tr>
<tr>
<td>(\bar{k})</td>
<td>8.76e-5 mm/V</td>
</tr>
<tr>
<td>(\tau)</td>
<td>0 s</td>
</tr>
<tr>
<td>(\bar{\tau})</td>
<td>0.0125 s</td>
</tr>
</tbody>
</table>

Table 3.2 Parameter Values in Force Phase
Figure 3.13 Hysteresis Curves from Experimental Measured Data

Figure 3.14 Hysteresis Curves based on Variable Delay and Variable Gain Model
Comparing Figure 3.13 and Figure 3.14, we see that the model recaptures the actual hysteresis data in this case of sinusoidal voltage input. Hence this simple model may be used to develop advanced robust control strategies to control the system. Hence the overall model of the system shown in Figure 3.4 in terms of the voltage input to the piezoactuator and generated force may be written as

\[
\frac{F}{U_s}(s) = \bar{p}(s) = \frac{K_p K_b}{M_b} \left( \frac{k}{s^2 + 2\zeta\omega_n s + \omega_n^2} \right) e^{-\tau s}, \quad \frac{\bar{k}}{\bar{\tau}} \leq k \leq \bar{k}, \frac{\bar{\tau}}{\bar{\tau}} \leq \tau \leq \bar{\tau} \quad (3.34)
\]

For this system, the nominal values (after the preload of the brake pad) are listed in the table below. The generated force \(F\) is assumed to be proportional to the displacement of the spring/brake pad, the proportional constant being the spring constant \(K_b\) of the brake pad.

### 3.3 Experimental Setup

A simple experimental setup is built following the design principles discussed in detail in the previous sections. Figure 3.15 shows the experimental setup. The setup consists of a controllable DC motor capable of producing large enough torque to provide the necessary preload driving a lead screw-nut assembly via a coupling and an axial bearing. The nut of the lead-screw assembly is constrained to move axially without rotating. This nut is connected to a piezoelectric actuator (purchased from APC International Inc.) capable of producing 12.5 kN of blocked force and 105 \(\mu\)m free displacement for a voltage range of –200 to 1000 V. The free end of the piezoelectric actuator is connected to a brake pad, which pushes against a load cell at the end of the setup. This experimental setup is used for two main purposes, namely,
i) Model verification and

ii) Testing and validation of advanced control system strategies

Two controllable DC power supplies are used to command different control voltages to the DC motor and the piezoelectric actuator. The power supply for the DC motor is capable of taking in analog control signals from 0 – 10 V and outputs a proportional voltage from 0 – 150 V. This power supply also has a maximum current limit of 8 amps and this is acceptable for the DC motor. The power supply for the piezoelectric actuator is capable of producing –1000 to + 1000 V for a corresponding analog control signal of –10 to + 10 V. This power supply has a current limit of +/- 45 mA and is acceptable for the piezoelectric stack actuator since they demand very little current even at frequencies of up to 15 – 20 Hz.
3.3.1 Model Verification

The models derived for the stroke phase and the force phase are verified here. For the stroke phase, a simple open loop control is applied for the DC motor with a constant control voltage of 30 V. A simple controller is built using MATLAB/Simulink and implemented using a DSpace 1102 board. Initially the brake pad and the piezoelectric actuator are a certain distance away (3-5 mm) from the load cell. Once the system is switched on, the DC power supply powers the DC motor. This in turn actuates the lead screw and nut assembly. The nut constrained to move axially pushes the piezoactuator and the brake pad forward towards the load cell. The controllable DC power supply senses the instantaneous current drawn by the DC motor and this signal is input to the controller using the analog I/O interface on the DSpace board. Once the load cell is reached the DC motor begins to stall and the current drawn by the DC motor increases rapidly and produces a preload on the brake pad. The controller cuts-off the voltage to the DC motor once a predefined current limit is reached. The final preload value depends on this current limit for the DC motor. Different current limits are set in the controller to obtain different preload force at the end of the stroke phase. Figure 3.16 shows the result of the stroke phase testing running the DC motor at 30 V input and different current limits. The value of 30 V is chosen (as opposed to higher voltages) so that the impact at the end of the stroke is not very fast to minimize the chance of any damage to the piezoactuator. The DC motor draws a minimum of around 1.15 amps to move the system forward. Hence there is no motion and hence no preload achieved for current limits less that 1.15 amps as illustrated in Figure 3.16.
A simple step response test is performed to check if the system response follows that of a second order system. Once a preload of around 1.25 kN is achieved using the DC motor, the piezoactuator is given a step voltage input. This is repeated using a slow square wave voltage input and the resulting generated force response is recorded. The voltage input is stepped up and down repeatedly between –200 to 900 V.

Figure 3.17 and Figure 3.18 shows the generated force response. It is seen that the system follows the typical response of a second order system with overshoot and oscillations. Based on this data, the values of the natural frequency and damping ratio of the
second order system are assigned. Also the open-loop hysteresis curves for the system are determined by applying a periodic sinusoidal voltage to the piezoelectric actuator.

Figure 3.17 Step voltage input
Figure 3.18 Response of piezoactuator-based force phase

Figure 3.19 shows the resulting hysteresis curves. It is seen that the large hysteresis effects give rise to poor tracking performance. The major goal of the controller design is to minimize the hysteresis effects thereby linearizing the closed loop performance without compromising the stability requirement. Another important point to be noted is that the preload from the first phase of the process using the DC motor must be high enough to make the mechanism meet the force requirement. This can however be avoided using a longer piezoactuator with extended stroke capabilities, or a plurality of smaller piezoelectric actuators. This in turn will reflect on the overall cost of the device.
3.4 Force Measurement/Estimation without Load Cell

In this section, a method where the force can be estimated with using the load cell is studied. The basic philosophy is to measure the instantaneous charge stored in the piezoelectric actuator and use it to estimate the force generated by the system in the second phase of the actuation process. It is a well-known property of piezoactuators to follow a hysteresis-free linear law between its free displacement (and blocked force) with respect to the stored charge rather than the applied voltage.

In this method we use a capacitor \( C^* \) in series with the piezoelectric actuator. This
method is shown in the below. This method utilizes the fundamental Kirchoff’s law that capacitors in series carry the same charge at all times.

![Diagram of capacitors in series](image)

**Figure 3.20 Charge Measurement using Capacitor in Series**

The equations concerning this circuit are

\[ V = V_p + V_c \]  
(3.35)

where \( V_p \) is the voltage across the piezoelectric actuator. Obviously, the drop across the measurement capacitor (\( C^* \)) must be negligibly small compared to the applied voltage so that it does not interfere with the system functioning. We may model the piezoelectric actuator as a time-varying capacitance element (\( C_p(t) \)). Hence the overall capacitance of the circuit including the piezoactuator and the measurement capacitor is

\[ C(t) = \frac{C^* C_p(t)}{C^* + C_p(t)} \]  
(3.36)
\[ C(t) = \frac{C_p(t)}{1 + \frac{C_p(t)}{C^*}} \]  

(3.37)

Hence we see that in order for the measurement capacitor to have very little effect in the overall circuit, we require

\[ C^* \gg C_p(t) \]  

(3.38)

The piezoactuators have a nominal capacitance of around 500 nF with a 10-15% variation with respect to time. Hence we may choose values above 500 \( \mu \text{F} \) for \( C^* \).

However, the problem in this method is that the voltage measured across the capacitor decays with time since the charge across this capacitor is dissipated across the leakage resistor and measurement system impedance that is present in closed measuring the circuit. Figure 3.21 and Figure 3.22 show this phenomenon clearly.

Figure 3.21 shows a simple smooth applied voltage to the piezoelectric actuator. As the voltage to piezoelectric actuator is applied, the voltage across the additional capacitor \( (C^*) \) is measured instantaneously. The piezoelectric charge is then estimated by dividing this measured voltage by the capacitor value \( (C^*) \). Figure 3.22 shows the resulting estimated charge. It is evident that the charge across the additional capacitor \( (C^*) \) decays slowly due to the leakage resistance and impedance in the measuring system.
Figure 3.21 Voltage applied across the piezoactuator vs. time

Though this decay is slow in time, this leads to incorrect estimations of the force and will eventually saturate the controller when this measurement system is used in a charge-based feedback control method. We see that though the applied voltage is held constant, the voltage measured across the capacitor ($C^*$) decays to zero. This decay can be modeled as the effect of a high pass filter system as shown in Figure 3.23.
The value of the parameter \( \tau \) is estimated using the decaying charge value plot shown in Figure 3.22. It is useful to note that the transfer function seen above has a relative order of zero and hence it is invertible. So in order to exactly estimate the charge across the piezoelectric actuator, we cascade an additional transfer function, which is the inverse of the transfer function shown in Figure 3.23, to the voltage \( (V_c) \) measured across the capacitor \( (C^* ) \), This is shown in Figure 3.24.
Figure 3.23 Model for decay of voltage across the measuring capacitor (C^*)

Figure 3.24 Inversion method to manipulate actual charge across the piezoelectric actuator

Figure 3.25 compares the estimated charge based on the voltage measured across the
capacitor \( (C^*) \) with the actual charge estimated after regeneration as shown in Figure 3.24.

![Figure 3.25 Comparison of charge before and after regeneration](image)

In the above figure, it is evident that the actual charge estimated stays constant as one expects for the applied voltage shown in Figure 3.21. It is clearly observed that the inversion method to regenerate charge is an effective way to estimate the charge stored in the piezoelectric actuator.

Once the charge stored in the piezoelectric actuators is estimated as discussed above, the next step is to relate this charge to the force output of the system. The system is
modeled as a second order system with the blocked force on the right hand-side as discussed in equation (3.15). The blocked force and the free displacement are known to follow a linear law with respect to the charge stored in the piezoactuator. Hence to estimate the force produced by the system in the force phase of the actuation process, the actual charge estimated above is input to a normalized second order transfer function based on the system model as shown in Figure 3.26.

A sinusoidal voltage input varying between –200 and 1000 volts is applied to the piezoelectric actuator. The dynamic charge output of the system shown above in Figure 3.26 is then plot against the actual force generated, measured using the load cell. Figure 3.27 shows the approximately linear one-to-one mapping between the estimated dynamic charge and the actual generated force in the system measured using the load cell. This one-to-one mapping is an attractive method that allows one to eliminate the load cell in designing and implementing simple and advanced control strategies. For a given reference force signal, one can relate what the required charge must be and the controller can be designed to force the actual dynamic charge to track this value.
Figure 3.27 One-to-one mapping between dynamic charge and actual force
CHAPTER 4

ROBUST INTERNAL MODEL CONTROL OF SYSTEM

This chapter discusses the development and implementation of Internal Model Control for the force phase of the dual-stage actuation system. The chapter begins with an introduction to the concept of Internal Model Control and its merits are compared to the classical controller structure. Then, a tutorial-like review and analysis is presented that will allow the user to extend the design of IMC to various systems. Then a robust IMC controller is developed for the force phase of the two-stage actuation system based on the variable delay model. Finally the experimental results for the dual stage actuation system of the IMC method are shown with detailed discussion.

4.1 Introduction

The second phase of the actuation process involves the piezoelectric actuator and its control. The control objective is to guide the system to track different reference force inputs during the second phase of the actuation process using the piezoactuator. Since the piezoelectric actuator system contain uncertainties due to modeling errors, it is necessary
to develop and implement a robust control strategy to maintain consistent tracking performance. Internal Model Control (IMC) is one such method that is widely used in chemical and process industry where uncertain models are quite common [Morari-1989]. Internal model control relies on the internal model principle, which states that a plant or a process can be controlled only if the control system incorporates or encapsulates, either implicitly or explicitly, some representation of the process. For example in an open loop control, the model of the process to be controlled is almost exactly known. Hence an inverse model is used for the controlling the plant in this case. However, an exact model of the plant is not known in almost all practical cases and process-model mismatch is very common. These uncertainties and unmodeled dynamics in the system usually affect system performance. In such cases Internal Model Control (IMC) is found to be very useful. The general structure of an internal model control methodology compared to the classical controller structure like PID is shown below in Figure 4.1.

Figure 4.1 A Comparison of Classical and IMC Controller Structure

It is noted that the system model is explicitly used in the IMC structure unlike the classical controller structure. Before a detailed analysis of the IMC system is performed, a few important definitions are presented.
**Definition 4.1:** A transfer function $T(s)$ is said to be ‘**proper**’ if there exists a real constant $C < \infty$ such that

$$\lim_{s \to \infty} |T(s)| < C$$

(4.1)

**Definition 4.2:** A transfer function $T(s)$ is said to be ‘**strictly proper**’ if

$$\lim_{s \to \infty} |T(s)| = 0$$

(4.2)

**Definition 4.3:** Any transfer function $T(s)$ that does not satisfy the conditions mentioned in definitions 4.1 and 4.2 is termed ‘**improper**’.

Rational transfer functions $T(s)$, that are ratio of two polynomials $\text{num}(s)$ and $\text{den}(s)$ of degree $m$ and $n$ respectively, are termed proper if $m \leq n$, strictly proper if $m < n$ and improper if $m > n$. Simple examples are listed in table 4.1.

<table>
<thead>
<tr>
<th>PROPER</th>
<th>Lead/Lag Controller</th>
<th>$T(s) = K \frac{s + \alpha}{s + \beta}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>STRICTLY PROPER</td>
<td>Ideal Integrator</td>
<td>$T(s) = \frac{1}{s}$</td>
</tr>
<tr>
<td>IMPROPER</td>
<td>Ideal Differentiator</td>
<td>$T(s) = s$</td>
</tr>
<tr>
<td></td>
<td>Ideal PID controller</td>
<td>$T(s) = K_p + \frac{K_i}{s} + K_ds$</td>
</tr>
</tbody>
</table>

Table 4.1 Examples of Proper, Strictly Proper and Improper Transfer Functions
While proper and strictly proper transfer functions are physically realizable, improper functions are not since improper functions output signals of arbitrarily high amplitude for input signals with arbitrary high frequencies.

**Definition 4.4:** A system is said to be ‘causal’, if the current output does not depend on the future values of the inputs.

In terms of transfer functions, all proper rational transfer functions that can be expressed as a ratio of two finite degree polynomials in ‘s’ are causal, while improper rational functions are not. In other cases, all transfer functions that have a positive time delay \( e^{-Ts}, T>0 \) are causal, while the ones with negative time-delays are not since they require prediction of future inputs to obtain the current output.

### 4.2 Limitations of Classical Controller Structure

We shall now analyze the classical controller structure and explicitly discuss its limitations. Figure 4.2 shows the general structure of a classical control system with reference, disturbance and noise inputs. Though the classical controller \( c(s) \) is designed based on a model of the system \( p(s) \), it is not explicitly used in the controller structure.

In the ensuing analyses, the Laplace variable “(s)” is omitted for brevity and \( R, D, M \) and \( N \) represent the Laplace transform of the inputs \( r, d, m \) and \( n \) respectively. For the classical controller structure shown in Figure 4.2, one may write

\[
Y = R \left( \frac{cp}{1+cp} \right) + D \left( \frac{p}{1+cp} \right) + M \left( \frac{1}{1+cp} \right) + N \left( \frac{-cp}{1+cp} \right) \quad (4.3)
\]

and
Hence for the system to be BIBO (bounded-input-bounded-output) stable with respect to all inputs, we require that the transfer functions \( \frac{c}{1+cp} \), \( \frac{p}{1+cp} \) and \( \frac{pc}{1+cp} \) be stable. For a stable \( p \), the requirement reduces to the stability of \( \frac{c}{1+cp} \). Moreover the sensitivity (\( S \)) and complementary sensitivity (\( T \)) functions of the classical controller structure are given by

\[
S = \frac{1}{1+cp} \quad \text{and} \quad T = \frac{cp}{1+cp}
\]  

(4.5)

For good reference tracking performance, it is desired to have \( T \) close to 1 and for good disturbance rejection, it is desired to have \( S \) close to 0. Both are complementary to each other since \( S + T = 1 \). But for good noise rejection, it is desirable to have \( T \) close to 0, after a certain cut-off frequency. This is fundamental trade-off between system performance.
(T=1) and noise rejection (T=0).

Also when model uncertainties are present, a measure for quantifying the model error must be specified. One usual method is to use a multiplicative uncertainty to the system model. Hence the actual plant model may be written as

\[ p(s) = \tilde{p}(s)(1 + \Delta(s)) \]  

(4.6)

where \( \Delta(s) \) represents the multiplicative uncertainty in the actual plant. Depending on the system model, and its uncertain parameters, frequency-dependent bound may be found on the uncertainty \( \Delta(s) \) such that

\[ |\Delta(j\omega)| \leq \Delta^*(\omega) \]  

(4.7)

Since most models are usually accurate only at lower frequencies, the frequency-dependent uncertainty bound \( \Delta^*(\omega) \) is expected to increase with frequency and eventually cross unity.

Using Nyquist theorem on stability, one may derive the required condition on the robust stability of the closed loop system. Based on Figure 4.3, since the Nyquist plot must avoid encircling the point \( X(-1, 0) \), the required robust stability criterion is

\[ XZ > \rho \]  

(4.8)

where \( \rho = |\tilde{c} \tilde{p}(j\omega)|\Delta^*(\omega) \) and \( XZ = |\tilde{c} \tilde{p} - (-1)| = |1 + \tilde{c} \tilde{p}| \).

Hence the closed-loop uncertain system is BIBO stable if and only if the nominal system is BIBO stable and the following condition is satisfied.

\[ \left| \frac{\tilde{c} \tilde{p}}{1 + \tilde{c} \tilde{p}} \right| < \frac{1}{\Delta^*(\omega)} \]  

(4.9)
Hence the limitations of the classical controller structure are

i) The controller $c(s)$ appears in an inconvenient manner in equation (4.5) so that it is difficult to design it to use it as a scale/measure to incorporate the trade-off between system performance and noise rejection.

ii) The controller $c(s)$ appears in an inconvenient manner in equation (4.9) so that it is difficult to design it to make the system is robustly stable while simultaneously maintaining good reference tracking performance and disturbance rejection.

iii) If the controller has saturation limits, it is not trivial to account for it in the design of $c(s)$ so as to maintain stability of the closed loop system, since saturation of controller is a well-known cause of system instability in the classical control structure.
The IMC structure provides convenient solutions to these limitations of the classical controller. The following section provides a detailed analysis and design procedure of the IMC.

4.3 IMC Analysis

Again, in the ensuing analyses, the Laplace variable “$(s)$” is omitted for brevity and R, D, M and N represent the Laplace transform of the inputs r, d, m and n respectively. Based on the IMC structure shown in Figure 4.4, one may write

\[
Y = R \left( \frac{pq}{1 + (p - \tilde{p})q} \right) + D \left( \frac{p(1 - \tilde{p}q)}{1 + (p - \tilde{p})q} \right) + M \left( \frac{1 - \tilde{p}q}{1 + (p - \tilde{p})q} \right) + N \left( \frac{-pq}{1 + (p - \tilde{p})q} \right) \tag{4.10}
\]

Assuming a perfect model, i.e., $p = \tilde{p}$, we may reduce the above equation into

\[
Y = R \left( pq \right) + D \left( p(1 - pq) \right) + M \left( 1 - pq \right) + N \left( -pq \right) \tag{4.11}
\]

and

\[
\tilde{Y} = R \left( pq \right) + D \left( -p^2q \right) + M \left( -pq \right) + N \left( -pq \right) \tag{4.12}
\]
and \[ U = R(q) + D(-pq) + M(-q) + N(-q) \] (4.13)

Hence based on equations (4.10) to (4.13), one may see that in order for the system to be BIBO stable, one requires \( p(s) \) and \( q(s) \) to be stable. For stable plants, meaning stable \( p(s) \), one is left with a simpler task of choosing any stable \( q(s) \) in order that the nominal closed-loop system is stable. Also we see that the sensitivity \( (S) \) and complementary sensitivity \( (T) \) functions have a convenient form given by

\[ S = 1 - \hat{p}q, \quad \text{and} \quad T = \hat{p}q \] (4.14)

This explicitly shows the trade-off between nominal system performance and model uncertainty in a simple form, which is utilized in the IMC design in the subsequent analysis. By rearranging the IMC system to the classical controller structure, it may be noted that the equivalent classical controller for the IMC structure is given by

\[ c(s)|_{IMC} = \frac{q}{1 - \hat{p}q} \] (4.15)

Now, when model uncertainties are present, using equation (4.15) in equation (4.9) the robust stability criterion for the IMC structure reduces to

\[ \left| \hat{p}q \right| < \frac{1}{\Delta^*(\omega)} \] (4.16)

where \( \Delta^*(\omega) \) represents the frequency-dependent bound on the multiplicative model uncertainty given by equation (4.7). Consequently it is easy to see that, by using IMC, the task of choosing a Internal Model controller \( q(s) \) that satisfies equation (4.16) to provide robust closed-loop stability is simplified to a great extent as compared to the complicated task of choosing a classical controller \( c(s) \) to satisfy equation (4.9).
One other advantage of the IMC structure is its ability to incorporate actuator constraints in its structure. It can be easily proven that when the constrained controller output is sent to the plant and the model in the IMC structure, and when the plant and the model are perfectly matched, the system will always be stable as long as $q(s)$ and $p(s)$ are stable as show. This is illustrated in Figure 4.5.

![Figure 4.5 IMC structure with actuator constraints for stability guarantee](image)

### 4.4 IMC Design Procedure

Now that the advantages of IMC structure over the classical controller structure are known, we proceed to discuss the unique design procedure of Internal Model Control for dynamic systems with and without uncertainties. Two major objectives of the IMC design procedure include i) Nominal Performance and ii) Robust Stability and Performance. Nominal performance implies that the designed IMC controller must satisfy a required performance criterion for the nominal system, when the model exactly matches
with the plant \( p = \tilde{p} \). The performance criterion is usually minimizing the tracking error using a standard measure like Integral Absolute Error (IAE) of Integral Square Error (ISE). Robust stability and performance implies that the system must provide a stable performance with zero steady state error for step or ramp reference inputs even if the uncertain plant varies within a predefined uncertainty bound as discussed in the previous sections. Based on equation (4.11) the tracking error is given by

\[
E(s) = R(s) - Y(s)
\]

or

\[
E(s) = (1 - \tilde{p}q)(R - D\tilde{p} - M) + pqN
\]

Using ISE optimization, we would like to choose \( q(s) \) that minimizes

\[
J = \int_0^\infty (e(t))^2 dt = \int_0^\infty (r(t) - y(t))^2 dt
\]

Using Parseval’s theorem [Gradshteyn-2000], one may write

\[
J = \|E(s)\|^2
\]

where \( \|\cdot\|^2 \) represents the standard 2-norm, i.e.,

\[
\|E(s)\|^2 = \frac{1}{2\pi} \int_{-\infty}^{\infty} |E(j\omega)|^2 d\omega
\]

and \( E(s) \) is given by equation (4.18).

Now, assuming that \( \tilde{p}(s) \) and the Laplace transform of the cumulative input \( R'(s) = (R - D\tilde{p} - M) \) are rational transfer functions without time-delays, we may split them into its minimum phase and non-minimum phase portions. Hence,

\[
\tilde{p}(s) = \tilde{p}_N(s)\tilde{p}_M(s) \text{ and } R'(s) = R_N(s)R_M(s)
\]

where \( \tilde{p}_N(s) \) and \( R_N(s) \) include all zeros in the right-hand side of the imaginary axis. It is
assumed that $R'$ is at least marginally stable since only then the inputs are bounded. It is also convenient if $|\bar{p}_N(s)| = |R_N(s)| = 1$, which will make the derivation of the Internal Model Controller $q(s)$ easier. Mathematically,

\[
\bar{p}_N(s) = \frac{(z_{p1} - s)(z_{p2} - s)\cdots}{(s + z_{p1})(s + z_{p2})\cdots} \text{ and } R_N(s) = \frac{(z_{R1} - s)(z_{R2} - s)\cdots}{(s + z_{R1})(s + z_{R2})\cdots}
\]  

(4.22)

It is noted that the complex conjugate of each of the non-minimum phase zeros are added to the denominators to maintain a unit norm and to account for higher order zeros. It is also noted that $z_{p1}$ and $z_{R1}$ are $> 0$.

The following theorem gives the Internal Model Controller $q(s)$ that minimizes ‘$J$’ given above in equation (4.20) for transfer functions $\bar{p}(s)$ and $R(s)$.

**Theorem 4.1:** For a stable plant given by $p(s)$, the internal model controller $q(s)$ that minimizes ‘$J$’ as defined in equations (4.19) and (4.20), is given by

\[
q(s) = \bar{p}_M^{-1}(s) R_M^{-1}(s) \left\{ \bar{p}_N^{-1} R_M \right\}_*
\]  

(4.23)

where $\left\{ \right\}_*$ represents only the parts which do not contain the poles of $\bar{p}_N^{-1}(s)$ after a partial fraction expansion of the expression.

**Proof:**

This proof can be appropriately extended when noise inputs are also present in addition to the disturbance inputs.

Suppose that there exists a $q_o(s)$ that makes $(1 - \bar{p}q_o)R'$ stable. Then we write

\[
\|E(s)\|^2 = \|(1 - \bar{p}q)R\|^2
\]  

99
\[
= \left\| (1 - \tilde{p} q_o) R' - \tilde{p} (q - q_o) R \right\|^2 \\
= \left\| (1 - \tilde{p} q_o) R_N R_M - \tilde{p} N \tilde{p} M (q - q_o) R_N R_M \right\|^2 \\
= \frac{1}{2\pi} \int_{-\infty}^{+\infty} \left\| (1 - \tilde{p} (j\omega) q_o (j\omega) - \tilde{p} N (j\omega) \tilde{p} M (j\omega) (q (j\omega) - q_o (j\omega)) \right\|^2 \cdot |R_N (j\omega) R_M (j\omega)|^2 d\omega
\]

From now, the argument \((j\omega)\) is omitted for brevity. Using the fact that \(|R_N (j\omega)| = |\tilde{p} N (j\omega)| = 1\), we rewrite the above equation as

\[
\left\| E(s) \right\|^2 = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \left\| \tilde{p}^{-1} N \tilde{p}^{-1} M - \tilde{p} M (q - q_o) \right\|^2 |R_M|^2 d\omega
\]

Now, since \(\tilde{p} M, q\) and \(q_o\) are all stable, and \(R_M\) is assumed to be atleast marginally stable, \(\tilde{p} M (q - q_o) R_M\) must be atleast marginally stable. But since the standard 2-norm is not defined for transfer functions with marginally stable poles, it is imperative that \(\tilde{p} M (q - q_o) R_M\) be stable. Also since \((1 - \tilde{p} q_o) R'\) is assumed to be stable, \((1 - \tilde{p} q_o) R_M\) is also stable. This implies that the only unstable poles of \((1 - \tilde{p} q_o) \tilde{p}^{-1} N R_M\) are exactly the unstable poles of \(\tilde{p}^{-1} N\). Hence to minimize \(\left\| E(s) \right\|^2\), using the projection theorem in Hilbert Spaces [Luenberger-1997], we require

\[
\tilde{p} M (q - q_o) R_M = \left\{ (1 - \tilde{p} q_o) \tilde{p}^{-1} N R_M \right\}_* \\
\]

where \(\left\{ \right\}_*\) represents only the parts which do not contain the poles of \(\tilde{p}^{-1} N (s)\) after a partial fraction expansion of the expression. This satisfies the requirement that
\( \tilde{p}_M (q - q_o) R_M \) be stable. Denoting \( \tilde{q}(s) \) as the optimal \( q(s) \), we have

\[
\tilde{q}(s) = q_o + (\tilde{p}_M R_M)^{-1} \left\{ (1 - \tilde{p}_o q_o) \tilde{p}_N^{-1} R_M \right\}_s
\]

Since the final result is desired to be independent of \( q_o \), we write

\[
\tilde{q}(s) = (\tilde{p}_M R_M)^{-1} \left\{ (\tilde{p}_M R_M) q_o + \left\{ \tilde{p}_N^{-1} R_M \right\}_s - \left\{ \tilde{p}_M \tilde{p}_N q_o \tilde{p}_N^{-1} R_M \right\}_s \right\}
\]

This further reduces to

\[
\tilde{q}(s) = (\tilde{p}_M R_M)^{-1} \left\{ (\tilde{p}_M R_M) q_o - \left\{ (\tilde{p}_M R_M) q_o \right\}_s + \left\{ \tilde{p}_N^{-1} R_M \right\}_s \right\}
\]

This is rewritten as

\[
\tilde{q}(s) = (\tilde{p}_M R_M)^{-1} \left\{ \left\{ (\tilde{p}_M R_M) q_o \right\}_s^* + \left\{ \tilde{p}_N^{-1} R_M \right\}_s^* \right\}
\]

where \( \{ \}_s^* \) represents only the parts with unstable poles. But since \( \tilde{q}(s) \) is required to be stable, we ignore this term and hence

\[
\tilde{q}(s) = (\tilde{p}_M R_M)^{-1} \left\{ \tilde{p}_N^{-1} R_M \right\}_s
\]

Hence proved.

In most practical cases, the expression for \( \tilde{q}(s) \) given by equation (4.23) will be improper and hence cannot be physically realized. Therefore an appropriate filter needs to be cascaded to this expression to make it proper. Hence the final Internal Model Controller \( q(s) \) is given by

\[
q(s) = \tilde{q}(s) f(s)
\]  
(4.24)

where \( f(s) \) is a filter of an appropriate order that is chosen to make \( q(s) \) proper. Also this filter gives us a degree of freedom in maintaining robust stability and robust performance
with zero steady state tracking error. Simple filters that may be used are

\[ f_1(s) = \frac{1}{(\lambda s + 1)^n} \]  \hspace{1cm} (4.25)

and

\[ f_2(s) = \frac{n\lambda s + 1}{(\lambda s + 1)^n} \]  \hspace{1cm} (4.26)

where \( f_1(s) \) is used for zero steady state error for step inputs and \( f_2(s) \) is used for zero steady state errors for ramp inputs and ‘\( n \)’ is chosen appropriately to make \( q(s) \) proper. It is noted that ‘\( \lambda \)’ provides a degree of freedom in maintaining robust stability.

**Special Case 1:** For systems \((p(s))\) that only have minimum phase portions, it can be shown that the final internal model controller (IMC) is given by

\[ q(s) = p_M^{-1}(s) f(s) \]  \hspace{1cm} (4.27)

**Special Case 2:** For systems with time delays, the Internal Model Controller is computed for the part of the plant without the delay. This will remove the delay out of the feedback loop and hence will not affect the stability of the nominal system.

So in a nut shell the design steps for IMC is as follows:

**Step 1:** Using the plant model \( p(s) \) and the reference input \( R(s) \), split them into minimum phase and non-minimum phase portions as shown in equations (4.21) and (4.22).

**Step 2:** Then design \( q(s) \) the IMC using equations (4.23) and (4.24).

### 4.4.1 IMC Design Example 1

Design an IMC \( q(s) \) that tracks a step input without any steady state error for the system

\[ p(s) = \frac{1-s}{(s+2)^2} e^{-s} \]  \hspace{1cm} (4.28)
Using the discussion above (step 1), we have

\[ p_M(s) = \frac{s+1}{(s+2)^2}, \quad p_N(s) = \frac{1-s}{s+1}, \quad R_M(s) = \frac{1}{s} \text{ and } R_N(s) = 1 \]

(4.29)

Hence, using the theorem 4.1 and equation (4.24) (step 2), the controller \( q(s) \) is given by

\[ q(s) = \frac{(s+2)^2}{(s+1)} s \left[ \frac{(s+1)}{(1-s)} \right] \frac{1}{s} \left( \lambda s + 1 \right)^n \]

(4.30)

\[ \Rightarrow q(s) = \frac{(s+2)^2}{(s+1)} s \left[ \frac{2}{(1-s)} + \frac{1}{s} \right] \frac{1}{s} \left( \lambda s + 1 \right)^n \]

(4.31)

Neglecting the term \( \frac{2}{s-1} \) as per the theorem, and choosing \( n = 1 \), we have

\[ q(s) = \frac{(s+2)^2}{(s+1)(\lambda s + 1)} \]

(4.32)

Thus the transfer function from the input \( R(s) \) to the output \( Y(s) \) is

\[ \frac{Y}{R}(s) = \tilde{p}(s)q(s) = \frac{(1-s)}{(1+s)(\lambda s + 1)} e^{-s} \]

(4.33)

This Internal Model Controller will give an offset-free tracking of a step input since \( \frac{Y}{R}(0) = 1 \). Now, the question is how does one choose ‘\( \lambda \)’? Here we utilize the robust

stability criterion given by equation (4.16). This requires the knowledge of the frequency

dependent bound on the multiplicative uncertainty on the plant \( p(s) \). The following

example illustrates this method by finding the frequency dependent bound based on

the predefined uncertainties in the system.
4.4.2 IMC Design Example 2

Design an IMC \( q(s) \) that tracks a step input without any steady state error for the system

\[
p(s) = \frac{K}{(s+1)}e^{-\delta s}, 1-\alpha \leq K \leq 1+\alpha \text{ and } 1-\beta \leq \delta \leq 1+\beta
\]  

(4.34)

where \( \delta \) is in seconds. Based on the average value of the uncertain system parameters \( K \) and \( \delta \), we write the nominal plant model as

\[
\tilde{p}(s) = \left( \frac{1}{s+1} \right)e^{-s}
\]  

(4.35)

To obtain the frequency-dependent bound on the multiplicative uncertainty, we write

\[
\Delta^*(s) = \left| \frac{p_{\text{worst}}(s) - \tilde{p}(s)}{\tilde{p}(s)} \right|
\]  

(4.36)

where \( p_{\text{worst}}(s) = \frac{1+\alpha}{(s+1)}e^{-(1+\beta)s} \). Hence we write

\[
\Delta^*(s) = \left| \frac{1+\alpha}{(s+1)}e^{-(1+\beta)s} - \frac{1}{(s+1)}e^{-s} \right| = \left| (1+\alpha)e^{-\beta s} - 1 \right|
\]  

(4.37)

Therefore, from equation (4.16), the required condition for absolute stability is

\[
|\tilde{p}q| < \frac{1}{\Delta^*(s)} = \frac{1}{(1+\alpha)e^{-\beta s} - 1}
\]  

(4.38)

Now, we design the Internal Model Controller \( q(s) \) by considering the nominal system \( \tilde{p}(s) \) without the delay. Hence, for this plant, the IMC is given by

\[
q(s) = \frac{(s+1)}{(2s+1)}
\]  

(4.39)
Using equations (4.39), (4.35) and the fact that $|e^{-\beta s}| = 1$ in equation (4.38), we have

$$\left| \frac{1}{\lambda s + 1} \right| < \frac{1}{(1 + \alpha)} \left| e^{-\beta s} - 1 \right|$$

(4.40)

A simple approximate way of choosing ‘$\lambda$’ is to set the bandwidth of the filter at the frequency at which the uncertainty bound crosses unity, so that the product stays below unity at all frequencies thereby making the closed-loop system robustly stable. In this case,

$$\lambda > \frac{1}{\omega_c}, \text{ where } \Delta^*(\omega_c) = 1$$

(4.41)

To calculate $\omega_c$:

$$\Delta^*(\omega_c) = 1 \Rightarrow \left| (1 + \alpha) \cos (\beta \omega_c) - 1 + j (1 + \alpha) \sin (\beta \omega_c) \right| = 1$$

$$\Rightarrow \left( (1 + \alpha) \cos (\beta \omega_c) - 1 \right)^2 + \left( (1 + \alpha) \sin (\beta \omega_c) \right)^2 = 1$$

$$\Rightarrow \omega_c = \frac{1}{\beta} \cos^{-1} \left( \frac{1 + \alpha}{2} \right)$$

(4.42)

For $\alpha = \beta = 0.1$, we have $\omega_c = 9.88 \text{ rad/s}$ and $\lambda > 0.1011 \text{ s}$. Thus there is a lower limit on the choice of ‘$\lambda$’, which in turn means that the response to a step input cannot be made quicker than a certain limit without compromising on the stability of the closed-loop system. Figure 4.6 shows the simulation result for this system, when the actual plant is given by the worst-case in equation (4.36). It is seen that the system clearly goes unstable for a value of ‘$\lambda$’ smaller than the limit found above while it settles without any steady state error for a higher value of ‘$\lambda$’.
4.5 PID Tuning using IMC

PID control is a simple but very popular control technique that uses the classical controller structure. It is a very old concept but is still applied widely in most of practical industrial and commercial applications because of its simplicity and its ability to control steady state and transient response. The simplest form of a PID controller is the Proportional controller and was first used in 1788 by James Watt in a flyball governor for speed control of his steam engine. Much later, the integral and derivative controllers were
added to form the PID controller for pneumatic systems during the 1930’s. The ideal form of the PID controller is given by

$$c(s) = K_p + \frac{K_i}{s} + K_d s$$  \hspace{1cm} (4.43)

It can be immediately seen that this is improper and hence not physically realizable. But usually, the ideal differentiator does not exist and it usually is in the form of a filtered differentiator, which makes the controller proper. One disadvantage of using the PID controller in the classical controller form is that it is difficult to tune three parameters or gains $K_p$, $K_i$ and $K_d$ simultaneously to obtain the best performance. Moreover, when system uncertainties, disturbance and noise inputs are present it is not trivial to evaluate how these gains affect the system stability. Hence the IMC structure can be used to overcome these drawbacks of using the classical controller structure to tune the PID controller. It will be seen in the following examples that in some cases the actual IMC controller turns into a PID controller when converted to the classical controller form using equation (4.15). Also a generic method for arriving at the PID controller gains for any other systems is discussed in detail.

For the plant $\ddot{p}(s) = \frac{K}{s^2 + \tau}$, the internal model controller $q(s) = \frac{\tau s + 1}{K(\lambda s + 1)}$. When converted to its equivalent classical controller using equation (4.15), we obtain $c(s) = \frac{\tau s + 1}{K\lambda s}$. This can be visualized as a PI controller with

$$K_p = \frac{\tau}{K\lambda}, K_i = \frac{1}{K\lambda}$$  \hspace{1cm} (4.44)

The degree of freedom in choosing the controller is now reduced to one, namely ‘$\lambda$’. Also
by using this method, we can design a robust controller that will maintain closed-loop stability in spite of model uncertainty. While the controller turns out to be an exact PI controller for this particular example, the expressions for $c(s)$ is more complicated for different plants with higher order transfer functions. In such cases, a new method for tuning the PID controller using IMC needs to be developed.

Following the IMC design procedure, we choose

$$\tilde{p}(s) = \tilde{p}_N(s) \tilde{p}_M(s)$$ \hspace{1cm} (4.45)

where we choose the non-minimum phase part $\tilde{p}_N(s)$ such that $|\tilde{p}_N(s)|=1$. For cases, in which the non-minimum phase portion is only the part with time delay, the Internal Model Controller is designed as

$$q(s) = \frac{\tilde{p}_M^{-1}(s)}{(\lambda s + 1)^n}$$ \hspace{1cm} (4.46)

where ‘$\lambda$’ and ‘$n$’ are yet to be chosen appropriately. Now, the equivalent classical controller is equal to

$$c(s) = \frac{q(s)}{1 - \tilde{p}(s)q(s)}$$ \hspace{1cm} (4.47)

Substituting using equation (4.46), we have

$$c(s) = \frac{\tilde{p}_M^{-1}(s)}{(\lambda s + 1)^n} = \frac{\tilde{p}_M^{-1}(s)}{(\lambda s + 1)^n - \tilde{p}_N(s)}$$ \hspace{1cm} (4.48)

An interesting property to be noted in ‘$c(s)$’ is that the denominator in equation (4.48) vanishes when $s = 0$ since $\lim_{s \to 0} \tilde{p}_N(s) = 1$. This means $s = 0$ is one of the poles of
\( c(s) \). Hence there is always at least one integrator in the equivalent classical controller designed using the internal model control (IMC) methodology. Hence one may write

\[
c(s) = \frac{v(s)}{s}
\]  

(4.49)

where

\[
v(s) = \frac{s\hat{p}_M^{-1}(s)}{\left(\lambda s + 1\right)^n - \hat{p}_N(s)}
\]  

(4.50)

Now, one may write \( v(s) \) using a Taylor series expansion into

\[
v(s) = v(0) + sv'(0) + \frac{s^2}{2}v''(0) + \cdots
\]  

(4.51)

Using equations (4.49) and (4.51), we have

\[
c(s) = v'(0) + \frac{v(0)}{s} + s\left(\frac{v''(0)}{2}\right) + \cdots
\]  

(4.52)

Neglecting higher order terms, this can be written as a standard PID controller with

\[ K_p = v'(0), \quad K_i = v(0), \quad K_d = \frac{v''(0)}{2} \]

where \( v(s) \) is given by equation (4.50).

### 4.5.1 IMC-based PID tuning Example 1

Consider a first order plant with delay, whose transfer function is given by

\[
p(s) = \frac{k}{\tau s + 1} e^{-\delta s}
\]  

(4.53)

Based on equations (4.45) and (4.46) we write

\[
p_M(s) = \frac{k}{\tau s + 1}, \quad p_A(s) = e^{-\delta s}
\]  

(4.54)

And the corresponding IMC and its equivalent classical controller is given by
\[ q(s) = \frac{\tau s + 1}{K(\lambda s + 1)} \quad \text{and} \quad c(s) = \frac{\tau s + 1}{K[(\lambda s + 1) - e^{-\delta s}]} \]  \quad (4.55)

Using equation (4.49), we have

\[ v(s) = \frac{s(\tau s + 1)}{K[(\lambda s + 1) - e^{-\delta s}]} \]  \quad (4.56)

Therefore,

\[ K_p = v'(0) = \frac{\delta^2 + 2\tau(\lambda + \delta)}{2k(\lambda + \delta)^2} \quad \text{and} \quad K_i = v(0) = \frac{1}{k(\lambda + \delta)} \]  \quad (4.57)

By setting the time-delay \( \delta = 0 \), we obtain the same result as in equation (4.44), which confirms the result.

### 4.6 Variable Delay and Gain Model for the Piezoactuator System

Now that the general theory and analysis of IMC is complete, this section discusses the development of an appropriate model for the piezoelectrically actuated force phase of the dual-stage actuation system that can facilitate the design of IMC for the system.

Once the stroke phase of the actuation process is completed by the DC motor, the force phase is initiated using the piezoelectric actuator. Here we shall recap the variable delay model for the force phase of the system. The preload provided by the DC motor in the stroke phase must be high enough to make the second phase effective. As discussed in the previous chapter, the second phase of the actuation process is modeled by a second order system as shown in Figure 4.7.
This model leads to a second order system equation

\[ M_x \ddot{x} + b \dot{x} + \left( K_b + \frac{F_{bl}}{x_{\text{max}}} \right) x = F_{bl} \]  \hspace{1cm} (4.58)

The blocked force ‘\( F_{bl} \)’ and the free displacement ‘\( x_{\text{max}} \)’ of the piezoelectric actuator are functions of the applied voltage and their ratio is the voltage-based apparent stiffness (\( K_p \)) of the actuator. Assuming linear relationships, we have,

\[ F_{bl} = \frac{A d_{33} c_{33} E}{t} V \] and \[ K_p = \frac{A c_{33} E}{n t} \], where \( A \) is the surface area of the piezoelectric stack actuator, \( d_{33} \) is the piezoelectric constant, \( c_{33} \) is the material modulus of the actuator, \( n \) is the number of layers in the stack actuator and \( t \) is the thickness of each layer in the actuator. As is evident, the relation between the applied voltage and the blocked force of the piezoelectric actuator is of utmost importance here. Though linear relations between the applied voltage and blocked force are shown above, this is not valid due to the
nonlinearities of piezoactuators like hysteresis. As discussed in the previous chapter, the approximate model for the piezoactuator-based force phase system can be written as

\[
\frac{X(s)}{U(s)} = H(\cdot)G(s)
\]  \hspace{1cm} (4.59)

where the operator \( H(\cdot) \) is defined as a variable time delay and variable gain model. A similar but more detailed approach is followed here that illustrates the ability of this model to capture the hysteretic property of the piezoelectric actuator. Most of the modeling is performed based on the free displacement since it is very difficult to measure the blocked forces. It is again noted that the free displacement and the blocked force are related by a proportional constant for any given applied voltage. This reasonable assumption is valid since the piezoelectric actuator stiffness does not vary much. Hence the overall model of the system shown in Figure 4.7 in terms of the voltage input to the piezoactuator and generated force may be written as

\[
\frac{F}{U}(s) = \hat{p}(s) = \frac{K_p K_b}{M_b} \left( \frac{k}{s^2 + 2\zeta\omega_n s + \omega_n^2} \right) e^{-\tau s}, \quad k \leq k \leq \bar{k}, \tau \leq \bar{\tau} \leq \bar{\tau}
\]  \hspace{1cm} (4.60)

For this system, the nominal value (after the preload of the brake pad) of \( \frac{kK_p K_b}{M_b} = 3.5346 \) \( \text{N/(V.s}^2) \), \( \zeta = 0.75 \), \( \omega_n = 1.113 \times 10^4 \) rad/s, \( k = 6.33 \times 10^{-5} \) mm/V, \( \bar{k} = 8.76 \times 10^{-5} \) mm/V, \( \tau = 0 \) s and \( \bar{\tau} = 0.0075 \) s. The generated force \( (F) \) is assumed to be proportional to the displacement of the spring/brake pad, the proportional constant being the spring constant of the brake pad \( (K_b \text{ shown in Figure 4.7}) \).
4.7 Internal Model Control of Force Phase of the System

Now, for the purpose of robust controller design using IMC, we follow the design procedure described in detail in the previous chapters. One may write the actual plant/system “p(s)” as equal to the plant model “\( \tilde{p}(s) \)” with a multiplicative uncertainty. Mathematically,

\[
p(s) = \tilde{p}(s)(1 + \Delta(s))
\]  

where \( \Delta(s) \) is the multiplicative uncertainty of the plant. One may obtain a frequency-dependent bound of the multiplicative uncertainty using the extreme and nominal plant models available based on the uncertain parameters ‘\( k \)’ and ‘\( \tau \)’. Hence

\[
\Delta^*(s) = \left| \frac{\tilde{p}_{\text{worst}}(s) - \tilde{p}_{\text{nominal}}(s)}{\tilde{p}_{\text{nominal}}(s)} \right|
\]  

where \( \tilde{p}_{\text{nominal}}(s) \) and \( \tilde{p}_{\text{worst}}(s) \) are chosen based on the upper and lower bounds on ‘\( k \)’ and ‘\( \tau \)’. Though the time delay and gain are time-varying, they vary at a much lower rate compare to system’s natural frequency and hence are lumped into simple uncertainties. In this case \( \tilde{p}_{\text{worst}}(s) = k_{\text{max}}e^{-\tau_{\text{max}}s}G(s) \) and \( \tilde{p}_{\text{nominal}}(s) = kG(s) \), taking zero delay as the nominal value. Let \( k_{\text{max}} = k + \alpha \) and \( \tau_{\text{max}} = \beta \). Therefore,

\[
\Delta^*(s) = \left| \frac{(k + \alpha)e^{-\beta s}G(s) - kG(s)}{kG(s)} \right|
\]  

where \( G(s) = \frac{K_pK_b}{M_b}\left(\frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2}\right) \). Hence we have \( \Delta(s) = \left| 1 + \frac{\alpha}{k}e^{-\beta s} - 1 \right| \). Now we define \( \Delta^*(\omega) = |\Delta^*(i\omega)| \). Hence
\[ \Delta^* (\omega) = \left| 1 + \frac{\alpha}{k} e^{-i\beta \omega} - 1 \right| \]  

(4.64)

\[ \Delta^* (\omega) = \left| \left( 1 + \frac{\alpha}{k} \cos (\beta \omega) - 1 \right) + i \left( 1 + \frac{\alpha}{k} \sin (\beta \omega) \right) \right| \]  

(4.65)

Now we determine the critical frequency when \( \omega_c \) at which the value of \( \Delta^* (\omega) \) becomes unity, i.e., \( \Delta^* (\omega_c) = 1 \).

\[ \Rightarrow \left( 1 + \frac{\alpha}{k} \cos (\beta \omega_c) - 1 \right)^2 + \left( 1 + \frac{\alpha}{k} \sin (\beta \omega_c) \right)^2 = 1 \]  

(4.66)

\[ \Rightarrow \left( 1 + \frac{\alpha}{k} \right)^2 \left( \cos^2 (\beta \omega_c) + \sin^2 (\beta \omega_c) \right) + 1 - 2 \left( 1 + \frac{\alpha}{k} \right) \cos (\beta \omega_c) = 1 \]  

(4.67)

\[ \Rightarrow \cos (\beta \omega_c) = \frac{1}{2} \left( 1 + \frac{\alpha}{k} \right) \]  

(4.68)

Hence we may write the critical frequency \( \omega_c \) as

\[ \omega_c = \frac{1}{\beta} \cos^{-1} \left( \frac{1}{2} \left( 1 + \frac{\alpha}{k} \right) \right) \]  

(4.69)

Now, we write the process or plant model as a product of two transfer functions:

\[ \tilde{p}(s) = \tilde{p}_N(s) \tilde{p}_M(s) \]  

(4.70)

Here \( \tilde{p}_N(s) \) represents the non-invertible part, with non-minimum phase zeros and time delays and \( \tilde{p}_M(s) \) represents the invertible part with regular poles and minimum-phase zeros. It is also useful to make \( |\tilde{p}_N(i\omega)| = 1 \) by adding appropriate minimum phase parts.

This will ensure that the resulting controller is ISE (Integral Squared Error) optimized [Braatz-1995]. Now, we define the Internal Model Controller as
\[ \ddot{q}(s) = \hat{p}_{M}^{-1}(s) \quad (4.71) \]

Since this will be improper we add an appropriate filter so that the resulting controller is causal. Hence

\[ q(s) = \hat{p}_{M}^{-1}(s) f(s) \quad (4.72) \]

The system model is given by equation (4.60). Hence it is split into

\[ \hat{p}_{M}(s) = \frac{kK_pK_b}{M_b} \left( \frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2} \right) \quad (4.73) \]

\[ \hat{p}_{N}(s) = e^{-\tau s} \quad (4.74) \]

The internal model controller is then given by

\[ \ddot{q}(s) = \hat{p}_{M}^{-1}(s) = \frac{M_b}{kK_pK_b} \left( s^2 + 2\zeta\omega_n s + \omega_n^2 \right) \quad (4.75) \]

Since this has a second order numerator, a second order filter is chosen to make the IMC controller causal. Hence

\[ f(s) = \frac{1}{(\lambda s + 1)^2} \quad (4.76) \]

Therefore the final complete Internal Model Controller is

\[ q(s) = \frac{M_b}{kK_pK_b} \left( \frac{s^2 + 2\zeta\omega_n s + \omega_n^2}{(\lambda s + 1)^2} \right) \quad (4.77) \]

From robust stability theory, for a conventional unity negative feedback control system to be robustly stable within the defined bounds of the system parameters, we have

\[ \left| \frac{\hat{pc}}{1 + \hat{pc}} \right| < \frac{1}{\Delta^*(\omega)} \quad (4.78) \]

where ‘c’ represents the equivalent conventional classical controller and ‘\( \Delta^* \)’
represents the multiplicative uncertainty bound as defined in equations (4.47) and (4.65) respectively. In the case of IMC, one may see that \( c(s) = \frac{q(s)}{1 - \tilde{p}(s)q(s)} \). Substituting this and equations (4.70) and (4.72) in equation (4.78) and using the fact that \( |\tilde{p}_N(i\omega)| = 1 \), we have

\[
|f(s)| < \frac{1}{\Delta^*(\omega)}
\]  

(4.79)

A simple approximation to force this requirement is to set the bandwidth of the filter at the frequency \( \omega_c \) where \( \Delta^*(\omega_c) = 1 \). Hence, for a ‘3 dB accuracy’ in the bandwidth, we need \( \lambda > \frac{0.6243}{\omega_c} \). Using equation (4.69), we may therefore write

\[
\lambda > \frac{(0.6243)\beta}{\cos^{-1}\left(\frac{1}{2}\left(1 + \frac{\alpha}{k}\right)\right)}
\]  

(4.80)

This result when used in equation (4.77) gives the required constraint on the IMC controller \( q(s) \).

### 4.7.1 PID Gains

By using the analysis in section CHAPTER 0, the appropriate PID gains are computed for this system. In this case, in our case, the transfer function is as follows

\[
\tilde{p}(s) = \tilde{p}_M(s) \tilde{p}_N(s)
\]  

(4.81)

where
\[
\tilde{p}_M(s) = \frac{kK_p K_b}{M_b} \left( \frac{1}{s^2 + 2\zeta \omega_n s + \omega_n^2} \right), \quad \text{and} \quad \tilde{p}_N(s) = e^{-\tau s} \quad (4.82)
\]

Hence we have \( v(0) = \lim_{s \to 0} \frac{s \tilde{p}_M^{-1}(s)}{\lambda s + 1} - \tilde{p}_N(s) \). Using equation (4.82) and applying L’Hospital’s rule, we have

\[
v(0) = \lim_{s \to 0} \frac{1}{\tilde{p}_M'(s)(\lambda s + 1) - e^{-\tau s}} + \frac{\tilde{p}_M'(s)(\lambda s + 1) + \tilde{p}_M(s)(n\lambda + 1)^n + \zeta e^{-\tau s}}{\lambda s + 1} \quad (4.83)
\]

Proceeding similarly, we evaluate the first and second derivatives of \( v(s) \) at the origin and use \( n = 2 \) in the result. Thus

\[
v'(0) = -\frac{-2kK_p K_b \xi}{M_b \omega_n^3} (n\lambda + \tau) + \frac{kK_p K_b}{2M_b \omega_n^2} (n(n-1)\lambda^2 - \tau^2)
\]

and

\[
v''(0) = \left\{
\begin{array}{l}
\frac{2v'(0)}{v(0)} - \frac{2(1-4\xi^2)(n\lambda + \tau) - 2\xi \omega_n (n(n-1)\lambda^2 - \tau^2) + \frac{1}{3\omega_n^2} (n(n-1)(n-2)\lambda^3 + \tau^3)}{\left(\frac{-2\xi}{\omega_n} (n\lambda + \tau) + \frac{1}{2\omega_n^2} (n(n-1)\lambda^2 - \tau^2)\right)^2} v'(0)
\end{array}
\right.
\]

For \( n = 2 \), we have

\[
K_p = v'(0) = \frac{\omega_n \left(4\xi \omega_n (2\lambda + \tau) - \omega_n (2\lambda^2 - \tau^2)\right)}{2 \frac{kK_p K_b}{M_b} (2\lambda + \tau)^2},
\]

\[
K_i = v(0) = \frac{M_b \omega_n^2}{kK_p K_b (2\lambda + \tau)} \quad \text{and}
\]
\[ K_d = \frac{v''(0)}{2} = \left( \frac{2v'(0)}{v(0)} \right) \left( \frac{2(1-4\xi^2)(2\lambda + \tau) - 2\xi(2\lambda^2 - \tau^2) + \frac{1}{3\omega_n^2}(\tau^3)}{\omega_n^4} \right) \] 

4.8 Experimental results and Discussion

The IMC controller and the PID controllers are implemented in the test setup shown in the previous chapter. The controller is built in MATLAB/Simulink and a DSpace 1102 hardware-in-the-loop system is used to perform the real-time control task. The Real-Time-Workshop (RTW) module in the DSpace system converts the control system in Simulink into a corresponding optimized C code and downloads it into the microprocessor in the DSpace board. The sampling rate was fixed at 500 microseconds (\(\mu s\)). The analog inputs at the DSpace board receive different signals like the current drawn by the DC motor and the force measured by load cell. The DSpace system converts these analog inputs into digital inputs using built-in A/D converter with a 16-bit output resolution. The control system then computes the necessary control voltages to be applied to the DC motor and the piezoelectric actuator. These digital outputs are converted to analog outputs using built-in D/A converters in the DSpace board. These two analog outputs are fed to two different controllable power supplies that command proportional output voltages to the DC motor and the piezoelectric actuator respectively.
Figure 4.8 Step Input Tracking for various ‘$\lambda$’ values using IMC

Figure 4.9 Control Voltages for Step Input Tracking for various ‘$\lambda$’ values using IMC
Figure 4.10 Sinusoid Input (1 Hz) Tracking for various ‘$\lambda$’ values using IMC

Figure 4.11 Control Voltages for Sinusoid Input (1 Hz) Tracking for various ‘$\lambda$’ values
Figure 4.12 Sinusoid Input (10 Hz) Tracking for various ‘λ’ values using IMC

Figure 4.13 Control Voltages for Sinusoid Input (10 Hz) Tracking for various ‘λ’ values using IMC
Figure 4.14 Step Input Tracking using a PI controller tuned using IMC for ‘$\lambda$’ = 0.005

Figure 4.15 Control Voltage for PI controller tuned using IMC for ‘$\lambda$’ = 0.005
Figure 4.8 – 4.16 show the tracking response of the IMC system for various ‘λ’ values. Figure 4.8 and 4.12 show the step reference tracking while Figure 4.10 to 4.16 show the sinusoid reference tracking performance at different frequencies (1Hz and 10 Hz). For the step input tracking it is observed that the response is faster as expected when ‘λ’ is decreased. However, the response gets oscillatory as ‘λ’ is decreased since the system is pushed towards instability as ‘λ’ is decreased as discussed in the previous sections and explicitly given by equation (4.80). The overshoot is due to the fact that the system uncertainty model used for the IMC design does not exactly capture the hysteresis nonlinearity and other unmodeled dynamics like sensor and actuator dynamics. Thus when the response speed is increased the system becomes unstable after a certain point when the threshold value of λ is crossed. It is worth noting that the system responds with zero-steady state error for the step-reference input for all values of ‘λ’ since the IMC controller carries a virtual integrator that will force the system type to be atleast 1. For the sinusoidal inputs, it is noted that the steady state response is better for lower values of λ but the transient response is oscillatory. This again is due to the fact that the system uncertainty model used for the IMC design does not exactly capture the hysteresis nonlinearity and other unmodeled dynamics like sensor and actuator dynamics. Also it is noted that the choice of ‘λ’ is limited to ensure robust stability of the system. Thus an appropriate value of ‘λ’ is chosen based on the stability requirement along with the response requirements like settling time and overshoot. It is also noted that the classical trade-off in control system design between the steady-state and transient response is illustrated clearly in the IMC system for various values of ‘λ’.
Figure 4.14 and Figure 4.15 illustrate the response of the system using a PID controller tuned using IMC as discussed in the previous section of this chapter. It is seen that the response is similar to that of the IMC system for ‘λ’ = 0.005 as seen in Figure 4.8. The same PID controller tuning can be extended to different responses for different reference inputs. Thus it is seen that though IMC is an effective way of controlling uncertain systems, the various response show that better controllers may be developed to improve this system performance without affecting system stability.
CHAPTER 5

MODEL PREDICTIVE CONTROL OF SYSTEM

This chapter discusses the development and implementation of a model predictive control methodology for controlling the force phase of the two-stage actuation system in tracking various reference inputs. The chapter begins with a brief introduction to the concept of Model Predictive Control. Then a general analysis of MPC is presented for dynamic systems with discrete-time transfer function and state space models. The rest of the chapter involves the derivation of the Model Predictive Control for the force phase of the two-stage actuation system. An extended unit-step delayed disturbance estimate is used to virtually account for actuator hysteresis, other nonlinearities and unmodeled dynamics in the system. The chapter concludes with a discussion on the effectiveness of this control methodology by studying the experimental results in tracking various reference inputs.

5.1 Introduction

As cited in the previous chapter, the objective of the control system is to develop an appropriate controller that is to be designed for the second phase of the actuation process of system so that it robustly tracks a time-varying reference force signal. A settling time
of less than 0.2 s for a step reference is desirable with offset-free steady state output. Also most of the existing actuators have a tracking bandwidth requirement of up to 5-10 Hz. Hence the dual-stage actuator must also provide satisfactory tracking performance up to 10 Hz. Since the system involves constraints and an accurate hysteresis model that is practical for real-time control applications is not readily available, it is imperative that a real-time robust controller be designed and implemented for consistent performance. Chapter 4 discussed the design and implementation of an Internal Model Controller (IMC) for the second phase of the two-stage actuation system. While we see good tracking performance results, there is a limitation on the speed of response of the system. As the response speed is increased, the system tends to go more and more towards instability. Also the model used for the IMC design is very crude, since it approximates time-varying delay into bounded uncertainties. Also since a transfer function based method is used in IMC, it is difficult to extend it to MIMO systems and time-varying disturbances. Since the primary nonlinearity in the two-stage actuation system is the piezoelectric actuator hysteresis, which is time varying, the IMC method falls short of accounting for it reasonably accurately.

Model Predictive Control is one other model-based control methodology that can virtually account for unmodeled dynamics and other disturbance inputs to the system [Rossiter-2003]. This chapter discusses in detail the adaptation and implementation of this controller MPC methodology. This method has become quite popular among process control engineers and used in most of the chemical process industry [Pike-1995]. As the name suggests, this controller methodology uses a discrete-time model of the dynamic
system to predict the system performance in a predefined length of time in the future in order to arrive at the optimal control action at each instant of time. However, though this method has an implicitly built-in robustness property, it still requires a relatively accurate model to provide premium performance especially in the case of reference tracking [Rossiter-2003].

MPC, also known as receding horizon control, is an online sub-optimal feedback control methodology used to control linear and nonlinear systems with or without constraints. The advantages and merits of using MPC include the ability to handle constraints and to produce optimal solutions at each control move. The philosophy here is to develop a control algorithm that uses a receding horizon or window to predict the behavior of the system over a predefined length in the future and to adjust the system control action to produce an optimal solution with respect to a predefined cost function. MPC requires a representative discrete-time model of the process usually in state space form or transfer function form.

Figure 5.1 Schematic of Model Predictive Controller

Figure 5.2 Illustration of the working of MPC
The working procedure of MPC is to use a discrete time model for the dynamic system that allows a receding window prediction of future outputs based on the current value of the output and future value of inputs and to use this model to arrive at an optimal value of future inputs relative to a predefined cost criterion. Figure 5.1 and 5.2 highlight the concept of the MPC algorithm. The concept of MPC works is illustrated by citing the example of how a driver drives his/her car say from point A to point B in a city. The control inputs available to the driver are either acceleration (racing) or deceleration (braking) and the steering wheel position. In achieving the task of reaching point B, the driver does not optimize his control inputs for the whole trip from A to B and apply these inputs in a blindfolded manner. Instead, he/she looks 100 yards in front of him/her and optimizes his/her control inputs depending upon how clear the road in front of him/her is, for the next 100 yards and applies the first value of these optimal controls actions that he/she feels are required. After moving a yard towards point B, he/she performs the same optimization for the next 100 yards and repeats this process until he/she reaches point B. This concept of moving window of 100 yards is termed as the ‘receding horizon’. MPC adapts this concept for dynamic systems. One other difference of MPC compared to say a PID controller is that, the MPC uses the future values of the reference inputs to arrive at the current control action, whereas PID controller does not use the information of future reference inputs.

MPC is known for its ability to handle uncertainties, constraints and external and internal disturbances to provide excellent controller performance. These properties have been utilized in many process control problems to effectively control dynamic plants in the
chemical process industry [Garcia-1989]. However, an accurate system model is usually required to implement MPC successfully. Moreover, the time constants of chemical process systems are usually on the order of a few seconds to a few minutes, with a low sampling frequency, giving the controller plenty of time to compute the computationally intensive optimal control actions at each time step [Rossiter-2003, Morari-1997, Pike-1995]. Nevertheless, the dual stage actuation mechanism in this study is relatively much faster with a required settling time lower than 0.2 s and a sampling rate of 1 ms or lower. Hence this puts a huge bottleneck on the complexity of the optimization problem that the controller can perform at each time step. This means that the system equations must be formulated in a manner so that the optimization problem has a closed form solution that may be quickly applied at each time step and must also virtually account for the nonlinearities in the system like hysteresis.

Before a detailed analysis of MPC is presented, the following section provides a brief review of some necessary mathematical concepts and notations followed throughout this chapter.

5.2 Math Background and Notations for MPC

This section provides brief definitions of some key notations consistently followed in this chapter and some important math background that will help the reader understand and appreciate the philosophy and details of MPC.

5.2.1 Discrete Time Systems

The concept behind MPC requires a discrete-time model of the system. But
most practical dynamic systems are modeled in continuous-time based on first principles like Newton’s law for mechanical systems and Kirchoff’s and Ohm’s laws for electrical systems. The following definition and analysis shows the methods to arrive at discrete-time models of dynamic systems based on the continuous time models in both state space and transfer function forms.

### 5.2.1.1 Z-Transform

For a given discrete-time signal \( x(k) \), the corresponding Z-transform of the signal is given by

\[
Z \{ x(k) \} = X(z) = \sum_{i=0}^{\infty} x(i)z^{-i}
\]

where ‘\( z \)’ is a complex variable.

### 5.2.1.2 Transfer Function Model

Consider a system with a continuous-time transfer function model relating the input \( U \) to the output \( Y \) given by

\[
\frac{Y(s)}{U(s)} = G(s)
\]

Figure 5.3 Equivalent Discrete-Time Model of a Continuous Time System
When this system is used in a classical digital control system, the analog output signal $Y(t)$ is sampled at a predefined finite frequency with a period ($T$) using A/D converters. The input to the system is converted back to analog using D/A converters. A common D/A conversion method that maintains causality is the Zero-Order-Hold (ZOH) process, which maintains the value of the signal given at the previous sampling instant until the next sampling point is reached, at which point it updates the signal to the new value. Hence the signal flow diagram is as shown in Figure 5.3.

As seen above, in order to obtain the equivalent discrete time model $G(z)$ of the system, a unit impulse is input to the system (i.e., $U(k) = d(k)$). It is easy to visualize that the Laplace transform of $U(t)$, which after a ZOH of the unit impulse is a unit rectangular pulse of width equal to the sampling period $T$, is given by $L\{U(t)\} = \frac{(1-e^{-sT})}{s}$, where ‘$s$’ is the Laplace variable. The Laplace transform of the corresponding output $Y(t)$ is therefore equal to $L\{Y(t)\} = \frac{(1-e^{-sT})}{s}G(s)$. Since the output $Y(k)$ is arrived by sampling $Y(t)$ at the sampling frequency $(1/T)$, the corresponding Z-transform of the output is given by

$$Y(z) = Z\{Y(k)\} = Z\left\{L^{-1}\left[\frac{(1-e^{-sT})}{s}G(s)\right]\right\}$$

(5.3)

Since the Z-transform the unit-impulse input is unity, we have

$$G(z) = \frac{Y(z)}{U(z)} = Y(z) = Z\left\{\frac{(1-e^{-sT})}{s}G(s)\right\} = Z\{1-e^{-sT}\}Z\left\{\frac{G(s)}{s}\right\}$$

(5.4)

since the part containing $(e^{-sT})$ can be factored out because it represents just a
unit-delay, with a Z-transform of $z^{-1}$. Hence the final discrete-time transfer function of the system is given by

$$G(z) = (1 - z^{-1})\left\{ \frac{G(s)}{s} \right\}$$

(5.5)

### 5.2.1.3 State Space Models

Consider a continuous time system modeled in state-space as

$$\dot{x}(t) = Ax(t) + Bu(t)$$

(5.6)

and

$$y(t) =Cx(t) + Du(t)$$

(5.7)

where $x \in \mathbb{R}^n$ is the vector of state variables of the system, $u \in \mathbb{R}^m$ is vector of control variables and $y \in \mathbb{R}^p$ is the vector of outputs of the system that are usually termed as the measured states. $A$, $B$, $C$ and $D$ are system, control and output matrices respectively with appropriate dimensions.

Given a sampling period $T$, The equivalent discrete time state space equations may be written as

$$x(k+1) = A_dx(k) + B_du(k)$$

(5.8)

and

$$y(k) =C_dx(k) + D_d u(k)$$

(5.9)

where $x(k)$, $u(k)$, $y(k)$ are the state vector, control vector and output vector respectively at the time instant $t = kT$. Also $A_d$, $B_d$, $C_d$ and $D_d$ are the discrete-time system, control and output matrices respectively with appropriate dimensions. The following analysis shows the method to arrive at these matrices from the continuous-time models given in equations (5.6) and (5.7). The solution to the continuous state space equation (5.6) is
given by
\[ x(t) = e^{At}x(0) + e^{At} \int_{0}^{t} e^{-A\tau} Bu(\tau) d\tau \]  
(5.10)

Assuming a sampling interval of ‘\( T \)’, we may write \( t = kT \). Hence the value of the state ‘\( x \)’ at \( t = kT \) may now be written as
\[ x(kT) = x(k) = e^{AkT}x(0) + e^{AkT} \int_{0}^{kT} e^{-AkT} Bu(\tau) d\tau \]  
(5.11)

Similarly, one may extend this for the time \( t = (k+1)T \), and write
\[ x((k+1)T) = e^{A(k+1)T}x(0) + e^{A(k+1)T} \int_{0}^{(k+1)T} e^{-A\tau} Bu(\tau) d\tau \]  
(5.12)

\[ x(k+1) = e^{AT} \left( e^{AkT}x(0) + e^{AkT} \int_{0}^{kT} e^{-AkT} Bu(\tau) d\tau + \int_{kT}^{(k+1)T} e^{-A\tau} Bu(\tau) d\tau \right) 
+ \left( e^{A(k+1)T} \int_{0}^{(k+1)T} e^{-A\tau} Bu(\tau) d\tau \right) u(k) \]

\[ x(k+1) = e^{AT}x(k) + \left( e^{AT} \int_{0}^{T} e^{-AT} Bd\tau \right) u(k) \]  
(5.13)

\[ x(k+1) = A_d x(k) + B_d u(k) \]  
(5.14)

where \( A_d = e^{AT} \) and \( B_d = e^{AT} \int_{0}^{T} e^{-AT} Bd\tau = \left( \int_{0}^{T} e^{A(T-\tau)} d\tau \right) B = \left( e^{AT} - I \right) A^{-1} B \). It is also useful to know that \( e^{AT} = I + AT + \frac{A^2T^2}{2!} + \frac{A^3T^3}{3!} + \ldots \)

### 5.2.2 State Vectors of Past, Present and Future Values

Since it is understood that the concept behind MPC uses predictions of the
system behavior in the future based on a system model and its past behavior, it is imperative that we will encounter vectors of the system state that describe its values in the past, present and future. Table 5.1 describes these notations. It is noted that the prediction notation is up to ‘N’ points by default unless otherwise specified.

<table>
<thead>
<tr>
<th>NOTATION</th>
<th>EXPANSION</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_k$ or $x(k)$</td>
<td>-</td>
<td>This notation describes the value of the system state at the time instant $k$ (i.e., $t = kT$)</td>
</tr>
<tr>
<td>$\mathbf{x} \leftarrow k$</td>
<td>$\begin{bmatrix} x_k \ x_{k-1} \ \vdots \ x_{k-N+1} \end{bmatrix}$</td>
<td>This notation describes the values of the system state $\mathbf{x}$ up to $N$ steps in the past including the current value</td>
</tr>
<tr>
<td>$\mathbf{x} \rightarrow k$</td>
<td>$\begin{bmatrix} x_{k+1} \ x_{k+2} \ \vdots \ x_{k+N} \end{bmatrix}$</td>
<td>This notation describes the values of the system state $\mathbf{x}$ up to $N$ steps in the future</td>
</tr>
</tbody>
</table>

Table 5.1 Notations for Past and Present and Future State Values

5.2.3 Toeplitz and Hankel Matrices

Let us define a polynomial in $(z^{-1})$

$$n(z) = n_0 + n_1z^{-1} + n_2z^{-2} + \ldots + n_mz^{-m} = \sum_{j=0}^{m} n_jz^{-j} \quad (5.15)$$

Let us also define the matrices $\Gamma_n$, $C_n$ and $H_n$ associated with the polynomial $n(z)$ above as follows
One may simplify the matrices by using the following notation:

\[ \Gamma_n(i, j) = C_n(i, j) = n_{i-j}, \quad H_n(i, j) = n_{i+j-1}, \] where \( n_i = 0 \) if \( i < 0 \) or \( i > m \)

The matrices \( \Gamma_n, C_n \) are called Toeplitz matrices and the matrix \( H_n \) is termed Hankel. It is also noted that the matrix \( \Gamma_n \) can have as many columns as is required but the number of rows must be ‘\( m \)’ more than the number of columns. The matrix \( C_n \) is square but can be of any dimension. The number of rows and columns in the Hankel matrix \( H_n \) must be greater than or equal to \( m \).

**5.2.3.1 Multiplication of polynomials**

One major application of Toeplitz matrices is to reduce polynomial convolution into simple matrix multiplication.
To illustrate this, let us define another polynomial in ‘$z^{-1}$’ as follows

\[
d(z) = d_0 + d_1 z^{-1} + d_2 z^{-2} + \ldots + d_r z^{-r} = \sum_{j=0}^{r} d_j z^{-j}
\]  

(5.18)

The product of the two polynomials $n(z)$ and $d(z)$ is

\[
f(z) = n(z) d(z) = f_0 + f_1 z^{-1} + f_2 z^{-2} + \ldots + f_{m+r} z^{-m-r} = \sum_{j=0}^{m+r} f_j z^{-j}
\]  

(5.19)

One may derive the coefficients of $f(z)$ in terms of the coefficients of $n(z)$ and $d(z)$. Let $\tilde{f}, \tilde{n}$ and $\tilde{d}$ denote the vector of coefficients of the corresponding polynomials. This is done with the help of the Toeplitz matrix $\Gamma_n$ defined earlier. The order of the matrix $\Gamma_n$ for this case is $(m+r+1)(r+1)$.

\[
\begin{bmatrix}
  f_0 \\
  f_1 \\
  f_2 \\
  \vdots \\
  f_m \\
  f_{m+1} \\
  \vdots \\
  f_{m+r}
\end{bmatrix} = \Gamma_n \begin{bmatrix}
  d_0 \\
  d_1 \\
  d_2 \\
  \vdots \\
  d_r
\end{bmatrix} = \Gamma_n \tilde{d}
\]

(5.20)

Example:

\[
n(z) = 2 + z^{-1} - z^{-2} + 3z^{-3}; \Rightarrow m = 3
\]

\[
d(z) = 1 + z^{-1} + z^{-2}; \Rightarrow r = 2
\]

\[
f(z) = n(z) d(z) = (2 + z^{-1} - z^{-2} + 3z^{-3})(1 + z^{-1} + z^{-2})
\]

\[
\Rightarrow f(z) = 2 + 3z^{-1} + 2z^{-2} + 3z^{-3} + 2z^{-4} + 3z^{-5}
\]
5.2.3.2 Inversion of Toeplitz Matrix

It will be seen that inverse of the Toeplitz matrix plays an important role in MPC formulations. Hence it is useful to know that

\[
[C_n]^{-1} = C_\frac{1}{n}
\]

**Example:**

\[
n(z) = 1 + z^{-1}
\]

\[
\Rightarrow C_n = \begin{bmatrix}
1 & 0 & 0 & \cdots & 0 \\
1 & 1 & 0 & \cdots & 0 \\
0 & 1 & 1 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \cdots & 1
\end{bmatrix}
\]

\[
\frac{1}{n(z)} = \frac{1}{1 + z^{-1}} = 1 - z^{-1} + z^{-2} - z^{-3} + z^{-4} + \cdots
\]
5.3 Model Predictive Control Formulation

In this section, the formulation and derivation of MPC for general systems with transfer function and state-space models without disturbance inputs are described. This serves as a useful preview for the detailed derivation used for the two-stage actuation system. Different cost functions are used for the transfer function and state space formulations to illustrate the flexibility of MPC in providing a basic framework that allows the control engineer/researcher to appropriately choose and accomplish different tasks.

5.3.1 Transfer Function Model

Consider a transfer function model of a system as follows

\[ \frac{Y(z^{-1})}{U(z^{-1})} = \frac{N(z^{-1})}{D(z^{-1})} \]  

(5.21)
where $N(z^{-1})$ and $D(z^{-1})$ are the numerator and denominator polynomial respectively. It is noted that that argument of the polynomials is $z^{-1}$ and not $z$ since $z^{-1}$ represents the unit delay and is causal and hence it is more appropriate to write the transfer function terms of $z^{-1}$. Now one may assume

$$N(z^{-1}) = N_1 z^{-1} + N_2 z^{-2} + N_3 z^{-3} + \cdots + N_n z^{-n} \tag{5.22}$$

$$D(z^{-1}) = 1 + D_1 z^{-1} + D_2 z^{-2} + D_3 z^{-3} + \cdots + D_{n+1} z^{-(n+1)} \tag{5.23}$$

Therefore one may write the set of prediction equations that predict that output up to $N$ steps in the future as

$$y_{k+1} + D_1 y_k + \cdots + D_{n+1} y_{k-n} = N_1 u_k + N_2 u_{k-1} + \cdots + N_n u_{k-n+1}$$

$$y_{k+2} + D_1 y_{k+1} + \cdots + D_{n+1} y_{k-n+1} = N_1 u_{k+1} + N_2 u_k + \cdots + N_n u_{k-n+2} \tag{5.24}$$

One may rewrite these equations into a single matrix equation

$$\begin{bmatrix}
1 & 0 & \cdots & 0 \\
D_1 & 1 & \cdots & 0 \\
D_2 & D_1 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
D_k & D_{k-1} & \cdots & 0 \\
\end{bmatrix}
\begin{bmatrix}
y_{k+1} \\
y_{k+2} \\
\vdots \\
y_{k+N} \\
\end{bmatrix}
= 
\begin{bmatrix}
1 & D_1 & \cdots & D_{n+1} \\
D_2 & D_3 & \cdots & 0 \\
D_3 & D_4 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
D_{k-n+1} & D_{k-n} & \cdots & 0 \\
\end{bmatrix}
\begin{bmatrix}
y_k \\
y_{k-1} \\
\vdots \\
y_{k-n} \\
\end{bmatrix}
$$

This is further reduced into

$$y = H u + P u + Q y \tag{5.26}$$
where $H = C_D^{-1}C_N$, $P = C_D^{-1}H_N$, $Q = -C_D^{-1}H_D$. It is noted that $H_D$ and $H_N$ are Hankel matrices, while $C_D$ and $C_N$ are Toeplitz matrices.

Equation (5.26) serves as a tool for predicting future values of outputs as a function of past outputs and past and future inputs based on the transfer function model.

### 5.3.1.1 MPC Algorithm for Systems with Transfer Function Model

Now that the prediction equation is available, one proceeds to solve for the optimal control action using the MPC concept. The procedure involves the definition of cost function $J$ that is to be minimized over the receding horizon window up to a finite number of steps ($N$) into the future. The objective of the controller is to guide the output of the system ‘$y$’ towards zero from any given initial condition. Hence a good choice for the cost function in this case is

$$J = \| y \|^2_2 + \lambda \| u \|^2_2$$

(5.27)

where $\| \cdot \|^2_2$ represents the square of the standard 2-norm, i.e. $\| x \|^2_2 = x_1^2 + x_2^2 + \cdots x_n^2$.

Substituting the values of ‘$y$’ from equation (5.26), we have

$$J = \left\| H u + P u + Q y \right\|^2_2 + \lambda \left\| u \right\|^2_2$$

(5.28)

The subscripts ‘$k$’ and ‘$k-1$’ are omitted for brevity. This cost function is minimized to obtain the control moves. Mathematically,

$$\text{Minimize } J = u^T \left( H^T H + \lambda I \right) u + 2 u^T H^T \left[ P u + Q y \right] + \left\| P u + Q y \right\|^2_2$$

(5.29)

Taking the derivative of ‘$J$’ w.r.t. $u$ and setting it to zero, one obtains
Following the fundamental concept behind MPC, only the first value from the optimized predicted values of the future control inputs from equation (5.31). Hence

\[
 u_k = [I \ 0 \ 0 \ 0 \ ... \ 0] u
\]  

(5.32)

In the next step the same optimization procedure is carried out and again only the first value of the predicted control inputs is implemented. This is repeated step after step to obtain an optimized response from the system.

### 5.3.2 State Space Model

Consider a discrete state-space system model as follows

\[
x_{k+1} = Ax_k + Bu_k
\]  

(5.33)

Now one may extend equation (5.33) to obtain the system state value at time instant \( k+1 \) as

\[
x_{k+2} = A^2 x_k + ABu_k + Bu_{k+1}
\]  

(5.34)

\[
\Rightarrow x_{k+2} = A^2 x_k + ABu_k + Bu_{k+1}
\]  

(5.35)

Generalizing to the \( N^{th} \) step, we have

\[
x_{k+N} = A^N x_k + A^{N-1} Bu_k + A^{N-2} Bu_{k+1} + \ldots + Bu_{k+N-1}
\]  

(5.36)

Writing it in a matrix form, we have the future states as
Hence in a simpler format,\( x_{k} = P_{x} x_{k} + H_{x} u \) \( \rightarrow_{k} \) \( \rightarrow_{k-1} \) (5.38)

### 5.3.2.1 MPC Algorithm for Systems with State Space Model

Now that the prediction equation is available, one proceeds to solve for the optimal control action using the MPC concept for the state-space formulation. The procedure again involves the definition of cost function \( J \) that is to be minimized over the receding horizon window up to a finite number of steps \( (N) \) into the future. The objective of the controller is to guide the states of the system ‘\( x \)’ towards zero from any given initial condition. Hence a good choice for the cost function is

\[
J = \left\| x_{k} \right\|_{2}^{2} + \lambda \left\| u \right\|_{2}^{2}
\]

(5.39)

where \( \left\| \cdot \right\|_{2}^{2} \) represents the square of the standard 2-norm. Substituting the values of ‘\( x_{k} \)’ from equation (5.38), we have

\[
\Rightarrow J = \left\| P_{x} x_{k} + H_{x} u \right\|_{2}^{2} + \lambda \left\| u \right\|_{2}^{2}
\]

(5.40)

The subscripts ‘\( k \)’ and ‘\( k-1 \)’ are omitted for brevity. This cost function is minimized to obtain the control moves. Mathematically,

\[
\text{Minimize } J = u^{T} \left( H_{x}^{T} H_{x} + \lambda I \right) u + x_{k}^{T} P_{x} x_{k} + x_{k}^{T} P_{x} H_{x} u + x_{k}^{T} P_{x} H_{x} u
\]

(5.41)
Taking the derivative of \( J \) w.r.t. \( u \) and setting it to zero, one obtains

\[
\frac{dJ}{du} = 2 \left( H_x^T H_x + \lambda I \right) u + 2 H_x^T P_x x_k = 0
\]

\[
\Rightarrow u = - \left( H_x^T H_x + \lambda I \right)^{-1} H_x^T P_x x_k
\]

Following the fundamental concept behind MPC, only the first value from the optimized predicted values of the future control inputs from equation (5.31). Hence

\[
 u_k = [I \ 0 \ 0 \ ... \ 0] u
\]

In the next step the same optimization procedure is carried out and again only the first value of the predicted control inputs is implemented. This is repeated step after step to obtain an optimized response from the system.

Thus it is clear that MPC can be used for systems with transfer function and state-space models. The state space formulation however has a few advantages over the transfer function method including

i) Ability to easily incorporate MIMO systems

ii) Ability to handle unmodeled but time varying disturbances, which will be illustrated in the next section.

In the next section the extension of the concept of MPC for the second stage of the dual-stage actuation mechanism that enables the tracking of different reference force inputs to the system is discussed in detail.
5.4 Stability Analysis of MPC

Since the cost function $J$ in equation (5.39) is minimized at each and every sampling instant, it is basically an explicit function of states and control and an implicit function of time. Also it is noted that the cost function is a positive definite expression. It is therefore acceptable to use the cost-function as the Lyapunov candidate to analyze stability. The following analysis performs the detailed stability analysis for the nominal system without uncertainties as shown in equation (5.33). The following analysis is for a system with a single state only. It can be appropriately extended for systems with multiple states. The following analysis also uses a simple terminal constraint of $x_{k+N+1} = 0$.

Consider the sampling instant ‘k’. Let the MPC-based optimal control sequence up to N steps in future computed at this instant (k) be

$$u^* \rightarrow_{k-1} = \left[ u^*_k, u^*_{k+1}, \ldots, u^*_{k+N-1} \right]$$

Similarly at the next instant ‘k+1’, there will be a new sequence $u^*_{\rightarrow k+1}$ of optimal control values for N steps in future starting from k+1. Consider a different sequence at the instant (k+1) that is not optimal with respect to the cost function

$$u' \rightarrow_k = \left[ u^*_{k+1}, u^*_{k+2}, \ldots, u^*_{k+N-1}, 0 \right]$$

Now let the Lyapunov candidate be defined as

$$V_k = J\left( u^* \rightarrow_{k-1} \right) = \left( x \rightarrow_k \right)^T x + \lambda \left( u^* \rightarrow_{k-1} \right)^T \left( u^* \rightarrow_{k-1} \right)$$

Therefore at the next instant, keeping in mind that $u'_{\rightarrow k+1}$ is not the optimal control sequence, we have
\[ V_{k+1} = J \left( u^*_k \right) \]  
\[ \leq J \left( u^*_{k+1} \right) \]

\[ = \begin{pmatrix} x \to_{k+1} \\ \to_{k+1} \end{pmatrix} + \lambda \begin{pmatrix} u \to_k \\ u \to_k \end{pmatrix} \]

\[ = V_k - x^2_{k+1} - \lambda \left( u^*_k \right)^2 \]

\[ \therefore V_{k+1} < V_k \]  

Hence it is seen that the Lyapunov candidate decreases with each and every sampling instant. This confirms the stability of the MPC for nominal systems.

### 5.5 Model Predictive Control of Dual Stage Actuation System

From the second order model of the second stage of the actuation system derived in the previous chapter, the nominal equation of motion for the force phase can be reproduced in state space form as

\[ \dot{X} = A_c X + B_c U + d(t) \]  

where the state \( X = \begin{bmatrix} f \\ f' \end{bmatrix} \) is the vector of the generated force and its derivative, 

\[ A_c = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\xi\omega_n \end{bmatrix}, \quad B_c = \begin{bmatrix} 0 \\ \beta \end{bmatrix} \]  

with \( U \) being the control voltage. Note that an additional term \( d(t) \) is added to account for unmodeled dynamics and nonlinearities in the system. It is also assumed that this disturbance and its derivative are bounded. The controller is designed so that the total force from the system, which is the sum of the force produced in the stroke phase and the force phase, tracks a time-varying
reference force requirement. At each time instant, it is assumed that the force produced is equal to the product of the brake pad stiffness and the system displacement. The following procedure describes the methodology in implementing MPC for this system.

Consider a continuous time dynamic system with disturbance as shown in equation (5.50), where ‘\(X\) \(\in \mathbb{R}^n\) is the state vector, ‘\(U\) \(\in \mathbb{R}^m\) is the control, ‘\(d(t)\) \(\in \mathbb{R}^n\) is the time-varying disturbance vector. \(A_c\) and \(B_c\) are the system and control matrices of appropriate dimensions. The disturbance is added to model the uncertainties and unmodeled dynamics in the system. Given a sampling time ‘\(T\)’, one may convert this continuous-time system given by equation (5.50) into an equivalent discrete time system using a standard discretization procedure \([Ogata-1987]\). This yields

\[X_{k+1} = AX_k + BU_k + d_k\] (5.51)

\[Y_k = CX_k\] (5.52)

where \(A = e^{A_T}\), \(B = \int_0^T e^{A \tau} B d\tau\) and \(d_k = \int_0^T e^{A \tau} d(kT + T - \tau) d\tau\), with \(B\) and \(d_k\) being \(O(T)\), meaning it is proportional to \(T\), if \(d(t)\) is bounded and \(Y\) defined as the measured output. Defining a new state that augments the control input we have

\[Z_k = \begin{bmatrix} X_k \\ U_{k-1} \end{bmatrix}\] (5.53)

Now the new state equation is

\[
\begin{bmatrix}
X_{k+1} \\
U_k
\end{bmatrix} = \begin{bmatrix}
A & B \\
0 & I
\end{bmatrix} \begin{bmatrix}
X_k \\
U_{k-1}
\end{bmatrix} + \begin{bmatrix}
B \\
I
\end{bmatrix} \Delta U_k + \begin{bmatrix}
0 \\
0
\end{bmatrix} d_k
\] (5.54)

In other words, this can be written as

\[Z_{k+1} = \hat{A}Z_k + \hat{B} \Delta U_k + \hat{D} d_k\] (5.55)
with \[ Z_k = \begin{bmatrix} X_k \\ U_{k-1} \end{bmatrix} \in \mathbb{R}^{n+m}, \quad \hat{A} = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix} \in \mathbb{R}^{(n+m)\times(n+m)}, \quad \hat{D} = \begin{bmatrix} I \\ 0 \end{bmatrix} \in \mathbb{R}^{(n+m)\times(n)} \] and
\[ \hat{B} = \begin{bmatrix} B \\ I \end{bmatrix} \in \mathbb{R}^{(n+m)\times(m)}. \] The new output equation is given by
\[ Y_k = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} X_k \\ U_{k-1} \end{bmatrix} = \hat{C}Z_k \tag{5.56} \]

The reason behind reformulating the system in terms of the control moves \( \Delta U \) and not \( U \) is to facilitate the MPC method to enable reference input tracking. This will be made clearer when the cost function \( J \) is defined later on.

Extending equation (5.56) by one and two steps ahead in time gives
\[ Y_{k+1} = \hat{C}Z_{k+1} = (\hat{C}\hat{A})Z_k + (\hat{C}\hat{B})\Delta U_k + (\hat{C}\hat{D})d_k \tag{5.57} \]
and
\[ Y_{k+2} = \hat{C}Z_{k+2} = \hat{C}\hat{A}Z_{k+1} + \hat{C}\hat{B}\Delta U_{k+1} + \hat{C}\hat{D}d_{k+1} \tag{5.58} \]
\[ \Rightarrow Y_{k+2} = (\hat{C}\hat{A}^2)Z_k + (\hat{C}\hat{A}\hat{B})\Delta U_k + \hat{C}\hat{B}\Delta U_{k+1} + (\hat{C}\hat{A}\hat{D})d_k + \hat{C}\hat{D}d_{k+1} \tag{5.59} \]

Extending this prediction up to \( N \) steps ahead in time gives the following matrix relationship
\[ Y = \begin{bmatrix} \hat{C}\hat{A} \\ \hat{C}\hat{A}^2 \\ \vdots \\ \hat{C}\hat{A}^N \end{bmatrix} Z_k + \begin{bmatrix} \hat{C}\hat{B} & 0 & \ldots & 0 \\ \hat{C}\hat{A}\hat{B} & \hat{C}\hat{B} & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \hat{C}\hat{A}^{(N-1)}\hat{B} & \hat{C}\hat{A}^{(N-2)}\hat{B} & \ldots & \hat{C}\hat{B} \end{bmatrix} \Delta U + \begin{bmatrix} \hat{C}\hat{D} & 0 & \ldots & 0 \\ \hat{C}\hat{A}\hat{D} & \hat{C}\hat{D} & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \hat{C}\hat{A}^{(N-1)}\hat{D} & \hat{C}\hat{A}^{(N-2)}\hat{D} & \ldots & \hat{C}\hat{D} \end{bmatrix} \Delta d \tag{5.60} \]

where \[ Y = \begin{bmatrix} Y_{k+1} \\ Y_{k+2} \\ \vdots \\ Y_{k+N} \end{bmatrix} \] and \[ \Delta U = \begin{bmatrix} \Delta U_k \\ \Delta U_{k+1} \\ \vdots \\ \Delta U_{k+N-1} \end{bmatrix} \] and \[ \Delta d = \begin{bmatrix} d_k \\ d_{k+1} \\ \vdots \\ d_{k+N-1} \end{bmatrix} \]. It is noted that the matrices \( H \) and \( L \) in equation (5.60) have a Toeplitz structure. We thus have used the state space model of the dynamic system to
provide a framework for prediction of the system output up to N steps in the future. The controller to be designed is required to track a reference input R. MPC uses the prediction equation (5.60) to define and solve an optimization problem that minimizes an appropriate cost function over the next N steps. Let’s define a positive-definite cost function

$$J = \left\| R - Y \right\|^2 + \lambda \left\| \Delta U \right\|^2$$  \hspace{1cm} (5.61)$$

where $R$ represents the future values of the reference inputs up to N steps ahead in time that are to be tracked by the system output. It is noted that the cost function puts a weighting on the change in control effort relative to each step rather than the control effort itself. This is justified for piezoelectric actuator systems since the power consumed by the actuator is approximately proportional to the rate of change of the voltage. In addition, the steady state value of the terms in the cost function must tend to zero. This happens only when the change in control effort is used instead of the actual control input.

Substituting equation (5.60) into (5.61) gives,

$$\Rightarrow J = \left\| R - PZ_k - H \Delta U - Ld \right\|^2 + \lambda \left\| \Delta U \right\|^2$$

where the matrices $P$, $H$ and $L$ are shown in equation (5.60). In order to perform the optimization, we use the condition, $\frac{\partial J}{\partial \Delta U} = 0$. This yields

$$\left( H^T H + \lambda I \right) \Delta U = \left[ H^T \rightarrow R - H^T PZ_k - H^T L d \right]$$  \hspace{1cm} (5.62)$$

Hence the optimal MPC-based control moves are given by
\[ \Delta U = (H^T H + \lambda I)^{-1} \left[ H^T R - H^T P Z_k - H^T L d \right] \]  

(5.63)

The above equation gives the optimal value of the control changes up to ‘\(N\)’ steps ahead in time including the change to be applied at the current instant based on the dynamic system model. The MPC concept however, requires that we only apply the first value in the list of ‘\(N\)’ controller values given by equation (5.63). Once the system responds to this control action, the same procedure is performed at the next sampling instant and so on. The value of the control effort at the instant \(k\) is given by

\[ U_k = U_{k-1} + \Delta U_k \]  

(5.64)

where \(\Delta U_k = [I \ 0 \ \cdots \ 0] U\). This formulation works well for even time-varying reference signals unlike most other formulations, which assume a steady state fixed set point as the reference signal. However, the problem with this formulation and MPC in general, is that one requires knowledge of the future ‘\(N\)’ values of the disturbances \(d\) in order to use the MPC method exactly. In many cases, when a linear model for the disturbance exists, one may implement it in the formulation and use it to derive the MPC controller as shown above. But in cases, when a disturbance model is not available, one may have to use some kind of disturbance estimation in equation (5.63). In the case of the dual-stage actuation system in this document, the procedure required to arrive at a valid model for nonlinearities like hysteresis is non-trivial and too complicated for control design. Hence, it is not possible to estimate the exact disturbance in the system accurately. A simple way of overcoming this difficulty is to obtain the previous value of the disturbance and assume that the disturbance remains the same for the next \(N\) steps,
which is basically an extended zero-order estimate. In this case, we replace $d$ by its estimate

$$d_{est} = (d_{k-1}, d_{k-1}, \ldots, d_{k-1})^T \quad (5.65)$$

where $d_{k-1} = X_k - AX_{k-1} - BU_{k-1}$. The actual error between the actual and estimated values of $d_{k+N-1}$, the final disturbance value, is given by

$$d_{k+N-1} - d_{k-1} = (d_{k+N-1} - d_{k+N-2}) + (d_{k+N-2} - d_{k+N-3}) + \cdots + (d_{k} - d_{k-1}) \quad (5.66)$$

Considering one general term on the right hand side of the above equation, we have

$$d_{k+j} - d_{k+j-1} = \int_0^T e^{A_T} \left( d((k+j)T + T - \tau) - d((k+j)T - \tau) \right) d\tau \quad (5.67)$$

Now since $(f(b) - f(a)) = \int_a^b df = \int_a^b \frac{df}{d\theta} d\theta$, we write

$$d_{k+j} - d_{k+j-1} = \int_0^T e^{A_T} \left( \int_{(k+j)T - \tau}^{(k+j)T + T - \tau} \dot{d}(\theta) d\theta \right) d\tau \quad (5.68)$$

where ‘$\theta$’ is a dummy variable. This error is $O(T^2)$ (meaning of the order of $T^2$) if $d(t)$ and its time-derivative $\dot{d}(t)$ are bounded. The total error in the estimation of the disturbance can be given by

$$d_{k+N-1} - d_{k-1} = N \cdot O(T^2) = O(T^2) \quad (5.69)$$

if $N << \frac{1}{T}$, which is assumed and is valid for this system, $N = 10, T \leq 1 \text{ ms}$ is used.

The MPC controller is then designed as discussed in detail above. The final control law is exactly given by equations (5.63) and (5.64). We also saturate the control effort by limiting the controller values to lie between its maximum and minimum values. Hence
the MPC methodology implemented in the dual-stage actuation system is a clipped-sub-optimal control.

![Figure 5.4 Experimental Setup](image)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\omega_n$</td>
<td>Natural Frequency</td>
<td>1.113e4 rad/s</td>
</tr>
<tr>
<td>$\xi$</td>
<td>Damping ratio</td>
<td>0.75</td>
</tr>
<tr>
<td>$\beta$</td>
<td>Constant in $B$ matrix</td>
<td>0.132 m/V.s²</td>
</tr>
<tr>
<td>$\lambda$</td>
<td>Control Weight in MPC</td>
<td>varied as shown in results</td>
</tr>
<tr>
<td>$U_{\text{max}}$</td>
<td>Maximum Control Voltage</td>
<td>1000 V</td>
</tr>
<tr>
<td>$U_{\text{min}}$</td>
<td>Minimum Control Voltage</td>
<td>-200 V</td>
</tr>
</tbody>
</table>

Table 5.2 Values of parameters used in the MPC system
5.6 Experimental Results

A simple experimental setup is built following the design principles discussed in detail in the previous chapters. Figure 5.4 shows the simple setup. The setup consists of a controllable DC motor capable of producing large enough torque to provide the necessary preload driving a lead screw-nut assembly via a coupling and an axial bearing. The nut is connected to a piezoelectric actuator capable of producing 12.5 kN of blocked force and $105 \, \mu m$ free displacement for a voltage range of –200 to 1000 V. The free end of the piezoelectric actuator is connected to a brake pad, which pushes against a load cell at the end of the setup. A high voltage power supply is used to power the piezoelectric actuator. A DSpace controller board is used to interface Simulink to the system during the real-time control.

5.6.1 Control Law Implementation and Testing

The model predictive controller designed in the previous section is applied to the force phase of the actuation process of the system. The controller is designed to force the system to track different reference signals like step and sinusoidal signals at different frequencies. Figure 5.5 to Figure 5.11 show the tracking results for different reference inputs to the system. It is noted that the figures show the results of the controller after the first phase of the actuation system is completed. From the figures, it is observed that the model predictive controller provides excellent tracking performance. Different values for the parameter ‘$\lambda$’ in the model predictive controller, which is the weight on the control changes in the cost function $J$, are chosen to illustrate the effect of choosing different
weighting on control effort changes. For step reference tracking shown in Figure 5.5 and Figure 5.6, the response time decreases as the value of ‘$\lambda$’ decreases and though the lower value of ‘$\lambda$’ means faster response and settling times, it is observed that that for ‘$\lambda$’ = 0.00025, overshoot is present in the system response. This is because the error in the disturbance estimate is high when very fast response times are desired. However a response time of less than 0.1 s is very desirable and hence the system response for $\lambda = 0.0005$ is excellent. Figure 5.7 to Figure 5.10 show the tracking response for sinusoidal reference inputs at two different frequencies, namely 1 Hz and 10 Hz. Again, it is noted that the steady state tracking performance improves as the ‘$\lambda$’ is decreased. However, the transient performance deteriorates as ‘$\lambda$’ is decreased, because at lower ‘$\lambda$’ the weight on the control effort change is low, which allows faster and higher changes in the control input. This leads to undesirable transient performance since the error in the disturbance estimates becomes higher due to higher and quicker control changes. This also highlights the classical trade-off in control system design between transient and steady state performances. However, the performance of MPC is much better compared to a simple PID controller since the PID controller does not account for disturbances explicitly in its structure. Moreover, a PID controller will need to be retuned frequently at different operating conditions to provide a consistent performance, while no frequent tuning is required for MPC. For sinusoidal reference tracking inputs, the controller performance deteriorates as the frequency increases. This again may be attributed to the accuracy of the disturbance estimates at high frequencies. However, based on the overall performance ‘$\lambda$’ = 0.0005 may be chosen as the best value for the model predictive controller among the various values shown. Figure 5.11 shows the steady state closed loop relation
between the reference and actual force for \( \lambda = 0.0005 \) for a simple sinusoidal voltage at 10 Hz.

### 5.7 Concluding Remarks

The development and implementation of the Model Predictive Control concept for the second stage of the dual stage actuation system has been discussed in detail. After a brief introduction to the concept of MPC, the adaptation and reformulation of MPC to suit our application has been described. An extended one-step delayed disturbance estimate is used to account for nonlinearities in the system like the hysteresis and other internal and external disturbances. The results show that this MPC controller is an excellent candidate for controlling piezoelectric actuator systems and may be extended to control other smart material systems like magnetostrictive material devices and MR fluid devices. However, the drawbacks of MPC include

i) Inability to explicitly relate \( \lambda \), the weight on control moves to the tracking response of the system like the settling time. It is obtained only through simulation and testing.

ii) Moreover, the information of future reference inputs is required to implement MPC. While this is available for simulation and laboratory based testing using simple reference inputs, it is not readily available in many practical cases, where the reference inputs are known at the current time instant.

iii) Inability to provide formal stability guarantees for the non-ideal case with unmodeled disturbance inputs.
Figure 5.5 Step Reference Tracking – Force vs Time for various values of ‘λ’

Figure 5.6 Step Reference Tracking – Control Voltage vs Time for various values of ‘λ’
Figure 5.7 Sinusoidal Reference (1 Hz) Tracking – Generated Force vs Time for various values of ‘$\lambda$’

Figure 5.8 Sinusoidal Reference (1 Hz) Tracking – Control Voltage vs Time for various values of ‘$\lambda$’
Figure 5.9 Sinusoidal Reference (10 Hz) Tracking – Generated Force vs Time for various values of ‘λ’

Figure 5.10 Sinusoidal Reference (10 Hz) Tracking – Control Voltage vs Time for various values of ‘λ’
Figure 5.11 Closed loop Relation between reference and actual force
CHAPTER 6

MODEL PREDICTIVE SLIDING MODE CONTROL OF SYSTEM

This chapter discusses in detail the development and implementation of a new control methodology called the Model Predictive Sliding Mode Control (MPSMC). A detailed introduction section discusses the basic ideas behind the popular DSMC (discrete-time sliding mode) and MPC (Model Predictive Control) methodologies. This section also discusses in detail the problems and drawbacks of these methodologies. Then the new concept of MPSMC is introduced as a way of overcoming the drawbacks of conventional DSMC by integrating these two control strategies (DSMC and MPC). Detailed analysis is carried out using two different approaches for dynamic systems with unmodeled disturbances. Simulation results of the MPSMC method are shown to illustrate its effectiveness in stabilizing as compared to the conventional DSMC strategy. Finally the method is applied to the experimental system and results are shown illustrating the benefits of this new methodology.


6.1 Introduction

Sliding mode control (SMC) is one of the most widely used nonlinear control schemes popular for its robustness and order reduction properties [Utkin-1977; Utkin-1999]. Continuous-time sliding mode control may be categorized as a special-case of variable structure control systems. Variable structures are defined as continuous systems that consist of boundaries in which the controller switches its feedback gains. The sliding mode control design procedure for continuous-time systems consists of two steps. The first step involves the selection of a discontinuity surface (sometimes referred to the switching surface or switching manifold) \( s(x) = 0 \) in the state space where the control undergoes discontinuities. This surface is determined so that the system behaves in a desired manner (i.e. has the desired eigenvalues). In step two the continuous control functions \( u^+ (x,t) \) and \( u^- (x,t) \) beyond the discontinuity surface are selected such that the control law satisfies the sufficient conditions for the existence and reachability of the sliding mode. The continuous time SMC then switches between these two control functions in order to enforce the sliding mode.

However, it is usually not possible to exactly meet all the requirements of the ideal continuous-time SMC. For example, the application of continuous-time SMC requires perfect switching of the control action on either side of the sliding mode [Utkin-1977; Utkin-1999]. This is not practically possible since perfect switching requires infinite actuation energy but actuators usually have some dynamics resulting in slower response and imperfect switching. Other unmodeled dynamics usually neglected in system modeling include sensor dynamics, minor nonlinearities and disturbances that can
cause problems in the convergence of the sliding mode. These lead to the phenomenon of chattering, which is the oscillation of the system states about the sliding surface with a finite amplitude and frequency [Utkin-1999]. Chattering often results in poor system performance and actuator degradation with time. Different methods have been proposed in overcoming the phenomenon of chattering. These include the boundary layer method and the observer-based solution [Young-1999]. One other method of overcoming the phenomenon of chattering is the application of discrete-time sliding mode control (DSMC) [Utkin-1999; Su-2000]. DSMC is studied and used more often these days since most of the modern-day controllers are implemented using digital computers via A/D and D/A converters and microcontrollers that function with a finite sampling rate. Consequently the discontinuous controller designed using SMC for the continuous time system model will lead to discretization chatter about the sliding mode when implemented by directly discretizing the discontinuous controller. Hence a different approach is utilized for applying DSMC based on the discrete-time model of a dynamic system. In DSMC, usually the control law is designed to force the system to reach the sliding surface at the very next sampling instant [Utkin-1999]. This DSMC method provides a $O(T^2)$ (an order of $T^2$) convergence to the sliding mode when the system disturbances are known at each time instant as compared to the $O(T)$ convergence that is obtained by directly digitizing the discontinuous controller of the continuous-time SMC. However in most systems the disturbances due to unmodeled dynamics and other nonlinearities are not known exactly at each time step. This problem is addressed by using a unit-step delayed disturbance estimate that uses the value of the disturbance at the previous time instant $(k-1)$ for the disturbance at the current instant $(k)$ [Su-
This leads to a $O(T^2)$ convergence to the sliding mode in the presence of unmodeled disturbances.

However, this method assumes three important requirements, namely

i) The bound on the disturbance and its derivative

ii) Availability of all system states and

iii) Independence of the disturbance with respect to the control input, i.e., the disturbance must not be a function of the control input.

The first assumption is valid in most cases, while the second and third ones are not. It is not trivial to design a robust discrete time observer like a sliding mode observer to estimate various unmeasured system states at based on the system output when unmodeled disturbances are present. In cases where the actual unmeasured system states are estimated using crude approximations using measured states, the $O(T^2)$ convergence of the system to its sliding mode is not guaranteed. This problem can be noticed easily in the usual DSMC law, where the sliding mode is enforced at the very next instant ($S_{k+1} = 0$) [Su-2000]. Also in many cases where actuator nonlinearities are present, the system can be usually modeled as a linear system with unknown disturbances. But the unknown disturbances are function of the control input and these disturbances account for the nonlinearity in the system. In such cases, the third requirement is not valid. Piezoelectric actuator systems fall under this category.

Hence there is a need for modifying the DSMC law to enforce sliding mode in such cases. Moreover, it is neither necessary nor optimal to force the system to reach the sliding mode at the very next instant since this usually results in saturating the controller or over-compensating the system, forcing the system to cross the sliding surface at
the next instant. This in turn may lead to oscillatory motion about the sliding surface known as the discretization chatter [Utkin-1999].

Model Predictive Control is a robust discrete-time control methodology that explicitly uses the system model. This method has become quite popular among process control engineers and used in most of the chemical process industry [Rossiter-2003]. As the name suggests, this controller methodology uses a discrete-time model of the dynamic system to predict the system performance in a predefined length of time in the future in order to arrive at the control action that is optimal with respect to a pre-defined cost criterion at each instant of time. MPC is known for its ability to handle constraints, time-delay systems and disturbances while providing an optimal control action. However, though this method has an implicitly built-in robustness property, it still requires a relatively accurate model to provide premium performance especially in the case of reference tracking. Moreover, it is difficult to relate the system performance in terms of time constant, settling time for a particular choice of cost function [Morari-1997] without testing or simulation. Also, MPC requires the knowledge of future reference signals [Rossiter-2003]. While this is available for simple simulations, future reference signals may not be accurately available for real-time applications. However, its abilities to produce optimal control action at each step, to enhance tracking performance and to handle MIMO, unstable and non-minimum phase systems remain the main attractions of this methodology [Pike-1996]. This chapter discusses ways to integrate both these popular control methodologies (DSMC and MPSC) to arrive at a new control strategy (MPSMC) that aims to derive the advantages of both the original methods. The idea is to force the system to reach the sliding surface in an optimal manner as compared
to the regular DSMC, wherein the system is forced to reach the sliding mode at the very next step, while the optimization is carried out using the MPC strategy. In the recent past, researchers have worked on developing a ‘predictive sliding mode control methodology’ [Corradini-1997; Wang-2000; Zhou-2001; Partel-2002]. Corradini et al. have proposed a predictive variable structure control methodology for systems with discrete-time transfer function models [Corradini-1997]. But the drawbacks of this method include failure to provide robustness in the presence of model uncertainties and unmodeled disturbances and inability to deal with MIMO systems. Wang et al. have proposed a predictive control methodology to control a hot strip mill. However, the predictive part of their method is used in observing the states to overcome time-delays while the discrete sliding mode controller still has the conventional formulation [Wang-2000]. Zhou et al. have worked on developing a dual mode control algorithm using MPC and DSMC. In their algorithm, MPC is used to drive the system to a predefined terminal region and conventional sliding mode control is used while the system states are within the terminal region [Zhou-2001]. However, the disadvantages of the standard DSMC discussed previously still holds for this method. The method proposed by Camacho et al. uses a partial optimization of the cost function with respect to only the continuous part of the controller. However, it is the discontinuous part that drives the system to the sliding mode and hence the effect of MPC is not fully utilized in this method. Also this method fails to account for the effect of model uncertainties and unmodeled disturbances in the system [Partel-2002].

This chapter provides two new and different approaches in integrating discrete time sliding mode control and model predictive control for systems with unmodeled but bounded disturbances and model uncertainties. While one method applies
direct optimization of a cost function criterion with respect to the control effort change using control augmentation, the other method involves the splitting of the controller into an equivalent control part (that ensures that the system stays on the sliding mode once reached) and a reaching control part (that guides the system towards the sliding mode). The idea behind both these methods is to guide the system towards the sliding mode in a smooth manner and the MPC-based optimization technique is used as a tool to achieve this. Both these methods utilize state space formulation and hence are valid for MIMO systems.

6.2 Conventional Discrete-Time Sliding Mode Control

Consider a continuous time dynamic system with disturbance

\[ \dot{X} = A_cX(t) + B_cU(t) + D_c f(t) \]  

(6.1)

where \( X \in \mathbb{R}^n \) is the state vector, \( U \in \mathbb{R}^m \) is the control, \( f(t) \in \mathbb{R}^p \) is the time-varying disturbance vector. \( A_c, B_c \) and \( D_c \) are the system, control and disturbance matrices of appropriate dimensions. The disturbance is added to model the uncertainties and unmodeled dynamics in the system. To control the system to track a reference input \( R \), we define the instantaneous tracking error \( E \) as

\[ E(t) = X(t) - R(t) \]  

(6.2)

The corresponding system differential equation in terms of the instantaneous tracking error is

\[ \dot{E} = A_cE + B_cU + d(t) \]  

(6.3)

where
This accumulated sum is the total disturbance in the system that accounts for uncertainties, unmodeled system dynamics and the effect of reference inputs. Given a sampling time \( T \), one may convert this continuous-time system given by equation (6.3) into an equivalent discrete time system using standard procedure. This yields

\[
E_{k+1} = AE_k + BU_k + d_k
\]  

(6.5)

where

\[
A = e^{AT}, \quad B = \int_0^T e^{A\tau} Bd\tau \quad \text{and} \quad d_k = \int_0^T e^{A\tau} d(kT + T - \tau)d\tau, \quad \text{with} \quad B \quad \text{and} \quad d_k \quad \text{being} \quad O(T), \quad \text{if} \quad d(t) \quad \text{is bounded.}
\]

We proceed to design an appropriate sliding surface as

\[
S_k = GE_k
\]  

(6.6)

where \( G \in \mathbb{R}^m \), is a vector of constants chosen in advance, in order to assign appropriate eigenvalues to the system once the sliding mode is reached. Our control design objective is to force the system to reach the sliding mode quickly and stay on it. In order that the sliding mode is enforced at the very next step, we write

\[
S_{k+1} = 0 \Rightarrow GE_{k+1} = 0
\]  

(6.7)

\[
\Rightarrow G(AE_k + BU_k + d_k) = 0
\]  

(6.8)

Hence the control effort known as the equivalent control [Utkin-1999], required to force sliding mode to zero at the \( k+1^{th} \) instant is equal to

\[
U_{k-eq} = -(GB)^{-1} G(AE_k + d_k)
\]  

(6.9)

As one may see, it is evident that one requires the knowledge of the current value of the disturbance, \( d_k \) in order to obtain the value of the control input \( U_{k-eq} \). But for
uncertain time-varying systems, it is not possible to know the instantaneous value of the disturbance exactly. Hence one may use an estimate of the disturbance based on the previous values assuming that the disturbance is smooth and its derivatives are bounded. Hence

\[ U_{k\text{-eq-est}} = - (GB)^{-1} G(AE_k + d_{k\text{-est}}) \] (6.10)

A zero-order estimate and a first order estimate are the simplest examples one may consider. The zero-order estimate is

\[ d_{k\text{-est}} = d_{k-1} \] (6.11)

and the first order estimate is

\[ d_{k\text{-est}} = d_{k-1} + (d_{k-1} + d_{k-2}) \] (6.12)

where \( d_{k-1} = X_k - AX_{k-1} - BU_{k-1} \) and \( d_{k-2} = X_{k-1} - AX_{k-2} - BU_{k-2} \) are exactly known at the time instant ‘k’. It is obvious that one expects to induce some error by using an estimate of the disturbance rather than the actual disturbance itself. The resulting sliding mode value will not be exactly equal to zero but will be a function of the error in the disturbance estimate. In other words,

\[ S_{k+1} = G(AE_k + BU_{k\text{-eq-est}} + d_k) \] (6.13)

\[ \Rightarrow S_{k+1} = G(d_k - d_{k\text{-est}}) \] (6.14)

For a zero order estimate,

\[ d_k - d_{k-1} = \int_0^\tau e^{A\tau} \left( d(kT + T - \tau) - d(kT - \tau) \right) d\tau \] (6.15)
\[ d_k - d_{k-1} = \int_0^T e^{A_k T} \left( \int_{kT}^{kT+T} \dot{d}(\theta) d\theta \right) d\tau \]  

(6.16)

This error is \( O(T^2) \) if \( d(t) \) and its derivative \( \dot{d}(t) \) are bounded. Hence from equations (6.14) and (6.16), we have

\[ S_{k+1} = O(T^2) \]  

(6.17)

This results shows that using the one-step delayed disturbance estimate drives the states toward in the \( O(T^2) \) vicinity of the desired sliding mode \( S_k = 0 \). In order to account for controller saturation limits, the final discrete time sliding mode control law for is given by

\[
U = \begin{cases} 
U_{k-eq-est}, & \text{if } U_{\min} < U_{k-eq-EST} < U_{\max} \\
U_{\max}, & \text{if } U_{k-eq-EST} > U_{\max} \\
U_{\min}, & \text{if } U_{k-eq-EST} < U_{\min} 
\end{cases}
\]  

(6.18)

where \( U_{k-eq-est} \) is given by equation (6.10)

### 6.2.1 Limitations of Conventional DSMC

As highlighted in the introduction section, the conventional DSMC as derived above is valid only for systems where the unknown disturbance \( d_k \) does not depend on the control input \( U_k \). However, when the disturbance does depend on the control input the system can either go unstable or produce chattering. In cases when the disturbance can be written as

\[ d_k = d_k^* + (\Delta B) U_k, \] 

(6.19)

where \( d_k^* \) is independent of the control input and hence does not affect the sliding mode convergence, it has been shown [Su-1993, Su-1996] that the system sliding mode is governed by a second order difference equation given by
\[ S_{k+1} = -G\Delta B \left( GB \right)^{-1} \left[ 2S_k - S_{k-1} \right] + O(T^2) \]  

(6.19)

Hence if the uncertainty in the control matrix \((\Delta B)\) of the system is high enough, the poles of the equation (6.19) can have a magnitude greater than unity. Hence the system diverges from the sliding mode and this eventually destabilizes the system. The reason for this instability is because the conventional DSMC attempts to enforce the sliding mode at the very next instant, i.e., \((S_{k+1} = 0)\) even when unknown disturbances are present. While this is attractive when the disturbances are either known or do not depend on the control, it can destabilize the system when the disturbance is a function of the control input. The rest of this chapter discusses the development of a new Model Predictive Sliding Mode Controller that combines the concept of DSMC and MPC to optimally enforce the sliding mode in an optimal manner even when the disturbance depends on the control input.

### 6.3 Model Predictive Sliding Mode Control – Method 1

Consider a continuous time dynamic system with disturbance

\[ \dot{X} = A_c X(t) + B_c U(t) + D_c f(t) \]  

(6.20)

where \(X \in \mathbb{R}^n\) is the state vector, \(U \in \mathbb{R}^m\) is the control, \(f(t) \in \mathbb{R}^p\) is the time-varying disturbance vector. \(A_c, B_c\) and \(D_c\) are the system, control and disturbance matrices of appropriate dimensions. The disturbance is added to model the uncertainties and unmodeled dynamics in the system. To control the system to track a reference input \(R\), we define the instantaneous tracking error \(E\) as
\[ E(t) = X(t) - R(t) \]  
(6.21)

The corresponding system differential equation in terms of the instantaneous tracking error is

\[ \dot{E} = A_c E + B_c U + d(t) \]  
(6.22)

where

\[ d(t) = A_c R(t) + D_c f(t) - \dot{R}(t) \]  
(6.23)

This accumulated sum is the total disturbance in the system that accounts for uncertainties, unmodeled system dynamics and the effect of reference inputs on the error dynamics. Given a sampling time ‘\( T \)’, one may convert the continuous-time system in equation (6.22) into an equivalent discrete time system using standard procedure. This yields

\[ E_{k+1} = A E_k + B U_k + d_k \]  
(6.24)

where \( A = e^{A_T} \), \( B = \int_0^T e^{A \tau} B d\tau \) and \( d_k = \int_0^T e^{A \tau} d(kT + T - \tau) d\tau \), with \( B \) and \( d_k \) being \( O(T) \), meaning it is of the order of \( T \), if \( d(t) \) is bounded and \( Y \) defined as the measured output. Defining a new state, that augments the control input to the original state, we have

\[ Z_k = \begin{bmatrix} E_k \\ U_{k-1} \end{bmatrix} \]  
(6.25)

Now the new state equation is

\[
\begin{bmatrix} E_{k+1} \\ U_k \end{bmatrix} = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix} \begin{bmatrix} E_k \\ U_{k-1} \end{bmatrix} + \begin{bmatrix} B \\ I \end{bmatrix} \Delta U_k + \begin{bmatrix} I \\ 0 \end{bmatrix} d_k
\]  
(6.26)

In other words, this can be written as

\[ Z_{k+1} = AZ_k + \hat{B} \Delta U_k + \delta_k \]  
(6.27)
with \( Z_k = \begin{bmatrix} E_k^T & \end{bmatrix} \in \mathbb{R}^{n+m}, \quad \hat{A} = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix} \in \mathbb{R}^{(n+m) \times (n+m)}, \quad \hat{B} = \begin{bmatrix} \hat{B}^T \\ \hat{B} \end{bmatrix} \in \mathbb{R}^{(n+m) \times (m)} \) and
\[
\delta_k = \begin{bmatrix} 1 \\ 0 \end{bmatrix} d_k \in \mathbb{R}^{(n+m)} \quad \text{and} \quad \Delta U_k \quad \text{is the control effort change or move at the current time instant ‘k’}. \]

Now one may define the sliding mode as
\[
S_k = GZ_k \quad \text{(6.28)}
\]

where the matrix \( G \in \mathbb{R}^{m \times (n+m)} \) is chosen so that the resulting system along the sliding manifold has certain eigenvalues [Utkin-1999]. One may extend equation (6.28) into
\[
S_{k+1} = GZ_{k+1} = G\hat{A}Z_k + G\hat{B}\Delta U_k + G\delta_k \quad \text{(6.29)}
\]

and
\[
S_{k+2} = GZ_{k+2} = G\hat{A}Z_{k+1} + G\hat{B}\Delta U_{k+1} + G\delta_{k+1}
\]

Continuing up to ‘N’ steps, we obtain
\[
S_{k+1} = GZ_{k+1} = G\hat{A}Z_k + G\hat{B}\Delta U_k + G\delta_k \quad \text{(6.28)}
\]

where \( S_{k+1} = \begin{bmatrix} S_{k+1} & S_{k+2} & \cdots & S_{k+N} \end{bmatrix} \), \( \Delta U_{k+1} = \begin{bmatrix} \Delta U_k & \Delta U_{k+1} & \cdots & \Delta U_{k+N-1} \end{bmatrix} \) and \( \delta_{k+1} = \begin{bmatrix} \delta_k & \delta_{k+1} & \cdots & \delta_{k+N-1} \end{bmatrix} \) represent the future values of the sliding mode, control input moves and the disturbance vectors respectively up to \( N \) steps ahead in time. It is observed that the matrices in the equation (6.31) are
\[
P = \begin{bmatrix} G\hat{A} \\ G\hat{A}^2 \\ \vdots \\ G\hat{A}^N \end{bmatrix}, \quad H = \begin{bmatrix} G\hat{B} & 0 & \cdots & 0 \\ G\hat{A}\hat{B} & G\hat{B} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ G\hat{A}^{(N-1)}\hat{B} & G\hat{A}^{(N-2)}\hat{B} & \cdots & G\hat{B} \end{bmatrix}
\]

and \( L = \begin{bmatrix} G & 0 & \cdots & 0 \\ G\hat{A} & G & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ G\hat{A}^{(N-1)} & G\hat{A}^{(N-2)} & \cdots & G \end{bmatrix} \)

where \( H \) and \( L \) have a Toeplitz structure. In the conventional discrete time sliding
mode controller, the sliding mode is enforced at the very next step [Utkin-1999; Su-2000]. Mathematically, this is written as $S_{k+1} = 0$. This usually results in saturating the controller and is not an efficient way of enforcing sliding mode. Also since a unit-step delayed disturbance is used, the error in the disturbance estimate is usually high at the very first step. Hence the resulting system performance will be different from the desired sliding mode. The formulation proposed in this paper will allow an efficient way of enforcing the sliding mode without saturating the controller while at the same time provide improvement in the system performance. The MPC strategy is used to arrive at optimal control actions to smoothly enforce sliding mode. Defining a cost function $J$, the following optimization algorithm may be applied.

$$
\min_{\Delta U} J = \left\| S \right\|^2 + \lambda \left\| \Delta U \right\|^2
$$

(6.32)

where ‘$\lambda$’ is the weighting on the control moves and $\|\cdot\|$ refers to conventional the $L^2$ norm of a vector. As one may note, the value of $\Delta U$ is used in the cost function instead of the total control $U$, as $\Delta U$ tends to zero when $S$ tends to zero, so that the value of $J$ at steady state will be zero. Substituting for $S$ from equation (6.31), we have

$$
\min_{\Delta U} J = \left(PZ_k + H\Delta U + L \delta \right)^T \left(PZ_k + H\Delta U + L \delta \right) + \lambda \left(\Delta U \right)^T \Delta U
$$

(6.33)

$$
\frac{\partial J}{\partial \left(\Delta U \right)} = 0 \Rightarrow \Delta U = -\left(H^T H + \lambda I \right)^{-1} H^T \left(PZ_k + L \delta \right)
$$

(6.34)

$$
\Rightarrow \Delta U_k = -e_1^T \left(H^T H + \lambda I \right)^{-1} H^T \left(PZ_k + L \delta \right)
$$

(6.35)

where $e_1^T = [1, 0, 0 \cdots 0]$, since only the first value of the estimated control sequence is
applied at the instant ‘\(k\)’ and the process is repeated at every time instant. As one may see, the value of future disturbances is required to compute the sliding mode control value. This can be easily applied if a model for the disturbance exists, thereby allowing us to predict the future. Otherwise, the value of the disturbance at the previous instant (\(k-1\)) is computed exactly and the value of the future disturbances from the instant ‘\(k\)’ is estimated to remain the same for the next \(N\) steps. In other words, using an extend zero-order estimate for the disturbance, we have

\[
\begin{align*}
    d_{\text{est}} & = (d_{k-1}\ d_{k-1} \ldots \ d_{k-1})^T \\
    \text{where } \ d_{k-1} & = E_k - AE_{k-1} - BU_{k-1} \text{ is exactly known at time instant ‘}k\text{’ assuming that all the states are available.}
\end{align*}
\]  

(6.36)

The actual error between the actual and estimated values of \(d_{k+N-1}\) is given by

\[
    d_{k+N-1} - d_{k-1} = (d_{k+N-1} - d_{k+N-2}) + (d_{k+N-2} - d_{k+N-3}) + \cdots + (d_{k} - d_{k-1})
\]  

(6.37)

Considering one general term on the right hand side of the above equation, we have

\[
    d_{k+j} - d_{k+j-1} = \int_0^T e^{A \tau} (d((k+j)T+T-\tau) - d((k+j)T-\tau)) d\tau
\]  

(6.38)

Now since \(\int (b-f(a)) \frac{df}{d\theta} \, d\theta = \int_a^b \frac{df}{d\theta} \, d\theta\), we rewrite

\[
    d_{k+j} - d_{k+j-1} = \int_0^T e^{A \tau} \left( \int_{(k+j)T-\tau}^{(k+j)T+\tau} d(\theta) d\theta \right) d\tau
\]  

(6.39)

where ‘\(\theta\)’ is a dummy variable. This error is \(O(T^2)\) if \(d(t)\) and its time-derivative \(\dot{d}(t)\) are bounded. Hence the total error in the estimation of the disturbance can be given by

\[
    d_{k+N-1} - d_{k-1} = N \cdot O(T^2) = O(T^2)
\]  

(6.40)
if \( N << \frac{1}{T} \). This extends to the augmented disturbance ‘\( \delta \)’ as per the definition in equation (6.27). In the analysis above, it is assumed that the full-state error information \((E_k)\) is available for computing the control action at each step. However in many cases, this is not possible, since only a few of the state variables are measured and the rest are estimated. For example, in a position control system, the position \((x_k)\) is measured while the velocity \((v_k)\) is usually estimated. Designing a discrete-time robust observer for systems with unmodeled disturbances is non-trivial, and in such cases, a crude approximation may be used for estimating the other states. In this example, the velocity can be estimated using a backward difference method based on the position information, i.e., \( v_k \approx \frac{x_k - x_{k-1}}{T} \). This again induces an error in the computation process. The following simulation shows the effectiveness of this method in overcoming all these shortcomings in providing an effective solution compared to the existing methods of DSMC.

6.3.1 Simulation Example 1

Since a first order \((n = m = 1)\) integrator plant with scalar control has been used by [Young-1999], the same system is used here to illustrate the effectiveness of MPSMC over DSMC. Equations (6.41) and (6.42) describe the equations for the actual uncertain system and the model.

\[
\text{ACTUAL SYSTEM:} \quad \dot{x} = (1+\alpha)u \quad (6.41)
\]

where ‘\( \alpha \)’ is unknown but bounded.
SYSTEM MODEL: \[ \dot{x} = u + d(t) \] (6.42)

The disturbance term \( d(t) \) is added to account for unknown properties of the system. Here it is evident that the disturbance \( d(t) = \alpha u(t) \), represents the quantity unknown about the system; but it is important to note that the disturbance depends on the control \( u(t) \). This system is discretized using a sampling instant of 0.05 s and a simple sliding mode matrix \( G = 1 \) is used. Thus the desired sliding mode is \( x = 0 \). The sliding mode controller’s objective is to drive the system to the sliding mode (i.e., \( x = 0 \)) from an initial condition and make sure the system states stay on the sliding mode. An initial condition of \( x(0) = 10 \) is used. After discretization, the actual plant and model equations are roughly

ACTUAL DISCRETIZED SYSTEM: \[ x_{k+1} = x_k + (1 + \alpha)Tu_k \] (6.43)

DISCRETIZED SYSTEM MODEL: \[ x_{k+1} = x_k + Tu_k + d_k \] (6.44)

where \( T \) is the sampling instant. For the simulation, \( T = 0.05 \) is used. From equation (6.19), it can be seen that the unit-step delayed disturbance estimate using conventional DSMC will lead to instability if the unknown ‘\( \alpha \)’ is high enough. Here, the characteristic equation of the sliding mode dynamics based on equation (6.19) is given by

\[ z = -\alpha T(T)^{-1} \left( 2 - z^{-1} \right) \] (6.45)

\[ \Rightarrow z^2 + 2\alpha z - \alpha = 0 \] (6.46)

Thus the roots of the equation are \( z_{1,2} = -\alpha \pm \sqrt{(\alpha^2 + \alpha)} \). Thus the DSMC system will diverge from the sliding mode and go unstable if either \( |z_1| \) or \( |z_2| > 1 \). In other words the conventional DSMC system will be unstable if
\[
\alpha > \frac{1}{3}
\] (6.47)

For the simulation, a value of \( \alpha = 0.5 \) and \( N = 10 \) is used to illustrate the effectiveness of MPSMC in stabilizing the system towards the sliding mode unlike the conventional DSMC when the system goes unstable. The following plots show the simulation results of the conventional Discrete-Time Sliding Mode Control (DSMC) along with the new Model Predictive Sliding Mode Control (MPSMC - method 1) for different values of \( \lambda \).

Figure 6.1 Simulation result showing divergence and convergence of system to and away from the desired sliding mode using DSMC and 1st MPSMC method respectively.

Figure 6.1 shows the value of sliding mode \( S = Gx = x, \) since \( G = 1 \) for the conventional DSMC with unit-step delayed disturbance estimate and the new MPSMC for two
different values of ‘$\lambda$’, namely 0.001 and 0.01. It is clearly seen that the sliding mode value diverges from the desired sliding mode ($x = 0$), and the system goes unstable when the conventional DSMC is used. However, for the MPSMC controller, the system converges to the desired sliding mode ($x = 0$) for both values of ‘$\lambda$’. It is also observed that the convergence of the system is quicker and smoother for the higher value of ‘$\lambda$’. This is explained by the fact that the disturbance is a function of the control input ($u$) and hence a higher weight ‘$\lambda$’ on the control change will lead to smoother convergence to the sliding mode.

Figure 6.2 Simulation result showing control input ($u$) for DSMC (unstable) and 1st MPSMC method (stable)
Figure 6.2 shows the value of control input \((u_k)\) as a function of time for both DSMC and MPSMC methods. It can be clearly seen that while DSMC system goes unstable, MPSMC remains stable and converges to the origin. Again the MPSMC controller with a higher weight on the control change allows the system to converge to the origin faster.

### 6.3.2 Experimental Result

The effectiveness of MPSMC’s first method over the DSMC has been demonstrated using the above simulation, and we now proceed to apply the concept for the force phase of the two-stage actuation system. In this case the controller (voltage input to the Piezoelectric Actuator) has a saturation limit of \(-200\) to \(1000\) Volts. The system model and sliding mode parameters are

\[
A_c = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\xi\omega_n \end{bmatrix}, \quad B_c = \begin{bmatrix} 0 \\ k \end{bmatrix}
\]

and

\[
G = \begin{bmatrix} 150 & 1 & 0 \end{bmatrix}
\]

where \(\omega_n = 1.1e3\) \(\text{rad.s}^{-1}\), \(\xi = 0.72\) and \(k = 3.51e3\). Also sampling time and horizon window length are chosen to be \(T = 0.5\text{ms}\) and \(N = 10\). The system states are the generated force by the piezoactuator and its derivative. The controller is designed to force the system to track a constant reference input of \(R = 4.5\) kN. The force is measured while its derivative is computed using the backward difference method. The control is saturated to lie between 0 and 1000 Volts. Figure 6.3 shows the schematic of the first MPSMC method as applied to this system.
Figure 6.3 Simple Schematic of the 1st MPSMC method

Figure 6.4 Generated force vs time for DSMC and MPSMC’s 1st method
Figure 6.5 Generated force vs time (Zoomed in)

Figure 6.6 Sliding mode value vs time for DSMC and MPSMC’s 1st method
Figure 6.7 Phase plane plot showing elimination of chattering by using MPSMC

Figure 6.8 Applied control voltage to piezo vs. time for DSMC and MPSMC’s 1st method
Figure 6.9 Applied control voltage vs time (Zoomed in)

Figure 6.10 Control Moves ($\Delta U$) for DSMC and MPSMC’s 1st method
Figure 6.4 to Figure 6.10 show the experimental results of the system applying DSMC and the first method of the Model Predictive Sliding Mode Control. The results compare the performance of the 2\textsuperscript{nd} MPSMC method with two different values (1 and 6) for the weight on the control move ‘\(\lambda\)’, as defined in equation (6.32) and that of the DSMC.

Figure 6.4 shows the resulting force tracking performance for different values of ‘\(\lambda\)’. It is seen that the performance is similar with respect to the settling time (about 56 ms) for all the three cases, however the oscillations along the response are reduced considerably for higher values of ‘\(\lambda\)’. Figure 6.5 shows the zoomed-in force tracking performance to clearly highlight the oscillations in the DSMC controller and the smoother tracking in the MPSMC controllers. Figure 6.6 and Figure 6.7 show the time history of the sliding mode value and the phase-plane trajectory of the tracking error and its derivative respectively. It is observed that the sliding mode is reached with lesser oscillations in the MPSMC cases with higher ‘\(\lambda\)’ value as compared to the DSMC case. While the oscillations stay on for a longer period in the case of the conventional discrete-time sliding mode control (DSMC), they die out rather quickly when ‘\(\lambda\)’ is increased. Figure 6.8 shows that the net control effort in each of the three cases. Figure 6.9 shows the zoomed-in plot of the control input values. It is noted that the control input is smooth for the MPSMC cases with higher ‘\(\lambda\)’ (>0) but is rather oscillatory for the conventional DSMC. Figure 6.10 shows the applied control moves (\(\Delta U\)) for the first 0.05 s for different values of ‘\(\lambda\)’. It is noted that the control moves are smaller for higher values of ‘\(\lambda\)’. This is explained by the optimization algorithm that adds higher weighting on the control moves for higher values of ‘\(\lambda\)’ thereby resulting in smaller control moves. In a nutshell, about 56 ms settling time
is achieved without any overshoot and any unnecessary control and state oscillations using the new MPSMC method.

Figure 6.11 Experimental result showing response deterioration when faster settling time (< 50 ms) is desired

However when the response time is increased further, using a higher sliding mode matrix $G$, like $G = [250 \ 1]$, the response deteriorates as shown in Figure 6.11, showing considerable overshoot. The reason for this is that the disturbance estimates for faster response is inaccurate due to other unmodeled dynamics in the system like sensor and power supply dynamics. Nevertheless, a settling time of 56 ms is achieved smoothly in the previous responses is very attractive and desirable and is rarely achieved by conventional and other means on controllable actuators for clutches and brakes. Thus, this first MPSMC method improves the system performance in reaching and
staying on the sliding mode through a smoother trajectory by imposing a weight on the control move at each instant when the sliding mode is enforced in spite of the unmodeled dynamics and nonlinearities like hysteresis in the system.

6.4 Model Predictive Sliding Mode Control – Method 2

Consider a continuous time dynamic system with disturbance

\[ \dot{X} = A_c X(t) + B_c U(t) + D_c f(t) \] (6.48)

where \( X \in \mathbb{R}^n \) is the state vector, \( U \in \mathbb{R}^m \) is the control, \( f(t) \in \mathbb{R}^p \) is the time-varying disturbance vector. \( A_c, B_c \) and \( D_c \) are the system, control and disturbance matrices of appropriate dimensions. The disturbance is added to model the uncertainties and unmodeled dynamics in the system. To control the system to track a reference input \( R \), we define the instantaneous tracking error \( E \) as

\[ E(t) = X(t) - R(t) \] (6.49)

The corresponding system differential equation in terms of the instantaneous tracking error is

\[ \dot{E} = A_c E + B_c U + d(t) \] (6.50)

where

\[ d(t) = A_c R(t) + D_c f(t) - \dot{R}(t) \] (6.51)

This accumulated sum is the total disturbance in the system that accounts for uncertainties, unmodeled system dynamics and the effect of reference inputs. Given a sampling time \( T \), one may convert this continuous-time system given by equation (6.50) into an equivalent discrete time system using standard procedure. This yields
\[ E_{k+1} = AE_k + BU_k + d_k \]  \hspace{1cm} (6.52)

where one may define the sliding mode as

\[ S_k = GE_k \]  \hspace{1cm} (6.53)

Hence one may extend equation (6.53) as \( S_{k+1} = GE_{k+1} \). In continuous time one may obtain the expression for the equivalent control by equating the time derivative of the sliding mode to zero. In discrete time systems, one may write

\[ \dot{S} = 0 \Rightarrow \frac{S_{k+1} - S_k}{T} = 0 \]  \hspace{1cm} (6.54)

where \( T \) is the sampling time interval. Hence to obtain the equivalent control value, we set

\[ S_{k+1}|_{U_{k\text{-eq}}} = S_k|_{U_{k\text{-eq}}} \Rightarrow (GE_{k+1})|_{U_{k\text{-eq}}} = (GE_k)|_{U_{k\text{-eq}}} \]  \hspace{1cm} (6.55)

Now using the equation (6.52) in equation (6.55), we have

\[ \Rightarrow G(AE_k + BU_{k\text{-eq}} + d_k) = GE_k \]  \hspace{1cm} (6.56)

\[ \Rightarrow U_{k\text{-eq}} = -(GB)^{-1}(-S_k + (GA)E_k + Gd_k) \]  \hspace{1cm} (6.57)

However, when the disturbances are unmodeled, the exact value of \( d_k \) is unknown at time instant \( k \). Hence a unit-step delayed estimate of the disturbance may be used \[Su-2000\]. Hence we write

\[ U_{k\text{-eq}} = -(GB)^{-1}(-S_k + (GA)E_k + Gd_{k-1}) \]  \hspace{1cm} (6.58)

where \( d_{k-1} = E_k - AE_{k-1} + BU_{k-1} \) is known exactly at the time instant \( k \), assuming that all state information is available.
6.4.1 Equivalent Control Augmentation

Now one may design the actual total control at the time instant ‘$k$’ as

$$U_k = U_k^* + U_{k-eq}$$  \hspace{1cm} (6.59)

Here, the first part in equation (6.59) is used to force the system to reach the sliding mode while the equivalent control is used to force the system to stay on the sliding mode. By applying this control input at time instant $k$, the value of the sliding mode $S$ at time instant $k+1$ is given by

$$S_{k+1} = G(AE_k + B(U_{k-eq} + U_k^*) + d_k)$$  \hspace{1cm} (6.60)

Simplifying using equation (6.58), one may write

$$\Rightarrow S_{k+1} = S_k + (GB)U_k^* + \varepsilon_k$$  \hspace{1cm} (6.61)

where $\varepsilon_k = G(d_k - d_{k-1})$, represents the error due to the application of a unit-step delayed estimate of the disturbance. This equation describes the dynamics of the sliding mode. One may use this equation to design an optimal way to enforce sliding mode. The degree of freedom in choosing $U_k^*$ allows us to control the way the system approaches the sliding mode. Equation (6.61) is basically a one step prediction of the sliding mode dynamics. One may extend this equation to obtain a general ‘$N$’ step prediction as follows

$$S_{k+2} = S_{k+1} + (GB)U_{k+1}^* + \varepsilon_{k+1} = S_k + (GB)(U_k^* + U_{k+1}^*) + \varepsilon_{k+1} + \varepsilon_k$$  \hspace{1cm} (6.62)

$$S_{k+3} = S_{k+2} + (GB)U_{k+2}^* + \varepsilon_{k+2} = S_k + (GB)(U_k^* + U_{k+1}^* + U_{k+2}^*) + \varepsilon_{k+2} + \varepsilon_{k+1} + \varepsilon_k$$  \hspace{1cm} (6.63)

Continuing in this fashion one may write a general prediction equation as follows

187
where ‘$S^k$’ and ‘$U^k$’ represent the predicted future estimates of the sliding mode ‘$S$’ and the control ‘$U^k$’ for the next ‘$N$’ sampling intervals. Writing the equation in a simpler format and omitting the ‘$k$’ subscripts for brevity we have,

$$ S = P_s S_k + H_s U^k + L_s \varepsilon $$  \hspace{1cm} (6.65)

Since only $U^k$ is used in driving the system states to the sliding mode, we may optimize the sliding mode dynamics using this part of the controller. The following partial optimization algorithm using MPC strategy may be applied

$$ \min_{U^k} J = \|S\|^2 + \lambda \|U^k\|^2 $$  \hspace{1cm} (6.66)

It is noted that only the partial control $U^k$ is used in the cost function, since both the terms must tend to zero in the steady state, and since $U^k$ is the part that enforces sliding mode, it is expected to tend to zero as the sliding mode is enforced. Using equation (6.65) one may rewrite this optimization problems as

$$ \min_{U^k} J = \left( P_s S_k + H_s U^k + L_s \varepsilon \right)^T \left( P_s S_k + H_s U^k + L_s \varepsilon \right) + \lambda \left( U^k \right)^T U^k $$  \hspace{1cm} (6.67)

Applying the optimization criteria by setting $\frac{\partial J}{\partial U^k} = 0$, we have
\[ U^* = - \left( H_s^T H_s + \lambda I \right)^{-1} H_s^T \left( P_s S_k + L_s \varepsilon \right) = -K^* \left( P_s S_k + L_s \varepsilon \right) \quad (6.68) \]

\[ \Rightarrow U_k^* = -e_1^T K^* \left( P_s S_k + L_s \varepsilon \right) \quad (6.69) \]

where \( K^* = \left( H_s^T H_s + \lambda I \right)^{-1} H_s^T \) and \( e_1^T = [1, 0, 0 \cdots 0] \), since only the first value of the estimated control sequence is applied here. Finally the total control value is obtained using equations (6.58), (6.59) and (6.69) as

\[ U_k = -(GB)^{-1} \left( -S_k + (GA)E_k + Gd_{k-1} \right) - e_1^T K^* \left( P_s S_k + L_s \varepsilon \right) \quad (6.70) \]

and again the since the exact value of \( \varepsilon \) is not exactly known since it contains the future disturbance values. Hence we replace this by an extended-zero-order hold estimation, thereby yielding

\[ U_k = -(GB)^{-1} \left( -S_k + (GA)E_k + Gd_{k-1} \right) - e_1^T K^* \left( P_s S_k + L_s \varepsilon_{\text{est}} \right) \quad (6.71) \]

where \( \varepsilon_{\text{est}} = [e_{k-1}, e_{k-1}, \ldots, e_{k-1}]^T \in \mathbb{R}^n \).

### 6.4.2 Simulation Example 2

The uncertain integrator plant used in the first simulation example in the previous sections for the 1st MPSMC method is again used here for the illustration of the working of the 2nd MPSMC methodology. Hence

**ACTUAL SYSTEM:**

\[ \dot{x} = (1 + \alpha)u \quad (6.72) \]

where \( \alpha \) is unknown but bounded.

**SYSTEM MODEL:**

\[ \dot{x} = u + d(t) \quad (6.73) \]
After discretization, we have

**ACTUAL DISCRETIZED SYSTEM:**
\[
x_{k+1} = x_k + (1 + \alpha) T u_k
\]  
(6.74)

**DISCRETIZED SYSTEM MODEL:**
\[
x_{k+1} = x_k + T u_k + d_k
\]  
(6.75)

The parameters used here are \( T = 0.05, \alpha = 0.5, G = 1, x(0) = 10 \) and \( N = 10, \lambda = 0.5, 10 \)

The simulation results are shown in Figure 6.12 to Figure 6.14. It is again noted that while the DSMC system diverges from the desired sliding mode and goes unstable, the 2\textsuperscript{nd} MPSMC method guides the system to the sliding mode \((x = 0)\). Also the effect of adding a weight on the partial control input \( U^* \) is noted in Figure 6.14.

From the figures, it is observed that the second method is an alternative way to integrate MPC and DSMC effectively to encounter uncertainty in the system even when the disturbance is a function of the control input.

![Figure 6.12 Simulation result showing divergence and convergence of system to and away from the desired sliding mode using DSMC and 2\textsuperscript{nd} MPSMC method respectively](image-url)
Figure 6.13 Simulation result showing control input ($u$) for DSMC (unstable) and 1$^{\text{st}}$ MPSMC method (stable)

Figure 6.14 Simulation Result showing the partial control input ($U^*$) that drives the system to the sliding mode using 2$^{\text{nd}}$ MPSMC method for two $\lambda$ values
6.4.3 Experimental Result

The effectiveness of MPSMC’s 2nd method over the DSMC has been demonstrated using the above simulation, and we now proceed to apply the concept for the force phase of the two-stage actuation system. In this case the controller (voltage input to the Piezoelectric Actuator) has a saturation limit of –200 to 1000 Volts. The system model and sliding mode parameters are

\[
A_c = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\xi\omega_n \end{bmatrix}, \quad B_c = \begin{bmatrix} 0 \\ k \end{bmatrix}
\]

and \( G = [150 \ 1] \) where \( \omega_n = 1.1e3 \text{ rad/s} \), \( \xi = 0.72 \) and \( k = 3.51e3 \). Also sampling time and horizon window length are chosen to be \( T = 0.5ms \) and \( N = 10 \). The system states are the generated force by the piezoactuator and its derivative. The controller is designed to force the system to track a constant reference input of \( R = 4.5 \text{ kN} \). The force is measured while its derivative is computed using the backward difference method. The control is saturated to lie between 0 and 1000 Volts. Figure 6.15 shows the schematic of the second MPSMC method as applied to this system.

![Figure 6.15 Simple Schematic of the 2nd MPSMC method](image)

192
Figure 6.16 Generated force vs time for DSMC and MPSMC’s 2\textsuperscript{nd} method

Figure 6.17 Generated force vs time (Zoomed in)
Figure 6.18 Sliding mode value vs time for DSMC and MPSMC’s 2\textsuperscript{nd} method

Figure 6.19 Phase Plane Plot showing elimination of chattering by using MPSMC’s 2\textsuperscript{nd} method
Figure 6.20 Applied control voltage to piezo vs. time for DSMC and MPSMC’s 2nd method

Figure 6.21 Applied control voltage vs time (Zoomed in)
Figure 6.22 Partial control input \((U^*)\) vs time for DSMC and 2\textsuperscript{nd} MPSMC method

Figure 6.16 to Figure 6.22 show the experimental results of the system applying DSMC and the 2\textsuperscript{nd} method of the Model Predictive Sliding Mode Control. The results compare the performance of the 2\textsuperscript{nd} MPSMC method with two different values (5 and 10) for the weight on the partial control input, ‘\(\lambda\)’, as defined in equation (6.66) and DSMC. Figure 6.16 shows the resulting force tracking performance for different values of ‘\(\lambda\)’. It is seen that the performance is similar with respect to the settling time (about 60 ms) for all the three cases, however the oscillations along the response are reduced considerably for higher values of ‘\(\lambda\)’. Figure 6.17 shows the zoomed-in force tracking performance to clearly highlight the oscillations in the DSMC controller and the smoother tracking in the MPSMC controllers. Figure 6.18 and Figure 6.19 show the time history of the sliding
mode value and the phase-plane trajectory of the tracking error and its derivative respectively. It is observed that the sliding mode is reached with lesser oscillations in the MPSMC cases with higher ‘λ’ value as compared to the DSMC case. While the oscillations stay on for a longer period in the case of the conventional discrete-time sliding mode control (DSMC), they die out rather quickly when ‘λ’ is increased. Figure 6.20 shows that the net control effort in each of the three cases. Figure 6.21 shows the zoomed-in plot of the control input values. It is noted that the control input is smooth for the MPSMC cases with higher ‘λ’ (>0) but is rather oscillatory for the conventional DSMC. Figure 6.22 shows the applied partial control input (U*) for the first 0.066 s for different values of ‘λ’. It is noted that the control input (U*) is smoother for the MPSMC controller with higher values of ‘λ’. In a nutshell, about 60 ms settling time is achieved without any overshoot and any unnecessary control and state oscillations using the 2nd MPSMC method.

Thus, this 2nd MPSMC method also improves the system performance in reaching and staying on the sliding mode through a smoother trajectory by imposing a weight on the partial control input that drives the system to the sliding mode at each instant when the sliding mode is enforced in spite of the unmodeled dynamics and nonlinearities like hysteresis in the system.

6.5 Concluding Remarks

A new method called Model Predictive Sliding Mode Controller (MPSMC) has been proposed and discussed in detail in this chapter that overcomes the limitations of the
conventional discrete-time sliding mode control (DSMC) in the presence of special uncertainties and disturbances in the system. The primary philosophy is to drive the system to the sliding mode in an optimal manner unlike DSMC. This optimization is carried out using the popular MPC strategy using a receding horizon window. Two different methods using the basic idea have been analyzed. The main difference between the working of the 1st and 2nd MPSMC controller is the different ways the optimization problem is defined and performed. In the first case, the cost function involves both the sliding mode value and the control moves so that in steady state, the cost function tends to zero when optimized. In the second case, the control input is split into two parts, with one part ($U^*$) driving the system to the sliding mode and one part ($U_{eq}$) that ensures the system stays on the sliding mode. It is evident that the part that ($U^*$) plays a key role in determining the convergence to the sliding mode. Thus the optimization problem is defined and solved using this part of the controller. The results suggest that both these methods can be used effectively in controlling dynamic systems with uncertainties to arrive at robust pre-defined system performance.
CHAPTER 7

CONCLUSIONS

This chapter summarizes all the research work done for this thesis. The chapter begins with a review of the research conducted in this thesis with brief outlines from each of the chapters. Then the chapter ends with some recommendations and scope for future research in related topics.

7.1 Summary

The primary objectives of this research were to design, develop and control a high-force and high-stroke actuation mechanism using smart material actuators that can potentially replace existing inefficient controllable power transmission mechanisms used in the automobiles. In order to meet the proposed objectives, important preparatory work was completed. This included a comprehensive literature review of i) existing controllable clutching and braking mechanisms and clutching and braking requirements, ii) modeling of smart material actuator systems and iii) advanced control strategies suitable for smart material systems.
Based on the typical requirements of a clutch actuation process and its unique stiffness characteristics, a new concept of clutch actuation mechanism was developed by splitting the total actuation process into two phases, namely the stroke and force phases with contrasting force-displacement requirements. The stroke phase required more displacement and less force while the force phase required relatively higher force and lower stroke. The stroke phase produced the necessary stroke or displacement (about 2-5 mm) during the initial part of the actuation process and was operated using a classical DC motor. The force phase produced the majority of the force during the compressive action of the friction elements of the clutch or brake pads and was carried out using a smart material actuator like piezoelectric actuator. The dynamic modeling and design of this mechanism was carried out for the two phases individually. The stroke phase was modeled using the classical DC motor relating the applied armature voltage to the motor speed and current drawn. The various elements of the stroke phase were designed based on the stroke to be produced and the initial preload required. The force phase was modeled using a simple second order system incorporating the model of the piezoactuator. The piezoactuator was chosen based on the required force to be generated by the system (about 5 kN). The piezoactuator modeling focused on capturing its primary nonlinearity, namely the hysteresis property. Different models were studied including the domain-wall model and Preisach model. However, their limitations initiated the development of a simpler variable delay and variable gain model that provides a basis for robust controller design. A simple experimental system was built to validate the model and test the control strategies developed later. The model was validated by running simple tests on the system and the model parameters were chosen to suit the
system response. Also a method to estimate the force generated by the system without a
mechanical force sensor like a load cell was developed.

The rest of the thesis focused on the development of appropriate advanced control
strategies to control the two-stage actuation system to track different reference force
requirements. Different control methodologies like Internal Mode Control (IMC), Model
Predictive Control (MPC) and Discrete-time Sliding Mode Control were studied. The
different control algorithms developed were implemented in the system and tested using
the MATLAB/Simulink® software combined with a real-time hardware-in-the-loop and
fast-prototyping tool called DSpace®.

IMC, a popular control strategy among chemical engineers was adapted to design a
robust controller for the force phase of the system. The unique structure of the IMC
control system was utilized to explicitly use the model for the controller design. The
variable delay model developed earlier in the thesis was used to obtain a bound on the
uncertainty in the system model. This uncertainty was used to appropriately design a
robust IMC controller. The IMC controller was tested on the system and was unable to
provide good tracking performance for a step reference tracking. However, the IMC
system was able to track periodic signals like sinusoidal inputs very accurately. The
limitation of the IMC with respect to step input tracking was directly associated to the
model used in the system, which was primarily based on a quasi-static sinusoidal voltage
input to the piezoelectric actuator.

MPC, again a robust control strategy was tailored to design a reference force-tracking
controller for the piezoelectric actuator system. MPC, also known as the receding-horizon
controller, was designed to produce optimal control actions based on minimizing a
predefined cost function up to a finite moving window into the future at each and every sampling instant. The controller was designed to account for unmodeled disturbances in the system including the nonlinear piezoelectric actuator property using a time-varying disturbance updated at each and every sampling instant. Also since the system had a really fast sampling instant of < 1 ms the control equations were customized so that most of the computations involved in the optimization was carried out offline. The MPC controller was implemented in system and able to provide considerable improvement in the system performance as compared to the IMC controller. A settling time less than 0.1 s was achieved without considerable overshoot. The MPC controller also provided excellent time-varying reference tracking. The limitations of the MPC controller were two-fold. One was that it required the knowledge of future reference inputs, which may not be readily available in practical applications. The second limitation was that the controller parameters like the weight on the control moves in the cost-function cannot be directly chosen based required system performance like settling time. The parameters were chosen based on a trial and error method using simulation and testing of the system.

Finally, a new control strategy called Model Predictive Sliding Mode Control (MPSMC) was developed to overcome the shortcomings of the conventional Discrete-Time Sliding Mode Control (DSMC). A DSMC control system using a unit-step delayed disturbance estimate can go unstable or produce chattering when the disturbance in the system depends on the control input to the system, as witnessed in many dynamic systems that include actuator nonlinearities like piezoelectric and smart material systems. This new MPSMC method used a discrete-time state-space approach and hence may be suitable for MIMO systems. This new control strategy, fundamentally a sliding
mode controller, was designed to enforce the sliding mode optimally unlike the conventional DSMC, where the sliding mode is enforced at the very next sampling instant. The optimization was carried out using MPC’s receding window strategy. Two different methods using different cost function definitions were proposed using this concept. The first method used a cost-function based on the sliding mode control value and the control moves at each and every step. This method used the direct implementation of the MPC strategy. The second method was developed by splitting the control input into two parts, namely the part that drives the system to the sliding mode and the part that ensures the system stays on the sliding mode. The cost function defined in this case used the sliding mode value and the part of the control input that drives the system to the sliding mode. The effectiveness of both the methods was illustrated using simulation examples of a simple first order integrator plant, with an uncertain control gain. The results were compared to the conventional DSMC system and clearly showed that the DSMC method was unstable while the new MPSMC methodologies stabilized the system and led to the convergence of the system to the sliding mode. The same concepts were applied to the two-stage actuation experimental system and the results showed the best performance among the three controllers with a settling time of about 60 ms without any overshoot. Further increase in the speed of response led to deterioration of the system performance.
7.2 Scope for Future Work

Likely extensions of the work performed in this thesis will concern some key issues not addressed in this thesis concerning the practical application of the two-stage actuation mechanism.

One major requirement in clutch and braking application that was not considered in this dissertation was the need for a cooling system to cool the actuation mechanism and the slipping surfaces involved in clutches and brakes. A typical commercially available piezoelectric actuator can maintain consistent performance from very low temperatures around –200 degree Celsius up to a maximum temperature called the Curie temperature. Curie temperatures are usually in the range of 250 degree Celsius for state-of-the-art piezoelectric actuators. This range is usually sufficient for a clutching application wherein the required temperature range is from –40 degree to 120 degree Celsius. However, higher temperatures may be encountered in a braking application and appropriate research must be conducted to design enclosures and cooling system for the piezoelectric actuators. Also in a clutching application, a thrust bearing is required to couple the stationary piezoelectric actuators to the rotating clutch plates. Hence some of the issues to be addressed in this area include operating-speed limits, drag losses and heating of the thrust bearing. The thrust bearings used between the rotating clutch plates and the piezo actuators must carry the axial load on the clutch plates and also simultaneously satisfy the rotating speed requirements. This might lead to power losses in the thrust bearings during torque transmission. This may be avoided to some extent by appropriately selecting a ball thrust bearing or a spherical roller thrust bearing instead of a hydrodynamic thrust bearing. Spray oil lubrication may be used to cool the thrust bearing during operation.
One other issue to be properly addressed is the compact packaging of the various elements in the actuation mechanism. Since space is a big constraint in automobiles, it is essential that the size of the actuation mechanism be made as small as possible. One possible solution is proposed here, wherein the DC motor is placed in parallel to the piezoactuator thereby reducing the overall length of the mechanism. This is illustrated in Figure 7.1.

![Figure 7.1 Proposed Design for Size-Reduction](image)

One other study that was not performed in detail in this application is the cost-worthiness analysis of this device in automotive applications. Though the current price (as of 2005) of a single piezoelectric actuator is very high ($500 – 1000$), this is expected to come down drastically as the volume of production increases. However, more research is required to arrive at estimates of overall cost of the device. This will help arrive at meaningful conclusions on the feasibility of carrying this product successfully into production.
REFERENCES


[24] Han, W., Yi, S-J. “A study of shift control using the clutch pressure pattern in


[57] Park, J-K., “Hysteresis modeling of piezoceramic actuators using a hybrid modeling strategy”, *MS Thesis*, The Ohio State University, 2004


[62] Pike, A.W., Grimble M.J., Johnson, M.A., Ordys, A.W.,


[76] Soeterboek, R., “Predictive Control, a Unified Approach”,


212
[89] www.americanpiezo.com (APC International, Ltd.)

[90] www.bts.gov (Bureau of Transportation Statistics, US Department of Transportation)

[91] www.epa.gov (US Environmental Protection Agency)


[93] www.physikinstrumente.com (Physik Instrumente, GmbH Co. & K)
