FRACTIONAL ORDER MODELING AND CONTROL: DEVELOPMENT OF ANALOG STRATEGIES FOR PLASMA POSITION CONTROL OF THE STOR-1M TOKAMAK

by

Shayok Mukhopadhyay

A thesis submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

Electrical Engineering

Approved:

Dr. YangQuan Chen
Major Professor

Dr. Donald Cripps
Committee Member

Dr. Ajay Singh
Committee Member

Dr. Byron R. Burnham
Dean of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah
2009
Abstract

Fractional Order Modeling and Control: Development of Analog Strategies for Plasma Position Control of the STOR-1M Tokamak

by

Shayok Mukhopadhyay, Master of Science
Utah State University, 2009

Major Professor: Dr. YangQuan Chen
Department: Electrical and Computer Engineering

This work revolves around the use of fractional order calculus in control science. Techniques such as fractional order universal adaptive stabilization (FO-UAS), and the fascinating results of their application to real-world systems, are presented initially. A major portion of this work deals with fractional order modeling and control of real-life systems like heat flow, fan and plate, and coupled tank systems. The fractional order models and controllers are not only simulated, they are also emulated using analog hardware. The main aim of all the above experimentation is to develop a fractional order controller for plasma position control of the Saskatchewan torus-1, modified (STOR-1M) tokamak at the Utah State University (USU) campus. A new method for plasma position estimation has been formulated. The results of hardware emulation of plasma position and its control are also presented. This work performs a small scale test measuring controller performance, so that it serves as a platform for future research efforts leading to real-life implementation of a plasma position controller for the tokamak.

(167 pages)
To my family and friends for making my life a pleasurable experience.
Acknowledgments

I would like to thank my advisor, Dr. Chen, for his ideas and guidance. Without his motivation and insight this work would have been impossible. He always answered my questions and helped me out when I was stuck with a problem.

I would like to thank the Utah Science Technology and Research Initiative (USTAR), the Electrical and Computer Engineering (ECE) Department, Utah State University (USU), and Dr. Chen for their support which made this work possible. I would like to thank my committee members: Dr. Cripps for his comments; and Dr. Singh for explaining the basics of plasma physics, the operation of the tokamak, helping with practical work in the plasma physics lab, and answering my questions. I would also like to thank Professor Farrell Edwards for providing me the opportunity to work on this project and helping me understand its scope and aim. He is the person responsible for making this work take shape. I would like to thank Jared Bench for bringing me up to speed initially, getting me familiarized with the tokamak, and also explaining the details about the various electrical systems. I would like to thank Jeremy Bishop for answering all my practical, as well as theoretical, questions during my work at the plasma physics lab and helping me out on my projects; Heidi Johnson for helping me out on some of the physics; and Tom Apedaile for his help with some of the practical hands-on work on the tokamak.

I would like to thank Dr. Yan Li for being my fractional calculus mentor and also being a very good friend who was always ready to help me with my problems. Understanding fractional calculus would have been a monumental task without him. I would like to thank Calvin Coopmans for his help with the fractional order controller and model boards. Without his hardware expertise none of the experiments would have seen light of day. I would like to thank Yiding Han for the countless late night rides back home, which helped me a lot; Hu Sheng for his help with the fan and plate experiments; Amin Hernandez for his help with one of the controller boards; and Justin Hamm for my great new computer. I would like to thank all the other members of CSOIS for making my time here enjoyable, and
the members of the Applied Fractional Calculus (AFC) reading group for creating a solid platform to discuss ideas. The constructive criticism obtained from the members greatly helped me with my problems encountered during the course of this work. I would like to thank Dr. Chen again for guiding me through my ECE 6930 independent study course in fractional calculus, which formed the backbone for this work, and also Varsha Bhambhani for her notes on the subject. Dr. Bruce J. West’s visit to CSOIS, USU, organized by Dr. Chen and the ECE Department, deserves commendation since it provided me with new perspectives towards fractional calculus.

Finally, I would like to thank my family for always standing by my decisions and having faith in me, and my fiancée for being infinitely patient.

Shayok Mukhopadhyay
# Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Abstract</strong></td>
<td>iii</td>
</tr>
<tr>
<td><strong>Acknowledgments</strong></td>
<td>v</td>
</tr>
<tr>
<td><strong>List of Tables</strong></td>
<td>x</td>
</tr>
<tr>
<td><strong>List of Figures</strong></td>
<td>xi</td>
</tr>
<tr>
<td><strong>Acronyms</strong></td>
<td>xvi</td>
</tr>
<tr>
<td><strong>1 Introduction</strong></td>
<td>1</td>
</tr>
<tr>
<td>1.1 Motivation</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Literature Review</td>
<td>2</td>
</tr>
<tr>
<td>1.3 Contribution and Organization</td>
<td>3</td>
</tr>
<tr>
<td><strong>2 Fractional Order Calculus in Controls: A First Glance</strong></td>
<td>6</td>
</tr>
<tr>
<td>2.1 Fractional Order PID Controllers: An Overview</td>
<td>6</td>
</tr>
<tr>
<td>2.2 Fractional Order Universal Adaptive Stabilization (FO-UAS)</td>
<td>7</td>
</tr>
<tr>
<td>2.2.1 Mathematical Preliminaries</td>
<td>7</td>
</tr>
<tr>
<td>2.2.2 Nussbaum Function</td>
<td>9</td>
</tr>
<tr>
<td>2.2.3 Stability Analysis of the Fractional Order UAS</td>
<td>10</td>
</tr>
<tr>
<td>2.2.4 Experimental Setup</td>
<td>11</td>
</tr>
<tr>
<td>2.2.5 Results I: FO-UAS vs. IO-UAS</td>
<td>12</td>
</tr>
<tr>
<td>2.2.6 Results II: Effect of Different Nussbaum Functions on FO-UAS</td>
<td>13</td>
</tr>
<tr>
<td>2.2.7 Results III: Other Experiments</td>
<td>18</td>
</tr>
<tr>
<td>2.2.8 Results IV: Effect of Varying $b$, $c$, and $\alpha_2$</td>
<td>22</td>
</tr>
<tr>
<td>2.2.9 Results V: Very Low-Speed Regulation and External Noise</td>
<td>23</td>
</tr>
<tr>
<td>2.3 Observations</td>
<td>23</td>
</tr>
<tr>
<td>2.4 Conclusion and Future Work</td>
<td>24</td>
</tr>
<tr>
<td><strong>3 Verification: Validation Platform 1</strong></td>
<td>28</td>
</tr>
<tr>
<td>3.1 The Coupled Tank System</td>
<td>28</td>
</tr>
<tr>
<td>3.1.1 Overview</td>
<td>28</td>
</tr>
<tr>
<td>3.1.2 System Identification</td>
<td>30</td>
</tr>
<tr>
<td>3.2 Experimental Setup</td>
<td>30</td>
</tr>
<tr>
<td>3.2.1 Hardware-in-the-Loop Implementation of a FO-PI Control System</td>
<td>31</td>
</tr>
<tr>
<td>3.2.2 Analog Hardware Implementation</td>
<td>31</td>
</tr>
<tr>
<td>3.3 Determining The Fractional Order $\alpha$ of a ‘Fractor’</td>
<td>33</td>
</tr>
<tr>
<td>3.4 Designing an Optimal FO-PI Controller</td>
<td>36</td>
</tr>
<tr>
<td>3.5 Approximation of a FO-PI Controller Using Practically Available Orders</td>
<td>36</td>
</tr>
<tr>
<td>3.6 Results</td>
<td>38</td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
</tr>
<tr>
<td>3.7</td>
<td>Conclusion</td>
</tr>
<tr>
<td>3.7.1</td>
<td>Integer Order Control vs. Fractional Order Control</td>
</tr>
<tr>
<td>3.7.2</td>
<td>Analog Implementation vs. Digital Implementation</td>
</tr>
<tr>
<td>3.8</td>
<td>Future Work</td>
</tr>
<tr>
<td>4</td>
<td>Verification: Validation Platform 2</td>
</tr>
<tr>
<td>4.1</td>
<td>The Heat Flow System: An Overview</td>
</tr>
<tr>
<td>4.2</td>
<td>System Modeling</td>
</tr>
<tr>
<td>4.2.1</td>
<td>Integer Order Modeling</td>
</tr>
<tr>
<td>4.2.2</td>
<td>Fractional Order Modeling</td>
</tr>
<tr>
<td>4.3</td>
<td>Controller Design</td>
</tr>
<tr>
<td>4.4</td>
<td>Experimental Setup</td>
</tr>
<tr>
<td>4.5</td>
<td>Results</td>
</tr>
<tr>
<td>4.6</td>
<td>Conclusion</td>
</tr>
<tr>
<td>5</td>
<td>Verification: Validation Platform 3</td>
</tr>
<tr>
<td>5.1</td>
<td>The Fan and Plate System: An Overview</td>
</tr>
<tr>
<td>5.2</td>
<td>System Modeling</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Integer Order Modeling</td>
</tr>
<tr>
<td>5.2.2</td>
<td>Fractional Order Modeling</td>
</tr>
<tr>
<td>5.3</td>
<td>Controller Design</td>
</tr>
<tr>
<td>5.4</td>
<td>Experimental Setup</td>
</tr>
<tr>
<td>5.5</td>
<td>Results</td>
</tr>
<tr>
<td>5.6</td>
<td>Conclusion and Future Work</td>
</tr>
<tr>
<td>6</td>
<td>Plasma Position Control of The STOR-1M Tokamak</td>
</tr>
<tr>
<td>6.1</td>
<td>A Tokamak - An Introduction</td>
</tr>
<tr>
<td>6.2</td>
<td>The STOR-1M Tokamak - System Overview</td>
</tr>
<tr>
<td>6.3</td>
<td>Basic Operation</td>
</tr>
<tr>
<td>6.4</td>
<td>Plasma Position Estimation</td>
</tr>
<tr>
<td>6.5</td>
<td>Plasma Position Modeling</td>
</tr>
<tr>
<td>6.6</td>
<td>Controller Design</td>
</tr>
<tr>
<td>6.7</td>
<td>Experimental Setup</td>
</tr>
<tr>
<td>6.8</td>
<td>Results</td>
</tr>
<tr>
<td>6.8.1</td>
<td>Simulation Results</td>
</tr>
<tr>
<td>6.8.2</td>
<td>Analog Hardware Emulation Results</td>
</tr>
<tr>
<td>6.9</td>
<td>Difference Based Modeling and Control</td>
</tr>
<tr>
<td>6.10</td>
<td>Conclusion</td>
</tr>
<tr>
<td>7</td>
<td>Conclusion</td>
</tr>
<tr>
<td>7.1</td>
<td>Summary</td>
</tr>
<tr>
<td>7.2</td>
<td>Future Work</td>
</tr>
<tr>
<td>References</td>
<td></td>
</tr>
</tbody>
</table>
## Appendices

<table>
<thead>
<tr>
<th>Appendix</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Fractional Order Derivatives and Integrals</td>
<td>103</td>
</tr>
<tr>
<td></td>
<td>A.1 Fractional Order Derivatives</td>
<td>103</td>
</tr>
<tr>
<td></td>
<td>A.2 Properties of Fractional Derivatives</td>
<td>105</td>
</tr>
<tr>
<td></td>
<td>A.3 Fractional Integrals and Laplace Transforms of Fractional Derivatives</td>
<td>106</td>
</tr>
<tr>
<td>B</td>
<td>Bode Plots of Available ‘Fractors’ - July 18, 2008</td>
<td>107</td>
</tr>
<tr>
<td>C</td>
<td>Bode Plots of Available ‘Fractors’ - June 22, 2009</td>
<td>113</td>
</tr>
<tr>
<td>D</td>
<td>Fractional Order Plant Board Details</td>
<td>118</td>
</tr>
<tr>
<td></td>
<td>D.1 Model Board Operational Waveforms and Schematics</td>
<td>118</td>
</tr>
<tr>
<td></td>
<td>D.2 Model Board Factsheet</td>
<td>119</td>
</tr>
<tr>
<td>E</td>
<td>Fractional Order Controller Board Details</td>
<td>121</td>
</tr>
<tr>
<td>F</td>
<td>Analog Signal Division - Schematics</td>
<td>123</td>
</tr>
<tr>
<td>G</td>
<td>STOR-1M Tokamak System Details</td>
<td>124</td>
</tr>
<tr>
<td>H</td>
<td>Mittag - Leffler Function: Code</td>
<td>126</td>
</tr>
<tr>
<td>I</td>
<td>Fractional Order System Modeling: Code</td>
<td>130</td>
</tr>
<tr>
<td>J</td>
<td>Approximation of an FO-PI Controller: Code</td>
<td>132</td>
</tr>
<tr>
<td></td>
<td>J.1 Parameter Setter Script</td>
<td>133</td>
</tr>
<tr>
<td></td>
<td>J.2 Position Estimator Model File</td>
<td>134</td>
</tr>
<tr>
<td></td>
<td>J.3 Plasma Position Calculator</td>
<td>135</td>
</tr>
<tr>
<td></td>
<td>J.4 Plasma Current Averaging and Validation</td>
<td>139</td>
</tr>
<tr>
<td></td>
<td>J.5 Plasma Current De-trending</td>
<td>140</td>
</tr>
<tr>
<td></td>
<td>J.6 Plasma Position Estimation Using Mirnov Coils</td>
<td>140</td>
</tr>
<tr>
<td></td>
<td>J.7 Vector Downsampler</td>
<td>151</td>
</tr>
</tbody>
</table>
List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Parameters of the ZN–PID controller for the coupled tank.</td>
<td>31</td>
</tr>
<tr>
<td>3.2</td>
<td>Parameters of the optimal FO-PI controller for the coupled tank.</td>
<td>36</td>
</tr>
<tr>
<td>3.3</td>
<td>Parameters of the AFO-PI controller for the coupled tank.</td>
<td>37</td>
</tr>
<tr>
<td>4.1</td>
<td>System parameters for the heat flow plant.</td>
<td>45</td>
</tr>
<tr>
<td>4.2</td>
<td>Heat flow plant - FO-model parameters.</td>
<td>47</td>
</tr>
<tr>
<td>4.3</td>
<td>Controller parameters for the heat flow system.</td>
<td>49</td>
</tr>
<tr>
<td>5.1</td>
<td>Initial parameters of the fan and plate system.</td>
<td>57</td>
</tr>
<tr>
<td>5.2</td>
<td>Fan and plate system - FO model parameters.</td>
<td>58</td>
</tr>
<tr>
<td>5.3</td>
<td>Controller parameters for the fan and plate plant.</td>
<td>64</td>
</tr>
<tr>
<td>6.1</td>
<td>STOR 1-M tokamak parameters.</td>
<td>74</td>
</tr>
<tr>
<td>6.2</td>
<td>Controller parameters for the STOR-1M tokamak.</td>
<td>84</td>
</tr>
<tr>
<td>6.3</td>
<td>Controller parameters for the STOR-1M tokamak - voltage difference based.</td>
<td>92</td>
</tr>
<tr>
<td>C.1</td>
<td>‘Fractor’ characteristics comparison.</td>
<td>113</td>
</tr>
<tr>
<td>G.1</td>
<td>STOR 1-M, bank voltages (nominal values).</td>
<td>124</td>
</tr>
<tr>
<td>G.2</td>
<td>STOR 1-M, horizontal and vertical field parameters.</td>
<td>124</td>
</tr>
<tr>
<td>G.3</td>
<td>STOR 1-M, Rogowski coil parameters.</td>
<td>125</td>
</tr>
<tr>
<td>G.4</td>
<td>STOR 1-M, plasma position magnetic pickup coil parameters.</td>
<td>125</td>
</tr>
<tr>
<td>G.5</td>
<td>STOR 1-M, bank timing and duration (ms).</td>
<td>125</td>
</tr>
<tr>
<td>G.6</td>
<td>STOR 1-M, mini shot log.</td>
<td>125</td>
</tr>
</tbody>
</table>
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>H-I-L FO-UAS control scheme.</td>
<td>12</td>
</tr>
<tr>
<td>2.2</td>
<td>Actual H-I-L setup.</td>
<td>13</td>
</tr>
<tr>
<td>2.3</td>
<td>Integer order UAS - squarewave tracking.</td>
<td>14</td>
</tr>
<tr>
<td>2.4</td>
<td>Integer order UAS - sinewave tracking.</td>
<td>14</td>
</tr>
<tr>
<td>2.5</td>
<td>Fractional order UAS - squarewave tracking, $\alpha = 0.6$.</td>
<td>15</td>
</tr>
<tr>
<td>2.6</td>
<td>Fractional order UAS - squarewave tracking, $\alpha = 0.1$.</td>
<td>15</td>
</tr>
<tr>
<td>2.7</td>
<td>Fractional order UAS - squarewave tracking, $\alpha = 0.3$.</td>
<td>16</td>
</tr>
<tr>
<td>2.8</td>
<td>Fractional order UAS - squarewave tracking, $\alpha = 0.75$.</td>
<td>16</td>
</tr>
<tr>
<td>2.9</td>
<td>Fractional order UAS - sinewave tracking, $\alpha = 0.5$.</td>
<td>17</td>
</tr>
<tr>
<td>2.10</td>
<td>Fractional order UAS - sawtooth wave tracking, $\alpha = 0.5$.</td>
<td>17</td>
</tr>
<tr>
<td>2.11</td>
<td>Fractional order UAS - Nussbaum function $= k^2 \cos(k)$.</td>
<td>18</td>
</tr>
<tr>
<td>2.12</td>
<td>Fractional order UAS - Nussbaum function $= k \cos(\sqrt{</td>
<td>k</td>
</tr>
<tr>
<td>2.13</td>
<td>Fractional order UAS - Nussbaum function $= \cos(\frac{\pi k}{2}) e^{k^2}$.</td>
<td>19</td>
</tr>
<tr>
<td>2.14</td>
<td>$\dot{k}(t) =</td>
<td>y(t)</td>
</tr>
<tr>
<td>2.15</td>
<td>$\dot{k}(t) = y^4(t)$.</td>
<td>21</td>
</tr>
<tr>
<td>2.16</td>
<td>$u(t) = +N(k(t))y(t) + (cb)^{-1}a_2 \alpha D_1^\alpha y(t)$.</td>
<td>21</td>
</tr>
<tr>
<td>2.17</td>
<td>Squarewave tracking - $\frac{a_2}{cb} = 0.5$.</td>
<td>22</td>
</tr>
<tr>
<td>2.18</td>
<td>Squarewave tracking - $\frac{a_2}{cb} = 2$.</td>
<td>24</td>
</tr>
<tr>
<td>2.19</td>
<td>Squarewave tracking - $\frac{a_2}{cb} = -0.5$.</td>
<td>25</td>
</tr>
</tbody>
</table>
2.20 Squarewave tracking ±0.1 rad/s. ........................................ 25
2.21 Squarewave tracking ±0.05 rad/s. ................................. 26
2.22 Squarewave tracking ±1 rad/s, external noise added. ......... 26
3.1 The coupled tank apparatus – courtesy of: KentRidge Instruments. .... 29
3.2 The coupled tank system - schematic. ............................. 29
3.3 Coupled tank sensor calibration. ..................................... 30
3.4 Simulink® model for digital FO-PID control. ..................... 31
3.5 Overall analog control layout. ..................................... 32
3.6 Analog FO-PID controller board - schematic. ................... 33
3.7 Analog FO-PID controller board - prototype. .................... 34
3.8 Bode plot of a ‘Fractor’ with α = 0.9. ................................. 34
3.9 Bode plot of a ‘Fractor’ with α = 0.868. .............................. 35
3.10 Bode plot of a ‘Fractor’ with α = 0.385. .............................. 35
3.11 Comparison of AFO-PI and FO-PI controllers. .................. 38
3.12 Comparison of control strategies - 1. ............................... 39
3.13 ITAE comparison - 1. ............................................. 40
3.14 Comparison of control strategies - 2. ............................... 41
3.15 ITAE comparison - 2. ............................................. 42
4.1 The heat flow plant. ................................................. 44
4.2 Fractional order vs. integer order modeling – comparison. ......... 45
4.3 Heat flow FO-model Bode plot. ..................................... 48
4.4 Heat flow system simulation setup. .................................. 49
4.5 FO vs. IO control simulation - heat flow system. .................. 50
4.6 Digital FO-PI control - heat flow system (H—I—L). ............... 51
4.7 Analog FO-PI control - heat flow system, experiment 1. ......... 52
4.8 Analog FO-PI control - heat flow system, experiment 2. 52
5.1 The fan and plate system. 54
5.2 The fan and plate system — step response. 55
5.3 Kalman filter implementation for the fan and plate system. 56
5.4 IO vs. FO modeling simulation - fan and plate system. 58
5.5 Fan and plate FO - model Bode plot. 59
5.6 Fan and plate FO-model board schematic. 60
5.7 Generalized analog FO-model board. 60
5.8 Generalized analog FO-model board schematic. 61
5.9 Fan and plate plant analog FO-model vs. plant data. 62
5.10 Analog fractional FO-PII controller board. 63
5.11 Digital FO-PI control setup - fan and plate system (H–I–L). 64
5.12 Analog FO-PI control of the analog FO-model board - setup. 65
5.13 FO-PI control vs. ZN-PID control - fan and plate system (H–I–L). 66
5.14 Analog FO-PI control of the analog FO-model board - result #1. 67
5.15 Analog FO-PI vs. analog PID control of the analog FO-model board. 68
5.16 Analog FO-PI control of the analog FO-model board - result #2. 69
6.1 The tokamak as a transformer. 72
6.2 The STOR-1M tokamak. 73
6.3 Various bank currents for the STOR-1M tokamak. 75
6.4 Plasma current and voltage. 75
6.5 The position measurement system. 76
6.6 Position estimation from the geometry. 77
6.7 Estimated position. 79
6.8 Plasma position modeling - $V_c = 123V$. 80
6.9 Bode plot for $G_{iT}(s)|_{V_c=123V}$

6.10 Analog hardware model of the tokamak for $V_c = 123V$

6.11 Plasma position control - simulation for $V_c = 123V$

6.12 Plasma position control (ratio) - hardware emulation for $V_c = 123V$

6.13 Plasma position control (error) - hardware emulation for $V_c = 123V$

6.14 Plasma position control (absolute) - hardware emulation for $V_c = 123V$

6.15 Plasma position control (test) - hardware emulation for $V_c = 123V$

6.16 Plasma position control (comparison) - hardware emulation for $V_c = 123V$

6.17 Modeling based on pickup coil signal difference

6.18 Controller performance for the signal difference case, reference = 0V

B.1 ‘Fractor-0’ Bode plot, $\alpha = 0.5175$

B.2 ‘Fractor-1’ Bode plot, $\alpha = 0.5645$

B.3 ‘Fractor-2’ Bode plot, $\alpha = 0.7425$

B.4 ‘Fractor-3’ Bode plot, $\alpha = 0.7900$

B.5 ‘Fractor-4’ Bode plot, $\alpha = 0.9440$

B.6 ‘Fractor-5’ Bode plot, $\alpha = 0.8630$

B.7 ‘Fractor-6’ Bode plot, $\alpha = 0.9190$

B.8 ‘Fractor-7’ Bode plot, $\alpha = 0.8680$

B.9 ‘Fractor-8’ Bode plot, $\alpha = 0.7460$

B.10 ‘Fractor-9’ Bode plot, $\alpha = 0.3850$

B.11 ‘Fractor-10’ Bode plot, $\alpha = 0.6495$

C.1 ‘Fractor-0’ Bode plot, $\alpha = 0.966$

C.2 ‘Fractor-1’ Bode plot, $\alpha = 0.715$

C.3 ‘Fractor-4’ Bode plot, $\alpha = 0.99$

C.4 ‘Fractor-5’ Bode plot, $\alpha = 0.905$
C.5 ‘Fractor-6’ Bode plot, $\alpha = 0.922$. .......................... 116
C.6 ‘Fractor-8’ Bode plot, $\alpha = 0.7705$. ......................... 116
C.7 ‘Fractor-9’ Bode plot, $\alpha = 0.832$. .......................... 117
C.8 ‘Fractor-10’ Bode plot, $\alpha = 0.772$. .......................... 117
D.1 Analog FO model board test I/O waveform. ............... 118
D.2 Analog FO model board - enlarged schematic. ............. 120
E.1 Analog FO-PII controller board - detailed schematic. .......... 122
F.1 Analog divider 1 - courtesy of National Semiconductor. ......... 123
F.2 Analog divider 2 - courtesy of EDN, Europe. ................. 123
K.1 Simulink$^\text{®}$ model for plasma position estimation. ......... 134
### Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IO</td>
<td>Integer Order</td>
</tr>
<tr>
<td>FO</td>
<td>Fractional Order</td>
</tr>
<tr>
<td>H-I-L</td>
<td>Hardware-In-The-Loop</td>
</tr>
<tr>
<td>UAS</td>
<td>Universal Adaptive Stabilization</td>
</tr>
<tr>
<td>FO-UAS</td>
<td>Fractional Order Universal Adaptive Stabilization</td>
</tr>
<tr>
<td>IO-UAS</td>
<td>Integer Order Universal Adaptive Stabilization</td>
</tr>
<tr>
<td>F-MIGO</td>
<td>Fractional $M_s$ Constrained Integral Gain Optimization</td>
</tr>
<tr>
<td>FO-PI</td>
<td>Fractional Order Proportional, Integral</td>
</tr>
<tr>
<td>FO-PID</td>
<td>Fractional Order Proportional, Integral and Derivative</td>
</tr>
<tr>
<td>FO-PII</td>
<td>Fractional Order Proportional, with Dual Integral Channels</td>
</tr>
<tr>
<td>AFO-PI</td>
<td>Approximated Fractional Order Proportional, Integral</td>
</tr>
<tr>
<td>PID</td>
<td>Proportional, Integral and Derivative</td>
</tr>
<tr>
<td>FOPDT</td>
<td>First Order Plus Delay Time</td>
</tr>
<tr>
<td>ZN-PID</td>
<td>Ziegler Nichols Tuned PID Controller</td>
</tr>
<tr>
<td>A/D</td>
<td>Analog-to-Digital</td>
</tr>
<tr>
<td>D/A</td>
<td>Digital-to-Analog</td>
</tr>
<tr>
<td>STOR</td>
<td>Saskatchewan Torus</td>
</tr>
<tr>
<td>BT</td>
<td>Toroidal Field Bank</td>
</tr>
<tr>
<td>OH</td>
<td>Ohmic Heating Bank</td>
</tr>
<tr>
<td>VE</td>
<td>Vertical Equilibrium Bank</td>
</tr>
<tr>
<td>HC</td>
<td>Horizontal Compensation Bank</td>
</tr>
<tr>
<td>VC</td>
<td>Vertical Compensation Bank</td>
</tr>
<tr>
<td>$V_c$</td>
<td>Vertical Field Compensation Voltage</td>
</tr>
<tr>
<td>$V_i$</td>
<td>Magnetic Pickup Coil Voltage - Inner Position</td>
</tr>
<tr>
<td>$V_o$</td>
<td>Magnetic Pickup Coil Voltage - Outer Position</td>
</tr>
<tr>
<td>$V_u$</td>
<td>Magnetic Pickup Coil Voltage - Upper Position</td>
</tr>
<tr>
<td>$V_l$</td>
<td>Magnetic Pickup Coil Voltage - Lower Position</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

1.1 Motivation

The process of questioning the reasons for the occurrence of natural phenomenon can be attributed as the cornerstone of all science in general. Scholars from myriad different fields dedicate their lifetimes to have a better understanding of our world. Notable contributions in the world of mathematics came from great people like Sir Isaac Newton, G.W. Leibniz, Leonhard Euler, to name a few. They laid down the basic concepts for calculus, a mathematical tool that forever changed our outlook in several aspects. It is only when one understands that all modern science depends on work that is centuries old, that one feels truly humbled.

Students of engineering, however, get exposed early on to the fact that understanding concepts are as important as putting them into use in real life. Getting things to work for real is a challenge in itself, since nature does not always like to play by the rigid rules of mathematics. The modern day devices, such as cell phones and computers, perform a variety of mathematical calculations like wavelet transforms, fourier transforms, etc. Seeing math in action is an infinitely fulfilling experience, almost indistinguishable from magic.

While most modern day devices and commercial equipment make tremendous use of classical calculus and the integer order school of thought, there exists another close parallel, i.e., fractional calculus. Fractional calculus is as old as classical calculus itself, but it has not gained as much rapport (comparatively) because one runs into the difficulty of not being able to comprehend the physical meanings of a fractional derivative or an integral.

The basic motivation for this work is to be able to see and experience the effects of the use of fractional calculus on real-life experiments. Resistance to any arcane science is a characteristic of the human mind, I am, however, overjoyed by the fact that my time spent
performing the experiments with fractional calculus has transformed me into a true believer of mathematics.

1.2 Literature Review

Oldham and Spanier, in their book [1], provide a good grounding in the basic concepts required for fractional calculus. Podlubny’s book [2] provides a lot of useful definitions and results regarding fractional calculus. It also highlights certain applications and methods of calculation. The Gamma, Beta, and Mittag-Leffler functions form the base for fractional calculus. The Mittag-Leffler function turns out to be a generalization of the widely used exponential function, $e^x$, and the scope of fractional calculus becomes apparent. Additionally, Samko [3], Miller and Ross [4] provide a good introduction to fractional calculus.

The work of Ziegler and Nichols [5], provides a usable controller which is ready to be tweaked. The ease of use of their control strategy and tuning rules however makes their work a solid foundation of control science, so solid that almost every process imaginable and in use in the industry still has a proportional, integral and derivative (PID) controller which is tuned based on their work. The authors Aström et al. [6, 7] delve into further details about PID controllers, their tuning and design, in the face of certain specifications and constraints.

The physical meaning of fractional order derivatives and integrals is hard to imagine but there are analytical results to help with the calculations. However when implementing the same in the simulation or a hardware-in-the-loop (H-I-L) setting, the use of Oustaloup’s recursive approximation [8, 9] is indispensable. The beginning of this work talks about nonidentifier-based universal adaptive stabilisation, Ilchmann’s work [10] provides the details about the same. The authors Li et al. [11] deal with the FO-UAS. Their results have been applied to some of the experiments in this work and the conclusions obtained are very good.

Fractional order PID (FO-PID) control is a useful control strategy since it provides five parameters to be tuned as opposed to the three available in ordinary PID control. The work of the authors Chen and Bhaskaran [12] is extremely useful for tuning fractional
order proportional integral (FO-PI) controllers for a wide class of plants. The methodology used therein results in a readily usable and practical approach for tuning FO-PI controllers. Auto-tuning strategies for simple FO-PI controllers, and also tuning rules for other types of controllers like FO-[PI], and their uses are presented by Wang et al. [13]. Varying time delay systems are a challenge to tune, which is dealt by Bhambhani et al. in their work [14].

This work also deals with a simple frequency domain fitting approach for obtaining fractional order models of systems. The authors Djamah et al. [15] and Das [16] provide an insight into fractional order system modeling methodologies. As the title indicates, Das [16] has a more practical and direct approach.

This work uses special analog devices called ‘Fractors.’ Bohannan’s work [17] provides details about the implementation and the characteristics of these devices. Petras et al. provide an insight into other methods for analog realization of fractional controllers in their work [18].

Wesson, White, and Kadomstev [19–21] provide general details about the tokamak, the fundamentals of plasma physics and some theory about plasma confinement in toroidal magnetic confinement devices. A good definitive reference on the subject is the book by Artsimovich [22], where as Ariola and Pironti’s work [23] deals with the controls aspect of the plasma confinement problem. This work heavily relies on Wolfe’s and Emami’s works [24, 25] for the operational parameters of the STOR-1M tokamak, and details about the previous plasma position control system, respectively. In order to understand the magnetic disturbances and magnetohydrodynamical (MHD) instabilities that affect the plasma column, consulting the book [26] by Bateman is recommended.

1.3 Contribution and Organization

As mentioned earlier, this work concentrates on the application of fractional calculus to real-life systems. One of the contributions of this work is to test the performance of the FO-UAS control strategy on a direct current (DC) motor and to observe the performance benefits obtained. Chapter 2 provides the mathematical details and the experimental
results. The ability to track sinusoidal, sawtooth and square wave signals with almost negligible steady-state error and near instantaneous rise times is a good finding.

The next contribution comes in the form of the prototype FO-PID controller board used for the analog control experiments on the coupled tank system. Chapter 3 provides an indepth coverage of the experimental setup and the results. This chapter also definitively shows the advantages purely analog control has over the digital versions. Finding an equivalent FO-PI controller by using two or more of the available ‘Fractors’ to approximate the optimal FO-PI controller designed according to Chen and Bhaskaran’s tuning rules [12], is the main mathematical contribution of this chapter.

Chapter 4 deals with fractional order modeling and control of the heat flow plant. This chapter presents a frequency domain fitting procedure to model a plant using fractional calculus. Experimental results verifying the same are presented in this chapter.

Chapter 5 deals with the real-life hardware implementation of the fractional order modeling methodology presented in Chapter 4. The plant modeled is the fan and plate system. The details of the fractional order (FO) model board and the fractional order proportional, dual integral (FO-PII) controller board are presented in this chapter. The results of a FO controller controlling a FO plant (in the purely analog hardware setting) have also been presented.

This entire work is presented in such a fashion that the experiments performed and the results obtained, build up towards controlling the plant described in Chapter 6. The plant dealt with here is the STOR-1M tokamak. The tuning rules used in the previous chapters, the analog hardware setup used, etc., are all used in this chapter to verify that the FO-PI controller proposed for plasma position control of the STOR-1M tokamak achieves the requirements in a real-life analog hardware emulation setting. Estimation of plasma position prior to applying any control theory however is the most interesting part of this chapter since it does not use any of the complex mathematics normally used by plasma physicists [25], but relies on simple results from basic physics and the geometry of the device to obtain a good estimate of the plasma position.
Chapter 7 forms the conclusion of this work, it summarizes the results obtained and presents some possibilities for future work.

The appendices contain valuable information pertaining to this work. Appendix A provides certain basic definitions used in fractional calculus. Appendices B and C present the Bode plots of the ‘Fractors’ taken at different dates. Details about the fractional order model board are included in Appendix D. Details about the fractional order controller board are presented in Appendix E. Appendix F provides schematics for implementing analog signal division. Appendix G presents operational details about the STOR-1M tokamak. The code used for various experiments in this work are included in Appendices H, I, J, and K.
Chapter 2

Fractional Order Calculus in Controls: A First Glance

The purpose of this chapter is to introduce basic elements of fractional calculus to the reader. The first section establishes basic ideas about fractional order PID control. The second section is dedicated to a more sophisticated control strategy called the FO-UAS. Various real-time hardware-in-the-loop experiments have been performed on a DC motor for comparing the integer order (IO) and fractional order (FO) approaches of universal adaptive stabilization (UAS). Data obtained from the experiments presented in this chapter shows that fractional calculus is indeed useful in real life.

2.1 Fractional Order PID Controllers: An Overview

A simple traditional PID controller can be written in the following form:

\[ u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{d}{dt} (e(t)) , \]  

(2.1)

where \( u(t) \) is the output of the PID controller, \( e(t) \) is the error and \( K_p, K_i, K_d \) are the proportional, integral and the derivative gains, respectively. There are several different methods for tuning PID controllers that have been explored by researchers over the years. The work of Aström et al. [6, 7] can be referred to as an example. The tuning rules by Ziegler and Nichols [5], however, are the most prevalently used since its use is simple. The relay auto-tuning procedure is convenient and it provides for a good initial controller to start out with.

Replacing the integer order derivatives and integrals in eq. (2.1) by integral and derivative operators of arbitrary real order provides a fractional order PID or FO-PID controller.
The mathematical representation of such a controller is as follows:

\[
    u_f(t) = K_{pf} e(t) + K_{if} \left( aD_t^\lambda e(t) \right) + K_{df} \left( aD_t^\mu e(t) \right),
\]

(2.2)

where \( u_f(t) \) is the output of the FO-PID controller, \( e(t) \) is the error. \( \lambda, \mu \in \mathbb{R} \) are the orders of the integral and differential operators, respectively. \( K_{pf}, K_{if}, K_{df} \) are the proportional, integral, and the derivative gains for the FO-PID controller. The operator of the form \( aD_t^\rho \) shown here is the Grünwald-Letnikov (GL) form of the fractional differintegral operator where \( a, t \) are the terminals. Preliminary information about such operators can be found in Appendix A. Podlubny’s book [2] must be consulted if one desires delving into the details. As seen from eq. (2.2), one can tune five separate parameters of the FO-PID controller, thus providing more finesse.

### 2.2 Fractional Order Universal Adaptive Stabilization (FO-UAS)

All real-life systems are not necessarily simple or linear. There are many complex systems with a high degree of uncertainty. Adaptive control is interesting from the point of view of controlling such systems. UAS is a special case of adaptive control. Li et al. [11] study the UAS for systems with state space representations. The work presented in the following sections is the application of the results from Li et al. [11] to real-life systems. Mukhopadhyay et al. [27] provide an abridged version of the experimental details and results shown here.

#### 2.2.1 Mathematical Preliminaries

This subsection provides some elementary information about the mathematics necessary to describe the FO-UAS system. Riemann-Liouville and Caputo fractional operators, the Mittag-Leffler function, and its Laplace transform are discussed mainly since they are widely used. The Riemann-Liouville and Caputo fractional operators are the primary mathematical tools used for working out the mathematics related to FO-UAS. Some details about the mathematical results used in fractional calculus can be found in Appendix A. For a thor-
ough understanding, however, Podlubny’s book [2] is recommended. The uniform formula of a fractional integral with \( \alpha \in (0, 1) \) is defined as follows:

\[
aD_t^{-\alpha} f(t) = \frac{1}{\Gamma(\alpha)} \int_a^t \frac{f(\tau)}{(t-\tau)^{1-\alpha}} d\tau,
\]

where \( f(t) \) is an arbitrary integrable function, \( aD_t^{-\alpha} \) is the \( \alpha \)th fractional integrator on \([a, t]\), and \( \Gamma(\cdot) \) denotes the Gamma function. For an arbitrary real number \( p \), the Riemann-Liouville and Caputo fractional operators are defined as follows:

\[
aD_t^p f(t) = \frac{d^{[p]+1}}{dt^{[p]+1}}[aD_t^{-(\lfloor p \rfloor - p + 1)} f(t)],
\]

and

\[
C_aD_t^p f(t) = aD_t^{-(\lfloor p \rfloor - p + 1)}[\frac{d^{[p]+1}}{dt^{[p]+1}} f(t)],
\]

respectively. Here \([p]\) stands for the integer part of \( p \), \( D \) and \( C \) are Riemann-Liouville and Caputo fractional operators, respectively.

The exponential function is frequently used in the solutions of integer order systems. Similarly, a function frequently used in the solutions of fractional order systems is the Mittag-Leffler function. It is defined as follows:

\[
E_\alpha(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(k\alpha + 1)},
\]

where \( \alpha > 0 \). The Mittag-Leffler function in two parameters has the following form:

\[
E_{\alpha,\beta}(z) = \sum_{k=0}^{\infty} \frac{z^k}{\Gamma(k\alpha + \beta)},
\]

where \( \alpha \) and \( \beta \) are arbitrary complex numbers. When \( \alpha > 0 \) and \( \beta = 1 \), \( E_\alpha(z) = E_{\alpha,1}(z) \).
Moreover, the Laplace transform of the Mittag-Leffler function in two parameters has the following form:

\[ L\{t^{\beta-1} E_{\alpha,\beta}(-\lambda t^\alpha)\} = \frac{s^{\alpha-\beta}}{s^\alpha + \lambda}, \quad (\Re(s) > |\lambda|^{1/\alpha}), \quad (2.8) \]

where \( s \) is the variable in Laplace domain, \( \Re(s) \) denotes the real part of \( s \), \( \lambda \in \mathbb{R} \) and \( L\{\cdot\} \) stands for the Laplace transform.

### 2.2.2 Nussbaum Function

**Definition 2.2.1** A piecewise right continuous and locally Lipschitz function \( N(\cdot) : [k', \infty) \rightarrow \mathbb{R} \) is called a Nussbaum function if

\[
\sup_{k > k_0} \frac{1}{k - k_0} \int_{k_0}^{k} N(\tau) d\tau = +\infty,
\]

and

\[
\inf_{k > k_0} \frac{1}{k - k_0} \int_{k_0}^{k} N(\tau) d\tau = -\infty,
\]

for some \( k_0 \in (k', \infty) \), where \( k' \in \mathbb{R} \).

It is easy to see that Definition 2.2.1 holds true for some \( k_0 \in (k', \infty) \) if and only if it is valid for all \( k_0 \in (k', \infty) \) [10]. The Nussbaum function plays a crucial role in UAS [10, 28–33], also Li et al. [11, 34] show that the Mittag-Leffler function \( E_\alpha(-\lambda t^\alpha) \) for \( \lambda > 0 \) and \( \alpha \in (2, 3] \) is a Nussbaum function. Details about the following results can be found from the work by Li et al. [11].

**Lemma 2.2.1** Suppose that \( f(k) \) is bounded for all \( k \) and \( N(k) \) is a Nussbaum function with respect to \( k \). Then \( f(k) + N(k) \) is also a Nussbaum function with respect to \( k \).

**Lemma 2.2.2** \( e^k \cos(k) \) is a Nussbaum function with respect to \( k \).
From Lemmas 2.2.1 and 2.2.2 the following theorem can be proved.

**Theorem 2.2.2** \( E_\alpha(-\lambda^k) \) is a Nussbaum function with respect to \( k \), where \( \alpha \in (2,3] \) and \( \lambda > 0 \).

### 2.2.3 Stability Analysis of the Fractional Order UAS

This section discusses the asymptotic stability of the following two fractional scalar systems by using the method of fractional UAS:

- Fractional dynamics with integer order control strategy,
- Fractional dynamics with fractional control strategy.

The mathematical results presented here are from the work on FO-UAS by Li et al. [11]. Consider the real fractional system as in (2.9) with the control law given by (2.10). In reality, such a model would correspond to the case of a vehicle running on water or any agent moving about in a complex environment. Consider the fractional dynamics

\[
\begin{align*}
\dot{x}(t) &= a_1 x(t) + a_2 \, _0^D_\alpha t \, x(t) + bu(t) \\
y(t) &= cx(t), \\
x(0) &= x_0,
\end{align*}
\]

with the fractional control strategy

\[
\begin{align*}
u(t) &= -N(k(t))y(t) - (cb)^{-1}a_2 \, _0^D_\alpha t \, y(t) \\
k(t) &= y^2(t), \\
k(0) &\in \mathbb{R},
\end{align*}
\]

where \( a_1, a_2, b, c, x_0 \in \mathbb{R} \) are unknown. \( N(k) \) is an arbitrary Nussbaum function with respect to \( k \), and \( \alpha \in (0,1) \). The effect of varying \( a_2, b, \) and \( c \) on the performance of the FO-UAS scheme can be seen in sec. 2.2.4. Note that if \( a_2 = 0 \), then the system described by eq. (2.9) reduces to an integer order system. The corresponding integer order control law would be
obtained by substituting $a_2 = 0$ in eq. (2.10). Applying eq. (2.10) to eq. (2.9) yields the closed-loop system

$$\begin{cases}
\dot{y}(t) = [a_1 - cbN(k(t))]y(t) \\
k(t) = y^2(t), \quad k(0) \in \mathbb{R}.
\end{cases} \tag{2.11}$$

By using Lemma 3.1 [11], it is obvious that $k(t)$ is bounded and $y(t)$ converges to zero as $t \to \infty$.

### 2.2.4 Experimental Setup

Based on the theoretical results in sec. 2.2.3 the experiments have been performed on a DC motor, with both integer order UAS and fractional order UAS control strategies. The fractional order UAS was actually tested on a hardware-in-the-loop setup. Figure 2.1 shows the details of the entire experimental system and fig. 2.2 shows the actual experimental setup.

The diagram shown in fig. 2.1 uses the control strategy as mentioned in eq. (2.10). The plant used is a DC motor. It is interfaced to the Quanser Q4 board through a combined analog-to-digital (AD)/digital-to-analog (DA) converter and power amplifier unit [35]. The saturation bound of the plant input is $\pm5V$. The Nussbaum function used here is the Mittag-Leffler function. It is represented by the ‘s-function’ block in fig. 2.1. The Mittag-Leffler function is coded in ‘C’ as a c-mex ‘*.dll’ file [36], since the existing Simulink® block for implementing custom functions is not yet supported by the real-time workshop (RTW). However, coding it in ‘C’ has a performance benefit since it operates much faster than a normal MATLAB® function call. The blocks to the right of the fractional derivative block are used to change the type of the derivative from Riemann-Liouville to the Caputo form. The error $e(t)$ is used to obtain the function $k(t)$. $k(t)$ is fed to the Mittag-Leffler function to produce the Nussbaum function $N(k(t))$. Note that the parameter $\alpha$ used for the Mittag-Leffler function is different from the fractional order $\alpha$ used in the control loop. All the parameters $a, b, c$, and $a_2$ are initially set to 1. The effect of varying these parameters are discussed in sec. 2.2.5. A signal generator block is used so that the reference can be changed
as required. The performance of the fractional order UAS can thus be studied for different reference signals.

### 2.2.5 Results I: FO-UAS vs. IO-UAS

The experiments are first conducted with the integer order UAS control law. They are then repeated with the fractional order UAS and the results are compared. As can be seen from fig. 2.3, the integer order UAS has a finite steady-state error, which is quite big. Also it has a slow response time. From fig. 2.4 it is clear that the integer order UAS does not track a sinusoidal reference well. It has a slow response, and it fails to track the reference midway, after which the error is around 0.1275.

It can be seen from fig. 2.5 that for $\alpha = 0.6$ the fractional order UAS performs really

---

**Fig. 2.1: H-I-L FO-UAS control scheme.**
well. It has a fast response and the steady-state error is of the order of 0.02. The control effort is highest when the squarewave switches. This behavior is obvious since the error is the highest at that point.

From figs. 2.5, 2.6, 2.7, and 2.8 it can be seen that as $\alpha$ decreases, the steady-state error increases. As $\alpha$ increases a lot of noise is introduced into the system and the settling time increases.

From fig. 2.9 it is seen that the fractional order UAS can track a sinewave reference. The error is in the range of 0.02 to 0.04. Figure 2.10 shows that the fractional order UAS can track a sawtooth reference. The steady-state error is around 0.005.

### 2.2.6 Results II: Effect of Different Nussbaum Functions on FO-UAS

The results shown so far are obtained using the Mittag-Leffler function as a Nussbaum
Fig. 2.3: Integer order UAS - squarewave tracking.

Fig. 2.4: Integer order UAS - sinewave tracking.
Fig. 2.5: Fractional order UAS - squarewave tracking, $\alpha = 0.6$.

Fig. 2.6: Fractional order UAS - squarewave tracking, $\alpha = 0.1$. 
Fig. 2.7: Fractional order UAS - squarewave tracking, $\alpha = 0.3$.

Fig. 2.8: Fractional order UAS - squarewave tracking, $\alpha = 0.75$. 
Fig. 2.9: Fractional order UAS - sinewave tracking, $\alpha = 0.5$.

Fig. 2.10: Fractional order UAS - sawtooth wave tracking, $\alpha = 0.5$. 
function. This section examines the effects of using different functions as a Nussbaum function. From fig. 2.11 it is clear that using $k^2 \cos(k)$ as the Nussbaum function has unpleasant effects. Noise is clearly visible in the output and the control effort remains beyond saturation for a long time. Extreme control effort saturation is not observed when the functions $k \cos(\sqrt{|k|})$ and $\cos\left(\frac{\pi k}{2}\right)e^{k^2}$ are used as Nussbaum functions. But they produce a lot of output noise as shown in figs. 2.12 and 2.13, respectively. A similar effect is observed when these cases are repeated with a sinusoidal reference. When the above mentioned Nussbaum functions are used in the fractional order UAS to track a sinusoidal reference, a lot of noise is introduced initially and the error is a lot higher than observed in the previous cases.

2.2.7 Results III: Other Experiments

Various experiments have been performed for different values of the parameters. For certain values of $b, c$, and $a_2$ the performance is not affected much, however, other values have bad effects. Different functions like $\dot{k}(t) = |y(t)|$, $y^4(t)$ or $\lambda y^\gamma(t), \gamma \in \mathbb{R},$ and $\lambda$ is a constant, have been tried. Increasing the power of $y$ or certain values for $\lambda$ affect the
Fig. 2.12: Fractional order UAS - Nussbaum function \( = k \cos(\sqrt{|k|}) \).

Fig. 2.13: Fractional order UAS - Nussbaum function \( = \cos(\pi k^2/2) e^{k^2} \).
performance adversely. Some of the other cases tried out are operational, but they do not give the best performance. A change in sign of the terms on the right hand side of the control input in eq. (2.10) does not have bad effects, but it increases the steady-state error. The same experiments have been tried out with a high gear and low gear configuration for the motors. It is observed that the gear configuration does not affect the performance of the fractional order UAS.

From fig. 2.14 it is observed that changing the second part of eq. (2.10) to $\dot{k}(t) = |y(t)|$ has no drastic effect, however, the steady-state error increases by a small amount and so does the control effort. However, if it is changed to $\dot{k}(t) = y^4(t)$, the output becomes very noisy and a large steady-state error is introduced as shown in fig. 2.15. Also the control effort is always beyond the saturation bounds, which is not good. If the sign of the control effort in eq. (2.10) is changed, then the system is still operational due to the Nussbaum functions ability of searching, as observed in fig. 2.16. However, it can be clearly seen that the steady-state error increases and so does the control effort.

![Fig. 2.14: $\dot{k}(t) = |y(t)|$.](image)
Fig. 2.15: $\dot{k}(t) = y^4(t)$.

Fig. 2.16: $u(t) = +N(k(t)y(t) + (cb)^{-1}a_2 \frac{D^\alpha}{\Delta^\alpha} y(t))$. 
2.2.8 Results IV: Effect of Varying $b, c$, and $a_2$

In the previous experiments the values of the variables $b, c$, and $a_2$ are all taken equal to one. Thus it is obvious that the fractional order UAS works when the co-efficient $\frac{a_2}{c b}$ of the second term in eq. (2.10) equals one. The following experimental results explore the performance of the fractional order UAS when the value of $\frac{a_2}{c b}$ is changed.

From fig. 2.17 it can be seen that the fractional order UAS works well even when $\frac{a_2}{c b} = 0.5$. Also when the value of $\frac{a_2}{c b} = 2$, as seen in fig. 2.18, it is found that the performance of the fractional order UAS does not deteriorate. However, when $\frac{a_2}{c b} = -0.5$, the performance is a bit slower, i.e. the settling time is a bit higher as seen in fig. 2.19. Also in this case, the error is a a little larger in the beginning, but it is seen to reduce with time.

The system, however, does not follow the squarewave as well as the previous cases because each cycle of the squarewave has a certain small amount of disturbance in the beginning.

![Fractional order control law - squarewave tracking, $\frac{a_2}{c b} = 0.5$, $u = 0.5$](image1)

![Tracking error](image2)

![Control effort](image3)

Fig. 2.17: Squarewave tracking - $\frac{a_2}{c b} = 0.5$. 

2.2.9 Results V: Very Low-Speed Regulation and External Noise

Experiments were performed in which the reference signals were set as low as 0.1 rad/s and 0.05 rad/s. Another experiment was performed when the reference was set to 1 rad/s, but additional band-limited white noise was introduced into the loop. The results are as follows.

In fig. 2.20, the fractional order UAS control scheme works well even for a low reference of 0.1 rad/s. Initially the error, while tracking the negative half cycle of the square wave, is large but the error decreases with time. Also the control effort never goes beyond the saturation limit and no erratic behaviour or noisy output is seen. Figure 2.21 shows the case when the reference is set to 0.05 rad/s. Here the fractional UAS starts out with an error larger than the previous case, but over time the error decreases. The output is a bit noisier than the previous case which may be attributed to the fact that frictional resistance and sensor noise play a vital role when tracking such low speeds. However, it should be noted that the fractional UAS cannot track any reference having a magnitude lower than 0.05 rad/s for this particular motor. Figure 2.22 shows the case when additional band-limited white noise is injected into the loop. Even with the deliberately added noise, the fractional UAS does a good job. For the most part, the error and the output is similar to the results obtained without noise and the minimum steady-state error obtained is of the order of 0.01. However when disrupted by noise, some spikes are seen in the error which quickly settle down. The maximum steady-state error with external added noise is found to be around 0.275 which decreases promptly to 0.06.

2.3 Observations

The results of the experiments conducted can be summarized as follows. The fractional order UAS definitely works better than the integer order case. The experiments have been performed on a DC motor hence, it is not essential to have a model/plant with fractional dynamics for conducting experiments. The performance deteriorates when the fractional order $\alpha$ is made too less or too large. When properly tuned it tracks a squarewave with ease and is also found to track sawtooth and sinusoidal references. The performance is
largely dependent on the selection of a proper Nussbaum function and it is seen that a bad choice may adversely effect the results. The performance changes slightly for different values of the ratio $\frac{a_2}{cb}$. However, the fractional order UAS still works when different values are used for $\frac{a_2}{cb}$. Using different functions and different gains on the right hand side of the eq. $\dot{k}(t) = y^2(t)$, has a profound impact on the performance. Changing the sign of the terms on the right hand side of eq. (2.10) generally changes the steady-state error by a sizeable amount. The fractional order UAS performs well even when tracking a reference of very small magnitude or when additional noise is induced deliberately into the loop.

2.4 Conclusion and Future Work

When properly tuned, with the right parameters, the fractional order UAS works better than the integer order case. The experiments have been performed on a DC motor, hence its application is not only restricted to models with fractional dynamics. The response of the fractional UAS is fast, and it works remarkably well even when tracking sinusoidal and
Fig. 2.19: Squarewave tracking $\frac{a_2}{cb} = -0.5$.

Fig. 2.20: Squarewave tracking $\pm 0.1 \text{rad/s}$. 
Fig. 2.21: Squarewave tracking ±0.05 rad/s.

Fig. 2.22: Squarewave tracking ±1 rad/s, external noise added.
sawtooth references.

Finding the optimal values for the parameters $a, b, c, a_2$, and $\alpha$ is a definite direction for future work. Also, the reason as to why the output is different for different Nussbaum functions, and why the fractional order UAS works best when a Mittag-Leffler function is used as a Nussbaum function, need to be researched. An explanation needs to be found for the fractional order UAS's ability to track sinusoidal references.
Chapter 3
Verification: Validation Platform 1

This chapter is dedicated to purely analog fractional order proportional integral (PI) control of the coupled tank system. A custom circuit board has been built to perform analog PID/FOPID control. Specialized circuit elements called ‘Fractors’ [17] are used to perform fractional order integration/differentiation of the error signals. The design of the board, subsequent experimentation, and its comparison to similar digital control techniques have been presented. The benefits of purely analog control when dealing with nonlinearities are also discussed. Further, a method of approximating a particular fractional order $\alpha^*$, with any number of practically available orders $\alpha_1, \alpha_2, ..., \alpha_n$, has been presented. This chapter forms a base for the other analog fractional order proportional integral (FO-PI) control experiments performed as a part of this work.

3.1 The Coupled Tank System

3.1.1 Overview

Figure 3.1 shows the plant being used [37]. It consists of a tank which can be partitioned into two separate tanks by the use of a baffle plate. The baffle plate may be raised or lowered in order to adjust the coupling of the two tanks. Water is fed from pipes at the top. The water pumps are located in a reservoir below the tanks. The pumps take analog inputs. They have internal circuitry to generate the pulse width modulated signals required to drive the motors. Outlets at the bottom of each tank drain the water. The rate of drainage can be adjusted using clamps. Capacitive sensors monitor the water level. For our experiments, tank no. 2 is the actuating tank and the water level of tank no. 1 is controlled. Figure 3.2 shows the schematic of the system.
Figure 3.3 shows the relation between the voltage output by the water level sensors and the actual height of the water column. The lower range of the sensors is extremely nonlinear, as seen from the plots. One of the objectives of this work is to test how different control strategies work in the face of such nonlinearities.
3.1.2 System Identification

A simple step response system identification method is used to model the coupled tank as a first order plus delay time (FOPDT) system. The model of the plant obtained is as follows:

$$G(s) = \frac{K}{Ts + 1} e^{-Ls} = \frac{0.8148}{3.634s + 1} e^{-0.999s}. \quad (3.1)$$

The relative dead time is calculated as $\tau = \frac{L}{L + T} = 0.2156$. Also from eq. (3.1) it can be inferred that the system takes quite a long time to react to stimulus.

3.2 Experimental Setup

From the FOPDT model in eq. (3.1), the parameters of the Ziegler Nichols (ZN) tuned PID controller are obtained as shown in table 3.1. These values will be used to carry out the
Table 3.1: Parameters of the ZN–PID controller for the coupled tank.

<table>
<thead>
<tr>
<th>$K_p$</th>
<th>$K_i$</th>
<th>$K_d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.26878</td>
<td>0.5</td>
<td>0.4995</td>
</tr>
</tbody>
</table>

ZN-PID controller experiments in order to compare it against the fractional order variants. The ZN step response tuning rules can be found in any standard PID tuning reference.

3.2.1 Hardware-in-the-Loop Implementation of a FO-PI Control System

Figure 3.4 shows the Simulink model used for digital ZN-PID and digital FO-PI control of the coupled tank system. This is a H-I-L system that is interfaced to the coupled tank through a Quanser Q4 board [38]. The WinCon toolbox [39] is used to send signals to and capture data from the plant. The Quanser Q4 board converts the digital signals from the computer into analog and feeds it to the plant. It also samples the incoming analog signals and converts them back to digital so they can be read using the computer. A sampling time of 1 millisecond (ms) is used for all the digital control experiments. Digital control implies a setup having a controller implemented in Simulink®, and the control effort, plant feedback signals pass through A/D, D/A converters, respectively, on the Quanser board.

3.2.2 Analog Hardware Implementation

Figure 3.5 shows the setup for analog PID/FO-PI control. Everything including the

![Simulink model for digital FO-PID control](image-url)
reference signals and the controller is replaced with analog hardware. The Quanser board and the computer is used only for the purpose of data monitoring.

The schematic shown in fig. 3.6 is used to create the board shown in fig. 3.7. The board has three multi-turn 20KΩ potentiometers which allows the user to tweak the proportional, integral, and derivative gains ($K_p$, $K_i$, and $K_d$). The board has circuitry that computes the error signal based on the reference and the plant outputs. A regular analog PID controller [40] uses capacitors whereas our board shown in fig. 3.6 uses ‘Fractors.’ This board can be used for analog PID control or analog FO-PID control. It also has channel selector switches which enable one to switch off the proportional (P), integral (I), and derivative (D) channels at will. Two ‘TLV2774i’ rail-to-rail output, single supply, quad op-amps [41], are used for performing all the mathematical operations. Capacitors are placed in series with the ‘Fractors.’ This enables one to perform analog PID control by shorting out the ‘Fractors’ using the switches across them. The disjoint portion of the schematic showing the part labeled ‘IC7805’ is the power supply unit for the board. The schematic shown in fig. 3.6 belongs to the prototype controller board. The prototype board has a derivative (D) channel. The updated and final version of the analog FO-PID controller board has two integrating (I) channels, as opposed to one on the prototype board. A bigger schematic and

![Overall analog control layout](image)
other details about the final version of the controller board are provided in Appendix E.

### 3.3 Determining The Fractional Order $\alpha$ of a ‘Fractor’

Figure 3.8 shows the Bode plot of the ‘Fractor’ with $\alpha = 0.919$. It is used to perform some of the the analog FO-PI control experiments. The mathematical relations used to calculate the order of the fractional integrator from the Bode plots are as follows:

\[
\text{Magnitude} = -\alpha \times 20\text{dB/Decade}, \quad (3.2)
\]
\[
\text{Phase} = -\alpha \times \frac{\pi}{2}. \quad (3.3)
\]

Comparing the values of $\alpha$ obtained by using eq. (3.2) and eq. (3.3) shows that they match closely for the flat phase portion from $10^3$ to $10^4\text{rad/sec}$ as seen in fig. 3.8. The value of the fractional order $\alpha$ obtained from the magnitude plot and subsequent verification from the phase plot can be easily seen in fig. 3.9. It must be noted that the Bode plot shown in fig. 3.9 is for one of the ‘Fractors’ used for the analog approximated fractional order proportional, integral (AFO-PI) control experiments. Also, fig. 3.10 shows a zoomed in view of the Bode plot of the ‘Fractor’ with $\alpha = 0.385$. This is the second ‘Fractor’ used.

Fig. 3.6: Analog FO-PID controller board - schematic.
Fig. 3.7: Analog FO-PID controller board - prototype.

Fig. 3.8: Bode plot of a ‘Fractor’ with $\alpha = 0.9$.

for the AFO-PI control experiments. It can be seen clearly from this figure that the phase angle remains fairly constant at the value obtained using eq. (3.2) and eq. (3.3). Note that this particular ‘Fractor’ has the smallest bandwidth among all the other ones used to perform the experiments.
For Fractor with $\alpha = 0.868$

Fig. 3.9: Bode plot of a ‘Fractor’ with $\alpha = 0.868$.

Fig. 3.10: Bode plot of a ‘Fractor’ with $\alpha = 0.385$. 
3.4 Designing an Optimal FO-PI Controller

The procedure to obtain an optimal FO-PI controller is detailed in Chen and Bhaskaran’s work [12]. The gains for the optimal FO-PI controller are presented in table 3.2. Their tuning rules are as follows:

\[ K^* = \frac{1}{K_p} \left( \frac{0.2978}{\tau + 0.000307} \right), \]
\[ T_i^* = T \left( \frac{0.8578}{\tau^2 - 3.402\tau + 2.405} \right), \]

where \( \alpha^*, K^*, \) and \( T_i^* \) form the parameters of the optimal FO-PI controller. To know more about the process of selecting \( \alpha^* \), please consult Chen and Bhaskaran’s work [12].

3.5 Approximation of a FO-PI Controller Using Practically Available Orders

The ‘Frac tors’ available may not always have the same order as obtained for the designed optimal FO-PI controller. Hence, it is required to approximate the optimal FO-PI controller by a different FO-PI controller such that the new controller is as good as the original one. The obtained optimal FO-PI controller can be represented as follows:

\[ C_o(s) = K_{p_o} + \frac{K_{i_o}}{s^{\alpha^*}}. \]

Say \( \alpha_1, ..., \alpha_m \), represent the fractional orders of all the available ‘Frac tors’ subject to the following conditions:

\[ m \geq 2, \ \alpha_i \neq \alpha^* \ \forall \ i \in [1, 2, ..., m]. \]

Table 3.2: Parameters of the optimal FO-PI controller for the coupled tank.

<table>
<thead>
<tr>
<th>( \alpha^* )</th>
<th>( K^* )</th>
<th>( K_i^* )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9</td>
<td>1.6926</td>
<td>0.5511</td>
</tr>
</tbody>
</table>
Let:

\[ C_i(s) = K_P + \frac{1}{s^{\alpha_i}} \]  \hspace{1cm} (3.8)

now the approximated FO-PI controller \( \tilde{C}_o(s) \) can be written as follows:

\[ \tilde{C}_o(s) = \sum_{i=1}^{m} K_I C_i(s). \]  \hspace{1cm} (3.9)

A fit in the frequency domain is thus obtained by minimizing the following index:

\[ J = \int_0^\infty \left| \tilde{C}_o(j\omega) - C_o(j\omega) \right|^2 d\omega, \]  \hspace{1cm} (3.10)

\[ J \approx \sum_{k=0}^{N_\omega} \left| \tilde{C}_o(j\omega_k) - C_o(j\omega_k) \right|^2, \]  \hspace{1cm} (3.11)

\[ \omega_k = k(\Delta \omega) + \omega_L, \]

\[ \Delta \omega = \frac{\omega_H - \omega_L}{N_\omega}. \]

The ‘fminsearch’ function in MATLAB® is used to find a good approximation to the optimal FO-PI controller using the values of \( \alpha \) practically available. The values for which the index ‘\( J \)’ is minimized is selected. The parameter of the AFO-PI controller obtained are shown in table 3.3.

Figure 3.11 shows that the approximation of the optimal FO-PI controller using the available ‘Fractors’ matches the original controller in the frequency domain. To perform the AFO-PI control experiments the ‘Fractor’ in the D channel of the board has been moved over to a different location so it acts like another fractional integrator, in addition to the existing I channel on the board. The results presented for analog FO-PI control have been

Table 3.3: Parameters of the AFO-PI controller for the coupled tank.

<table>
<thead>
<tr>
<th>Fractor No.</th>
<th>( \alpha_i )</th>
<th>( K_{P_i} )</th>
<th>( K_{I_i} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.8680</td>
<td>1.1730</td>
<td>0.3065</td>
</tr>
<tr>
<td>2</td>
<td>0.3850</td>
<td>0.3065</td>
<td>0.4467</td>
</tr>
</tbody>
</table>
carried out with a ‘Fractor’ having $\alpha = 0.9$.

### 3.6 Results

From fig. 3.12 it is clear that digital PID performs the worst followed by the digital FO-PI controller. It may occur to readers that the oscillations in the output may be due to integrator wind-up. However, that is clearly not the case, since the same gains work perfectly well in the analog case as seen from the plots. Also, the digital experiments have been carried out with anti-windup mechanisms only to find no appreciable difference in performance. From the plots it is evident that analog PID control does a good job, however, it has a big steady-state error. The optimal FO-PI controller does a very good job, but it is a bit slower compared to the analog AFO-PI controller. The approximated controller converges quickly to the reference with infinitesimally small steady-state error.
This fact is amply clarified from the comparison of the integral of time and absolute error (ITAE) shown in fig. 3.13. Figure 3.14 compares the performance of the controllers when they track a very low reference. In this zone the sensor is extremely nonlinear. From the results it can be seen that the digital PID and FO-PI controllers fail completely in this case. The analog PID controller is severely affected by the nonlinearities and it completely loses track of the reference and starts tracking at a much higher level. However, the analog FO-PI and the analog AFO-PI controllers do a better job. The analog FO-PI controller shows a faster response, however, it has a big steady-state error. The approximated analog FO-PI controller has a slower response, but it has a lower steady-state error. From fig. 3.15 it is clear that the AFO-PI controller does a better job compared to the others in the presence of nonlinearities.
3.7 Conclusion

It must be noted the term “digital controller” for this work implies a controller implemented on a computer as a part of a hardware-in-the-loop system having a A/D and D/A converter in the loop. Dedicated digital controllers implemented on microcontrollers with advanced algorithms may be similar or better when compared to analog variants. The findings of this work cannot be generalized to all digital controllers.

3.7.1 Integer Order Control vs. Fractional Order Control

It is evident from figs. 3.13 and 3.15, that the fractional order control strategies do a better job as compared to their analog or digital counterparts. The reason for the analog AFO-PI controller performing better than the others may be attributed to the fact that
it has two fractional order integrators. In the integer order math world two fractional integrators would represent a large number of terms, thus enabling the AFO-PI controller to react to very slight differences in error in a multitude of different ways.

3.7.2 Analog Implementation vs. Digital Implementation

Based on the plots obtained with a digital control strategy one may think that there might be a case of integrator windup. However, the same gains and parameters work perfectly for the analog setting. Digital control tends to converge slower to the steady state as compared to its analog counter parts, which explains the reason for the comparatively low control outputs for the analog controllers in fig. 3.12. Analog ZN-PID has a short rise time, but, it slows down later with a big steady-state error. Both the analog FO-PI control strategies perform well. The optimal controller with fractional $M_s$ constrained integral
Fig. 3.15: ITAE comparison - 2.

Gain optimization (F-MIGO) design is faster initially, however, it has a small steady-state error. The analog AFO-PI controller with approximated terms and two ‘Fractors’ has a slightly slower rise time, but it has virtually no steady-state error. When dealing with nonlinearities, the analog AFO-PI controller does the best job among the others although it has a substantial steady-state error. The others have even bigger errors. The digital strategies fail completely. With digital control for low references in the nonlinear region, it has been observed that the tubes which carry water to the tank take a long time to fill up. Thus inhibiting any water flow into the tank for a long time. This can be explained by the fact that the periodic sampling coupled with nonlinear feedback from the plant is just too slow a combination. Noise compounds the problem further. At such low references, noise plays a vital role and the digital control strategies get even slower if a filter is used.
Contrary to all of the above, the analog controllers show immediate response.

### 3.8 Future Work

The criteria presented in eq. (3.10) can be used to model plants with fractional order transfer functions. Also, a circuit board with fractional order integrators can be used to emulate the plant in hardware, and another such board may be used to control it. The results of such experimentation indeed holds the possibility of being interesting. Another direction of work is to use the analog FO-PI controller where digital control is not possible due to speed and time requirements, i.e., plasma position control of a tokamak [42]. The plasma lifetime is of the order of a few milliseconds. The sampling rate, the lag induced due to A/D-D/A conversion, and processing time in the computer will be large enough as to render a standard digital control solution useless. It is thus needed to discuss better analog control strategies.
Chapter 4
Verification: Validation Platform 2

4.1 The Heat Flow System: An Overview

The heat flow plant shown in fig. 4.1 consists of a plastic duct fitted with a blower and a heating coil at one end. There are also three fast settling platinum temperature transducers placed at equal intervals along the length of the duct. The transducers are used to measure temperature at these points along the duct and fan speed is measured using a tachometer.

4.2 System Modeling

Figure 4.2 shows a plot of the real plant data as well as the integer order and fractional order models. It must be noted that just the data obtained from the first sensor (the one nearest to the heating coil) has been used for modeling and controlling the temperature of the heat flow plant. Details about the system parameters can be found in table 4.1. A

Fig. 4.1: The heat flow plant.
Table 4.1: System parameters for the heat flow plant.

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fan Input Signal</td>
<td>3V (Step)</td>
</tr>
<tr>
<td>Heater Input Signal</td>
<td>3V (Step)</td>
</tr>
<tr>
<td>Output Sensor Name</td>
<td>T0 (Sensor−1)</td>
</tr>
</tbody>
</table>

similar exercise is possible with data from any other sensor. The details concerning each type of modeling can be found in the following sections.

4.2.1 Integer Order Modeling

The ‘System Identification Toolbox’ available in MATLAB® is used to obtain an integer
order model for the heat flow plant. The toolbox has an easy to use interface where one can provide the input and output data vectors, specify the sampling interval for the data, and also try out various types of models. Of all the available methodologies, an autoregressive exogenous (ARX) model worked best for the heat flow plant. The transfer function obtained is shown in eq. (4.1). Please note the use of the suffix ‘i’ used in the nomenclature of the transfer function. From this point forward a suffix of ‘i’ signifies an integer order model and similarly a suffix of ‘f’ signifies a fractional order model unless otherwise specified.

\[ G_{i, f}(s) = \frac{0.02467s^2 + 0.006552s + 0.0002862}{s^3 + 0.4129s^2 + 0.04732s + 0.0009909} \] (4.1)

4.2.2 Fractional Order Modeling

Let:

\[ P_i(s) = \sum_{i=1}^{j} a_i s^i \sum_{i=1}^{k} b_i s^i, \quad i, j, k \in \mathbb{Z}, j \leq k, \] (4.2)

represent an integer order transfer function of a system. Let \( \alpha_1, ..., \alpha_m \), represent the fractional orders of all the available ‘Fractors’ subjected to the following condition:

\[ m \geq 2. \] (4.3)

Let:

\[ P_f(s) = \frac{k_1}{s^{\alpha_1} + c_1} + \frac{k_2}{s^{\alpha_2} + c_2}, \quad \alpha_1, \alpha_2 \in \mathbb{R}, \] (4.4)

be the fractional order model approximating eq. (4.2). Note that \( \alpha_1, \alpha_2 \) in eq. (4.4) are from eq. (4.3). A fit in the frequency domain is thus obtained by minimizing the following:

\[ J = \int_0^\infty |P_f(j\omega) - P_i(j\omega)|^2 d\omega, \] (4.5)

\[ J \approx \sum_{k=0}^{N_\omega} |P_f(j\omega_k) - P_i(j\omega_k)|^2, \] (4.6)

\[ \omega_k = k(\Delta \omega) + \omega_L, \]

\[ \Delta \omega = \frac{\omega_H - \omega_L}{N_\omega}. \]
The ‘fminsearch’ function in MATLAB® is used and the values of $k_1, k_2, \alpha_1, \alpha_2, c_1, c_2$ for which the index ‘J’ is minimized is selected. Thus we can now have a fractional order transfer function that can be implemented in analog hardware, since, the orders $\alpha_1, \alpha_2$ used are among the practically available orders of the ‘Fractors.’ Note that, in the event none of the practically available orders of the ‘Fractors’ are suitable, eqs. (3.9) and (3.11) can be used to optimize a model obtained with any arbitrary fractional order to the ones available.

Figure 4.3 shows the Bode plots for the integer order model and the fractional order model of the heat flow system. The fractional order model used for modeling the heat flow system is of the following form:

$$G_{hf}(s) = \frac{k_1}{s^\alpha + a} + \frac{k_2}{s^\beta + b}, \quad (4.7)$$

where $k_1, k_2, a, b, \alpha, \beta \in \mathbb{R}$. The index represented in eq. (4.5) is minimized to find the optimal values for the constants $k_1, k_2, a, b, \alpha$, and $\beta$ for the heat flow system. The values of the constants used for modeling the heat flow system using a fractional order transfer function are shown in table 4.2. Figure 4.4 shows the Simulink® implementation of the fractional order transfer function whose parameter values are presented in table 4.2.

The transfer function attained by the use of the parameters in table 4.2 presents only the relation between the input voltage to the plant, and the output voltage obtained from the sensors. In order to convert it to temperature, multiplication by a certain output gain, and adding appropriate room temperature offsets is essential, as shown in fig. 4.4. It is also observed from fig. 4.3 that the frequency response of the fractional order and integer order models has a maximum difference of three units until about a frequency of 100 rad/sec. For higher frequencies the difference diminishes and the frequency response of both the models coincide. Thus it can be concluded that the methodology used to obtain a fractional order

Table 4.2: Heat flow plant - FO-model parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>$k_1$</th>
<th>$k_2$</th>
<th>$a$</th>
<th>$b$</th>
<th>$\alpha$</th>
<th>$\beta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>0.0463</td>
<td>-0.0299</td>
<td>0.1762</td>
<td>0.9605</td>
<td>0.6495</td>
<td>0.5645</td>
</tr>
</tbody>
</table>
model works fairly well.

4.3 Controller Design

The plant described in eq. (4.2) can be approximated by a FOPDT model. The gain $K$, time constant $T$, and the delay $L$ of the obtained FOPDT model are used to design the integer order and fractional order controllers. The parameters of the ZN-PID controller used are shown in table 4.3. The eqs. (3.4) and (3.5) described in sec. 3.4 are used to design the FO-PI controller. The parameter values of the FO-PI controller used for the heat flow system are also shown in table 4.3.

4.4 Experimental Setup

Figure 4.4 shows the Simulink® models used to test the designed controllers on the

![Bode Diagram](image)

Fig. 4.3: Heat flow FO-model Bode plot.
integer order and fractional order models. Given the fact that IO/FO mathematics may be used to design a controller or estimate a plant model, there exist four possible combinations for controllers and plants. However, only two of them are considered in fig. 4.4 because for a real-life scenario, it is only the design methodology used for the controller that matters. Models are only approximations, so even if we can conclude that a particular combination of a model and controller gives the best results, such a conclusion would have no physical meaning. Hence, the other two cases have not been considered for simulation.

The analog hardware control experiments performed with the analog controller board have the same setup as discussed for the coupled tank in sec. 3.2.2. The only differences being the controller parameters (shown in table 4.3), and a different plant being controlled. The details have thus been omitted.

Table 4.3: Controller parameters for the heat flow system.

<table>
<thead>
<tr>
<th>Controller</th>
<th>$K_p$</th>
<th>$K_i$</th>
<th>$K_d$</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>FO–PI</td>
<td>2.6847</td>
<td>0.1472</td>
<td>0</td>
<td>0.7</td>
</tr>
<tr>
<td>ZN–PID</td>
<td>10.7652</td>
<td>1.3262</td>
<td>0.1885</td>
<td>1</td>
</tr>
</tbody>
</table>

Fig. 4.4: Heat flow system simulation setup.
4.5 Results

This section discusses the experimental results. As seen in fig. 4.5 the FO-PI controller works much better than the ZN tuned PID controller. The FO-PI controller has a faster rise time, shorter settling time, and much lesser overshoot. Hence, all the H-I-L experiments and the analog control experiments have been performed using the FO-PI controller. Figure 4.6 shows the results obtained when the FO-PI controller is implemented in Simulink® as a part of a H-I-L setup using the Quanser board. From the results it is observed that the actual H-I-L system responds faster than the simulation. However, a lot of oscillation is seen in the output when it reaches close to the reference. This behavior can be attributed to the time delay induced in the system as a result of the A/D, D/A conversion taking place as the signals pass from the computer to the plant and vice versa. A disturbance is induced in the plant temperature by unplugging the controller input. It is seen that the controller responds promptly once it is plugged back in. However, it is seen that the control effort has very large and continuous fluctuations between the entire output supply voltage range.

![Fig. 4.5: FO vs. IO control simulation - heat flow system.](image-url)
Figure 4.7 shows the results of the analog FO-PI control experiment conducted by connecting the analog FO-PI controller board directly to the heat flow plant. Simulink® and the Quanser system are used only for data acquisition. It is clearly observed that the rise time is the same for both the H-I-L controller and the true analog FO-PI controller. However, the output obtained with the analog FO-PI controller has very small fluctuations about the reference. Also, the control effort shows a much lesser amplitude of fluctuations. It also reacts to the disturbance more quickly and has a much lesser overshoot. The smooth operation of the controller is evident from the error curve which is almost flat for a majority of the time period. It is thus clearly obvious that the analog implementation of the FO-PI controller works better than the digitally implemented controller in the H-I-L setting. The benefits of cutting down the delays induced due to A/D, D/A conversion are thus clearly visible.

Figure 4.8 shows the results obtained when a higher reference is set. The heat flow system has a very large time constant, thus it takes a long time for the temperature to rise.

Fig. 4.6: Digital FO-PI control - heat flow system (H–I–L).
Fig. 4.7: Analog FO-PI control - heat flow system, experiment 1.

Fig. 4.8: Analog FO-PI control - heat flow system, experiment 2.
Another factor being the limited control effort of the analog FO-PI controller board which saturates at 5 volts. Thus, the controller output remains at the saturation limit till the plant temperature attains the reference and then the control effort drops down. Experiments have been performed for checking disturbance rejection. For such a high reference it is observed that there is a very small but finite steady-state error.

4.6 Conclusion

It can thus be concluded from the experimental results presented in the previous section that FO-PI control performs better than ZN-PID control. Analog FO-PI control has lesser output oscillations as compared to the digital variants. Hence, analog FO-PI control is the way to go for control applications where the operating time and operational delays are of critical importance.

From the results obtained from fractional order modeling of the heat flow plant, it is evident that a FO approach to system modeling is indeed possible. If the values from table 4.2 and the form of the FO transfer function presented in eq. (4.7) are compared to the IO eq. (4.1), it is found that the order of the model has been reduced from three to less than one. Thus, demonstrating that FO modeling can be a valuable tool for reducing the order of complex high order IO models.
Section 5: Verification: Validation Platform 3

5.1 The Fan and Plate System: An Overview

Figure 5.1 shows the real setup of the fan and plate system. As is evident from the name it is a simple system which consists of a fan blowing air towards a plate. The control objective is to maintain the angle of the fan at a desired constant value by changing the rate of flow of air. The actuator is the fan and the problem is essentially of motor speed control. The compressibility of air, turbulence from other sources, and the time it takes air to travel from the fan to the plate makes this a very interesting control problem. The first step is system modeling which is discussed in detail in the next section.

5.2 System Modeling

Figure 5.2 shows that the output of the fan and plate system is very noisy. The observed noise can be attributed to the aerodynamics of the system. The air escaping around the

Fig. 5.1: The fan and plate system.
sides of the plate and the gushing stream of air from the fan causes oscillations in the plate angle. These oscillations are manifested as noise. Filtering is one way to get rid of the noise, however, any low-pass filter slows down the system greatly. This effect is more pronounced in this plant because the control is initially implemented in a H-I-L setup using the Quanser board so the delay induced by A/D, D/A converters is also added to it. The fan oscillates violently when a controller is tested with a low-pass filter in the loop. Hence, a Kalman filter is used to reduce the noise, which may provide a better system model and help us view the output with reduced noise. Kalman [43] discusses the details of the filtering strategy. Welch and Bishop [44] explain the fundamentals about the same. Figure 5.3 shows the
setup used for noise removal with the help of a Kalman filter. Yi Cao’s work [45] has been used for implementing the same. The initially sampled noisy data is processed with the help of the ‘System Identification Toolbox’ present in MATLAB®. A simple first order model is obtained, the transfer function for this single input single output (SISO) system is transformed into a state space equivalent to obtain the required matrices for the Kalman filter.

As seen in fig. 5.2 the estimated output obtained from the Kalman filter is subtracted from the actual plant output. It was observed that the mean estimated output was off from the mean actual output by a small factor. This difference can be adjusted by using a gain equal to the ratio of the mean actual output, and the mean estimated output. Finally, the output from the model is added to the difference between the actual and estimated outputs. The noise is thus reduced to a great extent without the lag of a low-pass filter. Table 5.1 shows the initial model used to obtain the state space matrices required for the Kalman filter to work.

![Fig. 5.3: Kalman filter implementation for the fan and plate system.](image-url)
Table 5.1: Initial parameters of the fan and plate system.

<table>
<thead>
<tr>
<th>Type of Model</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transfer Function</td>
<td>$2.459/\left(s + 0.7837\right)$</td>
</tr>
<tr>
<td>State Space</td>
<td>$A = -0.7837, B = 1, C = 2.4588, D = 0$</td>
</tr>
</tbody>
</table>

5.2.1 Integer Order Modeling

The filtered data obtained, as shown in fig. 5.2, can then be used to obtain either an integer order or fractional order model of the fan and plate system. For the integer order model the ‘System Identification Toolbox’ is used as mentioned earlier. Figure 5.4 shows the comparison between the simulated results obtained from the integer order and fractional order models. The details about fractional order modeling are presented in the next section, however, it can be observed that both the models fit the plant output quite well. Equation (5.1) represents the integer order FOPDT model for the fan and plate system.

$$G_{ifan}(s) = \frac{K e^{-Ts}}{Ts + 1} = \frac{3.1376 e^{-0.62s}}{1.2759s + 1}. \quad (5.1)$$

5.2.2 Fractional Order Modeling

The details of the modeling methodology used for obtaining a fractional order model have already been described in sec. 4.2.2. The fractional order transfer function obtained for the fan and plate system is similar to eq. (4.7). It can thus be represented as:

$$G_{ffan}(s) = \frac{g_1}{s^\gamma + c} + \frac{g_2}{s^\delta + d}, \quad (5.2)$$

where $g_1, g_2, c, d, \gamma, \delta \in \mathbb{R}$. The index represented in eq. (4.5) is minimized to find the optimal values for the constants $g_1, g_2, c, d, \gamma, \delta$ for the fan and plate system. Note that the procedure used is exactly similar to the one used for the heat flow system. The values of the constants used for modeling the fan and plate system using a fractional order transfer function are shown in table 5.2. The Simulink® model used to simulate the response of the fractional order model of the fan and plate system is similar to the one showed for the
heat flow plant in fig. 4.4. The parameters are the ones given in table 5.2. Figure 5.5 shows the comparison between the Bode plots of the integer order model and the fractional order model. It can be seen that the frequency response of the fractional order model is quite close to that of the integer order model for lower frequencies, but for higher frequencies there is some error between then two.

Table 5.2: Fan and plate system - FO model parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>$g_1$</th>
<th>$g_2$</th>
<th>$c$</th>
<th>$d$</th>
<th>$\gamma$</th>
<th>$\delta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>0.9747</td>
<td>1.8860</td>
<td>3.2322</td>
<td>0.6427</td>
<td>1.5645</td>
<td>0.944</td>
</tr>
</tbody>
</table>

Fig. 5.4: IO vs. FO modeling simulation - fan and plate system.
Another aim of this work is to be able to emulate the results obtained above by using analog hardware. Thus proving that fractional order modeling is not only possible theoretically, but also that it can be implemented in real life.

Figure 5.6 shows the actual analog hardware implementation for emulating eq. (4.7). It is specific to the parameters of the fan and plate system mentioned in table 5.2. The unused operational amplifiers appear in the schematic because quad op-amp chips have been used. The board shown in fig. 5.7, however, is a more general implementation. The schematic shown in fig. 5.6 has been implemented on a bread board to test its performance. Due to the excessive number of operational amplifiers used, tuning the hardware is a difficult task. When the hardware emulating the plant is connected to the controller the hardware starts behaving unreliably. The reason may be attributed to the fact that when a single
Fig. 5.6: Fan and plate FO-model board schematic.

Fig. 5.7: Generalized analog FO-model board.
signal passes through a large number of amplifiers on the controller board, and the plant board, there exists a very small range of values of gains for each amplifier which allow the signal to pass through as required, without getting railed or clipped. The schematic shown in fig. 5.6 is specific to the fan and plate system because of the presence of an integrator of order greater than one. It is thus more practical to design a general purpose board that can be used to emulate a greater number and variety of plants. Figure 5.8 shows a simpler, more general, and practical approach to emulating the fractional order models of any plant using hardware. The board shown in fig. 5.7 is based on the schematic shown in fig. 5.8. It implements a single term fractional order equation of the form $\frac{k}{s^\alpha + a}$. For details about the board Appendix D must be consulted. A model for the fan and plate plant having the form $\frac{k}{s^\alpha + a}$ can be obtained by minimizing the index given in eq. (4.5). Due to the presence of just a single term the frequency response is not as close to the frequency response of the integer order model. However, it is clearly seen from fig. 5.9 that the modeled output matches the plant output closely in the time domain. Due to the lesser number of operational amplifiers used in the generalized FO model board, the problem of feasible amplifier gains being in a very narrow range is eliminated. Another appealing feature is that one can easily cascade multiple such boards to attain FO models of the form shown in eq. (5.3) or connect them in parallel to obtain FO models of the form shown in

![Fig. 5.8: Generalized analog FO-model board schematic.](image-url)
Fig. 5.9: Fan and plate plant analog FO-model vs. plant data.

eq. (5.4). As it is general in nature and not strictly designed with the parameters of the fan and plate system in mind; the amplifier gains, the ‘Fractors,’ and the capacitors can be changed to be able to model any other plant as well.

\[ G_f(s) = \frac{k}{(s^\alpha + a)(s^\beta + b)(s^\gamma + c) \cdots (s^n + n)} \]  

(5.3)

\[ G_f(s) = \frac{k_1}{s^\alpha + a} + \frac{k_2}{s^\beta + b} + \frac{k_3}{s^\gamma + c} \cdots + \frac{k_n}{s^n + n} \]  

(5.4)
5.3 Controller Design

The board shown in fig. 5.10 is the final version of the FO-PI controller. The reason it is called an FO-PII controller is due to the fact that this board does not have a derivative channel, but it has two integrator channels. Another difference between this board and the one shown in fig. 3.7 is that this board can operate either in unipolar or bipolar configuration. The previous board is only capable of unipolar operation. As for all the other plants in this work, the eqs. (3.4) and (3.5) described in sec. 3.4 are used to design the FO-PI controller. One may not always have the ‘Fractor’ with the desired order at hand, and an approximation using the methodology described in sec. 3.5 maybe required. Hence, this board has two integrators. For details pertaining to this board please refer to Appendix E. Table 5.3 shows the parameters of the FO-PI controller and the ZN-PID controller used to carry out the experiments mentioned in this chapter. As for every other experiment in this work, the FO-PI controller has been tuned according to Chen and Bhaskaran’s

![Fig. 5.10: Analog fractional FO-PII controller board.](image)
Table 5.3: Controller parameters for the fan and plate plant.

<table>
<thead>
<tr>
<th>Controller</th>
<th>$K_p$</th>
<th>$K_i$</th>
<th>$K_d$</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>FO–PI</td>
<td>0.2899</td>
<td>1.2785</td>
<td>0</td>
<td>0.9</td>
</tr>
<tr>
<td>ZN–PID</td>
<td>0.7871</td>
<td>0.8064</td>
<td>0.31</td>
<td>1</td>
</tr>
</tbody>
</table>

rules [12].

5.4 Experimental Setup

Figure 5.11 shows the setup used for H-I-L FO-PI control experiments on the fan and plate system. A similar setup with a PID controller in place of the FO-PI controller has been used to perform the H-I-L ZN-PID control experiments on the fan and plate system. Figure 5.12 shows the purely analog control setup where the plant being controlled is the emulated fan and plate system using the analog FO-model board. The setup uses an analog controller board to control a plant model implemented using analog electronics. The Quanser system and the computer is used only to monitor the data and provide the reference input to the controller. The results of the experimental setups shown in fig. 5.11 and fig. 5.12 are presented in the next section.

5.5 Results

It is clearly seen from fig. 5.13 that the FO-PI controller performs much better than the ZN-PID controller. Although the PID controller has a slightly smaller overshoot, it has poor disturbance rejection characteristics. The output with the PID controller is seen

Fig. 5.11: Digital FO-PI control setup - fan and plate system (H–I–L).
to be slow to react to disturbances. Whereas, the output of the FO-PI controller reacts quickly to disturbances and settles faster to a value near the reference. Small oscillations in the output are observed due to air turbulence, currents, compressibility, and the delay between the time the air leaves the fan and hits the surface of the plate. Air escaping around the edges and the bottom of the plate also contribute to some of the effects. It should be noted that fins have been added on the side of the plate to reduce the effects due to air escaping around the sides. The delay induced by A/D, D/A conversion of the sensor, actuator signals also contributes to the observed oscillations. Figure 5.14 shows the results of one of the experiments performed with the setup shown in fig. 5.12. The controller has
Fig. 5.13: FO-PI control vs. ZN-PID control - fan and plate system (H–I–L).

A fast response and controls the plant output steadily at the reference with virtually no error. The disturbances seen at around a time of 15 seconds and 20 seconds are induced on purpose to test the disturbance rejection capabilities of the controller board. Disturbances are created by either shutting down the controller or the plant or both. The sections of the control effort plot which show a zero effort, correspond to the amount of time that the FO-PII controller board was turned off. Similarly, the effort is found to hit the 5 volt rail voltage when the plant is turned off. From the control effort plot it is also evident that the effort remains steadily at around 1V and is saturated for very short periods when the error is large. Figure 5.15 shows the zoomed in section of a square wave tracking experiment performed with the older FO-PID controller board as shown in fig. 3.7. This board operates only in the unipolar mode, hence the negative half cycles have been omitted, hence the time
Fig. 5.14: Analog FO-PI control of the analog FO-model board - result #1.

axis shows a start time of around 20 seconds. The analog PID control experiments on the analog FO-model board have been performed by replacing the ‘Factors’ by capacitors, so that the controller board can act like an analog integer order proportional, integral and derivative (IO-PID) controller. From the plots it is evident that analog ZN-PID control is slower even in the purely analog setting despite the absence of sampling and A/D, D/A conversion delays. It is also seen that the IO-PID controller output oscillates, whereas, the FO-PI controller reaches the set point and stays there firmly. The control effort for the analog IO-PID controller is also seen to be railed for a period of time longer than the FO-PI controller, and it is found to oscillate between both the upper and lower saturation limits even for small variations in the plant output. Figure 5.16 represents the results of
Fig. 5.15: Analog FO-PI vs. analog PID control of the analog FO-model board.

squarewave tracking experiments performed using the setup shown in fig. 5.12. As seen in fig. 5.16, the experiment is performed for two different amplitudes. The controller can track both the positive and negative half cycles, as the new FO-PII controller board can operate in bipolar mode. It can be seen that for the reference of 3°, the output follows the reference with almost no error (other than the error at the point where the squarewave switches signs). It also has a very fast rate of rise. For a reference of 4.5°, however, the output takes a comparatively longer time to reach the reference. This is possible since the reference being tracked is very close to the saturation limit of the controller output. It thus results in a slower rate of rise because the controller may be saturated for a longer period of time as compared to the previous case. The steady-state error in this case too is infinitesimally small.
5.6 Conclusion and Future Work

From the experimental results presented in the previous section, it can be clearly seen that FO-PI control is superior to integer order ZN-PID control in the H-I-L/simulation setting, and the actual analog hardware setting. The obvious benefits observed are: faster response, little or no overshoot, fast disturbance rejection, more robustness to noise, infintesimally small steady-state error, and lesser control effort.

This chapter also shows that fractional order modeling of systems is not only possible in theory, but it can also be implemented in real life. The results obtained from the experiments regarding analog FO control of the analog FO plant have very little or no noise. This
phenomenon is obvious, since, noise can easily be eliminated from properly tuned analog circuits. Also, this setup was unaffected by external factors like air turbulence, etc., which are the causes of disturbance and noise for the real fan and plate system.

Connecting many units of the FO model board, as mentioned in eqs. (5.3) and (5.4), and obtaining more accurate fractional order models representing real-life phenomena should definitely be considered as a direction for future work. Similarly, building hardware units for controllers based on other fractional order strategies, and testing them on real-life plants may provide interesting results with better performance over the current methodologies discussed.
Chapter 6

Plasma Position Control of The STOR-1M Tokamak

6.1 A Tokamak - An Introduction

The word tokamak is an acronym for the Russian words “toroidal’naya kamera v magnitnykh katushkakh” which means a toroidal chamber in magnetic coils [19–23]. The tokamak is thus a doughnut shaped device which continues to be the most promising means for magnetic confinement of heated plasma.

Nuclear fusion appears to be the most promising source of energy required to quench the world’s ever growing energy demands.

\[ E = (m_r - m_p)c^2 \] (6.1)

Equation (6.1) is the Einstein equation which provides a relation between the loss of mass and the resultant energy. \( E \) is the resultant energy produced due to nuclear fusion. \( m_r, m_p \) are the masses of the nuclei before and after the reaction, respectively, and \( c \) is the speed of light. It should be noted that the process occurring is a fusion reaction in which Hydrogen nuclei fuse together to form Helium, accompanied by a release of energy. Plasmas are known as the fourth state of matter. To generate fusion energy one must heat the fuel sufficiently high so that it transforms into plasma (an ionized gas containing positively and negative charged ions, which are good conductors of electricity, and are affected by a magnetic field), and confine the plasma for a sufficiently long time so that the energy produced is more than that used in heating. The tokamak is one such magnetic confinement device [21,22].

The ability to estimate the displacement of the plasma from the center of the toroidal chamber is important if one has to control it. Controlling the position of the plasma col-
umn, so that it remains centered in the toroidal chamber, is a requirement to ensure greater lengths of plasma discharge. Previous plasma position controllers used on the STOR-1M where simple PID controllers and the position was estimated using analytical and empirical results from physics literature [25]. This work, however, applies the control engineers black-box approach to the tokamak. The plasma lifetime is of the order of 3-5ms, which necessitates the use of analog components, since digital controllers will be too slow to react to the system in question. The proposed analog FO-PI controller has been tested on the coupled tank system [46]. Tests on the heat flow and fan and plate systems are demonstrated in Chapters 4 and 5, respectively.

6.2 The STOR-1M Tokamak - System Overview

The STOR 1-M [24] is a small size tokamak whose parameters are shown in table 6.1. Figure 6.1 shows the tokamak as a transformer. As observed, it consists of a laminated core with a primary winding similar to a conventional transformer. The plasma inside the toroidal shell forms the secondary winding. The toroidal shell of the STOR-1M is maintained at about $1.1 \times 10^{-4}$ Torr for achieving a stable breakdown. The gas currently used is Hydrogen. The toroidal shell is fitted with roughing and turbo pumps in order to

![Fig. 6.1: The tokamak as a transformer.](image)
be able to achieve the desired pressure in the high vacuum range. As shown in fig. 6.1 the
toroidal field coils carry current all around the outside of the shell which produces a toroidal
magnetic field along the center of the shell. When the primary of the transformer is energized
and a breakdown occurs inside the shell, plasma is formed and it starts conducting current.
This current flows through the plasma itself, hence it is known as the plasma current. The
plasma current flow gives rise to a poloidal magnetic field. The interaction of the poloidal
and toroidal magnetic fields produce the resultant helical magnetic field, which ensures
magnetic confinement of the plasma. The charged particles in the plasma follow this helical
field line. Control and vertical field coils are also shown in fig. 6.1, the vertical field coils
produce fields required for stable operation and plasma breakdown. The control coils are
used for plasma position control. The core and some of the coils are clearly visible in fig.
6.2, which shows the actual STOR-1M tokamak. Some of the capacitor banks which are

Fig. 6.2: The STOR-1M tokamak.
Table 6.1: STOR 1-M tokamak parameters.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Major radius</td>
<td>21 cm</td>
</tr>
<tr>
<td>Minor radius</td>
<td>5.0 cm</td>
</tr>
<tr>
<td>Toroidal magnetic field</td>
<td>0.5 – 1 T</td>
</tr>
<tr>
<td>Plasma current</td>
<td>5 to 10 kA</td>
</tr>
<tr>
<td>Electron density</td>
<td>$\approx 6 \times 10^{12}/\text{cm}^3$</td>
</tr>
<tr>
<td>Electron temperature</td>
<td>$\approx 300 \text{ eV}$</td>
</tr>
<tr>
<td>Confinement time</td>
<td>$\approx 1 \text{ ms}$</td>
</tr>
</tbody>
</table>

used to power the various fields, are also seen. Appendix G must be consulted for further details.

6.3 Basic Operation

Figure 6.3 shows the input current waveforms for a particular shot when plasma breakdown was achieved. It must be noted that the time at which a bank fires, the sequence of firing, the amplitude of current required, and the shapes of the current waveforms are required to achieve plasma breakdown. A difference of 10 to 20% in any of the above may result in no plasma. The banks displayed in fig. 6.3 are as follows: $B_T$ - Toroidal field bank, $I_{Oh}$ - Ohmic heating bank, $I_{Ve}$ - Vertical equilibrium bank, $I_{Hc}$ - Horizontal compensation bank, and $I_{Vc}$ - Vertical compensation bank. Each bank is essentially an array of large capacitors which are charged prior to firing a shot. When a bank fires, it discharges through its respective coils producing the waveforms shown. $I_{Vc}$ is negative because this current flows in a direction opposite to the other coils. Figure 6.3 shows the input quantities required to obtain a stable plasma discharge. Figure 6.4 shows the loop voltage $V_l$ and the plasma current $I_p$ measured as a result of a plasma discharge. Please note that these plots show data only in the range of 3ms to 7ms, because prior to 3ms one can find only noise generated due to stray magnetic fields emerging from the banks firing. The data post 7ms is also not trustworthy because the plasma has already lost its equilibrium and is now unstable. The ideal plasma current shape should be a flat topped wave with a ramp on either side marking the beginning and end, respectively [19].
Fig. 6.3: Various bank currents for the STOR-1M tokamak.

Fig. 6.4: Plasma current and voltage.
6.4 Plasma Position Estimation

Figure 6.5 is a representation of the toroidal chamber and the four magnetic pickup coils used to collect data for position measurement. Please note that the directions of the control current and plasma current shown in this figure are fictitious and for representational purposes only. When a plasma discharge occurs, there is current flowing through the toroidal shell. This current induces a certain voltage into the magnetic pickup coils. If the plasma is centered then the diametrically opposite pickup coils should have an equal amount of voltage induced in them. However, the plasma column is not very stable and keeps moving rapidly, thus inducing different voltages in the pickup coils. One can estimate the position of the center of the plasma column from the differences in the induced voltages. Emami [25], uses the difference between the signals for plasma position estimation and control. However, a difference can indicate the displacement only as a percent deviation from the center. Hence, we choose to use a ratio of voltages, which aided by some simple mathematics allows us to

Fig. 6.5: The position measurement system.
make a more practical estimation. Figure 6.6 shows the geometry of the toroidal chamber. The inner circle represents the inside of the chamber and the outer circle represents the outer surface of the steel shell. Note that the thickness has been exaggerated here for representational purposes, also note that fig. 6.6 is not drawn to scale. The distance between the magnetic pickup coils \( \approx 10\text{cm} \) (i.e. \( 2 \times \) the minor radius). \( B_o \) and \( B_i \) correspond to the magnetic field of the outer and inner horizontal pickup coils respectively, and the blue blocks show the approximate location of the pickup coils. Let the red circle represents an assumed position of the plasma column. Let \( r_o \) and \( r_i \) represent the distance of the center of the plasma column from the magnetic pickup coils. The following relations (refer to the book by Purcell [47], or any other standard reference for basic physics) can then be arrived at:

\[
B_o = \alpha_o V_o = \frac{\mu_o I_0}{2\pi r_o},
\]

(6.2)
\[ B_i = \alpha_i V_i = \frac{\mu_i I_p}{2\pi r_i}, \quad (6.3) \]

\[ r_i + r_o = 100, \quad (6.4) \]

where \( V_o, V_i \) are the voltages induced in the outer and inner pickup coils, respectively. \( I_p \) is the plasma current. \( \alpha_0, \alpha_i \) are the constant multiplication factors for the coils in Gauss/V, which are approximately equal (i.e., \( \alpha_0 \approx \alpha_i \)) [24]. If the plasma column were to hit the outer wall then \( r_o = 15 \) and \( r_i = 85 \), providing us with:

\[ \frac{V_o}{V_i} = \frac{B_o}{B_i} = \frac{r_i}{r_o} = 5.6667. \quad (6.5) \]

Thus we conclude that, when the pickup of one of the coils is 5.6667 times the other then the plasma column must have hit the wall. Similarly, the plasma column is centered if the ratio equals unity. This is based on the simplifying assumption that the plasma column can be treated as an infinitesimally small point, ignoring the plasma current distribution in the toroidal chamber. This approach is suitable for offline calculation purposes for position estimation. It also provides a better understanding of the physical movement of the plasma column. The estimated horizontal and vertical position can be seen from fig. 6.7.

A compensation voltage of 123V corresponds to stable operating conditions for the shots taken. The position measured is calculated using eq. (6.5). Data has been acquired over five successive good shots, for the same operating conditions. This data has been averaged for the purposes of calculating position. This eliminates stray effects and other uncontrollable circumstances, which may provide corrupt data and cause calculation errors. It is seen that as the voltage decreases, the horizontal position is affected. The plasma column starts diverging from the center. However, when the voltage is reduced to 95V or below, the plasma lifetime reduces greatly. Thus, the position can be seen to vary by large amounts. For modeling position for control purposes, the most stable operating condition at 123V is chosen. This is the nominal operating condition for the plant. For our work the horizontal
position is of major interest, because there are no external forces acting in the vertical di-
rection on the plasma column. This is due to the nature of the physical equations governing
the plasma dynamics [19, 20].

6.5 Plasma Position Modeling

Figure 6.7 shows the plots for horizontal and vertical position as well as the ratio of the
voltage signals of the concerned magnetic pickup coils. It can be seen that for the purpose
of position control, one does not need to know the absolute position, but just knowing the

Fig. 6.7: Estimated position.
ratio is sufficient. Maintaining the ratio of the pickup coil voltages at unity automatically ensures that the plasma column is centered. We thus concentrate on modeling the ratio of the horizontal field pickup coil voltages in this section. The ratio of the horizontal magnetic pickup coil voltages $\frac{V_i}{V_o}$ is shown in fig. 6.8. The plot shows that the first 0.8ms worth of data is composed of garbage values. The reason for this can be found in figs. 6.3 and 6.4. From fig. 6.4 it is seen that conduction of plasma current starts from about 3.4ms. The garbage data shown in fig. 6.8 is collected by the data acquisition cards prior to this time. The huge spike appearing in fig. 6.8 occurs at around 3.8ms as a result of the horizontal compensation (HC) bank, firing as shown in fig. 6.3. A better idea of the occurrence of

![Graph showing plasma position modeling - $V_c = 123V$.](image)
events can be obtained by comparing the spike on fig. 6.8 to those on fig. 6.7.

From fig. 6.8 it can be seen that the IO model obtained by using the ‘System Identification Toolbox’ available in MATLAB® models the steady-state dynamics rather well. However, it does not capture a lot of the transient behavior. The fractional order model has a certain offset from the steady-state value, however, it has a slower rate of rise as compared to the IO model. The frequency response of the FO model also does not match that of the IO model. The IO model is thus chosen to be the best representation for testing the controllers to be designed for plasma position control. A value of $123\, V$ for the vertical field compensation voltage $V_c$, is the value at which the plasma lasts the longest (if other fields and parameters are adjusted to the values mentioned in Appendix G). If the value of $V_c$ changes then the steady-state value of the plasma position changes, as is evident from fig. 6.7. The different transfer functions (IO models) found for different operating voltages for plasma position are as follows. It must be noted that the following models have been obtained with an aim of emulation using small signal analog electronics, hence a compensation voltage, $V_c = 123\, V$, has been represented by a $5\, V$ signal, and proportionally thereof for all the other compensation voltages.

$$\begin{align*}
G_{iT}(s)|_{V_c=123V} &= \frac{2.84s^3 + 1.14e7s^2 + 1.12e14s + 4.5e20}{s^4 + 1.51e4s^3 + 3.94e13s^2 + 4.37e17s + 2.147e021} \quad (6.6) \\
G_{iT}(s)|_{V_c=100V} &= \frac{115.7s^2 + 1.355e9s + 5.424e15}{s^3 + 6282s^2 + 5.901e12s + 1.631e16} \quad (6.7) \\
G_{iT}(s)|_{V_c=95V} &= \frac{-1.343e4s^5 - 7.79e10s^4 - 3.5e17s^3 - 7.93e23s^2 - 7.93e29s + 4.56e32}{s^6 + 1.65e4s^5 + 2.16e13s^4 + 2.93e17s^3 + 3.39e25s^2 + 3.04e29s + 1.19e33} \quad (6.8) \\
G_{iT}(s)|_{V_c=90V} &= \frac{-4.673e6s^5 - 1.773e13s^4 - 5.84e19s^3 - 6.22e25s^2 - 1.82e31s + 1.31e34}{s^6 + 1.49e6s^5 + 1.56e13s^4 + 1.31e19s^3 + 3.31e25s^2 + 1.05e31s + 2.9e34} \quad (6.9)
\end{align*}$$

As stated earlier, eq. (6.6) represents the most stable operating condition for the tokamak.
For all further analysis, experimentation and comments, eq. (6.6) will be used as the nominal system model. It can be seen from the Bode plot in fig. 6.9 that the magnitude remains constant almost over five decades. The phase remains almost at zero degrees for approximately the same length. Signifying that the model for the complex device known as the tokamak, is unfortunately nothing more than a pure DC gain. As a result a simple IO transfer function will suffice, not necessitating the use of a FO model. The FO model used in fig. 6.8 is obtained as described in sec. 4.2.2. Equation (6.10) shows the FO model for the IO model described in eq. (6.6).

$$G_{IT}(s)|_{V_c=123V} = \frac{-285.457}{s^{1.5645} + 242.071} + \frac{103.256}{s^{0.746} + 395.806}$$  (6.10)

The model represented in eq. (6.10) is not used for two reasons, the first being a poor fit in the frequency domain (which was the best fit ever obtained for an FO model for the tokamak). The second reason is the high gains required on the amplifiers in order to implement the transfer function. A gain of 400 for any operational amplifier not only means amplifying the input 400 times, but it also implies amplifying the noise by the same amount.

Fig. 6.9: Bode plot for $G_{IT}(s)|_{V_c=123V}$. 
amount. Hence for reasons of practicality, this model has been discarded. Figure 6.10 shows a comparative plot of the output obtained by using the transfer function represented by eq. (6.6), its FOPDT equivalent shown in eq. (6.11), and the corresponding analog hardware model.

\[ G_T(s) \big|_{V_c=123V} = \frac{0.2096e^{-0.0007s}}{0.0864s + 1} \]  

(6.11)

The output from the analog hardware model is obtained by using the FO-model board shown in fig. 5.7. However since this is a simple FOPDT model, the ‘Fractors’ are replaced by capacitors with capacitances of 1\(\mu\)F in order to achieve the right rate of rise. The potentiometers are adjusted to get the correct amplitude (gain) on the output signal as required. From fig. 6.10 it is evident that the output of the analog hardware model of the tokamak is slightly higher than the output obtained from eq. (6.6). This is done on purpose in order to test the robustness of the analog FO controller board.

A question may arise regarding the necessity of emulating the nominal transfer function for plasma position using analog hardware. The answer lies in the fact that an H-I-L test setup would be far too slow. Although the performance can be simulated using Simulink\textsuperscript{®},

![Comparison of plasma position models used for controller testing](image)

Fig. 6.10: Analog hardware model of the tokamak for \(V_c = 123V\).
yet it does not give any concrete evidence about the controllers reliability of performance when actually tested on field. A GP6 analog computer was also tried in order to circumvent the necessity of building separate hardware, however, the potentiometers on the GP6 limit the rise time to a value that is unacceptable for the process in question. The GP6 manual by Petersen [48] is an invaluable reference for learning more about the device and the ways to use it.

It appears from fig. 6.10 that the rise time of the analog hardware is nowhere near the rise times of the models describing the system. It must be noted that the minimum possible sampling interval allowed by the Quanser system (which is used to acquire data for this particular experiment) is 1ms. Hence, nothing can really be said about the data in between. But from the plot it is obvious that the output reaches its steady-state value after 1ms, which is in conformation to the output obtained from the IO model in eq. (6.6).

### 6.6 Controller Design

Equations (3.4) and (3.5) described in sec. 3.4 are used to design the FO-PI controller for the tokamak system. Table 6.2 shows the parameters of the FO-PI controller and the ZN-PID controller used to carry out the experiments mentioned in this chapter. As for previous experiments in this work, the FO-PI controller has been tuned according to Chen and Bhaskaran’s tuning rules [12].

### 6.7 Experimental Setup

The setup shown in fig. 5.12 has been used for the hardware experiments; the only changes being the adjustment of the plant parameters to model the tokamak system. The

<table>
<thead>
<tr>
<th>Controller</th>
<th>$K_p$</th>
<th>$K_i$</th>
<th>$K_d$</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>FO–PI</td>
<td>170.2649</td>
<td>32.0719</td>
<td>0</td>
<td>0.7425</td>
</tr>
<tr>
<td>ZN–PID</td>
<td>706.584</td>
<td>714.285</td>
<td>0.00035</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 6.2: Controller parameters for the STOR-1M tokamak.
controller gains are also tuned according to the values in table 6.2. The simulation results are obtained using a standard Simulink® model with a controller and a plant, designed as per the parameters provided in the preceding sections.

6.8 Results

This section presents the results of the experiments described in the preceding section. It must be noted that the hardware experiments are emphasized more for the tokamak system. Simulating a process for a length of time in the order of a few milliseconds is easy to do in software. However, testing the behavior of the hardware and seeing if it works in the same time frame are of paramount importance. Sampling time limitations, and the difficulty in adding disturbances (within the time span of 5 ms) to the hardware experiments, make the results from the hardware and the simulation appear different.

6.8.1 Simulation Results

From fig. 6.11 it can be seen that the FO-PI controller performs much better than

![Graph](image)

Fig. 6.11: Plasma position control - simulation for $V_c = 123V$. 
the ZN-PID controller. The IO-PID controller shows a lot of oscillations in the output and needs about 0.8ms to settle down to the required reference. The FO-PI controller achieves the same within 0.3ms, without any overshoot or oscillations. A disturbance of about 0.6cm on each side has been introduced into the simulation. The FO-PI controller rejects it almost immediately, whereas the ZN-PID controller shows some oscillations in this case too.

### 6.8.2 Analog Hardware Emulation Results

Figure 6.12 shows the results of one of the analog hardware emulation experiments performed with the analog FO-PI controller board and the analog FO model board for the tokamak system. It should be noted that fig. 6.12 represents the ratio of the horizontal magnetic pickup coil voltages. Thus, the objective is to maintain the ratio at unity. From fig. 6.12 it can be seen that the controller responds fast. In about 2ms from the start, the output reaches very close to the desired set point. There is an overshoot of about 0.5%, and at about 3ms it can be said that the output has settled close to the expected reference. Thus for a plasma lifetime of about 4 – 5ms, or even 3.5ms, it is seen that the proposed

![Graph](image)

Fig. 6.12: Plasma position control (ratio) - hardware emulation for $V_c = 123V$. 

FO-PI controller will work well. Figure 6.13 shows the error signal for the experiment whose results are displayed in fig. 6.12. From the error plot it is clear that after 2ms the error is almost zero, which shows that the controller acts fast and stays at the reference. Figure 6.14 shows the results displayed in fig. 6.12 in terms of the equivalent plasma position in centimetres. Note that the model and the controller are designed to control the ratio of the horizontal magnetic pickup coil voltages, since, one does need to know the absolute position for control purposes. Figure 6.14 has been included for better understanding of the real physical position control of the plasma column. All the experimental results are based on the assumption that the vertical field compensation voltage is maintained at 123V, and all other parameters are as mentioned in Appendix G, corresponding to the most stable operational condition observed. Figure 6.15 is shown just to verify controller operation for a different reference and also to check disturbance rejection. A disturbance is added at about 2ms, the controller is seen to recover from it and get back to the desired reference. Some oscillations are observed in this case. Figure 6.16 shows the comparison between the ZN-PID controller and the FO-PI controller for a different set of tests that were carried on.

![Error Plot](image)

Fig. 6.13: Plasma position control (error) - hardware emulation for $V_c = 123V$. 
Fig. 6.14: Plasma position control (absolute) - hardware emulation for $V_c = 123\, \text{V}$.

Fig. 6.15: Plasma position control (test) - hardware emulation for $V_c = 123\, \text{V}$. 
the analog hardware emulation setup. For this test, a disturbance was added at the start, hence the delayed operation (i.e., the output converges to the reference at 3\(ms\) as opposed to 2\(ms\) as observed in fig. 6.12). From fig. 6.16 it is clear that even for extremely high speed applications, the FO-PI controller is slightly faster than the ZN-PID controller tuned as per the gains presented in table 6.2. On measuring the time difference it is found that the FO-PI controller has an advantage of 1\(\mu s\) over the IO, ZN-PID controller. The gains required for the FO-PI controller are significantly lower than that of the ZN-PID controller, implying lesser control effort and lower amplification of noise.

6.9 Difference Based Modeling and Control

The models for plasma position presented in the previous section are based on the ratio of the voltages of the inner and outer magnetic position pickup coils. However, analog division of two signals is a challenging operation and it raises concerns of reliability and ‘divide by zero’ errors. Some possible implementations for analog signal division [49, 50] have been presented in Appendix F.

![Comparison between PID and FO-PI controllers](image)

**Fig. 6.16:** Plasma position control (comparison) - hardware emulation for \(V_c = 123V\).
However for purposes of control, one can consider the difference between the horizontal magnetic pickup coil signals, to get around the problems associated with division. Again, as one does not need to know the absolute position for the purposes of control, the reference to the controller can just be zero. A null difference between the voltages of the pickup coils would mean that the plasma column is centered. Figure 6.17 shows the actual (and

![Graph showing difference in horizontal pickup coil voltages](image)

Fig. 6.17: Modeling based on pickup coil signal difference.
modeled) difference in voltages as obtained from the horizontal magnetic pickup coils.

\[ G_{iT}(s)\big|_{\text{Diff},V_c=123V} = \frac{A(s)}{B(s)}, \text{ where,} \]

\[ A(s) = -1.035e4s^{14} - 1.381e11s^{13} - 1.16e18s^{12} - 7.616e24s^{11} \]

\[ -3.374e31s^{10} - 1.274e38s^9 - 3.13e44s^8 - 7.15e50s^7 \]

\[ -9.928e56s^6 - 1.02e63s^5 - 4.86e68s^4 - 8.78e73s^3 \]

\[ +1.68e78s^2 - 2.73e81s + 3.35e84 \]  \hspace{1cm} (6.13)

\[ B(s) = s^{15} + 9.535e6s^{14} + 8.74e13s^{13} + 6.027e20s^{12} + 2.662e27s^{11} \]

\[ +1.318e34s^{10} + 3.47e40s^9 + 1.205e47s^8 + 1.914e53s^7 \]

\[ +4.39e59s^6 + 3.7e65s^5 + 4.69e71s^4 + 1.29e77s^3 \]

\[ +8.96e80s^2 + 1.303e84s + 3.591e87 \]  \hspace{1cm} (6.14)

Equations (6.13) and (6.14) present the numerator and denominator polynomials of the model representing the difference in voltages of the horizontal position magnetic pickup coils. The model is obtained using a Box-Jenkins (BJ) model found in the ‘System Identification Toolbox’ available in MATLAB®. As is clearly observed, the model required is of a very high order, and it would be intuitively obvious that the difference between the coil voltages would change with almost every shot. It is also seen from fig. 6.17 that the difference between the coil voltages has a non-zero mean value. For testing the controller, this mean value is modeled as a FOPDT system and the variations in the voltage difference observed in fig. 6.17 are added as a disturbance, for judging the controller performance. The FOPDT equivalent for the model in eqs. (6.12), (6.13), and (6.14) is as follows:

\[ G_T(s)\big|_{\text{Diff},V_c=123V} = \frac{0.000676e^{-0.0007s}}{0.0001275s + 1}. \]  \hspace{1cm} (6.15)

It is observed from table 6.3 that the gains of the FO-PI controller are significantly higher than that for the previous case. This can be attributed to the fact that the values for \( K, L, \) and \( T \) in eq. (6.15) are very small. Also, the values for the PID controller are not much
different than that from the previous case. According to step response ZN-PID tuning rules, the values of $K_i$, and $K_d$ are dependent on the delay $L$, which is the same for both cases.

Figure 6.18 shows the comparison of the ZN-PID and FO-PI controllers tuned as per the gains presented in table 6.3. The output labeled ‘FO-PI2’ corresponds to the controller tuned according to the gains in table 6.2. All the controllers have near similar performance

Table 6.3: Controller parameters for the STOR-1M tokamak - voltage difference based.

<table>
<thead>
<tr>
<th>Controller</th>
<th>$K_p$</th>
<th>$K_i$</th>
<th>$K_d$</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>FO-PI</td>
<td>520.4958</td>
<td>2219.6163</td>
<td>0</td>
<td>1.1</td>
</tr>
<tr>
<td>ZN-PID</td>
<td>323.2761</td>
<td>714.2857</td>
<td>0.00035</td>
<td>1</td>
</tr>
</tbody>
</table>

Fig. 6.18: Controller performance for the signal difference case, reference = 0V.
for this case, and each of them attain the set-point of zero within 0.5\(ms\). A disturbance from the actual output data is also added, and the controllers are seen to recover quickly. It is interesting that the controller used for controlling the plasma position, designed based on the ratio of the magnetic coil pickup signals, works the best. It might be of practical value to use this controller, since the gains required are still within the practical limits laid down by hardware, noise issues, etc.

6.10 Conclusion

From the experimental results presented in the previous section, it can be concluded that the FO-PI controller works better than the ZN-PID controller. Despite the extremely fast operational requirements, the FO-PI controller manages to be faster than the ZN-PID controller. The results have been confirmed by hardware emulation, thus corroborating the fact that the proposed controller is feasible for implementation on the STOR-1M tokamak.
Chapter 7
Conclusion

The preceding chapters have presented a variety of applications of fractional calculus in control systems. The conclusions of this work can thus be summarized as follows.

7.1 Summary

The FO-UAS experiments provide some insight into the power of the application of fractional calculus to control engineering. The FO-UAS strategy outdoes the integer order UAS in every aspect possible. It also shows us that one can achieve extremely fast response times by using fractional derivatives and integrals in the control loop. The steady-state error is also minimal. What is more intriguing is that the FO-UAS can track sinusoidal and sawtooth references of varying frequencies along with the regular square wave. The use of the Mittag-Leffler function (which forms the base of fractional calculus) as the Nussbaum function shows great benefits as compared to other functions, in terms of steady-state error and control effort. The integer order UAS is known to be affected by noise and is also known to be a poor performer when tracking low references. The FO-UAS scheme however works without any problems for extremely low references, and also in the presence of external noise. Even in the presence of noise the control effort does not saturate, which is an extremely valuable feature of this control strategy.

Experiments on the coupled tank system provide some interesting results as well. The coupled tank system has a large time constant and the sensors are extremely noisy with a high degree of nonlinearity in the lower range of operation. From the results we can conclude that digital (H-I-L) control whether integer order PID or FO-PI, is inferior to their purely analog counterparts by a substantial margin. Another important conclusion is that any required FO-PI controller can be approximated by a number of the available
‘Factors.’ Surprisingly, the approximated analog FO-PI controller is seen to have the best performance as compared to all other controllers. The presence of a greater number of ‘Factors’ can thus be concluded to be beneficial. FO-PI control also results in a lot less actuator saturation and lesser average control output, it is thus more efficient.

From the experiments on the heat flow plant it can be concluded that fractional order modeling of systems is not only possible in theory, but it can also be implemented in real life. It can also be used as a tool to reduce the order of integer order models. The heat flow plant has a very slow response and the benefits of using an FO-PI controller are readily observable. Low control effort, lesser overshoot, lesser oscillations are some of them. The superiority of purely analog control over digital H-I-L control is obvious from the results obtained. Noise, oscillations, and delay are reduced to a great extent when one uses an FO-PI controller in the analog setting.

The fan and plate system provides an interesting challenge because it is affected by nonlinear phenomenon like air turbulence, compressibility of air, etc. The system is extremely noisy and has a certain delay. It can be easily concluded from the experimental results that the ZN-PID controller is not a prudent choice for such systems because it is affected by noise, the output does not follow the reference well, and it has poor disturbance rejection. Using the FO-PI controller solves the above problems, and even though the system is noisy, the output returns to the desired reference almost immediately after a disturbance occurs. A model of the fan and plate system is implemented in hardware and the analog FO-PI controller is tested on it to provide some interesting results. When the ‘Factors’ are replaced by capacitors to make the analog FO-PI board into an IO-PID board, the controller is seen to be slower. The control output is seen to be saturated longer, and a small but definite steady-state error is observed. Using an analog FO-PI controller overcomes all the problems mentioned. It can thus be concluded that the FO-PI controller is definitely better than the integer order controller.

All of the experimentation in this work has been performed with the sole aim of testing the FO-PI control strategy that is designed for the STOR-1M tokamak. Before the plasma
position can be controlled, it is important to be able to estimate the plasma position. Using the ratio of voltages from the magnetic pickup coils can be concluded to be the easiest and most useful method of estimating position offline. For the tokamak it is observed that integer order modeling works a lot better than the equivalent fractional order model. The simulation results show that the FO-PI controller is faster than the IO-PID controller, has lesser output oscillations and better disturbance rejection characteristics. Analog hardware experiments performed with the analog FO controller and model boards confirm the findings obtained from the simulations. The analog FO-PI controller is faster than the purely analog implementation of the ZN-PID controller, and has an advantage of about $1\mu s$ over the ZN-PID controller. In a process which lasts only about $4−5ms$, this can prove to be a significant advantage. The FO-PI controller designed for the tokamak system is also robust and handles disturbances well.

The final conclusion is that fractional order calculus is practically useful in control science; it helps to obtain much better results as compared to the presently widespread integer order methodologies.

### 7.2 Future Work

Understanding the reasons behind the superior operation of the FO-UAS when tracking sinusoidal, sawtooth references is a definite direction for future work. Also, understanding the reason behind the better performance of certain Nussbaum functions over the others promises to be an interesting area for future work.

Developing different controller structures for FO control, newer tuning rules is a future possibility. This work provides a nudge in the direction of FO modeling. Using the basic FO model boards, cascading them or using them in parallel can be used to get more accurate FO models of systems. Using strategies other than a frequency domain fit to model FO systems can be explored.

Developing the FO controller board into an independent unit which tunes itself either automatically or using programmed rules, and using this FO controller unit to control a variety of plants by simply connecting them to it is definitely an exciting idea. Implementing
the controller on the real STOR-1M tokamak system is a definite future step. Studying the effectiveness of fractional calculus on such a system in real life will be an exciting area for future work.
References


Appendices
Appendix A

Fractional Order Derivatives and Integrals

This Appendix provides an extremely condensed and brief introduction to the most important and commonly used definitions when dealing with fractional order calculus. The results quoted here are from Podlubny’s book [2]. The readers are suggested to use the book for further details.

A.1 Fractional Order Derivatives

Definition A.0.1 General expression for the Grünwald-Letnikov(GL) derivative, for \( p > 0 \).

\[
aD^p_t f(t) = \lim_{h \to 0} h^{-p} \sum_{r=0}^{n} (-1)^r \binom{n}{r} f(t - rh) \quad (A.1)
\]

\( nh = t - a \)

Definition A.0.2 GL fractional derivative of the power function.

\[
aD^p_t (t - a)^\nu = \frac{\Gamma(\nu + 1)}{\Gamma(-p + \nu + 1)} (t - a)^{\nu - p} \quad (A.2)
\]

Where \( (p < 0, \nu > -1) \) or \( (p \geq 0, \nu > [p]) \), where \([p]\) denotes the integer part of \( p \).

Definition A.0.3 The Riemann-Liouville(RL) fractional derivative.

\[
aD^p_t f(t) = \frac{1}{\Gamma(m + 1 - p)} \left( \frac{d}{dt} \right)^{(m+1)} \int_a^t (t - \tau)^{(m-p)} f(\tau) d\tau, \quad (m \leq p < m + 1) \quad (A.3)
\]

Where \( m = [p] \). (We can also use \( m \in \mathbb{Z} \), see the next definition).
Definition A.0.4  The fractional order Caputo derivative.

\[
\frac{C}{a}D^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \int_a^t \frac{f^{(n)}(\tau)d\tau}{(\tau-t)^{\alpha+1-n}}, (n-1 < \alpha < n), (n \in \mathbb{Z}) \tag{A.4}
\]

Assuming \(0 \leq n-1 < \alpha < n\) and \(f(t)\) has \(n+1\) continuous derivatives in \([a, T] \forall T > a\); as shown in eq. (A.5) the Caputo derivative becomes a conventional \(n^{th}\) derivative of \(f(t)\)

\[
\lim_{\alpha \to n} \left( \frac{C}{a}D^\alpha f(t) \right) = f^n(a) + \int_a^t f^{(n+1)}(\tau)d\tau = f^{(n)}(t), n = 1, 2, ...
\]  (A.5)

- **Practical advantages of the Caputo approach**

The Riemann-Liouville approach leads to initial conditions containing the limit values of the Riemann-Liouville fractional derivatives at the lower terminal \(t = a\) as shown in eq. (A.6)

\[
\lim_{t \to a} \left( aD_t^{\alpha-1}f(t) \right) = b_1 \tag{A.6}
\]

where \(b_1\) is a given constant. The condition given in eq. (A.6), although mathematically useful is practically useless hence the Caputo approach with initial conditions as given in eq. (A.5) is used for engineering problems.

Definition A.0.5  The left Riemann-Liouville derivative.

If \(k-1 \leq p < k\), then

\[
aD_t^p f(t) = \frac{1}{\Gamma(k-p)} \left( \frac{d}{dt} \right)^k \int_a^t (t-\tau)^{k-p-1} f(\tau)d\tau, (k \in \mathbb{Z}) \tag{A.7}
\]

Definition A.0.6  The corresponding right Riemann-Liouville derivative.

\[
tD_b^p f(t) = \frac{1}{\Gamma(k-p)} \left( -\frac{d}{dt} \right)^k \int_t^b (t-\tau)^{k-p-1} f(\tau)d\tau, (k \in \mathbb{Z}, k-1 \leq p \leq k) \tag{A.8}
\]
A.2 Properties of Fractional Derivatives

Linearity:

\[ D^p (\lambda f(t) + \mu g(t)) = \lambda D^p f(t) + \mu D^p g(t), (\lambda, \mu \in \mathbb{C}) \]  
(A.9)

**Leibniz rule:**

If \( f(t) \) is continuous in \([a, t]\) and \( \varphi(t) \) has \( n + 1 \) continuous derivatives in \([a, t]\) then:

\[ aD_t^p (\varphi(t) f(t)) = \sum_{k=0}^{n} \binom{p}{k} \varphi^{(k)}(t) aD_t^{p-k} f(t) - R^p_n(t) \]  
(A.10)

where,

\[ R^p_n(t) = \frac{1}{n!(-p)} \int_a^t (t - \tau)^{-p-1} f(\tau) d\tau \int_\tau^t \varphi^{n+1}(\epsilon)(\tau - \epsilon)^n d\epsilon. \]  
(A.11)

**Fractional derivatives of a composite function:**

\[ aD_t^p F(h(t)) = \frac{(t - a)^{-p}}{\Gamma(1 - p)} \varphi(t) + \sum_{k=1}^{\infty} \binom{p}{k} \frac{k!(t - a)^{k-p}}{\Gamma(k - p + 1)} \]  
\[ \times \sum_{m=1}^{k} F^{(m)}(h(t)) \sum_{r=1}^{k} \frac{1}{a_r!} \left( \frac{h^{(r)}(t)}{r!} \right)^{a_r} \]  
(A.12)

where \( \sum_{r=1}^{k} ra_r = k \) and \( \sum_{r} a_r = m. \)

**RL fractional differentiation of an integral depending on a parameter:**

\[ aD_t^\alpha \int_0^t K(t, \tau) d\tau = \int_0^t rD_t^\alpha K(t, \tau) d\tau + \lim_{\tau \to t^-} rD_t^{\alpha-1} K(t, \tau) \]  
(A.13)
A.3 Fractional Integrals and Laplace Transforms of Fractional Derivatives

Definition A.0.7Grünwald-Letnikov fractional integral.

\[ aD_t^{-p}f(t) = \lim_{h \to 0} h^n \sum_{r=0}^{n} \left[ \begin{array}{c} p \\ r \end{array} \right] f(t - rh) \] \hspace{1cm} (A.14)

\[ nh = t - a \]

Or

\[ aD_t^{-p}f(t) = \frac{1}{\Gamma(p)} \int_a^t (t - \tau)^{p-1} f(\tau) d\tau \] \hspace{1cm} (A.15)

If the derivative \( f'(t) \) is continuous in \([a, b]\) then integrating eq. (A.15) by parts we get

\[ aD_t^{-p}f(t) = \frac{f(a)(t - a)^p}{\Gamma(p + 1)} + \frac{1}{\Gamma(p + 1)} \int_a^t (t - \tau)^{p-1} f'(\tau) d\tau \] \hspace{1cm} (A.16)

Definition A.0.8 The Laplace transform of the RL derivative

\[ L\{aD_t^p f(t); s\} = s^p F(s) - \sum_{k=0}^{n-1} s^k \left[ aD_t^{p-k-1} f(t) \right] t=0, (n-1 \leq p < n) \] \hspace{1cm} (A.17)

Definition A.0.9 The Laplace transform of the Caputo derivative

\[ L\{	frac{C}{0}D_t^p f(t); s\} = s^p F(s) - \sum_{k=0}^{n-1} s^{p-k-1} f^{(k)}(0), (n-1 < p \leq n) \] \hspace{1cm} (A.18)

Definition A.0.10 The Laplace transform of the Grünwald-Letnikov derivative

\[ L\{aD_t^p f(t); s\} = s^p F(s) \] \hspace{1cm} (A.19)
Appendix B

Bode Plots of Available ‘Fractors’ - July 18, 2008

The following figures show the Bode plots of the various available ‘Fractors’ (FO integrators in which these devices were used). These devices were built during the years 2001 - 2003 and some appear to have aged. Signs of aging are differences between the expected and the current characteristics. Some of the ‘Fractors’ appear to have no flat phase at all (due to aging), however the ‘Fractors’ which show a gradual change in their phase plots still have the flat phase phenomenon occurring over a very small bandwidth. The eqs. (3.2) and (3.3) are useful and important with regard to the use of ‘Fractors.’

Fig. B.1: ‘Fractor-0’ Bode plot, $\alpha = 0.5175$. 
Fig. B.2: ‘Fractor-1’ Bode plot, $\alpha = 0.5645$.

Fig. B.3: ‘Fractor-2’ Bode plot, $\alpha = 0.7425$. 
Fig. B.4: ‘Fractor-3’ Bode plot, $\alpha = 0.7900$.

Fig. B.5: ‘Fractor-4’ Bode plot, $\alpha = 0.9440$. 
Fig. B.6: ‘Fractor-5’ Bode plot, $\alpha = 0.8630$.

Fig. B.7: ‘Fractor-6’ Bode plot, $\alpha = 0.9190$. 
Fig. B.8: ‘Fractor-7’ Bode plot, $\alpha = 0.8680$.

Fig. B.9: ‘Fractor-8’ Bode plot, $\alpha = 0.7460$. 
Fig. B.10: ‘Fractor-9’ Bode plot, $\alpha = 0.3850$.

Fig. B.11: ‘Fractor-10’ Bode plot, $\alpha = 0.6495$. 
Appendix C

Bode Plots of Available ‘Fractors’ - June 22, 2009

The following plots show the Bode plots of the ‘Fractors’, measured at a different date. As seen in table C.1, there has been some change in the characteristics of the ‘Fractors’ over time.

It must be noted that these devices have an impedance/resistance of the order of 1-5 MΩ. ‘Fractors’ 2, 3, and 7 appeared to be shorted at the time of the second measurement and had an impedance/resistance of about 2-4Ω. Hence, they can be declared unusable.

Table C.1: ‘Fractor’ characteristics comparison.

<table>
<thead>
<tr>
<th>‘Fractor’ No.</th>
<th>α-2008</th>
<th>α-2009</th>
<th>Change - %/yr</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.5175</td>
<td>0.966</td>
<td>86.66%</td>
</tr>
<tr>
<td>1</td>
<td>0.5645</td>
<td>0.715</td>
<td>26.66%</td>
</tr>
<tr>
<td>2</td>
<td>0.7425</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>3</td>
<td>0.79</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>4</td>
<td>0.9440</td>
<td>0.99</td>
<td>4.87%</td>
</tr>
<tr>
<td>5</td>
<td>0.8630</td>
<td>0.905</td>
<td>4.64%</td>
</tr>
<tr>
<td>6</td>
<td>0.9190</td>
<td>0.922</td>
<td>0.326%</td>
</tr>
<tr>
<td>7</td>
<td>0.8680</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>8</td>
<td>0.7460</td>
<td>0.7705</td>
<td>3.28%</td>
</tr>
<tr>
<td>9</td>
<td>0.3850</td>
<td>0.832</td>
<td>116.103%</td>
</tr>
<tr>
<td>10</td>
<td>0.6495</td>
<td>0.772</td>
<td>18.86%</td>
</tr>
</tbody>
</table>
Fig. C.1: ‘Fractor-0’ Bode plot, $\alpha = 0.966$.

Fig. C.2: ‘Fractor-1’ Bode plot, $\alpha = 0.715$. 
Fig. C.3: ‘Fractor-4’ Bode plot, $\alpha = 0.99$.

Fig. C.4: ‘Fractor-5’ Bode plot, $\alpha = 0.905$. 
Fig. C.5: ‘Fractor-6’ Bode plot, $\alpha = 0.922$.

Fig. C.6: ‘Fractor-8’ Bode plot, $\alpha = 0.7705$. 
Fig. C.7: ‘Fractor-9’ Bode plot, $\alpha = 0.832$.

Fig. C.8: ‘Fractor-10’ Bode plot, $\alpha = 0.772$. 
Appendix D

Fractional Order Plant Board Details

D.1 Model Board Operational Waveforms and Schematics

Figure D.1 shows the open loop output obtained for a input signal with unit amplitude and 1Hz frequency. Specific parameters for obtaining this output are $C_7 = 2200\mu F$, $\alpha = 0.5645$, gains for integrators 1, 2, and the output buffer are 1, 5, and 3, respectively. Feedback and input resistances are equal.

This setup can be used as a standard reference or as a check to see if the hardware is working properly prior to experimentation.

Fig. D.1: Analog FO model board test I/O waveform.
D.2 Model Board Factsheet

- All resistors used have a value of 10kΩ.
- All potentiometers used have a value of 200kΩ.
- Use testpoints TP1 and TP5 to measure the gain for integrator 1.
- Use potentiometer R6 to change the gain for integrator 1.
- Use testpoints TP4 and TP7 to measure the gain for integrator 2.
- Use potentiometer R7 to change the gain for integrator 2.
- Use testpoints TP3 and TP2 to measure the output gain.
- Use potentiometer R5 to change the output gain.
- Use testpoints TP8 and TP1 to measure the feedback gain.
- Use potentiometer R9 to change the feedback gain.
- Use power, input, and output polarities as marked on the circuit board.
- The Switch across the ‘Fractor’ must be used to short it after prolonged operation.
Fig. D.2: Analog FO model board - enlarged schematic.
Appendix E
Fractional Order Controller Board Details

- Use testpoints TP1 and TP4 to measure gain $K_p$.
- Use potentiometer R4 to change $K_p$.
- Use testpoints TP5 and TP4 to measure gain $K_{i1}$.
- Use potentiometer R13 to change $K_{i1}$.
- Use testpoints TP8 and TP4 to measure gain $K_{i2}$.
- Use potentiometer R18 to change $K_{i2}$.
- Output gain can be measured across pins 9 and 14 of the second chip (marked IC4 in the schematic and placed vertically on the board).
- Input power pin configuration ($-5V, GND, +5V$).
- Reference pin configuration ($-, +$).
- Feedback pin configuration ($-, +$).
- Output pin configuration ($+, -$).
- Pin configurations apply to the side with the printed text.
- Switches across ‘Factors’ must be used to short them after prolonged operation.
- Other switches are useful for turning channels on or off selectively.
Fig. E.1: Analog FO-PII controller board - detailed schematic.
Appendix F

Analog Signal Division - Schematics

The operational details for the schematic shown in fig. F.2 can be found in the March 2007 issue of the Electronics Design Strategy News (EDN), Europe [49].

Fig. F.1: Analog divider 1 - courtesy of National Semiconductor.

Fig. F.2: Analog divider 2 - courtesy of EDN, Europe.
Appendix G

STOR-1M Tokamak System Details

For the important system parameters pertaining to the STOR-1M tokamak please consult table 6.1. Table G.1 shows the nominal voltages for the various capacitor banks. These voltages represent to the nominal operating condition of the tokamak. The nominal values for the horizontal and vertical fields have been presented in table G.2. Table G.3 provides the calibration factors for the various Rogowski coils. The pickup coil parameters shown in table G.4 have been obtained from Wolfe’s thesis [24]. Table G.5 presents the details about bank firing times. Table G.6 shows the (most important) shots which were used to obtain the position model, controller etc. For all shots the operating pressure is required to be maintained at about $1.1 - 1.5 \times 10^{-4}$ Torr.

Table G.1: STOR 1-M, bank voltages (nominal values).

<table>
<thead>
<tr>
<th>Bank Name</th>
<th>BT</th>
<th>HC</th>
<th>VC</th>
<th>VEF</th>
<th>VES</th>
<th>CTB</th>
<th>CTF</th>
<th>CTS</th>
<th>TH</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voltage (V)</td>
<td>3000</td>
<td>30</td>
<td>123</td>
<td>470</td>
<td>290</td>
<td>2400</td>
<td>220</td>
<td>190</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Table G.2: STOR 1-M, horizontal and vertical field parameters.

<table>
<thead>
<tr>
<th>Bank Name</th>
<th>HC</th>
<th>VC</th>
<th>VEF</th>
<th>VES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rated Voltage (V)</td>
<td>450</td>
<td>150</td>
<td>750</td>
<td>450</td>
</tr>
<tr>
<td>Nominal Voltage (V)</td>
<td>30</td>
<td>123</td>
<td>470</td>
<td>290</td>
</tr>
<tr>
<td>Variac Setting</td>
<td>10/120</td>
<td>34/120</td>
<td>50/120</td>
<td>4.2/10</td>
</tr>
<tr>
<td>Voltmeter Multiplier</td>
<td>1</td>
<td>30</td>
<td>100</td>
<td>100</td>
</tr>
</tbody>
</table>
Table G.3: STOR 1-M, Rogowski coil parameters.

<table>
<thead>
<tr>
<th>Measured Parameter</th>
<th>Calibration Factor - A/V</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_P$</td>
<td>$127.075 \times 10^6$</td>
</tr>
<tr>
<td>$I_{BT}$</td>
<td>$1$</td>
</tr>
<tr>
<td>$I_{OH}$</td>
<td>$10.95 \times 10^6$</td>
</tr>
<tr>
<td>$I_{VE}$</td>
<td>$3.7 \times 10^6$</td>
</tr>
<tr>
<td>$I_{HC}$</td>
<td>$3.18 \times 10^6$</td>
</tr>
<tr>
<td>$I_{VC}$</td>
<td>$4.117 \times 10^6$</td>
</tr>
</tbody>
</table>

Table G.4: STOR 1-M, plasma position magnetic pickup coil parameters.

<table>
<thead>
<tr>
<th>Coil Name</th>
<th>a(mm)</th>
<th>l(mm)</th>
<th>N</th>
<th>R(Ω)</th>
<th>L(mH)</th>
<th>Cal. - Gauss/V</th>
</tr>
</thead>
<tbody>
<tr>
<td>Up</td>
<td>3.5</td>
<td>6</td>
<td>462</td>
<td>22</td>
<td>0.97</td>
<td>3.23</td>
</tr>
<tr>
<td>Down</td>
<td>3.5</td>
<td>6</td>
<td>460</td>
<td>22.2</td>
<td>0.93</td>
<td>3.37</td>
</tr>
<tr>
<td>In</td>
<td>3.5</td>
<td>6</td>
<td>460</td>
<td>22.6</td>
<td>0.96</td>
<td>3.26</td>
</tr>
<tr>
<td>Out</td>
<td>3.5</td>
<td>6</td>
<td>480</td>
<td>22.8</td>
<td>1.01</td>
<td>3.17</td>
</tr>
</tbody>
</table>

Table G.5: STOR 1-M, bank timing and duration (ms).

<table>
<thead>
<tr>
<th>Bank Name</th>
<th>Firing Time</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>BT</td>
<td>1.2</td>
<td>10</td>
</tr>
<tr>
<td>HC</td>
<td>3.6</td>
<td>10</td>
</tr>
<tr>
<td>VC</td>
<td>2.8</td>
<td>10</td>
</tr>
<tr>
<td>VEF</td>
<td>3.4</td>
<td>10</td>
</tr>
<tr>
<td>CTB</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>CTF</td>
<td>3.4</td>
<td>10</td>
</tr>
<tr>
<td>RF</td>
<td>3.1</td>
<td>0.3</td>
</tr>
</tbody>
</table>

Table G.6: STOR 1-M, mini shot log.

<table>
<thead>
<tr>
<th>$(V_c)$</th>
<th>Bank Voltage</th>
<th>Shot Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>90V</td>
<td>3535, 3536, 3537, 3538, 3539</td>
<td></td>
</tr>
<tr>
<td>95V</td>
<td>3560, 3561, 3562, 3563, 3564</td>
<td></td>
</tr>
<tr>
<td>100V</td>
<td>3581, 3582, 3583, 3584, 3585</td>
<td></td>
</tr>
<tr>
<td>123V</td>
<td>3601, 3602, 3603, 3604, 3605</td>
<td></td>
</tr>
</tbody>
</table>
Appendix H

Mittag - Leffler Function: Code

```c
#define S_FUNCTION_NAME mitlef
/* Defines and Includes - cMex version*/
#define S_FUNCTION_LEVEL 2
#include "math.h"
#include "simstruc.h"

%source of the gamma function obtained from Ooura's Mathematical
    Software Packages - Special Functions - Gamma / Error Functions

double dgamma(double x)
{
    int k, n;
    double w, y;
    n = x < 1.5 ? -((int) (2.5 - x)) : (int) (x - 1.5);
    w = x - (n + 2);
    y = ( (((((((((( -1.99542863674e-7 * w + 1.337767384067e-6) * w -
                 2.591225267689e-6) * w - 1.754539395205e-5) * w +
                 1.45596568617526e-4) * w - 3.60837876648255e-4) * w -
                 0.04329819255744e-4) * w + 0.008023273027855346) * w -
                 0.017645244547851414) * w - 0.233093736421782878) * w -
                 0.422784335098466784) * w + 0.99999999999999999;
    if (n > 0) {
        w = x - 1;
        for (k = 2; k <= n; k++) {
            w *= x - k;
        }
    } else {
```

w = 1;
for (k = 0; k > n; k--) {
    y *= x - k;
}
return w / y;

double one_norm(double argument) {
    // note in our case the argument is always a constant, hence the one norm is always the absolute value
    return fabs(argument);
}

static void mdlInitializeSizes(SimStruct *S) {
    int_T i;
    ssSetNumSFcnParams(S, 0);
    if (ssGetNumSFcnParams(S) != ssGetSFcnParamsCount(S)) {
        return; /* Parameter mismatch reported by the Simulink engine*/
    }
    ssSetNumInputPorts(S, 5); // we have 5 inputs
    for (i=0;i<5;i++){
        ssSetInputPortWidth(S, i, 1); // each port is just one wide
        ssSetInputPortDirectFeedThrough(S, i, 1);
    }
    ssSetNumOutputPorts(S, 1); // need just one output port
    ssSetOutputPortWidth(S, 0, 1); // width of output port is one
/* Take care when specifying exception free code - see sfuntmpl.doc */
ssSetOptions(S, SS_OPTION_EXCEPTION_FREE_CODE);
}
static void mdlInitializeSampleTimes(SimStruct *S)
{
    ssSetSampleTime(S, 0, INHERITED_SAMPLE_TIME);
    ssSetOffsetTime(S, 0, 0.0);
}
static void mdlOutputs(SimStruct *S, int_T tid)
{
    int_T i;
    double alpha, beta, wye, n, eps;
    int_T nInputPorts = ssGetNumInputPorts(S);
    real_T *y = ssGetOutputPortRealSignal(S, 0);
    double f = 0;
    double fa = 1;
    double j = 0;
    for (i = 0; i < nInputPorts; i++) {
        InputRealPtrsType uPtrs = ssGetInputPortRealSignalPtrs(S, i);
        switch (i){
            case 0:
                alpha = **uPtrs[0];
                break;
            case 1:
                beta = **uPtrs[0];
                break;
            case 2:
                wye = **uPtrs[0];
                break;
        } // since we already know that the number of inputs is 5
    }
}
case 3:
    n=*uPtrs[0];
    break;

case 4:
    eps=*uPtrs[0];
    break;

default:
    return;

//*y= alpha + beta + wye + n + eps; - test line
while (one_norm(fa) >= eps) {
    fa = ((dgamma(j + n + 1)/dgamma(j + 1)) / dgamma(alpha * j + alpha * n + beta)) * pow(wye, j);
    f = f + fa; j = j + 1;
}

//*y = dgamma(5); - test line
*y = f;

static void mdlTerminate(SimStruct *S) {}

/* Simulink/Real-Time Workshop Interface */
#ifdef MATLAB_MEX_FILE /* Is this file being compiled as a MEX-file?
*/
#include "simulink.c" /* MEX-file interface mechanism */
#else
#include "cg_sfun.h" /* Code generation registration function */
#endif
Appendix I

Fractional Order System Modeling: Code

```matlab
%% example usage
%% [x,fval,exif] = fminsearch(@() fmdhf(x,wh,wl,N),[1 1 1 1]);
function f = fmdhf(v,wh,wl,N)
    %v=k1 a k2 b;
    % all available fractional orders, you may accept these as an
    % argument or modify the code manually
    alpha10 = 0.6495;
    alpha9 = 0.3850;
    alpha8 = 0.7460;
    alpha7 = 0.8680;
    alpha6 = 0.9190;
    alpha5 = 0.8630;
    alpha4 = 0.9440;
    alpha3 = 0.7900;
    alpha2 = 0.7425;
    alpha1 = 0.5645;
    alpha0 = 0.5175;
    delta_w=(wh-wl)/N;
    sum_of_diff_sq=0;
    for j=1:N
        wk=j*(delta_w)+wl;
        sv=sqrt(-1)*wk;
        %% enter plant transfer function here
        num= 0.02467*sv^2 + 0.006552*sv + 0.0002862;
        den=sv^3 + 0.4129*sv^2 + 0.04732*sv + 0.0009909;
        trx=num/den;
```
sum = trx;
approx_term = (v(1)/(s v^ alpha(1)+v(2)))+(v(3)/(s v^ alpha(2)+v(4)));

% change above line manually to increase terms or take in % terms as argument

diff = sum - approx_term;
sum_of_diff_sq = sum_of_diff_sq + (diff^2);

end

f = abs(sum_of_diff_sq);
end
Appendix J

Approximation of an FO-PI Controller: Code

function f = fracsplit(Ks, w1, w0, N, m, Kpo, Kio, ao, ais)
    if ((length(ais) < m) || (length(Ks) < (2 * m)))
        display('Number of orders do not match actually supplied orders');
        f = inf;
    else
        delta_w = (w1-w0)/N;
        sum_of_diff_sq = 0;
        for j = 1:N
            wk = j*(delta_w)+w0;
            sum = 0;
            for i = 1:m
                Term = Ks(i+count)*Ks(i+count+1)+Ks(i+count+1)/((sqrt(-1)*wk)^ais(i));
                count = count + 1;
                sum = sum + Term;
            end
            diff = sum - (Kpo + Kio/((sqrt(-1)*wk)^ao));
            sum_of_diff_sq = sum_of_diff_sq + (diff^2);
        end
        f = sum_of_diff_sq;
    end
end
Appendix K

Plasma Position Estimation: Code

K.1 Parameter Setter Script

1 global stoptime;
2 global d1;
3 global d2;
4 global d3;
5 global d4;
6 global d5;
7 global d6;
8 global d7;
9 global d8;
10 global ScopeData4;
11 global ScopeData5;
12 global ScopeData8;
13 global ScopeData9;
14 global ord;
15 global f;
16 %These are global variables used by the position calculator do not
17 %change or delete
K.2 Position Estimator Model File

Fig. K.1: Simulink® model for plasma position estimation.
K.3 Plasma Position Calculator

1 \%\%SYNTAX :: [output]=plasmaposition(shot1,shot2,shot3,shot4,shot5,
   filter_order,pass_band,estimation_mode,suppress)
2 \%\%usage :: posparam;[y]=plasmaposition
   (3601,3602,3603,3604,3605,1,8000,1,0);
3 \%\%[output]=[t_horz horz_disp t_vert ver_disp h_ratio v_ratio];
   follow format to extract
4 \%\%TIP :: Best performance obtained with filter_order=1 and pass_band
   =8000. If estimation_mode = 1 then it uses a ratio of 1 as center
   and ratio of 5.667 as wall, however if estimation_mode = 0 then
   it uses a ratio of 1 as center but a ratio of 2 as wall (which
   should not be used).
5 \%\%====================================================================
6 \%\%NOTE : always use as posparam;plasmaposition(...arguments...);
7 \%\%posparam sets the variables needed by the simulink files
8 \%\%DO NOT be alarmed by warning messages for divide by zero errors,
   they occur when the signal is initially zero before the plasma
   starts, however it has been handled internally in the simulink
   files.
9 function [out_data]=plasmaposition(name1,name2,name3,name4,name5,
   order,freq,Md,suppress)
10 global stoptime;
11 global d1;
12 global d2;
13 global d3;
14 global d4;
15 global ScopeData4
16 global ScopeData5
17 global ScopeData8
18 global ScopeData9
19 global ord;
20 global f;
ord=order;
f=freq;

% read files
% read_6133 script developed by Dr. Singh
[head data1_comp]=read_6133(strcat(num2str(name1),'_c.txt'));
[head data2_comp]=read_6133(strcat(num2str(name2),'_c.txt'));
[head data3_comp]=read_6133(strcat(num2str(name3),'_c.txt'));
[head data4_comp]=read_6133(strcat(num2str(name4),'_c.txt'));
[head data5_comp]=read_6133(strcat(num2str(name5),'_c.txt'));

% extract time

% detrend plasma current and remove biases
Ipavg=ipvalidator(name1,name2,name3,name4,name5,1);

% get mean position coil data across 5 shots

C1avg=mean([data1_comp(:,2)';data2_comp(:,2)';data3_comp(:,2)';
             data4_comp(:,2)';data5_comp(:,2)';])
C2avg=mean([data1_comp(:,3)';data2_comp(:,3)';data3_comp(:,3)';
             data4_comp(:,3)';data5_comp(:,3)';])
C3avg=mean([data1_comp(:,4)';data2_comp(:,4)';data3_comp(:,4)';
             data4_comp(:,4)';data5_comp(:,4)';])
C4avg=mean([data1_comp(:,5)';data2_comp(:,5)';data3_comp(:,5)';
             data4_comp(:,5)';data5_comp(:,5)';])

% prepare data for simulink

d1=[t C1avg'];
d2=[t C2avg'];
d3=[t C3avg'];
d4=[t C4avg'];
stoptime=t(end);
mode = Md;

% assign zero for simple wall estimation, 1
% for accurate wall estimation
% sim('filterdiffpos.mdl');
49  \% no sign change position estimator by ratio
50  \% sim('filterposns.mdl');
51  \% no sign change position estimator by ratio
52  \% sim('filterpos.mdl');
53  \% position estimator by ratio - incorporating sign change
54  if mode ==1
55     sim('filterposns_final.mdl');
56  \% no sign change accurate wall detection
57  else
58     sim('filterpos.mdl');
59  \% position estimator by ratio - incorporating no sign change
60  end
61  bound=3.5*ones(length(t),1);
62  zeroline=0*ones(length(t),1);
63  if suppress==0  \% plots suppressed
64     figure
65     if mode==1
66         plot(t,c1avg./c2avg,'g',ScopeData4.time,ScopeData4.signals.
67             values,'r');
68     else
69         plot(t,c2avg./c1avg,'g',ScopeData4.time,ScopeData4.signals.
70             values,'r');
71     end
72     axis([0 t(end) -5 5]);
73     xlabel('Time - seconds');ylabel('Pickup ratio');
74     title('Vertical position coils');
75     figure
76     if mode==1
77         plot(t,c3avg./c4avg,'g',ScopeData5.time,ScopeData5.signals.
78             values,'r');
79     else
80         \% no sign change accurate wall detection
plot(t, c4avg./c3avg, 'g', ScopeData5.time, ScopeData5.signals.values, 'r');
end
axis([0 t(end) -5 5]);
xlabel('Time - seconds'); ylabel('Pickup ratio');
title('Horizontal position coils');
% above plots for filter data performance
% lower plots for displacement
figure
plot(t, zeroline, 'k:'); hold on;
plot(ScopeData8.time, ScopeData8.signals.values, 'r');
hold on
plot(t, bound, 'b--', t, -bound, 'k--');
legend('Zero', 'Displacement', 'Upper bound', 'Lower bound');
axis([0 t(end) -5 5]);
xlabel('Time - seconds'); ylabel('Centimetres from centre');
title('Approximate Vertical displacement');
figure
plot(t, zeroline, 'k:'); hold on;
plot(ScopeData9.time, ScopeData9.signals.values, 'r');
hold on
plot(t, bound, 'b--', t, -bound, 'k--');
legend('Zero', 'Displacement', 'Inner bound', 'Outer bound');
axis([0 t(end) -5 5]);
xlabel('Time - seconds'); ylabel('Centimetres from centre');
title('Approximate Horizontal displacement');
% autosaver enable if needed
% save('C:\Files\dat\pos.mat', 'ScopeData8', 'ScopeData9');
figure
plot(t, Ipavg);
xlabel('Time - seconds'); ylabel('Amperes');
title('Plasma current');
grid on;
end

data=[ScopeData9. time, ScopeData9. signals. values, ScopeData8. time, ScopeData8. signals. values, ScopeData5. signals. values, ScopeData4. signals. values];

% Scope9 Hpos
% Scope8 Vpos
% Scope5 Hratio
% Scope4 Vratio
end

K.4 Plasma Current Averaging and Validation

function ret=ipvalidator(name1,name2,name3,name4,name5,asfunction)
[head data1]=read_6133(strcat(num2str(name1),'.txt'));
t=data1(:,1); %time
ip_new_mean=Ip_cal_mean(name1,name2,name3,name4,name5);
cval=Ip_compensator(ip_new_mean)
ip_new_mean(cval)
if asfunction==0
    %method mode for display purposes
    figure
    plot(t,ip_new_mean);
    hold on
    plot(t(1:length(t)-1,:),diff(ip_new_mean),'r');
    hold on
    plot(t,ip_new_mean-ip_new_mean(cval),'m');
title(strcat('Shot nos. ',num2str(name1),' ',num2str(name2),'
           ',num2str(name3),'
           ',num2str(name4),'
           ',num2str(name5)));
    legend('Numerically integrated Ip', 'Approximate derivative',
           'Compensated Ip');
K.5 Plasma Current De-trending

function \[ ret\_number\]=Ip\_compensator(ipval)
  d=diff(ipval);
  for i=1:length(d)
    if ((d(i) <=0.005)\&\&(d(i-1)<0))
      \%hitting end of plasma
      display('time found');
      ret\_number=(i+1);
    end
  end
end

K.6 Plasma Position Estimation Using Mirnov Coils

%%SYNTAX :: \[output\]=plasmapositionmirnov(\text{shot1,shot2,shot3,shot4, shot5,filter\_order,pass\_band,estimation\_mode,suppress,cvolt, rotateby, time})

%%[output]=[t\_vert vert\_disp t\_h45 h45\_disp t\_horz horz\_disp t\_h135 h135\_disp]; follow format to extract

%%TIP:: Best performance obtained with filter\_order=1 and pass\_band =8000 if estimation\_mode = 1 then it uses a ratio of 1 as center and ratio of 5.667 as wall, however if estimation\_mode = 0 then it uses a ratio of 1 as center but a ratio of 2 as wall (which should not be used); cvolt is the compensation voltage valid
values are 123, 90, 95, 100 volts, suppress=1 does not show plots, use 0 to see plots, rotate by is the angle (degrees) to rotate all coils by, rotation is used to fit observed data to practical coil arrangement on the tokamak, time is the instant in milliseconds for the final scatter plot, if make_a_movie=1 then it makes a movie, use 0 for normal operation

%========================================================================
%NOTE : always use as
%posparam; plasmapositionmirnov(...arguments...);
%posparam sets the variables needed by the simulink files
%DO NOT be alarmed by warning messages for divide by zero errors, they occur when the signal is initially zero before the plasma starts, however it has been handled internally in the simulink files.

function [out_data]=plasmapositionmirnov(name1,name2,name3,name4,name5,order,freq,Md,suppress, cvolt,rot,instant,make_a_movie)

global stoptime;
global d1;
global d2;
global d3;
global d4;
global d5;
global d6;
global d7;
global d8;
global ScopeData4
global ScopeData5
global ScopeData8
global ScopeData9
global ord;
global f;
ord=order;
f = freq;
% read files
[head data1_comp] = read_6133(strcat(num2str(name1),'_c.txt'));
[head data2_comp] = read_6133(strcat(num2str(name2),'_c.txt'));
[head data3_comp] = read_6133(strcat(num2str(name3),'_c.txt'));
[head data4_comp] = read_6133(strcat(num2str(name4),'_c.txt'));
[head data5_comp] = read_6133(strcat(num2str(name5),'_c.txt'));
% extract time
t = data1_comp(: ,1);
Ipavg = ipvalidator(name1,name2,name3,name4,name5,1);
% get mean position coil data across 5 shots
[head data1] = read_6133(strcat(num2str(name1),'_d.txt'));
[head data2] = read_6133(strcat(num2str(name2),'_d.txt'));
[head data3] = read_6133(strcat(num2str(name3),'_d.txt'));
[head data4] = read_6133(strcat(num2str(name4),'_d.txt'));
[head data5] = read_6133(strcat(num2str(name5),'_d.txt'));
% for mirnov coil data 1 = time, coils are 2,3,4,5,6,7,8,9
[head id1] = read_6133(strcat(num2str(name1),'_b.txt'));
[head id2] = read_6133(strcat(num2str(name2),'_b.txt'));
[head id3] = read_6133(strcat(num2str(name3),'_b.txt'));
[head id4] = read_6133(strcat(num2str(name4),'_b.txt'));
[head id5] = read_6133(strcat(num2str(name5),'_b.txt'));
compensationvoltage = cvolt;
%Mirnov coil calibration factors obtained from
% in lab experimental data
factorvector = [448.9290;419.4897;449.5197;426.4592;486.5989;_
525.5160;493.7818;402.9175];
% following if-loops read files for shots when the
%mirnov coils where calibrated
if compensationvoltage==90
    [head c90a] = read_6133('3661_d.txt');
    [head c90b] = read_6133('3662_d.txt');
c90comp_coil1=mean([c90a(:,2)';c90b(:,2)']);
c90comp_coil2=mean([c90a(:,3)';c90b(:,3)']);
c90comp_coil3=mean([c90a(:,4)';c90b(:,4)']);
c90comp_coil4=mean([c90a(:,5)';c90b(:,5)']);
c90comp_coil5=mean([c90a(:,6)';c90b(:,6)']);
c90comp_coil6=mean([c90a(:,7)';c90b(:,7)']);
c90comp_coil7=mean([c90a(:,8)';c90b(:,8)']);
c90comp_coil8=mean([c90a(:,9)';c90b(:,9)']);
c90_comp_mat=[c90comp_coil1',c90comp_coil2',c90comp_coil3',
c90comp_coil4',c90comp_coil5',c90comp_coil6',c90comp_coil7',
c90comp_coil8'];
data1_comp=data1(:,2:9)-c90_comp_mat;
data2_comp=data2(:,2:9)-c90_comp_mat;
data3_comp=data3(:,2:9)-c90_comp_mat;
data4_comp=data4(:,2:9)-c90_comp_mat;
data5_comp=data5(:,2:9)-c90_comp_mat;

elseif compensationvoltage==95
[head c95a]=read_6133('3659_d.txt');
[head c95b]=read_6133('3660_d.txt');
c95comp_coil1=mean([c95a(:,2)';c95b(:,2)']);
c95comp_coil2=mean([c95a(:,3)';c95b(:,3)']);
c95comp_coil3=mean([c95a(:,4)';c95b(:,4)']);
c95comp_coil4=mean([c95a(:,5)';c95b(:,5)']);
c95comp_coil5=mean([c95a(:,6)';c95b(:,6)']);
c95comp_coil6=mean([c95a(:,7)';c95b(:,7)']);
c95comp_coil7=mean([c95a(:,8)';c95b(:,8)']);
c95comp_coil8=mean([c95a(:,9)';c95b(:,9)']);
c95_comp_mat=[c95comp_coil1',c95comp_coil2',c95comp_coil3',
c95comp_coil4',c95comp_coil5',c95comp_coil6',c95comp_coil7',
c95comp_coil8'];
data1_comp=data1(:,2:9)-c95_comp_mat;
data2_comp=data2(:,2:9)-c95_comp_mat;
```plaintext
data3_comp = data3(:,2:9) - c95_comp_mat;
data4_comp = data4(:,2:9) - c95_comp_mat;
data5_comp = data5(:,2:9) - c95_comp_mat;

else if compensationvoltage == 100
    [head c100a] = read_6133('3657_d.txt');
    [head c100b] = read_6133('3658_d.txt');
    c100comp_coil1 = mean([c100a(:,2)'; c100b(:,2)']);
    c100comp_coil2 = mean([c100a(:,3)'; c100b(:,3)']);
    c100comp_coil3 = mean([c100a(:,4)'; c100b(:,4)']);
    c100comp_coil4 = mean([c100a(:,5)'; c100b(:,5)']);
    c100comp_coil5 = mean([c100a(:,6)'; c100b(:,6)']);
    c100comp_coil6 = mean([c100a(:,7)'; c100b(:,7)']);
    c100comp_coil7 = mean([c100a(:,8)'; c100b(:,8)']);
    c100comp_coil8 = mean([c100a(:,9)'; c100b(:,9)']);
    c100_comp_mat = [c100comp_coil1', c100comp_coil2', c100comp_coil3',
                     c100comp_coil4', c100comp_coil5', c100comp_coil6',
                     c100comp_coil7', c100comp_coil8'];
data1_comp = data1(:,2:9) - c100_comp_mat;
data2_comp = data2(:,2:9) - c100_comp_mat;
data3_comp = data3(:,2:9) - c100_comp_mat;
data4_comp = data4(:,2:9) - c100_comp_mat;
data5_comp = data5(:,2:9) - c100_comp_mat;

else if compensationvoltage == 123
    [head c123a] = read_6133('3655_d.txt');
    [head c123b] = read_6133('3656_d.txt');
    c123comp_coil1 = mean([c123a(:,2)'; c123b(:,2)']);
    c123comp_coil2 = mean([c123a(:,3)'; c123b(:,3)']);
    c123comp_coil3 = mean([c123a(:,4)'; c123b(:,4)']);
    c123comp_coil4 = mean([c123a(:,5)'; c123b(:,5)']);
    c123comp_coil5 = mean([c123a(:,6)'; c123b(:,6)']);
    c123comp_coil6 = mean([c123a(:,7)'; c123b(:,7)']);
    c123comp_coil7 = mean([c123a(:,8)'; c123b(:,8)']);
```

c123comp_coil8 = mean([c123a(:,9)'; c123b(:,9)']);
c123_comp_mat = [c123comp_coil1', c123comp_coil2', c123comp_coil3', c123comp_coil4', c123comp_coil5', c123comp_coil6', c123comp_coil7', c123comp_coil8'];
data1_comp = data1(:,2:9) - c123_comp_mat;
data2_comp = data2(:,2:9) - c123_comp_mat;
data3_comp = data3(:,2:9) - c123_comp_mat;
data4_comp = data4(:,2:9) - c123_comp_mat;
data5_comp = data5(:,2:9) - c123_comp_mat;
end
c1avg = mean([data1_comp(:,1)'; data2_comp(:,1)'; data3_comp(:,1)'; data4_comp(:,1)'; data5_comp(:,1)';]);
c2avg = mean([data1_comp(:,2)'; data2_comp(:,2)'; data3_comp(:,2)'; data4_comp(:,2)'; data5_comp(:,2)';]);
c3avg = mean([data1_comp(:,3)'; data2_comp(:,3)'; data3_comp(:,3)'; data4_comp(:,3)'; data5_comp(:,3)';]);
c4avg = mean([data1_comp(:,4)'; data2_comp(:,4)'; data3_comp(:,4)'; data4_comp(:,4)'; data5_comp(:,4)';]);
c5avg = mean([data1_comp(:,5)'; data2_comp(:,5)'; data3_comp(:,5)'; data4_comp(:,5)'; data5_comp(:,5)';]);
c6avg = mean([data1_comp(:,6)'; data2_comp(:,6)'; data3_comp(:,6)'; data4_comp(:,6)'; data5_comp(:,6)';]);
c7avg = mean([data1_comp(:,7)'; data2_comp(:,7)'; data3_comp(:,7)'; data4_comp(:,7)'; data5_comp(:,7)';]);
c8avg = mean([data1_comp(:,8)'; data2_comp(:,8)'; data3_comp(:,8)'; data4_comp(:,8)'; data5_comp(:,8)';]);
c1int = cumtrapz(t, c1avg) * factorvector(1);
c2int = cumtrapz(t, c2avg) * factorvector(2);
c3int = cumtrapz(t, c3avg) * factorvector(3);
c4int = cumtrapz(t, c4avg) * factorvector(4);
c5int = cumtrapz(t, c5avg) * factorvector(5);
c6int = cumtrapz(t, c6avg) * factorvector(6);
c7int = cumtrapz(t, c7avg) * factorvector(7);
c8int = cumtrapz(t, c8avg) * factorvector(8);
d1 = [t c1int ‘];
d2 = [t c5int ‘];
d3 = [t c2int ‘];
d4 = [t c6int ‘];
stoptime = t(end);
mode = Md;
sim(’filterposns_final.mdl’); % no sign change accurate wall
    detection
bound = 3.5 * ones(length(t), 1);
zeroline = 0 * ones(length(t), 1);
if suppress == 0 % plots suppressed
    figure
    if mode == 1
        plot(t, c1int ./ c5int, ’g’, ScopeData4.time, ScopeData4.signals.
            values, ’r’);
    else
        plot(t, c5int ./ c1int, ’g’, ScopeData4.time, ScopeData4.signals.
            values, ’r’);
    end
    axis([0 t(end) -5 5]);
xlabel(’Time – seconds’); ylabel(’Pickup ratio’);
title(’Vertical position coils’);
figure
if mode == 1
    plot(t, c2int ./ c6int, ’g’, ScopeData5.time, ScopeData5.signals.
        values, ’r’);
else
    plot(t, c6int ./ c2int, ’g’, ScopeData5.time, ScopeData5.signals.
        values, ’r’);

end
axis([0 t(end) -5 5]);
xlabel('Time - seconds');ylabel('Pickup ratio');
title('Position coils at 135 degrees to horizontal');

% lower plots for displacement
figure
plot(t,zeroline,'k:');hold on;
plot(ScopeData8.time,ScopeData8.signals.values,'r');
hold on
plot(t,bound,'b--',t,-bound,'k- -');
legend('Zero','Displacement','Upper bound','Lower bound');
axis([0 t(end) -5 5]);
xlabel('Time - seconds');ylabel('Centimetres from centre');
title('Approximate Vertical displacement');
tv=ScopeData8.time;
v=ScopeData8.signals.values;

figure
plot(t,zeroline,'k:');hold on;
plot(ScopeData9.time,ScopeData9.signals.values,'r');
hold on
plot(t,bound,'b--',t,-bound,'k- -');
legend('Zero','Displacement','Inner bound','Outer bound');
axis([0 t(end) -5 5]);
xlabel('Time - seconds');ylabel('Centimetres from centre');
title('Approximate displacement at 135 degrees to horizontal');
t135=ScopeData9.time;
h135=ScopeData9.signals.values;

end

d1=[t c3int'];
197  d2=[t c7int’];
198  d3=[t c4int’];
199  d4=[t c8int’];
200  sim('filterposns_final.mdl'); %no sign change accurate wall detection
201  if suppress==0 %plots suppressed
202      figure
203      if mode==1
204         plot(t,c3int./c7int,’g’,ScopeData4.time,ScopeData4.signals.
205             values,’r’);
206      else
207         plot(t,c7int./c3int,’g’,ScopeData4.time,ScopeData4.signals.
208             values,’r’);
209      end
210  axis([0 t(end) -5 5]);
211  xlabel(’Time - seconds’);ylabel(’Pickup ratio’);
212  title(’Horizontal position coils’);
213  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
214  figure
215  if mode==1
216      plot(t,c4int./c8int,’g’,ScopeData5.time,ScopeData5.signals.
217             values,’r’);
218  else
219      plot(t,c8int./c4int,’g’,ScopeData5.time,ScopeData5.signals.
220             values,’r’);
221  end
222  axis([0 t(end) -5 5]);
223  xlabel(’Time - seconds’);ylabel(’Pickup ratio’);
224  title(’Position coils at 45 degrees to horizontal’);
225  %above plots for filter data performance
226  figure
224       plot(t,zeroline,'k:'); hold on;
225       plot(ScopeData8.time,ScopeData8.signals.values,'r');
226       hold on
227       plot(t,bound,'b--',t,-bound,'k--');
228       legend('Zero','Displacement','Inner bound','Outer bound');
229       axis([0 t(end) -5 5]);
230       xlabel('Time - seconds'); ylabel('Centimetres from centre');
231       title('Approximate Horizontal displacement');
232       th=ScopeData8.time;
233       h=ScopeData8.signals.values;
234
235       figure
236       plot(t,zeroline,'k:'); hold on;
237       plot(ScopeData9.time,ScopeData9.signals.values,'r');
238       hold on
239       plot(t,bound,'b--',t,-bound,'k--');
240       legend('Zero','Displacement','Inner bound','Outer bound');
241       axis([0 t(end) -5 5]);
242       xlabel('Time - seconds'); ylabel('Centimetres from centre');
243       title('Approximate displacement at 45 degrees to horizontal');
244       t45=ScopeData9.time;
245       h45=ScopeData9.signals.values;
246
247       figure
248       plot(t,Ipavg);
249       xlabel('Time - seconds'); ylabel('Amperes');
250       title('Plasma current');
251       grid on;
252       end
253       out_data=[tv,v,t45,h45,th,h,t135,h135];
254       %set this to be angle of rotation as per desired
255       rotation_angle=rot; %degrees
timedes = instant * 1e-3;
index = timedes / 5e-7;
plot(v(index) * cos(deg2rad(90 + rotation_angle)), v(index) * sin(deg2rad(90 + rotation_angle)), 'b*');
hold on;
plot(h45(index) * (cos(deg2rad(45 + rotation_angle))), h45(index) * (sin(deg2rad(45 + rotation_angle))), 'r*');
hold on;
plot(h(index) * cos(deg2rad(0 + rotation_angle)), h(index) * sin(deg2rad(0 + rotation_angle)), 'm*');
hold on;
plot(h135(index) * (cos(deg2rad(135 + rotation_angle))), h135(index) * (sin(deg2rad(135 + rotation_angle))), 'c*');
grid on
axis([-3.5 3.5 -3.5 3.5])
title(strcat('Position scatter plot from mirnov coil data at time = ', num2str(timedes), ' seconds'));
xlabel('Centimetres - X'); ylabel('Centimetres - Y')
if (make_a_movie == 1)
close all
pause(3);
figure;
%// moviemode
v = downsampleby(v, 100);
h45 = downsampleby(h45, 100);
h = downsampleby(h, 100);
h135 = downsampleby(h135, 100);
tvect = downsampleby(tv, 100);
j = 0;
for index = 1: length(v)
j = j + 1;
end
plot(v(index)*cos(deg2rad(90+rotation_angle)),v(index)*sin(deg2rad(90+rotation_angle)),'bo',h45(index)*(cos(deg2rad(45+rotation_angle)))+h45(index)*(sin(deg2rad(45+rotation_angle))),'bo',h(index)*cos(deg2rad(0+rotation_angle)),h(index)*sin(deg2rad(0+rotation_angle)),'mo',h135(index)*(cos(deg2rad(135+rotation_angle)))+h135(index)*(sin(deg2rad(135+rotation_angle))),'co');
grid on;
xlabel('Centimetres - X');ylabel('Centimetres - Y');title(strcat('Position at time t=',num2str(tvect(index)),' seconds'));
axis([-3.5 3.5 -3.5 3.5]);
F(j)=getframe(gcf);
end
movie(F,1);
movie2avi(F,'pos.avi');
display('movie written');
close all;
end

K.7 Vector Downsampler

function res=downsampleby(vect,step)
a=vect;
j=1:step:length(a);
for i=1:length(j)
  b(i)=a(j(i));
end
res=b;
end