Model Predictive Control of Induction Motor Drive with Constraints

Lu Gan

Submitted in fulfilment of the requirements
for the degree of Doctor of Philosophy

August 2014

School of Electrical and Computer Engineering
Royal Melbourne Institute of Technology
Victoria Australia
Declaration

I certify that except where due acknowledgement has been made, the work is that of the author alone; the work has not been submitted previously, in whole or in part, to qualify for any other academic award; the content of the thesis is the result of work which has been carried out since the official commencement date of the approved research program; and, any editorial work, paid or unpaid, carried out by a third party is acknowledged.

(sign here)

(name here)

(date here)
Abstract

Induction motor drives play a significant role in various industrial applications. It is an essential machine that converts the electrical energy to mechanical motion. Due to the widely use in the industrial applications, the control system design of the induction motor drives became more significant in last decades. Meanwhile, Model Predictive Control (MPC) is an optimal control algorithm developed for constrained control of Multi-Input-Multi-Output (MIMO) systems. In the past decade, the research of MPC in power electronics field has become popular due to the development of digital control platforms. In general, there are two categories of MPC methods which are applied to motor drives: the traditional MPC and the finite control set MPC. The former type of MPC has modulation-based implementation, where a cost function is defined based on the prediction of future state as well as the control trajectory within the defined prediction window. Thus, this type of MPC design generally requires sufficient computational time and a Linear Time Invariant (LTI) model. Complementarily, the Finite Control Set (FCS) predictive control method is proposed based on the benefit of finite switching states of the inverter in the induction motor drive. The concept has become increasingly popular in the research field due to its simplicity and robustness. In this thesis, both MPC methods will be investigated and discussed along with own contributions in both theory and applications to induction motor control.

For the modulation-based MPC design, the continuous-time model based predictive control scheme is selected due to its advantages, such as independent to sampling time in the design step, accuracy of physical model and multi-rate sampling implementation. In this thesis, there are two different approaches investigated for this type MPC: centralized and cascaded control structures. On one hand, the centralized MPC is proposed to achieve the induction motor speed control by using a single model predictive controller, the major challenge is found due to the non-linearity of the full order model of the induction motor. Thus, the Gain-Scheduling (GS) technique is proposed for MPC by pre-defining the operating conditions according to different equilibrium points of the system operations. By employing the Direct Field Oriented Control (FOC) concept, the Gain-Scheduled MPC is developed and validated for the Variable Speed Drive (VSD) of induction motor. On the other hand, the cascaded MPC control is proposed to further develop the continuous-time Model Predictive Control of the induction motor drive. Based on the Indirect FOC strategy, two MPC controllers are designed according to inner-loop electrical model and outer-loop mechanical model, respectively. Because of the high gain control strategy of the inner-loop, the influence of the model non-linearity
is eliminated for the inner-loop current control. Furthermore, position control is also investigated for servo drive application by using the cascaded MPC technique. Another major contribution of this thesis is on the advance of theory and applications of Finite Control Set MPC to induction motor control. Here, the original FCS-MPC is investigated in depth from the perspective of feedback control and steady-state compensation. The new analytical results have been obtained in terms of closed-loop feedback control gain and the closed-loop poles of the FCS-MPC system, based on which closed-loop stability is established for the system without constraints. More importantly, integrator has been incorporated in the FCS-MPC system to overcome the steady-state errors existed in the original system. The proposed approach not only maintains the simplicity of the original algorithm, but also improves its robust performance in the presence of parameter uncertainty. In the $\alpha\beta$ reference frame, a resonant controller is derived to overcome steady-state errors of the original FCS-MPC system by following the same thought process as introduced in the $dq$ reference frame. Both simulation and experimental results are used to support the theoretical results. All of proposed control strategies are compared with the conventional PI-based FOC method, the control performances of the experimental results are studied and analysed. Moreover, the robustness analysis of the proposed control methods is investigated by comparing the experimental results based on mismatched model parameter values.
Acknowledgements

Foremost, I would like to express my sincere gratitude to my supervisor Prof. Liuping Wang for the continuous support of my Ph.D study and research, for her patience, motivation, enthusiasm and immense knowledge. Her guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better supervisor and mentor for my Ph.D study.

My sincere thanks also goes to Dr. Peter Graszkiewicz, Dr. Brendan McGrath and Dr. Yong Feng, who provided encouraging and constructive feedback on my project.

I thank my fellow colleagues: Ki Ng, Shan Chai, Dae Yoo and Matt Chen, for the time we were working and discussing together about our research topics, for all the fun we have had together in the last four years. I also want to thank Dr. Yong Feng and Dr. Vu Truong Nguyen for all the helps to me when I started my Ph.D study. In addition, I want to thank RMIT technical officers, Ivan and Sinisa, for their technical supports for this project. Also I thank my friend Shang Lin, for the grammar checking of this thesis.

Last but not the least, I would like to thank my family: my parents Shuliang Gan and Hongxia Ruan, for your infinite love and support throughout my life; my beautiful and dear wife, Miaomiao Quan, for your understanding, care and supporting me to finish this journey.
Contents

Declaration ii

Abstract iii

Acknowledgements v

List of Figures xi

List of Tables xv

Abbreviations xvii

Symbols xix

1 Introduction 1
1.1 Historical Background 1
1.2 Motivation 3
1.3 Literature Review 5
1.3.1 Model Predictive Control 5
1.3.2 Gain Scheduling Method 7
1.3.3 Finite Control Set - Model Predictive Control 8
1.4 Contribution 9
1.4.1 Gain-Scheduled Model Predictive Control 9
1.4.2 Cascaded Model Predictive Control 9
1.4.3 Finite Control Set - Model Predictive Control 10
1.5 Publication 10
1.6 Outline of Thesis 11

2 Modelling of Induction Motor 15
2.1 Introduction 15
2.2 Overview: Structure of Induction Motor 15
2.3 Development of Physical Model Equation 17
2.3.1 Vector Representation of Induction Motor Equation 18
2.3.2 Representation in Stationary $\alpha\beta$ Reference Frame 20
2.3.3 Representation in Stationary $dq$ Reference Frame 21
## 3 Centralized Model Predictive Control

3.1 Introduction ........................................... 31

3.2 Continuous-time MPC ................................ 32
   3.2.1 Augmented Model .................................. 33
   3.2.2 Laguerre Function ............................... 34
   3.2.3 Optimal Control Design without Constraints .... 36
   3.2.4 Exponential Data Weighting ........................ 41
   3.2.5 CMPC with Constraints ............................ 43
   3.2.6 State Feedback Observer .......................... 45
   3.2.7 Simulation and Experimental Results ............. 46

3.3 Gain-Scheduled Continuous-time MPC .............. 50
   3.3.1 Introduction ...................................... 50
   3.3.2 Revised Approach of Constraints Implementation ... 54
   3.3.3 Gain-Scheduled Predictive Controller ............ 57
   3.3.4 Simulation and Experimental Results ............. 60

3.4 Summary .............................................. 64

## 4 Cascaded Model Predictive Control

4.1 Introduction ........................................... 65

4.2 Slip Estimation ....................................... 65

4.3 Cascaded MPC Speed Control ......................... 67
   4.3.1 Inner-loop Current Control ....................... 68
   4.3.2 Outer-loop Speed Control ......................... 71

4.4 Cascaded MPC Position Control ...................... 74
   4.4.1 Predictive Position Control ....................... 74
   4.4.2 Non-Minimal State-Space MPC Control .......... 77

4.5 Summary .............................................. 80

## 5 Finite Control Set - Model Predictive Control

5.1 Background Study .................................... 81

5.2 Study of Original FCS-MPC .......................... 85

5.3 Revised Finite Control Set Method .................. 88
   5.3.1 Analysis of Finite Control Set Method .......... 88
   5.3.2 Feedback Controller Gain Design ................. 91
   5.3.3 Constrained Optimal Control Design ............. 92

5.4 Revised FCS-MPC with Integral Action .............. 96
   5.4.1 Calculation of Integral Gain ..................... 98
   5.4.2 Derivation of I-FCS Controller without Constraints 100
   5.4.3 Optimal Control Design with Constraints .......... 104
5.4.4 Implementation of I-FCS Controller .......................... 106
5.4.5 Experimental Results and Discussions ......................... 107
5.5 FCS-MPC in \( \alpha\beta \)-Coordinates ............................... 110
  5.5.1 Original FCS-MPC Current Control ............................ 110
  5.5.2 Resonant FCS-MPC Current Control Design ...................... 114
  5.5.3 Derivation of Resonant Controller .............................. 118
  5.5.4 Experimental Evaluations ..................................... 121
5.6 Summary .................................................................. 124
  5.6.1 Summary of \( dq \)-frame FCS-MPC ................................. 124
  5.6.2 Summary of \( \alpha\beta \)-frame FCS-MPC .............................. 124
6 PID Control Strategy .................................................. 127
  6.1 Introduction ................................................................ 127
  6.2 Torque Control .......................................................... 127
    6.2.1 Linearization ....................................................... 129
    6.2.2 PI Controllers Design ........................................... 130
    6.2.3 Implementation .................................................... 131
    6.2.4 Timing of Controllers ............................................ 135
    6.2.5 Experimental Results ............................................ 139
  6.3 Speed Control ........................................................... 140
    6.3.1 PI plus PI Control Structure ..................................... 141
    6.3.2 PI plus P Control Structure ..................................... 145
  6.4 Position Control ....................................................... 149
    6.4.1 PID plus PI Control Structure ................................... 150
    6.4.2 PID plus P Control Structure ................................... 153
  6.5 Summary .................................................................. 155
7 Comparison and Discussion of Control Algorithms using Experimental Results .... 157
  7.1 Introduction ............................................................. 157
  7.2 Current Control ........................................................ 157
    7.2.1 Comparison Analysis ............................................. 159
    7.2.2 Robustness Analysis .............................................. 161
  7.3 Velocity Control ....................................................... 166
    7.3.1 GS-MPC & PI-based FOC ....................................... 166
    7.3.2 Cascaded MPC & PI-based FOC ............................... 171
    7.3.3 MPC+FCS & PI+FCS ............................................. 176
  7.4 Position Control ....................................................... 178
  7.5 Summary .................................................................. 183
8 Conclusion ................................................................. 185
  8.1 Summary ................................................................. 185
  8.2 Future Research ....................................................... 187

A Motor Parameters Identification & Validation ........................................ 189
  A.1 Parameter Identification ............................................. 190
### Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A.2 Parameter Validation</td>
<td>191</td>
</tr>
<tr>
<td>B Simulation &amp; Experiment Setup</td>
<td>193</td>
</tr>
<tr>
<td>B.1 Simulation Setup</td>
<td>193</td>
</tr>
<tr>
<td>B.2 Experiment Setup</td>
<td>194</td>
</tr>
<tr>
<td>C Luenberger Observer</td>
<td>197</td>
</tr>
<tr>
<td>Bibliography</td>
<td>199</td>
</tr>
</tbody>
</table>
### List of Figures

1.1 Control methods of Induction Motor .............................................. 2

2.1 AC machine coordinates re-arrangement ........................................ 16

2.2 Equivalent circuit of induction motor ........................................... 18

2.3 Block diagram of induction motor model in $dq$-coordinates .............. 27

2.4 Open loop Simulator in Matlab Simulink ....................................... 29

2.5 Comparison of Model validation results. Key: line(1) current $i_{sd}$; line (2) current $i_{sq}$ ......................................................... 30

3.1 Speed control of induction using centralized MPC controller ............ 31

3.2 Prescribed degree of stability of $-\beta$ ........................................... 41

3.3 Closed-loop eigenvalues for CMPC. Key: (1) without exponential data weighting; (2) $\alpha = 1.2$ and $\beta = 20$ ................................. 43

3.4 Simulation results of speed control using continuous-time MPC. Key: line(1) Actual feedback measurement; line (2) Set-point signal .......... 47

3.5 Simulation results of CMPC with different $\beta$ values. Key: line(1) Set-point signal; line (2) $\beta = 20$; (3) $\beta = 15$; (4) $\beta = 10$; (5) $\beta = 5$ ................. 48

3.6 Experimental results of speed control using continuous-time MPC. Key: line(1) Actual feedback measurement; line (2) Set-point signal .......... 49

3.7 Experimental results of speed control using continuous-time MPC with state observer. Key: line(1) Actual feedback measurement; line (2) Set-point signal .................................................. 50

3.8 Variable speed reference controlled by MPC using single operating condition. Key: line(1) Actual feedback measurement; line (2) Set-point signal 51

3.9 Determinant closed-loop eigenvalues of $K^0_{mpc}$ with $A(t)$ ................ 53

3.10 Approximation of circular area using the area of octagon ................. 55

3.11 Weighting parameters ................................................................. 59

3.12 Simulation result of speed control using Gain-Scheduled MPC. Key: line(1) Actual feedback measurement; line (2) Set-point signal .......... 61

3.13 Experimental result of speed control using Gain-Scheduled MPC. Key: line(1) Actual feedback measurement; line (2) Set-point signal .......... 62

3.14 Experimental result of speed control using Gain-Scheduled MPC with constraints. Key: line(1) Feedback measurement without constraints; line (2) Set-point signal; line(3) Constraints case feedback ....................... 63

4.1 Block diagram of Cascaded MPC speed control ................................ 67

4.2 Closed-loop eigenvalues of MPC current control ............................. 69

4.3 Simulation results of inner-loop MPC control. Key: line(1) Actual feedback; line (2) set-point signal. ........................................... 70
4.4 Experimental results of inner-loop MPC control. Key: line(1) Actual feedback; line (2) set-point signal. ............................................ 70

4.5 Results of Cascaded MPC speed control. Key: line(1) Actual feedback; line (2) set-point signal. ............................................ 73

4.6 Block diagram of Cascaded MPC position control ................. 74

4.7 Results of Cascaded MPC position control. Key: line(1) Actual feedback; line (2) set-point signal. ............................................ 76

4.8 Results of Cascaded MPC position control using Non-Minimal State-Space model. Key: line(1) Actual feedback; line (2) set-point signal. .... 79

5.1 Topology of 2 Level-Voltage-Source-Inverter .......................... 82

5.2 Simulation result of revised FCS-MPC. Key: line(1) Actual feedback; line (2) set-point signal. ............................................ 86

5.3 Illustration of finite control set solution with $\theta_s = 0$ ............................................ 93

5.4 Simulation result of the revised FCS-MPC. Key: line(1) Actual feedback; line (2) set-point signal. ............................................ 95

5.5 Experimental results of the revised FCS-MPC. Key: line(1) Actual feedback; line (2) set-point signal. ............................................ 96

5.6 Implementation of feedback current control using I-FCS. ........ 98

5.7 Feedback current control using I-FCS. ............................................ 103

5.8 Experimental result of I-FCS in $dq$ frame. Key: line(1) Actual feedback; line (2) set-point signal. ............................................ 107

5.9 Step response during transient with different $k_I$ values. Key: line(1) $k_I = 0.05$; line (2) $k_I = 0.15$; line(3) $k_I = 0.25$; line(4) set-point signal .. 108

5.10 Finite Control Set Current Control in $\alpha\beta$ frame ................. 114

5.11 Finite Control Set (FCS) resonant current control in $\alpha\beta$ frame .... 115

5.12 Outer-loop system with inner-loop approximated by a sample of time delay116

5.13 Experimental results of revised FCS-MPC in $\alpha\beta$ coordinates. Key: line(1) Actual feedback; line (2) set-point signal. ............................................ 122

5.14 Experimental results of the resonant FCS in $\alpha\beta$ coordinates. Key: line(1) Actual feedback; line (2) set-point signal. ............................................ 122

5.15 Comparison of open-loop response of the motor speed. ............ 123

6.1 Torque control of induction motor using indirect vector control structure . 128

6.2 Closed-loop control of $i_{sd}$ stator current .......................... 130

6.3 Nonlinear compensation for $d$-axis current controller .............. 132

6.4 Nonlinear compensation for $q$-axis current controller .............. 132

6.5 Simulation results of torque control using PI controllers. Key: line(1) Actual feedback measurement; line (2) Set-point signal .................. 135

6.6 One-degree of freedom control system structure ....................... 135

6.7 Bode Plots for current control. Key: line(1) $\gamma_i = 0.5$; line(2) $\gamma_i = 0.8$; line(3) $\gamma_i = 0.9$; line(4) $\gamma_i = 0.95$ .................. 137

6.8 Nyquist plot of current control loop. Key: line(1) $\gamma_i = 0.5$; line(2) $\gamma_i = 0.8$; line(3) $\gamma_i = 0.9$; line(4) $\gamma_i = 0.95$ .................. 138

6.9 Root-loci for PI control of current system $(0.3 \leq \gamma_i \leq 0.98)$ ........ 138

6.10 Experimental results of torque control with different $\gamma_i$ values. Key: line(1) Actual feedback measurement; line (2) Set-point signal .................. 140

6.11 Velocity control of induction motor using indirect vector control structure 141
6.12 Experimental results of the PI+PI Speed control with tuning. Key: line(1) $\gamma_{\omega} = 0.9$; line(2) $\gamma_{\omega} = 0.95$; line(3) $\gamma_{\omega} = 0.98$; line(4) set-point signal. .................................................. 144

6.13 Experimental results of Speed control using PI controllers. Key: line(1) Actual feedback measurement; line (2) Set-point signal. .......... 145

6.14 q-axis of cascaded speed control ........................................... 145

6.15 Experimental results of the PI+P Speed control with tuning. Key: line(1) $\gamma_{\omega} = 0.9$; line(2) $\gamma_{\omega} = 0.95$; line(3) $\gamma_{\omega} = 0.98$; line(4) set-point signal. ... 147

6.16 Experimental results of Speed control using PI+P controllers. Key: line(1) Actual feedback measurement; line (2) Set-point signal. ........ 148

6.17 Position control of induction motor using indirect vector control structure ($D + F$ presents the derivative term of controller with embedded filter) . 149

6.18 Comparison of position control using PID+PI. Key: line(1) $\gamma_{\theta} = 0.95$; line (2) $\gamma_{\theta} = 0.98$; line (3) $\gamma_{\theta} = 0.99$; line (4) Set-point signal. .... 152

6.19 Experimental of inner-loop control using $\gamma_{\theta} = 0.99$. Key: line(1) Actual measurement; line (2) Set-point signal. .................. 152

6.20 Cascade feedback control of angular position of induction motor ... 153

6.21 Comparison of position control using PID+PI. Key: line(1) $\gamma_{\theta} = 0.95$; line (2) $\gamma_{\theta} = 0.98$; line (3) $\gamma_{\theta} = 0.99$; line (4) Set-point signal. .... 154

6.22 Inner-loop current control comparison. Key: line(1) Actual measurement; line (2) Set-point signal. .......................... 155

7.1 Comparison of control current methods. Key: line (1) Actual measurement; line (2) Set-point signal. ............................. 158

7.2 Speed Measurement for current control comparison. Key: line(1) MPC; line(2) FCS; line(3) PI. ........................................... 160

7.3 FFT analysis of the phase current. ........................................ 161

7.4 Comparison of current control methods with 0.5$L_h$. Key: line (1) Actual measurement; line (2) Set-point signal. .................. 162

7.5 Control signals of MPC and PI current controllers. ..................... 163

7.6 Speed Measurement for current control comparison with 0.5$L_h$. Key: line(1) MPC; line(2) FCS; line(3) PI. .......................... 164

7.7 Comparison of current control methods with 2$L_h$. Key: line (1) Actual measurement; line (2) Set-point signal. .................. 165

7.8 Speed Measurement for current control comparison with 2$L_h$. Key: line(1) MPC; line(2) FCS; line(3) PI. .......................... 166

7.9 Speed control comparison between GS-MPC and PI. Key: blue line: GS-MPC; black line: PI FOC; red dash: Set-point signal. ............ 168

7.10 Robustness analysis of GS-MPC. Key: line(1): set-point signal; line(2): actual measurement. ............................................. 169


7.13 Robustness analysis of the cascaded MPC method. Key: line(1): actual measurement; line(2): set-point signal. ........................ 175

7.14 Speed control comparison between MPC+FCS and PI+FCS using step reference. Key: line(1): actual measurement; line(2): set-point signal. .... 177
7.15 Speed control comparison between MPC+FCS and PI+FCS using ramp reference. Key: line(1): actual measurement; line(2): set-point signal. \ldots 178

7.16 Position control comparison between cascaded MPC and PID. Key: line(1): actual measurement; line(2): set-point signal. \ldots 180

7.17 Robustness analysis of MPC position control. Key: line (1) Actual measurement; line (2) Set-point signal. \ldots 181

7.18 Error signal of MPC position control robustness analysis. Key: line(1) $2J_m$; line (2) $J_m$; line (3) $0.5J_m$. \ldots 182

A.1 Induction motor \ldots 189

A.2 Equivalent circuit of no-load test \ldots 190

A.3 Equivalent circuit of rotor-block test \ldots 191

A.4 Motor parameter validation in Matlab Simulink \ldots 192

B.1 Set-up of induction motor model block \ldots 193

B.2 Open loop Simulator in Matlab Simulink \ldots 194

B.3 Test-bed of the induction motor control experiment \ldots 195
### List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Closed-loop eigenvalues with different $r_k$</td>
<td>41</td>
</tr>
<tr>
<td>3.2</td>
<td>Continuous-time MPC parameter definition</td>
<td>47</td>
</tr>
<tr>
<td>3.3</td>
<td>The values of the coordinates</td>
<td>54</td>
</tr>
<tr>
<td>3.4</td>
<td>Controller design and implementation parameters</td>
<td>61</td>
</tr>
<tr>
<td>4.1</td>
<td>Controller parameters of speed MPC controller</td>
<td>72</td>
</tr>
<tr>
<td>4.2</td>
<td>Controller parameters of position MPC controller</td>
<td>76</td>
</tr>
<tr>
<td>5.1</td>
<td>Switching States of Inverter</td>
<td>82</td>
</tr>
<tr>
<td>5.2</td>
<td>Output voltage of Inverter</td>
<td>82</td>
</tr>
<tr>
<td>5.3</td>
<td>Comparison between original FCS and I-FCS</td>
<td>109</td>
</tr>
<tr>
<td>5.4</td>
<td>Comparison of proposed methods in $\alpha\beta$ frame.</td>
<td>121</td>
</tr>
<tr>
<td>7.1</td>
<td>Controller parameters of current controllers</td>
<td>159</td>
</tr>
<tr>
<td>7.2</td>
<td>Comparison results of current controllers</td>
<td>159</td>
</tr>
<tr>
<td>7.3</td>
<td>Comparison results of current controllers with $0.5L_i$</td>
<td>163</td>
</tr>
<tr>
<td>7.4</td>
<td>Comparison results of current controllers with $2L_i$</td>
<td>164</td>
</tr>
<tr>
<td>7.5</td>
<td>Closed-loop eigenvalues of GS-MPC and PI</td>
<td>167</td>
</tr>
<tr>
<td>7.6</td>
<td>Closed-loop eigenvalues of cascaded MPC and PI</td>
<td>171</td>
</tr>
<tr>
<td>7.7</td>
<td>Closed-loop eigenvalues of cascaded MPC and PID</td>
<td>179</td>
</tr>
<tr>
<td>A.1</td>
<td>Name plant information of the experiment induction motor</td>
<td>192</td>
</tr>
</tbody>
</table>
## Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2L-VSI</td>
<td>Two Level Voltage Source Inverter</td>
</tr>
<tr>
<td>AC</td>
<td>Alternating Current</td>
</tr>
<tr>
<td>ADC</td>
<td>Analogue Digital Converter</td>
</tr>
<tr>
<td>DC</td>
<td>Direct Current</td>
</tr>
<tr>
<td>DMC</td>
<td>Dynamic Matrix Control</td>
</tr>
<tr>
<td>DSC</td>
<td>Direct Self Control</td>
</tr>
<tr>
<td>DSP</td>
<td>Digital Signal Processor</td>
</tr>
<tr>
<td>DTC</td>
<td>Direct Torque Control</td>
</tr>
<tr>
<td>FCS</td>
<td>Finite Control Set</td>
</tr>
<tr>
<td>FCS-MPC</td>
<td>Finite Control Set Model Predictive Control</td>
</tr>
<tr>
<td>FFT</td>
<td>Fast Fourier Transform</td>
</tr>
<tr>
<td>FOC</td>
<td>Field Oriented Control</td>
</tr>
<tr>
<td>FPGA</td>
<td>Field Programmable Gate Arrays</td>
</tr>
<tr>
<td>GPC</td>
<td>Generalized Predictive Control</td>
</tr>
<tr>
<td>GS</td>
<td>Gain Scheduling</td>
</tr>
<tr>
<td>GS-MPC</td>
<td>Gain Scheduled Model Predictive Control</td>
</tr>
<tr>
<td>I-FCS</td>
<td>Integrator Finite Control Set</td>
</tr>
<tr>
<td>IGBT</td>
<td>Insulated Gate Bipolar Transistor</td>
</tr>
<tr>
<td>IM</td>
<td>Induction Motor</td>
</tr>
<tr>
<td>IMP</td>
<td>Internal Model Principle</td>
</tr>
<tr>
<td>LQR</td>
<td>Linear Quadratic Regulator</td>
</tr>
<tr>
<td>LTI</td>
<td>Linear Time Invariant</td>
</tr>
<tr>
<td>MIMO</td>
<td>Multi Input Multi Output</td>
</tr>
<tr>
<td>MPC</td>
<td>Model Predictive Control</td>
</tr>
<tr>
<td>NMSS</td>
<td>Non Minimal State Space</td>
</tr>
<tr>
<td>PCC</td>
<td>Predictive Current Control</td>
</tr>
<tr>
<td>PID</td>
<td>Proportional Integral Derivative</td>
</tr>
<tr>
<td>PMSM</td>
<td>Permanent Magnet Synchronous Motor</td>
</tr>
<tr>
<td>PWM</td>
<td>Pulse Width Modulation</td>
</tr>
<tr>
<td>QP</td>
<td>Quadratic Programming</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>RPM</td>
<td>Revolutions Per Minute</td>
</tr>
<tr>
<td>SISO</td>
<td>Single Input Single Output</td>
</tr>
<tr>
<td>SSE</td>
<td>Steady State Error</td>
</tr>
<tr>
<td>VFC</td>
<td>Variable Frequency Control</td>
</tr>
<tr>
<td>VSD</td>
<td>Variable Speed Drive</td>
</tr>
<tr>
<td>VSI</td>
<td>Voltage Source Inverter</td>
</tr>
</tbody>
</table>
Symbols

\[ A, B, C \quad \text{Augmented model matrices} \]
\[ A_m, B_m, C_m \quad \text{Original model matrices} \]
\[ A_d \quad \text{Desired closed-loop polynomial} \]
\[ f_d \quad \text{Friction coefficient} \quad N \cdot m \cdot s \]
\[ i_{sd}, i_{sq} \quad dq\text{-axis stator current} \quad A \]
\[ i_{sa}, i_{sb} \quad \text{Stationary coordinates stator current} \quad A \]
\[ J \quad \text{Cost function} \]
\[ J_m \quad \text{Moment of inertia} \quad kg \cdot m^2 \]
\[ k_I \quad \text{Integral gain} \]
\[ K_c \quad \text{Proportional control gain} \]
\[ K_{fcs} \quad \text{Feedback controller gain} \]
\[ K_{mpc} \quad \text{Optimal state feedback gain} \]
\[ K_{ob} \quad \text{State observer gain} \]
\[ L_s, L_r \quad \text{Inductance of stator / rotor winding} \quad H \]
\[ L_h \quad \text{Machine mutual inductance} \quad H \]
\[ N_p \quad \text{Laguerre function parameter} \]
\[ q^{-1} \quad \text{Backforward shift operation} \]
\[ Q, R \quad \text{Weighting Matrices} \]
\[ R_s, R_r \quad \text{Resistance of stator / rotor winding} \quad \text{Ohm} \]
\[ S_i \quad \text{Switching state} \]
\[ T_e \quad \text{Motor Torque} \quad N \cdot m \]
\[ T_L \quad \text{Load Torque} \quad N \cdot m \]
\[ T_p \quad \text{Prediction horizon} \]
\[ u_{sd}, u_{sq} \quad dq\text{-axis stator voltage} \quad V \]
\[ u_{sa}, u_{sb} \quad \text{Stationary coordinates stator voltage} \quad V \]
\[ x(t) \quad \text{Augmented state variable} \]
\[ x_m(t) \quad \text{Original state variable} \]
\[ Z_p \quad \text{Number of pole pairs} \]
\[ \alpha, \beta \quad \text{Exponential weighting parameters} \]
\[ \gamma_i, \gamma_\omega, \gamma_\theta \quad \text{Tuning parameter of PI controllers} \]
\[ \delta \quad \text{Switching tolerance} \quad \text{RPM} \]
\[ \eta \quad \text{Control coefficient vector} \]
\[ \theta_e \quad \text{Electrical rotor position} \quad \text{radian} \]
\[ \theta_r \quad \text{Position of motor shaft} \quad \text{radian} \]
\[ \theta_s \quad \text{Position of synchronous flux} \quad \text{radian} \]
\[ \kappa_t \quad \text{Mechanical model constant} \]
\[ \lambda \quad \text{Discrete closed-loop pole} \]
\[ \lambda^l, \lambda^m, \lambda^h \quad \text{Scheduling parameter} \]
Symbols

\( \mu \)  
Disturbance vector

\( \xi \)  
Damping coefficient

\( \tau_D \)  
Derivative control time constant

\( \tau_f \)  
Derivative control filter time constant

\( \tau_I \)  
Integral control time constant

\( \psi_s \)  
Stator flux \( \text{Wb} \)

\( \psi_{rd, rq} \)  
\(dq\)-axis Rotor flux \( \text{Wb} \)

\( \omega_e \)  
Electrical motor speed \( \text{rad/s (or RPM)} \)

\( \omega_r \)  
Mechanical motor speed \( \text{rad/s (or RPM)} \)

\( \omega_s \)  
Speed of synchronous flux \( \text{rad/s (or RPM)} \)

\( \omega_n \)  
Nature frequency \( \text{rad/s} \)

\( \Omega, \Psi \)  
Off-line data matrices
Dedicated to my parents and my wife
for their constant support and unconditional love
Chapter 1

Introduction

Nowadays, the three phase Induction Motor (IM), also known as the asynchronous motor, has been widely used in industrial applications, such as heavy lifting, wind turbine and electrical vehicle. Since the first induction motor invented in 1880s, to date, more than 90% industrial drive applications are using induction motor. However, excluding applications which do not require the control system, most of these motors are not controlled precisely due to various reasons. One main reason in using uncontrolled IM is due to the level of difficulty in controlling it, for example, the non-linearity and complexity of the dynamic model.

1.1 Historical Background

The understanding of the historical background of induction motor control is essential and helpful, moreover, the contributions of this thesis can also be extended and applied in different fields. In the past few decades, for the general objectives, such as improved performance, high energy efficiency and increased safety levels, researchers have been investigating on the IM control from different directions. One of the approaches is from hardware aspect, such as improving the semiconductor switches; multi-level inverters and additional phases of motor winding. Another approach is the control theory development, due to the digital control platforms that are well developed and extensively applied in the power electronics applications. These digital control platforms provide powerful computational capacity for implementing more complex control methodologies. The modern control engineering had significantly developed since the early 20th century, due to the advance of technology. Initially, the control engineering arose when engineers attempted to ensure the productivity in the manufacturing industries by gathering the
Chapter 1. Introduction

Figure 1.1: Control methods of Induction Motor

plant information to plan operations [1]. Meanwhile, the feedback control is developed and analysed for automatic control systems [2].

Afterwards, the control theory is developed and applied in many different fields, such as process control, chemical control and power electronics. In the early days, the induction motor speed was adjusted by using the silicon controlled rectifier back in 1960s [3], after that, the variable-frequency $v/f$ control was investigated and even today still commonly used for applications with low performance requirements. The high performance control of induction motor drive was not found until the Field Oriented Control (FOC) become the industry standard for AC motor drive, since it transforms the AC motor dynamics close to a DC motor.

Figure 1.1 illustrates the categories of popular control theories for induction motor drive. Among them, FOC and Direct Torque Control (DTC) are more like methodology than theory. To differ with new control algorithms, they were placed under PI-based control. However, various control strategies are still using the same structure of FOC technique by applying different control algorithms, since R. H. Park [4] published the concept of rotating reference frame in 1929, in which the idea of FOC was developed based on the electromagnetic torque proportional to the cross product of stator current and rotor flux $\vec{i}_s \times \vec{\psi}_r$, which decoupled the control of torque and filed excitation, similar to a DC motor. After that, the Indirect FOC was presented by Hasse in 1968 [5] and the concept of Direct FOC was developed by Blaschke [6], both of the orientations are aligned with the rotor flux vector. Alternatively, the DTC employs hysteresis control directly with stator flux and torque references based on a look-up table, which was presented by Noguchi [7], later the similar method of Direct Self Control (DSC) was developed by Depenbrock [8].

Unlike the synchronous motors, which have identical position of flux and rotor shaft during operations, the induction motor has slip difference between the electromagnetic flux and the actual rotor shaft position. The estimation of flux position is essential for $dq$ rotating frame transformation. In the concept of FOC, the Indirect FOC does
not require the magnitude information of the rotor flux as a feedback signal, the $d$-axis current is defined based on the open-loop model and the rotor flux position is simply estimated from the reference values of currents and the real-time rotor shaft position. On the other hand, the idea of Direct FOC is investigated with the existence of rotor flux measurement in the initial contribution [6], which is not effective for most applications, hence various types of flux observers were developed for accommodating such situation. The study of flux observer is another aspect of induction motor control, the goal is to estimate the information of flux vector of motor based on the available measurements, such as current, voltage and speed. Direct FOC normally requires rotor flux observer, moreover, DTC generally needs the observer of stator flux. Once the flux information is observed, a torque estimation could be straightforward. The categories of observer could be divided into closed-loop and open-loop. A well-known closed-loop observer is the Luenberger observer [9], which is employed in Chapter 3 of this thesis and will be illustrated in Appendix C. Another type of closed-loop observers, such as Gopinahlth’s type observer [10] and Kalman Filter observer [11], has been developed in the research field. On the other hand, the open-loop observers are developed directly from the motor model without the measurements of error feedback, for example, the Voltage or Stator Model, Current or Rotor Model, Voltage-Current and Speed Model, and Voltage-Speed Model. The open-loop observers are normally employed in the research field for speed sensor-less control, which tends to observe the motor shaft speed and position directly, in order to accomplish for the applications where the speed and position sensors are undesired, due to cost, cabling, robustness and construction. Many contributions of speed sensor-less control were published for induction motor application [12–14]. Another popular research topic of induction motor control is the parameter identification, since the motor parameter information is significantly important for model-based control methods. Lots of techniques for parameter identification have been developed to this day, the types can be gathered as off-line and on-line identification, the initial proposal of off-line identification was presented in [15], the difference is when the identification procedure takes place, during commissioning for off-line or during normal operation for on-line identification. In this thesis, the major scope is focused on the high performance control of induction motor, especially optimal control using Model Predictive Control method.

1.2 Motivation

In the conventional FOC technique of induction motor control, Proportional-plus-Integral-plus-Derivative (PID) controllers were employed to accomplish the control objectives,
additionally, the Pulse Width Modulation (PWM) was employed to convert the numerical control signals into "on/off" commands for the inverter switches with desired switching frequency. Generally, the Indirect FOC speed control requires three PI controllers, two for inner-loop current control and the other one for outer-loop velocity control, where the Direct FOC needs four PI controllers, one more for outer-loop flux control, because the dynamics of induction motor speed model is a fourth order system and PID controller is designed for controlling Single-Input-Single-Output (SISO) system.

Model Predictive Control (MPC), as one type of optimal control, has been proven efficient in different fields, such as chemical engineering and process control [16]. Compared to PID controllers, due to the matrix based control design, MPC could effectively deal with Multi-Input-Multi-Output (MIMO) system with embedding constraints to deliver the optimal control solution. However, the weaknesses of MPC also appeared to open up different research questions. Firstly, the MPC method generally requires a linear model for optimization, and the model accuracy is essential to provide the desired control performance. Furthermore, the complex matrix calculations could increase the computational load. Recently, MPC has emerged to the research field of power electronics. There are two major directions classified [17, 18]: traditional MPC method and Finite Control Set (FCS)-MPC. The former category of methods generally requires the modulation techniques to generate the switching states based on the continuous control signals, which are computed from the feedback predictive controller. The latter group directly determines the optimal switching states of the inverter based on the error between the predictions and the outputs.

The modulation-based MPC control directly generates the continuous control signal using the traditional MPC algorithm [19], which inherits the advantages of MPC, such as optimization and robustness. Firstly, MPC is designed based on an Linear-Time-Invariant (LTI) model, and the induction motor model is strongly coupled and time-varying. In the traditional PI-based FOC method, the feedforward technique is applied to deal with non-linearity and coupling variables. Secondly, for model-based control, the completeness of dynamic model is essential, thus the Gain Scheduling (GS) method is developed for controlling non-linear system by using a set of linear controllers. The linear controllers could be MPC controllers based on linearised model at different operating conditions. Furthermore, the sampling rate of digital control for motor drive is fairly fast, to ensure the fast switching frequency of inverter. Thus, the continuous-time MPC will be chosen and directly employ the physical model without the influence of sampling rate selection.

In parallel, the FCS-MPC computes the prediction on-line then calculates the objective functions based on the finite control inputs, hence the optimal control solution will select the switch states, which minimizes the objective function. According to the control
technique, the on-line prediction is computed based on the non-linear model, since the time-varying parameters could be measured or estimated at the current sampling instance. Thus, the FCS-MPC method has been proven to be simple, efficient and robust, but there are still several research topics in this field waiting to be investigated, such as closed-loop performance, steady-state error and uncertain switching frequency.

Another major motivation of this thesis is on developing the MPC methods in both categories and assess their control performances using experimental results of induction motor drive control. The first approach is designed to achieve the speed control of IM using only one MPC controller where the control structure is based on the Direct FOC methodology, and the Luenberger observer is employed to estimate real-time information of the rotor flux vector. Within this framework, the Gain-Scheduling method is proposed in the model predictive control design to deploy multiple models to overcome nonlinearities. The second approach tends to use a cascaded structure for decoupling and linearization, in order to control the IM velocity. The third approach is to revise the current FCS-MPC method in order to resolve the steady-state error issue. Finally, the proposed control methods are assessed together with the traditional PI-based FOC method.

1.3 Literature Review

Based on the motivated research topics discussed previously, the related literature reviews have been given in this section. Firstly, the field of Model Predictive Control (MPC) has been briefly reviewed. The next subsection then studies the contributions using Gain-Scheduling method to control non-linear plants. The last part is dedicated to the field of Finite Control Set-MPC.

1.3.1 Model Predictive Control

Model Predictive Control (MPC), with its advantage in multi-variable constraint control system, had been widely used in industry for last few decades [16]. The initial idea of MPC was published in 1980s, on Model predictive heuristic control [20] and Dynamic Matrix Control (DMC) [21]. Following the same track, the Generalized Predictive Control (GPC) was presented in 1987 [22, 23]. The objectives of DMC and GPC are different due to the concentration on different applications, DMC was focused on multivariable constrained control for oil and chemical applications, on the other hand, the GPC was attempted to invent a new adaptive control. In the earlier years, the method of Linear Quadratic Regulator (LQR) had been extensively studied for optimal control without constraints [24], where the optimal control sequence was generated using
the state feedback law and the feedback gain matrix was calculated using an algebraic Riccati equation. In order to reduce the on-line computational load, a receding horizon implementation was formulated in [25]. Another research issue of MPC is the closed-loop stability, one approach which had been proposed was the contraction constraint [26], moreover, various approaches of the closed-loop stability had been extensively dedicated and addressed satisfactorily in the literature [27–29].

Furthermore, in the field of non-linear MPC, various model forms have been tested and discussed. In [30], an excellent survey had been summarized and updated, different approaches have been published to achieve the objective of non-linear MPC. [31] presents the infinite horizon/terminal constraint MPC with continuous-time model. Then, the technique of variable horizon/hybrid MPC was proposed in [32] to deal with the global optimality and feasibility problems. Another technique, called Quasi-infinite horizon MPC, was introduced in [33] using an infinite horizon without switching of controller. Additionally, the idea of contractive MPC was presented with complete algorithm and proven stability in [34]. Moreover, the mentioned approaches all require a non-linear program that is solved on-line during each sample-time, thus, the more applicable approach is to design the MPC after the system is linearised in some manner, then the techniques can be developed for linear system control on-line. The classical contribution of MPC with linearisation are presented with different methods. [35] presents the cascade arranged MPC by applying first feedback linearisation, which has obvious limitations when the system is low order and fulfilled the conditions required. Then, [25] applies the MPC to an industrial application by using different linear models, which are derived from local linearisation, at each sampling interval. The dissertation [36] presents the calculation of the predicted system trajectory at each sampling time using the time-invariant MPC algorithm. More approach is presented in [37], which approximated the non-linear MPC control law with a linear controller. Since, the research outcomes of the MPC design based on non-linear system are not essentially satisfactory, the field is still widely open for future development. In the thesis, the linear MPC design is computed referring to Wang’s book [19], which illustrates briefly the design and implementation of a continuous-time MPC based on a LTI model.

The early research of MPC design was generally applied to oil, chemical control or process control. however, it is still relatively new in the field of power electronics, Permanent-Magnet-Synchronous Motor (PMSM) and Induction Motor control [38, 39]. However, the majority of the model predictive control systems in electrical drives and power electronics are based on discrete-time systems [40–42]. It is well known that discrete-time models are ill-conditioned in a fast sampling environment with the poles of the discrete-time model converging to the unit circle of complex plane as the sampling interval reduces [43]. Thus, the continuous-time MPC based on linearised model
is employed in Chapter 3, moreover, the Gain-Scheduling method is introduced for the non-linearity of IM model, which will be reviewed in the following subsection.

1.3.2 Gain Scheduling Method

Gain scheduled control for non-linear plants had been proven as a successful design methodology in many engineering applications. The well-known application is the multiple model adaptive control of aircraft [44, 45], another engineering application of process control was presented in [46]. The basic ideas behind a gain scheduling type of control systems involve linear time-invariant approximation of a non-linear plant and the deployment of several linear models related to the operating conditions of the system to be controlled [47]. The Gain-Scheduling method was reviewed extensively in the survey paper [48], there are four general steps involved in the design of a gain scheduled control system. The first step is to identify the operating conditions of the non-linear system and obtaining a family of linear models with regard to these conditions. The second step is to perform linear control system design for the family of linear models with specified closed-loop performance for each linear system. The third step is the actual gain scheduling that forms an interpolation between the family of the linear closed-loop control systems. The fourth step is the validation and simulation of gain scheduled control system. The gain scheduled control system is to use linear control strategies to control a non-linear plant, and the family of closed-loop linear systems is stable in the vicinity of each linear model. However, the stability of each linear system is not sufficient to guarantee the closed-loop stability of entire gain scheduled control system because of the time-varying nature of the closed-loop systems [49]. With respect to slowly time-varying systems, the Gain-Scheduling (GS) method had been proven with guaranteed properties on the robustness, performance and nominal stability. Based on the classical work by Desoer, in [50, 51], the guaranteed properties of GS method had been presented for linear parameter-varying plants. Then, [52] presents the same analysis for non-linear systems with both types: scheduling on reference trajectory and scheduling on the plant output. In this thesis, the induction motor can be considered as a linear time-varying plant, Chapter 3 illustrates that Gain-Scheduling method is applied on the centralized MPC control design, which contribute the fact the Gain-Scheduling method is still functional under a relatively fast parameter varying system, due to the advantages of the MPC and robustness of the induction motor model.
1.3.3 Finite Control Set - Model Predictive Control

The control methodology of FCS-MPC to control machine drives and power converters arises in last decade, which differs from the MPC discussed in the previous section. The control objective is achieved by predicting the future outputs with the advantage of finite switching states of inverter. The early work of FCS-MPC was presented by J. Rodriguez [53–55] with comparison to other techniques, such as PWM and hysteresis. With PWM based implementation, the similar control method of deadbeat control was developed in earlier literature [56]. On the other hand, the hysteresis based current control was well developed in 1980s [7]. Moreover, the Predictive Current Control (PCC) methods are developed [57], which presents and compares two PCC methods with the synchronized on-off principle. Since the PCC and FCS-MPC are similar in early versions when the cost functions were defined based on the error between the prediction and the reference. Similar methods have also been published with different applications, such as Permanent Magnet Synchronous Motor [58–60] and power converters [61–64]. Recently, the latest work of FCS-MPC has been reviewed in [65, 66]. The hardware devices of control platforms are developed widely applied in Power Electronics, such as Digital Signal Processor (DSP) and Field Programmable Gate Arrays (FPGA) [67–70]. Thus, the powerful computational capacity has made the implementation of new and complex control methods become possible, especially the FCS-MPC, which is simple and effective but requires very fast sampling rate to ensure the control performance.

There are several research topics coexisted, such as weighting factor calculation, uncertain switching frequency and steady-state error. [71] presents a method to calculate the optimal weighting factor in cost function, for the purpose of reducing the torque ripples for induction motor control, the proposed method was formal and validated. [72] presents the method using Mean Square Error values of the controlled variables and [73] also calculates the weighting factor using trial procedure. Similar methods to calculate the weighting factor, or scalar factor were studied in [74, 75]. Moreover, an alternative way is presented in [76], where the selection of the weighting factor is unnecessary when the single cost function is replaced by a multi-objective optimization.

On the other hand, the steady-state error issue appeared since the minimization of cost function was computed in every sampling instant without considering the system performance between samples [66]. To counteract this issue, [77] presents approaches of intermediate sampling and integral error term in the cost function with application to a simple H-Bridge power converter. Another approach of embedding an on-line adaptive in the control system is studied in [78] for LCL Coupled Inverter-Based Distributed Generation Systems. Moreover, [62] presents the reduction of steady-state error for the predictive control of the DC-link voltage in an active-front-end rectifier. With the same
Chapter 1. \textit{Introduction}

objective, Chapter 5 presents a brief study of the original FCS-MPC and the revised approach with cascaded integral action to eliminate the steady-state error problem.

1.4 Contribution

As the title of this thesis, the main contribution is to develop theory and applications of the Model Predictive Control (MPC) methodologies for induction motor drive. In detail, three approaches are presented in this thesis, Gain-Scheduled Model Predictive Control (GS-MPC) for centralized structure; Cascaded MPC control structure and the Finite Control Set - Model Predictive Control (FCS-MPC) method.

1.4.1 Gain-Scheduled Model Predictive Control

Initially, the continuous-time MPC with constraints is designed based on linearized model of induction motor, but the pre-defined operating conditions are required for the controller design. In order to ensure the speed control of IM could handle various operating conditions, the Gain-Scheduling method is introduced to apply on the continuous-time MPC with constraints. Thus, the speed control using the centralized MPC could operate in different conditions. Due to the exponential data weighting method, the controller bandwidth can be tuned separately for different operation points, hence the control performance will be optimized. Furthermore, due to the advances of constrained predictive control method, non-linear constraints are more closely approximated to produce a higher accuracy. By applying the Direct FOC together with the Luenberger observer, the proposed method is validated on a real industrial sized induction motor. Furthermore, the GS-MPC method could also be extended to other non-linear plants for various applications.

1.4.2 Cascaded Model Predictive Control

In the research field of induction motor speed control, the cascaded structure is widely used with Indirect FOC technique. Here, the cascaded MPC method is proposed with two MPC controllers. The proposed structure has advantages when the dynamics have significant difference of the time constants. In the induction motor case, the dynamics of the electrical model is much faster than that of the mechanical model. Hence, the stator currents are controlled by an inner-loop controller where the velocity is controlled by an outer-loop controller. The contribution of this aspect is to design the velocity control using two continuous-time MPC with constraints. Instead of using Gain-Scheduling
method, the non-linearity issue is counteracted by using high-gain control feedback for inner-loop, thus the impact of non-linearity could be reduced. Moreover, the MPC controller is extended to torque control as well as position control of induction motor drive. The Non-Minimal State-Space (NMSS) model is introduced to eliminate the induced noise, which is generated from the continuous-time MPC feedback of the position angle.

1.4.3 Finite Control Set - Model Predictive Control

Another main contribution of this thesis is concentrated on the FCS-MPC method, which is to develop the MPC methodology to power electronics control field without using modulation techniques. The FCS-MPC method has advantages of fast response and simple implementation, however, lack of the integral action inside the control loop. The contribution of this thesis is to embed an integral action into the FCS-MPC control method. Firstly, the original FCS is revised to form a feedback control structure, then the incremental model can be simply derived for the integrator embedding. The contributions of FCS-MPC include control systems using both $dq$-frame and $\alpha\beta$-frame models. The $dq$ coordinate based control only requires the integrator to eliminate the steady-state error, since the outputs are constant or piecewise constant during the steady-state operation. On the other hand, the FCS-MPC based on $\alpha\beta$ model is more complicated, since the controlled outputs are sinusoidal signals during the steady-state. In order to resolve the steady-state error issue, the resonant controller is embedded into the FCS-MPC method. Both applications are validated for induction motor current with experimental results.

1.5 Publication

Based on the contribution of this thesis, the following publications are presented by the author during the PhD candidature.

Book Chapter


Journal Paper

Conference Paper


1.6 Outline of Thesis

Previously, the introduction of this thesis includes the historical background, motivation and literature review of this thesis, hence the organization of remaining chapters are introduced as follows.

Chapter 2 introduces the background of the induction motor structure, hence the physical model of induction motor is developed. Then, the necessary linearization of the mathematical model is presented for the control design in later chapters. Both of mathematical and simulation models are validated by comparing with the actual experimental results.

Chapter 3 details the algorithm design of the centralized continuous-time Model Predictive Controller. The augmented model is introduced to embed integral action into the MPC design procedure, in which the Laguerre functions are employed to describe the continuous-time trajectory of the control signals. Then, the optimal control technique is derived without considering the constraints, in order to achieve the desired closed-loop
bandwidth, the exponential data weighting method is introduced to obtain the prescribed degree of stability. The constrained control is one of the major advantages for MPC method, thus the constraints control implementation is designed using Quadratic Programming procedure. In addition, the state feedback observer is introduced for the estimation of augmented state variables. Another main contribution of this thesis is illustrated using the Gain-Scheduled MPC method. The modification of control algorithm and the determination of the operation conditions are discussed. Moreover, the revised approach of the modulation limit is introduced based on the advance of the MPC design.

Chapter 4 illustrates an alternative structure of the MPC design, which is the cascaded control using two controllers, thus the electrical and mechanical models are separated due to the different dynamics. Since the Indirect FOC technique is used for this chapter, the slip estimation method is introduced for the synchronous flux position observation. After that, the cascaded speed control scheme is detailed separately with respect to inner-loop current and outer-loop speed controllers. Furthermore, the MPC position control is discussed inclusively using cascaded structure. Another research question is addressed using the Non-minimal state-space model to filter the noise generated from the higher order derivative states.

Chapter 5 presents the recently arising predictive control technique in power electronics control field of the Finite Control Set (FCS) method. Firstly, the original method of FCS-MPC is firstly introduced, then the revised FCS method is presented for integral action embedding. In addition, the revised method is also analysed and validated by comparing to the original FCS. The next step would be that the integral action is derived to solve the research issue of the steady-state error, the derivation of I-FCS controller is detailed followed by the design and implementation. At the end, the simulation and experimental results are shown for comparison purpose. The FCS method design in $\alpha\beta$ coordinates is also investigated for alternative approach where the resonant controller is designed for completely tracking the sinusoidal signals in $\alpha\beta$ frame.

Chapter 6 details the traditional Field Oriented Control (FOC) method of induction motor drive using PID controllers. The design of the PID control method is described with the frequency response analysis, and the implementation of the current, velocity and position, which are illustrated with the experimental results. Moreover, another control architecture is introduced for using proportional controller only at the inner-loop current control. The results and comparison will be shown in Chapter 6.

Chapter 7 illustrates the comparison of all presented control algorithms in this thesis using the experimental results. The proposed control methods are compared and validated according to the traditional PI-based FOC technique. The current, velocity and position controllers are compared and analysed with the performance as well as the robustness, where the model parameter is mismatching for the control design.
At the end, Chapter 8 concludes the entire thesis according to different contributions and specifies the further developments related to the proposed control algorithms in this thesis.
Chapter 2

Modelling of Induction Motor

2.1 Introduction

In this chapter, the model of induction motor is derived for the subsequent control design. The induction motor discussed in this thesis is a typical 3-phase AC induction machine, firstly, the background knowledge is introduced, which includes the working principle, structure and the Park-Clarke transformation. Then, the development of the physical model is illustrated and reviewed [79–81]. Because of the non-linearity of the model dynamics, the linearization procedure is introduced according to its steady-state values using Taylor expansion [82], so that, the mathematical model is validated for control design applications.

The content of this chapter is shown as follows. In order to understand the non-linearity of the induction motor, its basic structure is introduced in Section 2.2. Then, Section 2.3 presents the procedure of the development of the mathematical model in different coordinates. The linearization of the derived model is discussed in Section 2.4 for control design. Section 2.5 validates the derived mathematical model and the simulation model, comparing with the actual induction motor.

2.2 Overview: Structure of Induction Motor

Induction motor generally contains two main components in its structure: stator and rotor windings. The stator winding is supplied by AC power source, which generates rotating magnetic field, so-called stator flux $\vec{\psi}_s$. Then, the constantly changing of the flux will cause current induced in the rotor winding based on Lenz’s Law, this induced current also generates magnetic field, so-called rotor flux $\vec{\psi}_r$. Since both fluxes are opposite to each other, a rotational force is generated to accelerate the rotor until the
magnetizing torque is balanced to the load torque. Because the actual rotor’s position is always lagging the flux position, in order to ensure the flux cutting through the rotor winding, there is a difference between the angular speed $\omega_s$ of the synchronous magnetic field and the electrical motor speed $\omega_e$, which is called slip with $\omega_{\text{slip}} = \omega_s - \omega_e$. Furthermore, the electrical motor speed is depending on the actual mechanical speed of the motor shaft $\omega_r$, which has the relationship of $\omega_e = \omega_r \times Z_p$, where $Z_p$ is the number of pole pairs for the stator winding.

The traditional three-phase AC induction motor has two types, wound and squirrel-cage, which describes the form of the rotor winding. The wound form rotor has brushed external connection, the example applications include the wind-farm generator and heavy lifting motor. The squirrel-cage rotor is more popular and general, it has the rotor winding connection as a short-circuit without any external connection, the induction motor discussed in this thesis is the squirrel-cage type.

Since the induction motor is supplied by three-phase AC, the use of space vector will simplify the indication of the motor electrical variables, such as current, voltage and flux. Generally, these vectors are plotted according to a complex plane, such as $\alpha\beta$ and $dq$ coordinates. For example, in Figure 2.1(a), the stator current vector of the 3-phase induction motor are converted into 2-dimensional quantities, which are represented by $i_\alpha$ and $i_\beta$ on the $\alpha\beta$ coordinates. However, because the current vector is constantly rotating respect to the fixed $\alpha\beta$ frame, the variables $i_\alpha$ and $i_\beta$ will appear to be sinusoidal. On the other hand, as shown in Figure 2.1(b) the $dq$ coordinates are introduced especially for vector control design. Generally, the $d$-axis is overlapping to the rotor flux vector $\vec{\psi_r}$, thus the rotational velocity of the $dq$ coordinates is identical to the synchronous flux speed $\omega_s$. Furthermore, the position of the $dq$ coordinates $\theta_s$ is not measurable, so that the estimation of the rotor flux vector $\vec{\psi_r}$ is required, which will be discussed in

\begin{figure}
\centering
\includegraphics[width=\textwidth]{fig2_1.png}
\caption{AC machine coordinates re-arrangement}
\end{figure}
later chapters.

By reflecting on the \( dq \) coordinates, the space vectors of the induction motor are represented in terms of \( dq \) components.

\[
\begin{align*}
\vec{u}_s &= u_{sd} + ju_{sq} \\
\vec{i}_s &= i_{sd} + ji_{sq} \\
\vec{\psi}_s &= \psi_{sd} + j\psi_{sq}
\end{align*}
\]

\[
\begin{align*}
\vec{u}_r &= u_{rd} + ju_{rq} \\
\vec{i}_r &= i_{rd} + ji_{rq} \\
\vec{\psi}_r &= \psi_{rd} + j\psi_{rq}
\end{align*}
\]

Since the value of above \( dq \) components is normally constant or piece-wise constant during the operation, in a way, these variables are linearized, which is convenient for vector control design.

The method of converting implementation between 3-phase space vectors and \( dq \) coordinates is called Park and Clarke transformation \([4, 83]\). As a well-known knowledge, the detail is not going to be discussed. Generally, Clarke transform is converting \( abc \) frame into \( \alpha\beta \) frame, Park transform is converting \( \alpha\beta \) into \( dq \) coordinates. By using current as an example, both transforms and their inverse transforms are displayed as follows:

\[
\begin{bmatrix}
  i_\alpha \\
  i_\beta \\
  i_d \\
  i_q
\end{bmatrix} =
\begin{bmatrix}
  \frac{2}{3} & \frac{-1}{3} & \frac{-1}{3} \\
  0 & \frac{\sqrt{3}}{3} & \frac{-\sqrt{3}}{3}
\end{bmatrix}
\begin{bmatrix}
  i_a \\
  i_b \\
  i_c
\end{bmatrix} ;
\begin{bmatrix}
  i_a \\
  i_b \\
  i_c
\end{bmatrix} =
\begin{bmatrix}
  1 & 0 \\
  -\frac{1}{2} & \frac{\sqrt{3}}{2} \\
  -\frac{1}{2} & -\frac{\sqrt{3}}{2}
\end{bmatrix}
\begin{bmatrix}
  i_\alpha \\
  i_\beta \\
  i_d \\
  i_q
\end{bmatrix}
\]

where \( \theta_s \) is the position angle of \( dq \) coordinates, the identical calculation could also be applied to voltage \( u \) and flux \( \psi \).

### 2.3 Development of Physical Model Equation

The physical model of induction motor has been well developed and analysed in the past literatures \([79, 84]\), in this section, the procedure of developing the induction motor mathematical model is reviewed and studied.

The equivalent circuit, also known as Steinmetz or T-equivalent circuit, of the induction motor is introduced in Figure 2.2, where \( R_s \) and \( R_r \) are the stator and rotor resistance, \( X_s \) and \( X_r \) are the stator and rotor reactances, respectively. \( X_m \) presents the mutual machine reactance. The circuit is similar to the circuit of a voltage transformer, but the resistance at the secondary side is altered, due to the short circuit of the rotor winding, where \( s \) represents the slip difference, which is defined as \( s = \frac{\omega_s - \omega_e}{\omega_s} \). In this thesis, an assumption is made for the modelling process, that is the parasitic effect such
as eddy currents, magnetic field saturation and so on are neglected.

### 2.3.1 Vector Representation of Induction Motor Equation

For the completeness of the thesis, the development of induction motor physical model is introduced briefly in this section. Firstly, the voltage equations of both stator and rotor respect to their own winding system are derived:

\[
\begin{align*}
\mathbf{u}_s^s &= R_s \mathbf{i}_s^s + \frac{d}{dt} \mathbf{\psi}_s^s \\
\mathbf{u}_r^r &= R_r \mathbf{i}_r^r + \frac{d}{dt} \mathbf{\psi}_r^r = 0
\end{align*}
\]  

where \( R_s \) and \( R_r \) are the stator and rotor resistance, respectively, \( (\cdot)^s \) denotes the space vector of variables. Due to the short-circuit of the rotor winding, the rotor voltage vector in (2.2) is always equal to zero.

In order to calculate the space vectors in the same equation, the respecting coordinates of these vectors have to be identical. In the induction motor system, the stator winding is always fixed in standstill position, where the rotor winding is rotating identical to the electrical rotor speed \( \omega_e \). Thus, a set of space vectors are defined to change the reference frame of the space vectors of the rotor:

\[
\begin{align*}
\mathbf{u}_r^r &= \mathbf{u}_r^r e^{j \theta_e} \\
\mathbf{i}_r^r &= \mathbf{i}_r^r e^{j \theta_e} \\
\mathbf{\psi}_r^r &= \mathbf{\psi}_r^r e^{j \theta_e}
\end{align*}
\]

where \( \theta_e = \omega_e t \), \( \omega_e \) is the electrical angular speed of the rotor. This transformation is based on the electrical field of the rotor lagging behind that of the stator winding in \( \theta_e \) radians, thus, in order to synchronize these two reference frames, the space vectors in the rotor reference frame are advanced with the angle \( \theta_e \).
Chapter 2. Modelling of Induction Motor

Now, by multiplying both sides of (2.2) with the factor $e^{j\theta_e}$, substituting the transformations into the rotor voltage equation leads to

$$
\overrightarrow{v}_r^s = R_r \overrightarrow{i}_r + \frac{d \overrightarrow{\psi}_r^s}{dt} - j\omega_e \overrightarrow{\psi}_r^s = 0
$$

(2.3)

where the following equality is used,

$$
\frac{d \overrightarrow{\psi}_r^s}{dt} e^{j\theta_e} = \frac{d \overrightarrow{\psi}_r^s}{dt} - j\omega_e \overrightarrow{\psi}_r^s
$$

With the space vectors of both currents in stator and rotor, the instantaneous fluxes of both windings are given based on their relationships to currents:

$$
\overrightarrow{\psi}_s = L_s \overrightarrow{i}_s + L_h \overrightarrow{i}_r
$$

(2.4)

$$
\overrightarrow{\psi}_r = L_h \overrightarrow{i}_s + L_r \overrightarrow{i}_r
$$

(2.5)

where $L_h$ is the mutual machine inductance, $L_s$ and $L_r$ are the stator and rotor inductance, respectively. Note that there are coupling terms in the stator flux (see (2.4)) and rotor flux (see (2.5)). Here it is to eliminate the rotor current $\overrightarrow{i}_r$ from the equations and find the relationship between the stator voltage and current.

Taking derivative of stator flux based on (2.4), and substituting the stator flux with stator and rotor currents, the stator voltage equation (2.1) becomes:

$$
\overrightarrow{u}_s^s = R_s \overrightarrow{i}_s + L_s \frac{d \overrightarrow{i}_s}{dt} + L_h \frac{d \overrightarrow{i}_r}{dt}
$$

(2.6)

To eliminate the rotor current $\overrightarrow{i}_r$, the rotor flux equation (2.5) is used to find

$$
\overrightarrow{i}_r = \frac{1}{L_r} \overrightarrow{\psi}_r - \frac{L_h}{L_r} \overrightarrow{i}_s
$$

Substituting this into (2.6), it yields

$$
\overrightarrow{u}_s^s = R_s \overrightarrow{i}_s + L_s \left(1 - \frac{L_h^2}{L_s L_r}\right) \frac{d \overrightarrow{i}_s}{dt} + \frac{L_h}{L_r} \frac{d \overrightarrow{i}_r}{dt}
$$

To eliminate the derivative of the rotor flux from the above equation, the voltage balance equation from the rotor (see 2.3) is used, which leads to

$$
\frac{d \overrightarrow{\psi}_r^s}{dt} = -\frac{R_r}{L_r} \overrightarrow{\psi}_r^s + \frac{R_r L_h}{L_r} \overrightarrow{i}_s + j\omega_r \overrightarrow{\psi}_r^s
$$

(2.7)

where the rotor current is replaced with stator flux and current.

Finally, it can be verified that the stator voltage equation is expressed in terms of the
rotor flux and stator current:

\[
\overrightarrow{u_s} = (R_s + R_r \frac{L_h^2}{L_r^2}) \overrightarrow{i_s} + L_s (1 - \frac{L_h^2}{L_s L_r}) \frac{d \overrightarrow{i_s}}{dt} + (\frac{L_h R_r}{L_r^2} + j \omega_r \frac{L_h}{L_r}) \overrightarrow{\psi_r}
\]

Although all physical parameters are defined in the above model, they could have more compact expressions. More specifically, define the following parameters used in the model.

leakage factor:

\[
\sigma = 1 - \frac{L_h^2}{L_s L_r}
\]

stator time constant:

\[
\tau_s = \frac{L_s}{R_s}
\]

rotor time constant:

\[
\tau_r = \frac{L_r}{R_r}
\]

coefficients:

\[
k_r = \frac{L_h}{L_r}
\]

\[
r_\sigma = R_s + R_r k_r^2
\]

\[
\tau'_\sigma = \frac{\sigma L_s}{r_\sigma}
\]

With these definitions of parameters, the voltage equation in space vector form is simply expressed as

\[
\overrightarrow{i_s} + \tau'_\sigma \frac{d \overrightarrow{i_s}}{dt} = \frac{k_r}{r_\sigma} \left( \frac{1}{\tau_r} - j \omega_r \right) \overrightarrow{\psi_r} + \frac{1}{r_\sigma} \overrightarrow{u_s}
\]

(2.8)

where the rotor flux satisfies the differential equation:

\[
\overrightarrow{\psi_r} + \tau_r \frac{d \overrightarrow{\psi_r}}{dt} = j \omega_r \tau_r \overrightarrow{\psi_r} + L_h \overrightarrow{i_s}
\]

(2.9)

2.3.2 Representation in Stationary \(\alpha\beta\) Reference Frame

Upon obtaining the electrical model in the space vector form, the next step is to convert it to the model in the \(\alpha\beta\) reference frame. The \(\alpha\beta\) reference frame is a stationary reference frame in the stator side with the real (\(\alpha\)) axis and the imaginary (\(\beta\)) axis in quadrature.

By decomposing the space vector voltage, current and flux on real and imaginary axes,
they can be represented by the complex notations,

\[
\overrightarrow{u}_s = u_{s\alpha} + ju_{s\beta} \quad (2.10)
\]
\[
\overrightarrow{i}_s = i_{s\alpha} + ji_{s\beta} \quad (2.11)
\]
\[
\overrightarrow{\psi}_r = \psi_{r\alpha} + j\psi_{r\beta} \quad (2.12)
\]

To obtain the dynamic model in the \(\alpha\beta\) reference frame, the above variables are substituted into the space vector model (2.8) and (2.9).

It can be readily verified that the electrical model of the induction motor in the \(\alpha\beta\) reference frame is described by the following four differential equations:

\[
\frac{di_{s\alpha}}{dt} = -\frac{1}{\tau_{\sigma}} i_{s\alpha} + \frac{k_r}{r_{\sigma}\tau_{\sigma}} \psi_{r\alpha} + \frac{k_r}{r_{\sigma}\tau_{\sigma}} \omega_e \psi_{r\beta} + \frac{1}{r_{\sigma}\tau_{\sigma}} u_{s\alpha} \quad (2.13)
\]
\[
\frac{di_{s\beta}}{dt} = -\frac{1}{\tau_{\sigma}} i_{s\beta} - \frac{k_r}{r_{\sigma}\tau_{\sigma}} \omega_e \psi_{r\alpha} + \frac{k_r}{r_{\sigma}\tau_{\sigma}} \psi_{r\beta} + \frac{1}{r_{\sigma}\tau_{\sigma}} u_{s\beta} \quad (2.14)
\]
\[
\frac{d\psi_{r\alpha}}{dt} = \frac{L_h}{\tau_r} i_{s\alpha} - \frac{1}{\tau_r} \psi_{r\alpha} - \omega_e \psi_{r\beta} \quad (2.15)
\]
\[
\frac{d\psi_{r\beta}}{dt} = \frac{L_h}{\tau_r} i_{s\beta} + \omega_e \psi_{r\alpha} - \frac{1}{\tau_r} \psi_{r\beta} \quad (2.16)
\]

### 2.3.3 Representation in Stationary \(dq\) Reference Frame

To change the reference frame to the \(dq\) frame, it is equivalent to rotate the space vector in \(\alpha\beta\) frame clockwise by \(\theta_s\), that is

\[
\overrightarrow{u}_s' = \overrightarrow{u}_s e^{-j\theta_s} = u_{s\alpha} + j u_{s\beta}
\]
\[
\overrightarrow{i}_s' = \overrightarrow{i}_s e^{-j\theta_s} = i_{s\alpha} + j i_{s\beta}
\]
\[
\overrightarrow{\psi}_r' = \overrightarrow{\psi}_r e^{-j\theta_s} = \psi_{r\alpha} + j \psi_{r\beta}
\]

where \(\overrightarrow{u}_s', \overrightarrow{i}_s'\) and \(\overrightarrow{\psi}_r'\) denote the space vectors referred to rotating \(dq\) frame. \(\theta_s = \omega_s t\) where \(\omega_s\) is the synchronous flux angular speed in the stator. In this rotating \(dq\) reference frame, the rotor flux vector is fixed to the real axis of the coordinate system. Therefore, the quadrature component of \(\overrightarrow{\psi}_r'\) is zero. Multiplying (2.8) with the factor \(e^{-j\theta_s}\) and substituting in the space vectors in \(dq\) frame give

\[
\overrightarrow{i}_s' + \tau'_\sigma (\frac{di_s'}{dt} + j \omega_s \overrightarrow{i}_s') = \frac{k_r}{r_{\sigma}} \left( \frac{1}{\tau_r} - j \omega_r \right) \overrightarrow{\psi}_r' + \frac{1}{r_{\sigma}} \overrightarrow{u}_s'
\]

where the following equality is used:

\[
\frac{d\overrightarrow{i}_s}{dt} e^{-j\theta_s} = \frac{d\overrightarrow{i}_s'}{dt} + j \omega_s \overrightarrow{i}_s'
\]
Based on the real and imaginary components of (2.20), the dynamic electrical model in the \(dq\) reference frame is obtained:

\[
\frac{di_{sd}}{dt} = -\frac{1}{r_{\sigma}} i_{sd} + \omega_s i_{sq} + \frac{k_r}{r_{\sigma} r'_{\sigma} r_{\tau}} \psi_{rd} + \frac{1}{r_{\sigma} r'_{\sigma}} u_{sd} \tag{2.21}
\]

\[
\frac{di_{sq}}{dt} = -\omega_s i_{sd} - \frac{1}{r_{\sigma}} i_{sq} - \frac{k_r}{r_{\sigma} r'_{\sigma} r_{\tau}} \psi_{rd} + \frac{1}{r_{\sigma} r'_{\sigma}} u_{sq} \tag{2.22}
\]

Similarly, it can be shown that the rotor flux in the \(dq\) reference frame satisfies:

\[
\frac{d\psi_{rd}}{dt} = \frac{L_h}{r_{\tau}} i_{sd} - \frac{1}{r_{\tau}} \psi_{rd} \tag{2.23}
\]

\[
0 = \frac{L_h}{r_{\tau}} i_{sq} - (\omega_s - \omega_r) \psi_{rd} \tag{2.24}
\]

where the \(q\) component of rotor flux \(\psi_{rq} = 0\).

Since the equation (2.24) is an algebraic equation, it’s not included for control design. However, it yields the relationship used for estimation of \(\omega_s\):

\[
\omega_s = \omega_e + \frac{L_h}{r_{\tau}} i_{sq} \frac{\psi_{rd}}{\psi_{rd}} \tag{2.25}
\]

which is also called slip estimation and \(\omega_e\) is the electrical angular velocity of the rotor that is measured.

### 2.3.4 Mechanical Model of Induction Motor

The mechanical model of induction motor is derived from the general motion equation of motor rotation, which is described as follows.

\[
J_m \frac{d\omega_r(t)}{dt} + f_d \omega_r(t) = T_e(t) - T_L(t) \tag{2.26}
\]

where \(J_m\) presents the torque of inertia, \(f_d\) is the friction coefficient, \(\omega_r\) is the mechanical speed of motor shaft, \(T_e\) and \(T_L\) denote the electromagnetic torque and the load torque, respectively.

The electromagnetic torque of induction motor is calculated using the cross product of the space vectors of rotor flux and and stator current in the \(dq\) reference frame, which is

\[
T_e = \frac{3}{2} Z_p \frac{L_h}{L_r} (\overrightarrow{\psi_r} \otimes i_s) \tag{2.27}
\]

where \(Z_p\) is the number of pole pairs. The cross product is calculated using two three dimensional vectors \([\psi_{rd} \ 0 \ 0]\) and \([i_{sd} \ i_{sq} \ 0]\) since \(\psi_{rq}\) is zero. The result of the cross product is the vector \([0 \ 0 \ \psi_{rd} i_{sq}]\). Thus, in the \(dq\) reference frame, the electromagnetic
torque is proportional to $\psi_{rd}i_{sq}$, which is

$$T_e = \frac{3}{2}Z_p \frac{L_h}{L_r} \psi_{rd}i_{sq}$$  \hspace{1cm} (2.28)$$

If the electromagnetic torque is calculated using the space vectors of rotor and stator current in the $\alpha\beta$ reference frame, then it is proportional to the cross product of the space vectors of rotor flux and stator current in the stationary frame,

$$T_e = \frac{3}{2}Z_p \frac{L_h}{L_r} (\vec{\psi}_r \otimes \vec{i}_s)$$

$$= \frac{3}{2}Z_p \frac{L_h}{L_r} (\psi_{r \alpha}i_{s\beta} - \psi_{r \beta}i_{s\alpha})$$  \hspace{1cm} (2.29)$$

Note that the expression of electromagnetic torque $T_e$ in the $dq$ reference frame is only related to $\psi_{rd}$ and $i_{sq}$. These two variables are DC variables, thus the torque control can be achieved by controlling $\psi_{rd}$ and $i_{sq}$ to their specified constant or piece-wise constant reference signals. On the other hand, in the $\alpha\beta$ reference frame, it is much difficult to achieve torque control because the expression in (2.29) is associated with the fluxes in $\alpha\beta$ reference frame that are sinusoidal signals.

The dynamic model of the mechanical equation is obtained by substituting equation (2.28) into the motion equation (2.26).

$$\frac{d\omega_r}{dt} = -\frac{f_d}{J_m} \omega_r + \frac{3}{2} \frac{Z_p L_h}{L_r J_m} \psi_{rd}i_{sq} - \frac{T_L}{J_m}$$  \hspace{1cm} (2.30)$$

2.4 Model Linearization

The control methods, which will be discussed in Chapter 3 and Chapter 4, are proposed based on a Linear Time-Invariant (LTI) model, thus the linearization of induction motor dynamic model is essential. The differential model equations (2.21) to (2.23) and (2.30) in $dq$ coordinates are used to form the state space model.

By applying the first-order Taylor series expansion, the nonlinear terms are approximated respect to its steady-state operating condition:

$$\omega_s(t)i_{sq}(t) \approx \omega_s^0 i_{sq}^0 + i_{sq}^0 (\omega_s(t) - \omega_s^0) + \omega_s^0 (i_{sq}(t) - i_{sq}^0)$$  \hspace{1cm} (2.31)$$

$$\omega_s(t)i_{sd}(t) \approx \omega_s^0 i_{sd}^0 + i_{sd}^0 (\omega_s(t) - \omega_s^0) + \omega_s^0 (i_{sd}(t) - i_{sd}^0)$$  \hspace{1cm} (2.32)$$

$$\omega_e(t)\psi_{rd}(t) \approx \omega_e^0 \psi_{rd}^0 + \psi_{rd}^0 (\omega_e(t) - \omega_e^0) + \omega_e^0 (\psi_{rd}(t) - \psi_{rd}^0)$$  \hspace{1cm} (2.33)$$

$$i_{sq}(t)\psi_{rd}(t) \approx i_{sq}^0 \psi_{rd}^0 + \psi_{rd}^0 (i_{sq}(t) - i_{sq}^0) + i_{sq}^0 (\psi_{rd}(t) - \psi_{rd}^0)$$  \hspace{1cm} (2.34)$$
Chapter 2. Modelling of Induction Motor

where \( \omega_s^0, \omega_c^0, i_{sq}^0, i_{sd}^0, \psi_{rd}^0 \) denote the steady-state operating condition of system states, whose values are pre-defined.

Note that, the electrical rotor speed \( \omega_e \), which appears in the electrical model equations, has relationship with the mechanical rotor speed \( \omega_r \) as \( \omega_e = Z_p \omega_r \). Thus, the mechanical model equation has to be modified before being included in the full-order state space model. By substituting \( \omega_r = \frac{\omega_e}{Z_p} \) into (2.30), it leads to

\[
\frac{d\omega_e}{dt} = -\frac{f_d}{J_m} \omega_e + \frac{3}{2} \frac{Z_p^2 L_h}{L_r J_m} \psi_{rd} i_{sq} - \frac{Z_p T_L}{J_m}
\]

Therefore, the linearised full-order state space model in continuous-time is obtained as follows:

\[
\begin{bmatrix}
\dot{i}_{sd}(t) \\
\dot{i}_{sq}(t) \\
\dot{\psi}_{rd}(t) \\
\dot{\omega}_e(t)
\end{bmatrix} =
\begin{bmatrix}
\frac{1}{r_{e}} & -\frac{k_r}{r_{e} \tau_{e}} & \frac{k_r}{r_{e} \tau_{e}} & 0 \\
\frac{k_r}{r_{e} \tau_{e}} & \frac{1}{r_{e}} & -\frac{k_r Z_p}{r_{e} \tau_{e} \sigma} & 0 \\
\frac{L_h}{r_{e}} & 0 & -\frac{1}{r_{e}} & 0 \\
0 & Z_p \kappa_t \psi_{rd}^0 & Z_p \kappa_t i_{sq}^0 & -\frac{J_r}{J_m}
\end{bmatrix}
\begin{bmatrix}
i_{sd}(t) \\
i_{sq}(t) \\
\psi_{rd}(t) \\
\omega_e(t)
\end{bmatrix} +
\begin{bmatrix}
\frac{1}{r_{e} \tau_{e} \sigma} & 0 \\
0 & \frac{1}{r_{e} \tau_{e} \sigma} \\
0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
u_{sd}(t) \\
u_{sq}(t)
\end{bmatrix}
\]

where \( \kappa_t = \frac{3 Z_p L_h}{2 L_r J_m} \), the last matrix presents the disturbance term, which contains the variable terms \( \omega_e(t) \) and \( T_L \), if the system is stable during steady-state, the variable term \( \omega_s(t) \) will converge to constant, and the load torque \( T_L \) is assumed to be constant.

Then, the disturbance matrix can be defined as constant during steady-state operation. However, since the gain schedule method in Chapter 3 is intended using one-dimension switching, which means only one switching parameter inside the non-linear model. Therefore, the equation (2.25) is substituted into the state-space model, in order to eliminate the unmeasurable term, synchronous angular velocity \( \omega_s(t) \). So that, the bilinear parts containing \( \omega_s \) are transformed in term of the parameter \( \omega_e(t) \).

\[
\omega_s(t) i_{sq}(t) = \omega_e(t) i_{sq}(t) + \frac{L_h}{\tau_r} \frac{i_{sq}^2(t)}{\psi_{rd}(t)}
\]

\[
\omega_s(t) i_{sd}(t) = \omega_e(t) i_{sd}(t) + \frac{L_h}{\tau_r} \frac{i_{sq}(t) i_{sd}(t)}{\psi_{rd}(t)}
\]
The nonlinear terms inside the equations (2.37) and (2.38) are approximated again by apply Taylor series expansion.

\[
\omega_e(t)i_{sq}(t) \approx \omega_e^0i_{sq} + i_{sq}(\omega_e(t) - \omega_e^0) + \omega_e^0(i_{rd}(t) - i_{rd}^0) \quad (2.39)
\]

\[
\omega_e(t)i_{sd}(t) \approx \omega_e^0i_{sd} + i_{sd}(\omega_e(t) - \omega_e^0) + \omega_e^0(i_{sq}(t) - i_{sq}^0) \quad (2.40)
\]

\[
\frac{i_{sq}^2(t)}{\psi_{rd}(t)} \approx \frac{i_{sq}^0}{\psi_{rd}^0} + \frac{i_{sq}^0}{\psi_{rd}^0}(i_{sd}(t) - i_{sd}^0) - \frac{i_{sq}^0}{\psi_{rd}^0}(i_{sd}(t) - i_{sd}^0)^2 + \frac{i_{sd}^0}{\psi_{rd}^0}(i_{sq}(t) - i_{sq}^0) \quad (2.41)
\]

\[
\frac{i_{sq}(t)i_{sd}(t)}{\psi_{rd}(t)} \approx \frac{i_{sq}^0i_{sd}^0}{\psi_{rd}^0} + \frac{i_{sq}^0}{\psi_{rd}^0}(i_{sd}(t) - i_{sd}^0) - \frac{i_{sq}^0i_{sd}^0}{\psi_{rd}^0}(i_{rd}(t) - i_{rd}^0)^2 + \frac{i_{sd}^0}{\psi_{rd}^0}(i_{sq}(t) - i_{sq}^0) \quad (2.42)
\]

Although the variables \(\omega_e(t), i_{sq}(t), i_{sd}(t), \psi_{rd}(t)\) are the actual physical variables, not the deviation variables, the approximation relations are only valid in the vicinity of the steady-state conditions as they are based on the Taylor series expansion. From (2.33)-(2.42), it is seen that the information about the steady-state values of \(\omega_e^0, i_{sq}^0, i_{sd}^0\) and \(\psi_{rd}^0\) is required to obtain the parameters for the linearised terms. Since the output variables are \(\omega_e(t)\) and \(\psi_{rd}(t)\), the steady-state parameters for these variables will be chosen to be equal to their desired reference signals. In particular, in the application of induction motor control, the reference signal to rotor flux is often fixed as a constant with its value dependent on the operating speed and load condition of the induction motor. For instance, the reference signal for \(\psi_{rd}\) is recommended to be constant for the energy efficient within the rated speed and load-free operating condition. The reference signal to the rotor velocity \(\omega_e(t)\) changes according to operating conditions. Therefore, the steady state conditions for \(\psi_{rd}^0\) and \(\omega_e^0\) are first determined according to the operating conditions of the induction motor. Next, from the model equation (2.23), by letting \(\frac{d\psi_{rd}}{dt} = 0\), the steady-state solution of \(i_{sq}^0\) is determined via the steady-state calculation

\[
i_{sq}^0 = \frac{1}{L_h}\psi_{rd}^0
\]

Furthermore, by letting \(\frac{d\omega_e}{dt} = 0\) in the mechanical equation (2.30), the steady-state operating condition for \(i_{sq}\) is calculated together with the linear approximation (2.34), leading to

\[
i_{sq}^0 = \frac{2L_r}{3Z_p L_h \psi_{rd}^0}(f_0\omega_e^0 + T_L^0)
\]

With all the steady-state operating parameters defined, the next step is to substitute approximations (2.33-2.42) into model equations (2.21-2.24) and (2.30) in order to obtain the linear time-invariant (LTI) model that is valid at the operating condition specified by the steady-state parameters \(\omega_e^0, i_{sq}^0, i_{sd}^0, \psi_{rd}^0\). By gathering all the appropriate terms,
Chapter 2. Modelling of Induction Motor

26

it can be verified that the linear model has the form,

\[
\frac{dx_m(t)}{dt} = A_m x_m(t) + B_m u(t) + \mu^0
\]

\[y(t) = C_m x_m(t) \quad (2.43)\]

where \(x_m(t) = [i_{sd}(t) \ i_{sq}(t) \ \psi_{rd}(t) \ \omega_e(t)]^T\), \(u(t) = [u_{sd}(t) \ u_{sq}(t)]^T\), and with the coefficient \(\kappa_t = \frac{Z_pL_h}{J_m}\), the matrices \(A_m\) and \(B_m\) are defined as

\[
A_m = \begin{bmatrix}
-\frac{1}{r_e} & \frac{L_h}{r_e} \frac{\psi_{rd}}{\psi_{rd}} & \frac{k_r}{r_e r_t} \frac{(\psi_{rd})^2}{\psi_{rd}} & -\frac{L_h}{r_e} \frac{(\psi_{rd})^2}{\psi_{rd}} \\
0 & \frac{1}{r_e} & -\frac{1}{r_e} & 0 \\
L_h & Z_p \kappa_t \psi_{rd} & Z_p \kappa_t \psi_{sq} & -f_d/J_m \\
0 & 0 & 0 & 0
\end{bmatrix}
\]

\[
B_m = \begin{bmatrix}
\frac{1}{r_e r_t} & 0 & 0 & 0 \\
0 & \frac{1}{r_e r_t} & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}; C_m = \begin{bmatrix}
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

The constant bias vector \(\mu^0\), consisting of the steady-state parameters, is given by,

\[
\mu^0 = \begin{bmatrix}
-\omega_e^0 t_{sq} + \frac{L_h}{r_e} \frac{\psi_{rd}}{\psi_{rd}} \psi_{rd}^0 + \frac{k_r}{r_e r_t} \psi_{rd}^0 \omega_e^0 \ t_{sd} + \frac{k_r}{r_e r_t} \psi_{rd}^0 \omega_e^0 \psi_{rd}^0 - f_d/J_m
\end{bmatrix}
\]

In this case, the disturbance vector \(\mu^0\) contains only operating condition terms, which could be rejected by embedding an integrator inside the feedback controller.

2.5 Model Validation

Up to this point, the continuous-time mathematical models of induction motor are derived. However, the validation of dynamic model is essential before the control design stage. In this section, the responses of the open-loop plant are examined based on the mathematical model, simulation model and the actual experimental test-bed, with the identical input signal \(u_{sd}\) and \(u_{sq}\). The results of all state variables, \(i_{sd}\), \(i_{sq}\), \(\psi_{rd}\) and \(\omega_r\), are compared for validation. Firstly, the implementation of these three models is discussed as follows.

2.5.1 Mathematical Model

Firstly, validation of the mathematical model is derived from Section 2.3, where \(dq\) frame dynamic model is illustrated in Figure 2.3, which is divided to three parts: electrical
model based on equations (2.21-2.23), mechanical model based on equation (2.30) and the slip estimation based on equation (2.25).

Electrical model is based on the model equations (2.21) to (2.23), which are rewritten in state-space model format as follows:

\[
\dot{x}_m(t) = A_m(\omega_s, \omega_e)x_m(t) + B_m u(t) \\
y(t) = C_m x_m(t)
\]  

(2.44)

where

\[
A_m(\omega_s, \omega_e) = \begin{bmatrix}
-\frac{1}{\tau_s} & \frac{k_p}{r_p} & \frac{k_s}{r_s} \\
\frac{1}{\tau_s} & -\frac{1}{\tau_e} & \frac{-1}{\tau_e} \\
\frac{1}{\tau_e} & 0 & 0
\end{bmatrix}, \\
B_m = \begin{bmatrix}
\frac{1}{\tau_s r_p} & 0 \\
0 & \frac{1}{\tau_e r_p} \\
0 & 0
\end{bmatrix}, \\
C_m = \begin{bmatrix}
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

the state variable is \(x_m(t) = [i_{sd}(t) \ i_{sq}(t) \ \psi_{rd}(t)]^T\), the input signal is \(u(t) = [u_{sd}(t) \ u_{sq}(t)]^T\) and the output is \(y(t) = [i_{sd}(t) \ \psi_{rd}(t)]^T\). Additionally, the electrical part of dynamic model is a time-varying linear model, where the system matrix \(A_m\) contains the time varying parameters \(\omega_s(t)\) and \(\omega_e(t)\).

The state space model is implemented by using the first order approximation, as \(\frac{dx(t)}{dt} = \frac{x(t_i) - x(t_i - \Delta t)}{\Delta t}\) at arbitrary sampling instant \(t_i\), with the sampling interval of \(\Delta t\), then the implementation equations are obtained:

\[
x_m(t_i) = x_m(t_i - \Delta t) + \Delta t(A_m(t_i - \Delta t)x_m(t_i - \Delta t) + B_m u(t_i - \Delta t)) \\
y(t_i) = C_m x_m(t_i)
\]

where the initial condition is defined as \(x_m(0) = [0 \ 0 \ 0]^T\), \(\omega_s(0) = 0\) and \(\omega_e(0) = 0\) at the starting point.

Mechanical model is derived based on the differential equation (2.30), which contains
one bilinear term $\psi_{rd}(t) \times i_{sq}(t)$, another input disturbance term $T_L(t)$. Thus, the implementation equation is derived as

$$\omega_r(t_i) = \omega_r(t_i - \Delta t) + \Delta t \left( -\frac{f_d}{J_m} \omega_r(t_i - \Delta t) + \frac{3}{2} \frac{Z_p L_h}{L_r J_m} \psi_{rd}(t_i - \Delta t) i_{sq}(t_i - \Delta t) - T_L \right)$$

where the zero initial condition is also defined, as $\omega_r(0) = 0$.

Slip estimation is developed to estimate the angular velocity of the synchronous $dq$ frame $\omega_s(t)$, the estimation is derived from the model equation (2.24), at the sampling instant $t_i$, the estimation is obtained as:

$$\omega_s(t_i) = \omega_e(t_i) + \frac{L_h \tau_r}{\psi_{rd}(t_i)} i_{sq}(t_i)$$

Note that the rotor flux $\psi_{rd}(t_i)$ is included as the denominator, to ensure the model is stable, $\psi_{rd}(t_i) \neq 0$.

Therefore, the implementation of the mathematical model in $dq$ coordinates is established, with a constant input values of $u_{sd}(t)$ and $u_{sq}(t)$, the results of the state variables are presented in Figure 2.5(a).

### 2.5.2 Simulation Model

In later chapters of this thesis, there are respective simulation results for different control methods presented for analysis and discussion, thus the validation of the simulation model compared to the actual motor become essential.

The simulation results are computed using MATLAB Simulink. The setup detail of the simulator is illustrated in Appendix B, in this section, the open loop simulator is implemented to obtain the state variables response.

The simulator of Simulink blocks is shown in Figure 2.4, where the induction motor model is selected from the SimPower Toolbox, as well as the IGBT inverter unit. By defining the input signals $u_{sd}$ and $u_{sq}$, four state variables should be obtained as the results.

The implementation detail includes: the Pulse Width Modulation (PWM) has a carrier frequency of 2kHz, the dc-link voltage is 520V and zero load is attached on the motor shaft, to keep the simplicity of the open loop test, the position angle of the $dq$-frame is defined independently at frequency of 25Hz (ie. $\omega_s = 157.0796$ rad/s).

As a result, the simulation results of the model response are presented in Figure 2.5(b).
2.5.3 Experimental Test-bed

The experimental results are obtained as a comparison, in which both of the mathematical and simulation models are seen to be valid if their model responses are close enough to the experimental results. The experimental results are demonstrated by using MATLAB Simulink together with xPC Target as controller. The setup of the test bed is presented in Appendix B for detail. The implementation parameters are kept identical to the simulation for comparison purpose, such as the PWM carrier frequency, the DC-link voltage and the synchronous frequency. Furthermore, both simulations and experiments have identical sampling interval of 100µs. Figure 2.5(c) presents the experimental results of the state variables response.

2.5.4 Comparison

By defining the identical input signals, as \(u_{sd} = 10V\) and \(u_{sq} = 100V\) to the three setups. Figure 6.5 presents the result of open loop control based on non-linear mathematical model, simulation SimPower model and the real experiment. Figure 2.5(a) demonstrates the simulation result with respect to the non-linear mathematical model where it is seen that the result is relatively ‘tidy’ comparing to other two cases due to the entire simulation is implemented based on \(dq\) coordinates. Here, neither PWM switching nor Park-Clarke transformation are included in this simulation, as a result, the steady-state difference respect to experimental results is more significant comparing to figure 2.5(b), which has precisely the same implementation setup with the experiment case 2.5(c) leading to similar results in the steady-state. Once including the switching noise in the
current $i_{sd}$ and $i_{sq}$, the longer delay during transient section in simulation result is seen in the simulation computation, where in experimental case, the real induction motor has faster dynamics response.
Chapter 3

Centralized Model Predictive Control

3.1 Introduction

The Continuous-time Model Predictive Control (MPC) theory is extensively discussed in this chapter. As one type of optimal control theory, MPC computes a series of optimal control signals based on a Linear-Time-Invariant (LTI) dynamic model within a prediction horizon. Since MPC is designed based on the state-space model, it has significant advantage controlling Multi-Input-Multi-Output (MIMO) system with constraints.

The control objective of this chapter is to achieve the induction motor speed control by using only one model predictive controller. Thus, the full-order model of the induction motor are applied in $dq$ coordinates. Due to the physical model being time varying, the Gain-Scheduling (GS) method is introduced for linearisation.

Figure 3.1 illustrates the block diagram of the centralized MPC for speed control, the

![Figure 3.1: Speed control of induction using centralized MPC controller](image-url)
controlled outputs include the rotor flux magnitude $\psi_{rd}$ and the mechanical motor speed $\omega_r$, the controller will directly compute the control signals, $u_{sd}$ and $u_{sq}$, based on respective model, which will be discussed in later sections. Since the Direct FOC technique structure is applied, a Luenberger Observer is introduced for two reasons, one is estimation of the synchronous flux position $\theta_s$ for $dq$ transformation, the other is observing the magnitude of the rotor flux $\psi_{rd}$ for feedback that is not measurable. The design of the Luenberger observer is not new, which is discussed in Appendix C. Another state observer is included to estimate the derivative of the state feedback variables, due to the incremental model is applied for the MPC design.

The organization of this chapter is shown as follows, Section 3.2 introduces the continuous-time MPC design based on one pre-defined operating condition. In Section 3.3, the Gain-Scheduled continuous-time MPC is design for time-varying operating condition. Finally, the discussion and conclusion are included in Section 3.4.

### 3.2 Continuous-time MPC

The Model Predictive Controller in this section is designed based on a continuous-time model. While the continuous-time MPC is employed, the physical model could be directly applied for control design, in order to reduce the modelling error, in addition, the influence of the sampling interval $\Delta t$ disappears unlike the discrete-time model based control. At the same time, the weakness includes that the estimation of the feedback states is required, because the derivative of the state feedback are not measurable when the incremental model is used.

To achieve the zero steady-state error and disturbance rejection, an integrator must be included in the feedback control loop, thus the augmented model is introduced in Section 3.2.1; Then, the Laguerre function is analysed to approximate the control trajectory in Section 3.2.2 and the optimal control strategy is investigated with receding horizon method in Section 3.2.3. The prescribed degrees of stability is discussed in Section 3.2.4. In Section 3.2.5, one important advantage of MPC, the constraints control design, is utilized for Multi-Input-Multi-Output (MIMO) system. The state observer design is introduced in Section 3.2.6. Finally, the implementation of predictive control system will be established for induction motor application and the results of both simulation and experiment will be illustrated for the control objective in Section 3.2.7. The continuous-time model predictive control algorithm was introduced in [19]. However, for completeness of the thesis, the algorithm is introduced and discussed in this Chapter.
3.2.1 Augmented Model

The purpose of introducing augmented model is embedding an integrator into the optimal control loop. The strategy is to modify the model structure, as a result, the model input will be the derivative of the original control signal. For the predictive control design or other state feedback control design, when using a LTI model, it is meant to use a so-called 'small' signal model \[85\]. Namely, we define the incremental variables for all the signals around their steady-state values. In the meanwhile, the output must remain the same for the feedback comparison.

For the induction motor control, the incremental variables are \( \tilde{x}_1(t) = i_{sd}(t) - i_{sd}^0 \), \( \tilde{x}_2(t) = i_{sq}(t) - i_{sq}^0 \), \( \tilde{x}_3(t) = \psi_{rd}(t) - \psi_{rd}^0 \), \( \tilde{x}_4(t) = \omega_e(t) - \omega_e^0 \), \( \tilde{u}_1(t) = u_{sd}(t) - u_{sd}^0 \), \( \tilde{u}_2(t) = u_{sq}(t) - u_{sq}^0 \), \( \tilde{y}_1(t) = \tilde{x}_3(t) \) and \( \tilde{y}_2(t) = \tilde{x}_4(t) \). The linearized state space model suitable for the design of a linear continuous-time predictive controller is obtained as:

\[
\frac{d\tilde{x}_m(t)}{dt} = A_m \tilde{x}_m(t) + B_m \tilde{u}(t) + \mu^1(t) \\
\tilde{y}(t) = C_m \tilde{x}_m(t) \tag{3.1}
\]

Note that in the small signal model, the system matrices \( A_m, B_m \) and \( C_m \) are identical to the linearized model (2.36), however, the signals are different unless one assumes zero steady-state conditions for all signals. Also, the bias vector \( \mu^1 \) is different from the last term in (2.36). By using the small signal model in the control system design, the steady-state conditions are required in the implementation stage of the control system. For instance, the actual measurements of the outputs will have to subtract their steady-state values to obtain \( \tilde{y}(t) \) and the incremental control signals \( \tilde{u}(t) \) will have to add their steady-state values to obtain the actual control signals. In a gain scheduled nonlinear control system, the changes in the steady-state values, when using (3.1), are required in the implementation for each linear control system, which could be quite clumsy.

Instead of taking the conventional pathway for the design of linear predictive controller using the small signal model (3.1), in this chapter, we will explore an alternative approach that will form a small signal model, yet without using all steady-state values of the signals in the implementation stage. This no doubt leads to the convenience in the implementation stage of the gain scheduled model predictive controller. The approach is based on the original proposal in [86] where integrators are embedded in the design of a continuous-time model predictive controller.

By taking a derivative operation on the state-space equation of the linearised continuous-time state space model (2.36), which leads to

\[
\dot{x}_m(t) = A_m x_m(t) + B_m \tilde{u}(t) \tag{3.2}
\]
where the derivative of the last term is zero. The dimensions of the model matrices $A_m$, $B_m$ and $C_m$ are described as: $4 \times 4$, $4 \times 2$ and $2 \times 4$, respectively.

To design the augmented model, firstly the auxiliary variables are defined:

$$
\begin{align*}
    z(t) &= \dot{x}_m(t) \\
    y(t) &= C_m x_m(t)
\end{align*}
$$

then, the new state variable is chosen as

$$
x(t) = \begin{bmatrix} z(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A_{m,4} \ 0_{4 \times 2} \\ C_m \ 0_{2 \times 2} \end{bmatrix} \begin{bmatrix} z(t) \\ y(t) \end{bmatrix} + \begin{bmatrix} B_{m,2} \ 0_{2 \times 2} \end{bmatrix} \dot{u}(t)
$$

where $I_{s \times s}$ and $o_{s \times s}$ are the identity and zero matrices with dimensions denoted by the sub-indices, respectively. The matrices $A$, $B$, $C$ are denoted in the augmented model for the notational simplicity [19].

Several comments are in order about the augmented state-space model. Firstly, in the augmented model, the first part of the state variables consists of $\dot{x}_m(t)$. Because the predictive controller is designed to follow step reference signals, the steady-state vector of $x_m(t)$ is a constant vector, and as a result, the steady-state vector of $\dot{x}_m(t)$, is ensured to be a zero vector for all operating conditions. This information provides the convenience in the implementation of gain scheduled predictive controller as it could be a non-trivial task to find the steady-state values of the original state vector with respect to a family of operating conditions. The second part of the state variables consists of the plant outputs whose steady-state values are the desired reference signals to the control system. At this point, we could write the corresponding small signal model for the augmented model (3.3) by subtracting the steady-state values of the outputs from the second part of state variables ($\tilde{y}_1(t) = \psi_{rd}(t) - \psi_{rd}^0$, $\tilde{y}_2(t) = \omega_e(t) - \omega_e^0$). Furthermore, the steady-state values $\psi_{rd}^0$ and $\omega_e^0$ are taken as the reference signals to the control system and they can be included inside the objective function in the sequel.

### 3.2.2 Laguerre Function

Laguerre function is one type of orthonormal basis function, which is a series expansion of real functions that satisfies sequence of defined properties. The Laguerre function is
applied to the MPC design for tracking the trajectory within the control horizon. From the literature [87, 88], suppose that a set of real functions \( l_i(t), i = 1, 2, \ldots \) is defined as an orthonormal set within \([0, \infty)\) if

\[
\begin{align*}
\int_0^\infty l_i^2(t)dt &= 1 \quad (3.4) \\
\int_0^\infty l_i(t)l_j(t)dt &= 0 \quad (3.5)
\end{align*}
\]

where \( i \neq j \). Given an arbitrary function \( f(t) \), which satisfies

\[
\int_0^\infty f(t)^2dt = 0 \quad (3.6)
\]

A complete orthonormal function set \( l_i(t) \) has relation as follows

\[
\int_0^\infty f(t)l_i(t)dt = 0 \quad (3.7)
\]

The objective to apply orthonormal basis function is to approximate the control trajectory in continuous-time domain, then the arbitrary function \( f(t) \) is approximated by applying orthonormal expansion [89].

\[
f(t) = \sum_{i=1}^\infty c_i l_i(t) \quad (3.8)
\]

where \( c_i, i = 1, 2, \ldots \) are the coefficients of the expansion approximation. However, in the real application computation, the number of the approximations \( i \) can be infinity. Therefore, with the finite number \( N \) of coefficients, the arbitrary function \( f(t) \) is approximated with respective error,

\[
\int_0^\infty (f(t) - \sum_{i=1}^N c_i l_i(t))^2dt < \varepsilon \quad (3.9)
\]

where the approximation error \( \varepsilon \) reduces while the integer \( N \) increases.

Laguerre functions are one set functions that satisfy the properties of orthonormal functions [90, 91] and have defined functions as, for any \( p > 0 , \)

\[
\begin{align*}
l_1(t) &= \sqrt{2p} \times e^{-pt} \\
l_2(t) &= \sqrt{2p}(-2pt + 1)e^{-pt} \\
\vdots &= \vdots \\
l_i(t) &= \sqrt{2p} \frac{e^{-pt}}{(i-1)!(i-1)e^{-2pt}} \quad (3.10)
\end{align*}
\]
Chapter 3. Centralized Model Predictive Control

The Laguerre function is generated by defining the initial condition, at \( t = 0 \), the state vector \( L(0) = \sqrt{2} [1 \ldots 1]^T \), then the Laguerre function state vector \( L(t) \) satisfies

\[
\begin{bmatrix}
\dot{l}_1(t) \\
\dot{l}_2(t) \\
\vdots \\
\dot{l}_N(t)
\end{bmatrix}
= \begin{pmatrix}
-p & 0 & \ldots & 0 \\
-2p & -p & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
-2p & \ldots & -2p & -p
\end{pmatrix}
\begin{bmatrix}
l_1(t) \\
l_2(t) \\
\vdots \\
l_N(t)
\end{bmatrix}
\]

Therefore, the Laguerre function has two parameters to be defined, \( p \) is the time scaling factor and \( N \) is the dimension of the stator vector.

The final control objective of MPC is computing the derivative of the optimal control trajectory \( \dot{u}(\tau) \), where \( \tau \) is within the moving time window from \( t_i \) to \( t_i + T_p \). Since the continuous-time control trajectory is finally computed, the Laguerre functions are generated to approximate the trajectory at the time \( \tau \). Therefore, for \( 0 \leq \tau \leq T_p \),

\[
\dot{u}(\tau) \approx \sum_{i=1}^{N} c_i l_i(\tau) = L(\tau)^T \eta
\]

where \( \eta = [c_1 \ c_2 \ \ldots \ c_N]^T \) is the optimal control coefficient. So that, the optimal MPC result is computing \( \eta \) vector instead of \( \dot{u}(\tau) \).

### 3.2.3 Optimal Control Design without Constraints

In order to compute the optimal control solution, several steps are discussed in this section. Firstly, at the current time \( t_i \), the predicted response trajectory of the error state variable vector \( \tilde{x}(t) = x(t) - x^0 \) based on the plant model, where \( x^0 \) is the steady-state vector of \( x(t) \). At future time \( \tau \), where \( \tau > 0 \), is described as follows

\[
\tilde{x}(t_i + \tau | t_i) = e^{A\tau} \tilde{x}(t_i) + \int_0^\tau e^{A(\tau - \gamma)} B \dot{u}(\gamma) d\gamma
\]

where \( \gamma \) presents the time variable within the prediction window to distinguish from \( \tau \), due to the previous analysis of the augmented model, the effect of the disturbance terms is neglected in the future prediction.

Since induction motor model has 2 inputs, the control signal vector and input matrix \( B \) can be written as

\[
\dot{u}(\tau) = [\dot{u}_1(\tau) \ \dot{u}_2(\tau)]^T \\
B = [B_1 \ B_2]^T
\]
The approximation by using Laguerre functions to both control signals leads to

\[ \dot{u}_1(\tau) = L_1(\tau)^T \eta_1 \]
\[ \dot{u}_2(\tau) = L_2(\tau)^T \eta_2 \]

Therefore, the prediction of future state at time \( \tau \) (3.13) is re-written with orthonormal expansion

\[ \tilde{x}(t_i + \tau | t_i) = e^{A\tau} \tilde{x}(t_i) + \phi(\tau)^T \eta \] (3.14)

where \( \phi(\tau)^T \) is the convolution integral

\[ \phi(\tau)^T = \int_0^\tau e^{A(\tau-\gamma)} [B_1 L_1(\gamma)^T B_2 L_2(\gamma)^T] d\gamma \]

and the coefficient vector \( \eta \) contains the two sub-coefficient vectors \( \eta_1 \) and \( \eta_2 \):

\[ \eta^T = [\eta_1^T \eta_2^T] \]

therefore, the matrix \( \phi(\tau)^T \) has dimension of \( n \times (N_1 + N_2) \) and the optimal control coefficient vector has dimension of \( N_1 + N_2 \).

By substituting the predicted state trajectory into the state-space model, the prediction of future output at time \( \tau \) is written as

\[ y(t_i + \tau | t_i) = Ce^{A\tau} \tilde{x}(t_i) + C\phi(\tau)^T \eta \] (3.15)

Secondly, the cost function is derived for the predictive control design. In the traditional predictive control design (for example,\([23]\)), at time \( t_i \), the cost function is often chosen as

\[ J = \int_0^{T_p} (r(t_i) - y(t_i + \tau | t_i))^T (r(t_i) - y(t_i + \tau | t_i)) + \dot{u}(\tau)^T R \dot{u}(\tau) d\tau, \] (3.16)

where \( r(t_i) = [r_1(t_i) \ r_2(t_i)]^T \) is the reference vector for the outputs \( \psi_{rd}(t) \) and \( \omega_e(t) \) at the sampling time \( t_i \). Without constraints, the objective of model predictive control in the case of set-point following is to find the control law that will drive the predicted plant output \( y(t_i + \tau | t_i) \) as close as possible, in a least squares sense, to the future trajectory of the set-point \( r(t_i) \). The assumption is that the set-point signal \( r(t_i) \) is a constant (or a set of constants) within the optimization window.

The cost function, which is similar to the classical Linear Quadratic Regulator (LQR), is used as follows

\[ J = \int_0^{T_p} (\tilde{x}(t_i + \tau | t_i)^T Q \tilde{x}(t_i + \tau | t_i) + \dot{u}(\tau)^T R \dot{u}(\tau) d\tau \] (3.17)
where $\hat{x}(t_i + \tau \mid t_i) = x(t_i + \tau \mid t_i) - x^0$, $Q$ and $R$ are semi-positive and positive definite matrices. Note as stated before that the first part of state vector $x(t)$ is $\dot{x}_m(t)$ having a zero steady-state vector and the second part of $x(t)$ has its steady-state vector corresponding to the reference signals because of the augmented model used in the design (see (3.3)). More specifically, for the induction motor control application, the steady-state vector is defined as

$$x^0 = \begin{bmatrix} 0 & 0 & 0 & \psi_0^0 & \omega_e^0 \end{bmatrix}^T$$  \hspace{1cm} (3.18)

This formulation is convenient for the gain scheduled predictive control system because when the operation condition changes, the steady-state vector $x^0$ is varied according to the set-point signals. We can either pre-define the operating conditions in the design stage of the predictive control system or use the actual reference signal values to define $x^0$ in real-time. The latter approach is used in the implementation here. The cost function (3.16) used in the traditional model is identical to the LQR type of cost function (3.17) when the weighting matrix $Q$ is chosen to be $C^TC$ where $C$ is the output matrix, $C = [o_{2\times 4} \ I_{2\times 2}]$, and the reference signals are used as the steady-state values of the output variables.

The cost function (3.17) contains two terms: predicted future system response and the change of control signal. The second term of cost function is reorganized by assuming $R$ is a diagonal matrix $R = diag\{r_k\}$, where $k = 1, 2$.

$$\int_0^{T_p} \dot{u}(\tau)^T R \dot{u}(\tau) d\tau = \sum_{k=1}^2 \int_0^{T_p} r_k \dot{u}_k(\tau)^2 d\tau$$  \hspace{1cm} (3.19)

Since the property of the orthonormal functions, $\int_0^{\infty} L_1(\tau)L_1(\tau)^T d\tau$ is equal to the identity matrix, which is also true for $\int_0^{\infty} L_2(\tau)L_2(\tau)^T d\tau$. So that, the second term of the cost function is expressed, by assuming $R$ to be a diagonal matrix and sufficiently large prediction horizon $T_p$:

$$\int_0^{T_p} \dot{u}(\tau)^T R \dot{u}(\tau) d\tau = \eta^T R_L \eta$$  \hspace{1cm} (3.20)

where $R_L$ is a block diagonal matrix with two blocks corresponding to the weights on the control signals.

By substituting the prediction equation (3.14) into the cost function (3.17), it becomes:

$$J = \int_0^{T_p} (e^{A\tau}x(t_i) + \phi(\tau)^T \eta)^T Q(e^{A\tau} \tilde{x}(t_i) + \phi(\tau)^T \eta) d\tau + \eta^T R_L \eta$$  \hspace{1cm} (3.21)
which is a quadratic function with respect to $\eta$:

$$
J = \eta^T \int_0^{T_p} \phi(\tau)Q\phi(\tau)^T d\tau + R_L \eta + 2\eta^T \int_0^{T_p} \phi(\tau)Qe^{A_T}d\tau \tilde{x}(t_i) \\
+ \tilde{x}(t_i)^T \int_0^{T_p} e^{A_T\tau}Qe^{A_T}d\tau \tilde{x}(t_i)
$$  \hspace{1cm} (3.22)

By defining two matrices as follows

$$
\Omega = \int_0^{T_p} \phi(\tau)Q\phi(\tau)^T d\tau + R_L
$$  \hspace{1cm} (3.23)
$$
\Psi = \int_0^{T_p} \phi(\tau)Qe^{A_T}d\tau
$$  \hspace{1cm} (3.24)

the objective function (3.22) has the compact expression:

$$
J = \eta^T \Omega \eta + 2\eta^T \Psi \tilde{x}(t_i) + \tilde{x}(t_i)^T \int_0^{T_p} e^{A_T\tau}Qe^{A_T}d\tau \tilde{x}(t_i)
$$  \hspace{1cm} (3.25)

completing the squares in the cost function leads to

$$
J = [\eta + \Omega^{-1}\Psi \tilde{x}(t_i)]^T \Omega [\eta + \Omega^{-1}\Psi \tilde{x}(t_i)] \\
+ \tilde{x}(t_i)^T \int_0^{T_p} e^{A_T\tau}Qe^{A_T}d\tau \tilde{x}(t_i) - \tilde{x}(t_i)^T \Omega^T \Omega^{-1}\Psi \tilde{x}(t_i)
$$  \hspace{1cm} (3.26)

Since the last two terms are independent of $\eta$ the parameter vector $\eta$, the minimum of the cost function is achieved if the first term is set to zero, that is,

$$
\eta = -\Omega^{-1}\Psi \tilde{x}(t_i)
$$  \hspace{1cm} (3.27)

After the computation of optimal control coefficient vector $\eta$, the control trajectory $\dot{u}(\tau)$ is re-constructed with the Laguerre functions

$$
\dot{u}(\tau) = \begin{bmatrix} L_1(\tau)^T & O_2 \\ O_1 & L_2(\tau)^T \end{bmatrix} \eta
$$

where $O_1$ and $O_2$ are zeros vectors with their dimensions equal to those of $L_1(\tau)^T$ and $L_2(\tau)^T$.

The principle of receding horizon control strategy is to use the information from the first sample of the control trajectory. Hence, at the sampling time $t_i$, the optimal control $\dot{u}(t_i)$ for the unconstrained problem is

$$
\dot{u}(t_i) = \begin{bmatrix} L_1(0)^T & O_2 \\ O_1 & L_2(0)^T \end{bmatrix} \eta
$$  \hspace{1cm} (3.28)
The actual control signal is computed using

$$u(t_i) = u(t_{i-1}) + \dot{u}(t_i)\Delta t$$  \hspace{1cm} (3.29)

where $\Delta t$ is the sampling interval used in the implementation of the continuous-time predictive control system.

Without constraints, the optimal control solution can also be expressed as state feedback control

$$\dot{u}(t) = -K_{mpc} \tilde{x}(t)$$  \hspace{1cm} (3.30)

where the feedback control gain matrix is

$$K_{mpc} = \begin{bmatrix} L_1(0)^T & O_2 \\ O_1 & L_2(0)^T \end{bmatrix} \Omega^{-1} \Psi$$  \hspace{1cm} (3.31)

The data matrices $\Omega$ and $\Psi$ are computed off-line as the process of continuous-time MPC design. Therefore, in the unconstrained case, while the feedback gain matrix $K_{mpc}$ is computed off-line as shown in (3.31), the on-line computation only includes equation (3.30). Furthermore, the location of closed-loop poles is evaluated by calculating the eigenvalues of $(A - BK_{mpc})$. In this design, the Laguerre scaling parameters $p_1, p_2$ and the numbers of terms used, $N_1, N_2$, are the performance tuning parameters. When the numbers of terms are large, with a long prediction horizon $T_p$, the derivative of the control trajectory $\dot{u}(.)$ closely matches the underlying optimal control trajectory defined by the linear quadratic regulator (LQR) ([19]).

In the application of the induction motor, for a given operating condition, the time varying components of the dynamic model are calculable, so that a LTI model will be obtained for the MPC design. For example, the velocity and flux references are set at the rated values: $\omega^* = 1400$ RPM and $\psi^*_{rd} = 0.6$ Wb. The MPC control setup includes: the Laguerre function parameters $N = 6$ and $p = 20$, prediction horizon $T_p = 0.5$, weighting factor $r_k = 1$, then the closed-loop eigenvalues are derived as

$$[-139.72 \pm j288.84 \hspace{0.5cm} -68.46 \hspace{0.5cm} -1.43 \pm j2.16 \hspace{0.5cm} -5.39 \times 10^{-6}]$$

where one pole is almost located at the origin and another pair of poles is very closed to the origin as well. Thus, the feedback control is considered as marginally stable, the change of tuning parameter $r_k$ could affect the closed-loop eigenvalues, Table 3.1 shows that smaller weighting parameter $r_k$ does push the dominant eigenvalues away from the origin but not significant. Therefore, the method called exponential data weighting is applied for such situation.
Table 3.1: Closed-loop eigenvalues with different \( r_k \)

<table>
<thead>
<tr>
<th>( r_k )</th>
<th>Closed-loop eigenvalues</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>(-139.72 \pm j288.84 ) − 68.46 − 3.74 ± j3.74 − 5.38 \times 10^{-3} )</td>
</tr>
<tr>
<td>0.01</td>
<td>(-139.72 \pm j288.84 ) − 68.46 − 6.57 ± j6.53 − 5.36 \times 10^{-4} )</td>
</tr>
<tr>
<td>0.001</td>
<td>(-139.72 \pm j288.84 ) − 68.45 − 11.92 ± j11.88 − 0.0053 )</td>
</tr>
<tr>
<td>0.0001</td>
<td>(-139.72 \pm j288.84 ) − 68.35 − 21.07 ± j20.88 − 0.0529 )</td>
</tr>
</tbody>
</table>

### 3.2.4 Exponential Data Weighting

From the previous section, the closed-loop eigenvalues are heavily dependent on weight matrices \( Q \) and \( R_L \). However, because there are 36 elements in the \( Q \) matrix, it is a complicated matter to find the individual elements and the combination of them to achieve desired closed-loop performance. As demonstrated, the variation of \( R_L \) does not sufficiently change the closed-loop performance as desired.

Exponential weighting is to use a time dependent weighting \( e^{-\alpha t} (\alpha > 0) \) in the cost function of the predictive control system design, in order to produce a numerically well-conditioned \( \Omega \) matrix \cite{92}. On the basis of exponential data weighting, a prescribed degree of stability is to ensure that the eigenvalues of the closed-loop predictive control system are on the left-hand side of the \(-\beta \) line (\( \beta > 0 \)). The locations of the desired closed-loop eigenvalues are illustrated in Figure 3.2. The detailed information about the

\[
\begin{align*}
\text{Im} & \quad \text{Re} \\
-\beta & \quad 0
\end{align*}
\]

**Figure 3.2**: Prescribed degree of stability of \(-\beta \)

exponential data weighting and prescribed degree of stability can be found in \cite{19}.

**Selection of \( \alpha \).** The idea behind the selection of \( \alpha \) is to make sure that the design model with \((A - \alpha I)\) is stable with all eigenvalues on the left-half of the complex plane. The computation of the prediction when using \( A - \alpha I \) is numerically sound.

From a given augmented state-space model \((A, B)\), the eigenvalues of \( A \) are determined. Because the induction motor is a stable system, the unstable eigenvalues of \( A \) come from the integrators that have been embedded in the model. In this case, any \( \alpha > 0 \) will serve the purpose of exponential data weighting.
Once the exponential weight factor $\alpha$ is selected, the eigenvalues of the matrix $A - \alpha I$ are fixed. Since this matrix is stable with an appropriate choice of $\alpha$, the prediction of the state variables is numerically sound. In general, if the eigenvalues of $A - \alpha I$ were further away from the imaginary axis on the complex plane, then a smaller $T_p$ would be required.

**Selection of degree of stability $\beta$** The closed-loop performance of a predictive control system so far is determined by the choice of $Q$ and $R$ matrices. The tuning could be very time consuming as it often requires finding the off-diagonal elements in $Q$ and $R$ to achieve satisfactory performance. Now, with the additional parameter $\beta$ that dictates the degree of stability, the closed-loop eigenvalues of the predictive control system are effectively positioned to the left-hand side of the $-\beta$ line on the complex plane. The parameter $\beta$ will be used to shift the closed-loop eigenvalues of the predictive control system.

With the parameter $\beta$ chosen, which is the degree of stability, the following Riccati equation is solved for the $P$ matrix:

$$P(A + \beta I) + (A + \beta I)^T P - PBR^{-1}B^T P + Q = 0. \quad (3.32)$$

MATLAB script can be used for this solution:

```matlab
[K,P,E] = lqr(A+beta*eye(n,n), B, Q, R);
```

The matrix $Q_\alpha$ is determined, with the values of $\alpha$, $\beta$ and $P$, using

$$Q_\alpha = Q + 2(\alpha + \beta)P.$$ 

The augmented state-space model $(A, B)$ is modified for use in the design. The matrix $B$ is unchanged, however, the matrix $A$ is modified to become

$$A - \alpha I$$

With this set of performance parameters $(Q_\alpha, R)$ and the design model $(A - \alpha I, B)$, the predictive control problem is converted back to the original problem stated in Sections 3.2-3.2.3.

**The Parameters in Laguerre Functions.** When $N$ increases, the predictive control trajectory converges to the underlying optimal control trajectory of the linear quadratic regulator. However, with a small $N$, the scaling factor in the Laguerre functions $p$ will affect the closed-loop response. $p$ could be chosen close to the smallest magnitude of the eigenvalue from the LQR design, then increasing the parameter $N$ until the closed-loop eigenvalues from predictive control system become close to those produced by the LQR.
system.
To demonstrate the effectiveness of the tuning procedure, the example presented in Table 3.1. For instance, if the parameters remain identical to the previous case, except the exponential data weighting with prescribed degree of stability is applied by choosing the tuning parameters $\alpha = 1.2$ and $\beta = 20$. The comparative results are illustrated in Figure 3.3. It is interesting to note that for this set of tuning parameters, the largest eigenvalue in the original design was located at $-0.0529$ and is shifted to $-20$ in the tuning procedure with $\alpha$ and $\beta$ parameters, but for those already located on the left of the $s = -\beta = -20$, little change occurred.

![Figure 3.3: Closed-loop eigenvalues for CMPC. Key: (1) without exponential data weighting; (2) $\alpha = 1.2$ and $\beta = 20$](image)

### 3.2.5 CMPC with Constraints

The strength of the continuous-time model predictive control system lies in the conceptual and computational simplicity when tackling the constrained control problem. It is paramount that suitable operational constraints are in place for the safety of the equipment. In this thesis, the constraints implementation is established by using on-line Quadratic Programming (QP) technique.

The input constraints on the control signal magnitude $u_{sd}(t)$ and $u_{sq}(t)$ will be imposed in the design and implementation. Assuming that the constraints are denoted for the
upper and lower limit of the control signal as

\[
[u_{sd}^{\min} u_{sq}^{\min}]; [u_{sd}^{\max} u_{sq}^{\max}]
\]

the following inequalities are specified for each control signal:

\[
\begin{align*}
    u_{sd}^{\min} &\leq u_{sd}(t) \leq u_{sd}^{\max} \\
    u_{sq}^{\min} &\leq u_{sq}(t) \leq u_{sq}^{\max}
\end{align*}
\]

From the optimal control section, the calculation of the amplitude of control signal is shown as

\[
u(t_i) = u(t_i - \Delta t) + L(0)^T \eta \Delta t
\]

where \( \Delta t \) is the sampling interval and \( L(0)^T \eta \) is the first element of the control derivative within the optimization window. Then, the inequality constraints become

\[
u^{\min} - u(t_i - \Delta t) \leq L(0)^T \eta \Delta t \leq u^{\max} - u(t_i - \Delta t)
\]

However, the above inequality only contains the initial information of the control trajectory, in the MPC constraints design, not only the present implemented control signal is limited, but the future predicted control trajectory is also considered within the constraints limit. Since at the arbitrary time \( \tau_i \), the control signal is obtained as

\[
u(\tau_i) = u(t_i) + \int_0^{\tau_i} L(\gamma)^T \eta d\gamma = u(t_i) + (L(\tau_i)^T - L(0)^T)A_p^{-T} \eta
\]

Finally, the constraints implementation is formulated as

\[
-C_u \eta \leq -u^{\min} + u(t_i - \Delta t) \\
C_u \eta \leq u^{\max} - u(t_i - \Delta t)
\]

where \( C_u = L(0)^T \Delta t + L(\tau_i)^T A_p^{-T} - L(0)^T A_p^{-T} \).

Then, the optimal solution of the MPC control with constraints is solved using quadratic programming, which has been extensively studied in the literature \([19, 93, 94]\). In this thesis, the detail of the QP theory is not included, but the relevant application will be discussed. The general expression of the objective function, with the decision variable \( x \), is obtained from the literature:

\[
J = \frac{1}{2} x^T E x + x^T F \\
M x \leq \lambda
\]
where $E$, $F$, $M$ and $\lambda$ are compatible matrices and vectors in QP problem. For MPC control situation, the problem is written as minimizing

$$J = \frac{1}{2} \eta^T \Omega \eta + \eta^T \Psi x(t)$$

subject to constraints

$$A_{cons} \eta \leq b$$

where

$$b = [-u_{min} + u(t_i - \Delta t) \quad u_{max} - u(t_i - \Delta t)]^T.$$ 

Thus, the quadratic programming procedure has to be computed on-line in real-time, due to the continuously updated state variable $x(t)$ and the previous sample of control signal $u(t_i - \Delta t)$. However, the computational load of the matrix operation is heavy, so that the Hildreth’s quadratic programming is introduced, to solve the optimal solution for one component only at a time, by using a for loop, the computational time could be reduced.

### 3.2.6 State Feedback Observer

In continuous-time MPC design, the state variable vector contains the derivatives of the current signals.

$$x(t) = [\dot{i}_{sd}(t) \quad \dot{i}_{sq}(t) \quad \dot{\psi}_{rd}(t) \quad \dot{\psi}_r(t) \quad \omega_r(t)]^T$$

Because these signals are generally noisy, differentiation of the current signals is to be avoided for the reason that the derivative operation will amplify the noise in the current signals. A strategy is thus to use a state observer for estimating the state variables, which is also acting as a filter to the measurement noise.

The observer equation for such an application has the following form:

$$\frac{d\hat{x}(t)}{dt} = A\hat{x}(t) + B\dot{u}(t) + K_{ob}(y(t) - C\hat{x}(t)) \quad (3.33)$$

where $\hat{x}(t)$ is an estimate of $x(t)$, $K_{ob}$ is the observer gain and $(A, B, C)$ are the system matrices of the augmented model. $\dot{u}(t)$ is obtained from the solution of the predictive control.

The observer gain matrix $K_{ob}$ is chosen according to closed-loop performance specification of the observer system and the pair of system matrices $(A, C)$ [92, 95]. For instance, the computation of the observer gain $K_{ob}$ could be performed using the MATLAB function lqr, as

$$K_{ob} = \text{lqr}(A', C', Q_{ob}, R_{ob})'$$

where $A$ and $C$ are system matrices of the augmented model, $Q_{ob}$ and $R_{ob}$ are weighting matrices of the observer. Design of an observer is a dual task of the design of a controller, thus the use of transposes of $A$ and $C$ matrices in the lqr function.
The continuous-time observer equation (3.34) is discretized for implementation, leading to

\[
\hat{x}(t_i + \Delta t) = \hat{x}(t_i) + (A\hat{x}(t_i) + Bu_i(t) + K_{ob}(y(t_i) - C\hat{x}(t_i)))\Delta t
\]

Thus, based on the current sample information of the optimal control solution \(\dot{u}(t_i)\) and the error signal \(y(t_i) - C\hat{x}(t_i)\), the next sample of state estimate \(\hat{x}(t_i + \Delta t)\) is computed.

To complete this section, Figure 3.1 is used to illustrate the configuration of the continuous-time MPC for speed control. The controlled outputs are the rotor flux \(\psi_{rd}\) and the mechanical motor speed \(\omega_m\), and the control signals are, \(u_{sd}\) and \(u_{sq}\). A Luenberger Observer is introduced for the estimation of the synchronous flux position \(\omega_s\) for \(d-q\) transformation, and the rotor flux \(\psi_{rd}\) for feedback. An observer is used to estimate the state variable vector \(x(t)\) in order to avoid differentiation of the current signals.

### 3.2.7 Simulation and Experimental Results

In this section, the continuous-time predictive control algorithm derived for induction motor is evaluated firstly using Simulink simulation program, secondly using the testbed. Since the settings of both simulation and experiment are explained in Appendix B, only the controller information is discussed here.

The steady-state parameters used to obtain the linear model for the induction motor are listed in Table 3.2. The Luenberger observer gain is defined as 1.3. In the design of the continuous-time predictive controller, the prediction horizon \(T_p\), the Laguerre parameters \(N_1 = N_2 = N\), \(p_1 = p_2 = p\), the exponential data weighting parameter \(\alpha\) and the prescribed degree of stability parameter \(\beta\) are also listed in Table 3.2. The weighting matrices \(R = I\) (\(I\) being the identity matrix) and \(Q = C^T C\) where \(C\) is the output matrix for the augmented model. Exponential data weighting is used to improve the numerical condition of the Hessian matrix \(\Omega\), where \(\alpha = 1.2\) is selected. Together with the prescribed degree of stability, the closed-loop eigenvalues of the predictive control system are positioned to the left of the \(s = -\beta\) line where \(\beta = 20\).

The constraints on both stator voltages are specified as

\[-90.1 \leq u_{sd} \leq 90.1; \quad -286.4 \leq u_{sq} \leq 286.4\]

Although the predictive controller is designed using the continuous-time model, the discretization occurs at the implementation stage. In general, a smaller sampling interval would be preferred for its results in a smaller approximation error associated with the discretization. However, due to the on-line computational cost that restricts how fast the sampling rate could be, the experimental setup only allows the sampling interval \(\Delta t\) not be less than 200\(\mu s\). Thus, in both simulations and experiments, the continuous-time
The model predictive controller is implemented using the lowest sampling interval possible ($\Delta t = 200\mu s$).

<table>
<thead>
<tr>
<th>$i_{sq}^0$</th>
<th>$T_L$</th>
<th>$\omega_r^0$</th>
<th>$\omega_s^0$</th>
<th>$\psi_{rd}^0$</th>
<th>$N$</th>
<th>$p$</th>
<th>$T_p$</th>
<th>$\alpha$</th>
<th>$\beta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.4194 A</td>
<td>0.5Nm</td>
<td>1400RPM</td>
<td>298.4 rad/s</td>
<td>0.6Wb</td>
<td>6</td>
<td>20</td>
<td>0.5</td>
<td>1.2</td>
<td>20</td>
</tr>
</tbody>
</table>

Table 3.2: Continuous-time MPC parameter definition

![Figure 3.4](image-url)

Figure 3.4: Simulation results of speed control using continuous-time MPC. Key: line(1) Actual feedback measurement; line (2) Set-point signal

**Simulation results.** The simulation results for the continuous-time model predictive control of induction motor are shown in Figure 3.4. As expected, the velocity response $\omega_m$ converges to the steady-state value of 1400RPM (see Figure 3.4(a)), while the rotor flux response $\psi_{rd}$ converges to 0.6Wb (see Figure 3.4(b)). Both output responses have no steady state error and have the settling times within 0.4 sec. Note that in the predictive controller design the dominant constant of the closed-loop predictive control system is approximately $1/\beta$. With $\beta = 20$, the closed-loop settling time could be estimated as $5/\beta = 0.25$ seconds. It seems that the settling time for both outputs is larger than 0.25 sec. This is because the constraint on $u_{sd}$ becomes active during the transient response (see Figure 3.4(c)) that resulted in slower closed-loop responses for both outputs. The
Chapter 3. *Centralized Model Predictive Control*

Stator currents are part of the state variables and there are no constraints imposed on the currents (see Figure 3.4(d)).

To demonstrate how the tuning parameter $\beta$ affects the closed-loop response, Figure 3.5 shows the closed-loop output responses for five different values of $\beta$ varying from 5 to 20. Because of the effects of the control signal constraints, although five sets of closed-loop poles, the closed-loop responses have similar response times for first four cases except the one associated with $\beta = 5$ which has longer setting time (see Figure 3.5). One comment is that with the continuous-time model predictive control system, the closed-loop response times for the electrical and the mechanical systems are in the same time scale, as demonstrated in the simulation studies.

In the experimental evaluation, the parameter for the prescribed degree of stability is selected as $\beta = 15$ while the rest of the parameters remain unchanged from the simulation evaluations. There are two sets of experimental results presented for the evaluation. The first set of control experiments is performed without the state observer where derivatives in the augmented state vector are calculated using their first order approximations:

$$\frac{dx_m(t)}{dt} = \frac{x_m(t_i) - x_m(t_i - \Delta t)}{\Delta t}$$

The second set of experiments is performed with a full state observer in which the observer is designed using MATLAB 'lqr' function with the weighting matrices chosen as $Q_{ob} = I$ and $R_{ob} = 10^{-5}I$. The experimental results for the first case are illustrated in Figure 3.6 and the second case in Figure 3.7. When comparing these two figures, it is seen that the closed-loop velocity response has about the same response time (see Figure 3.6(a) and 3.7(a)), the rotor flux response has a larger peak when the observer is used (see Figure 3.6(b) and 3.7(b)), the noise in the control signals has reduced when using observer (see Figure 3.6(c) and 3.7(c)). On the other hand, it seems that without the observer, although the noise effect is larger, the closed-loop dynamic system has a...
faster response. Particularly, this is seen in the response of the rotor flux, which has a smaller peak without using the observer (see Figure 3.6(b)). This is because the closed-loop poles of the state estimate predictive control system consist of the poles from the predictive control system as well as from the observer error system.

Overall, the experimental results are similar to the simulation results (see Figure 3.6 and 3.4). This means that the physical simulation model has a high fidelity when comparing with the actual test-bed.

Based on the results obtained in this section, the speed control using single MPC controller is evaluated using one operating condition. However, in the real life applications, the induction motor might be operated in different conditions. The research question remains how to linearize the induction motor model for the design of the linear MPC controller.
Chapter 3. Centralized Model Predictive Control

3.3 Gain-Scheduled Continuous-time MPC

In the section, the gain-scheduled method is proposed to counteract the time-varying model of the induction motor. The idea is to firstly design a set of linear controllers with respect to different operating conditions, then switch between these controllers according to the actual feedback of the system states.

Therefore, in order to apply the gain-scheduled method, a number of linear continuous-time MPC controllers are designed based on different operating conditions, then the controllers gain is switched based on the actual operating states, which is the mechanical motor speed in this case.

3.3.1 Introduction

From the previous section, the limitation of the controller design is that the operating speed and flux set-point has to be pre-defined, if speed reference changed to be different with the one defined in the model, then, the modelling error will be enlarged. Figure 3.8 illustrates the speed control with the variable set-point values by using the MPC
controller in the previous section, which has the operating condition defined at $\omega_0 = 1400\text{rpm}$ and the rest parameters are defined as identical to previous Table 3.2. The oscillations are obtained at the low speed range, since the set-point value is far away from the pre-defined value in the model that is used for control design. As a comparison, the result is much better when the motor is operating at the speed that is close to the pre-defined operating condition. Therefore, it is not appropriate to control the induction motor over all speed range using a single linear MPC controller. By applying gain-scheduling method, a number of different operating conditions or equilibrium points are pre-defined, in order to design number of linear MPC controllers, according to the real-time state, the feedback controller gain will alter respectively.

However, the mathematical model used previously has two time-varying parameters: $\omega_s(t)$ and $\omega_e(t)$. From Chapter 2, the alternative linearized model has been derived with only one determinate parameter $\omega_e^0$, since one-dimension switching is much simpler than two-dimension. Additionally, the parameter $\omega_s(t)$ is not measurable in the real-time implementation. Thus, the following linear model is applied for the Gain-Scheduled...
continuous-time MPC design.

\[
\begin{align*}
\frac{dx_m(t)}{dt} &= A_m x_m(t) + B_m u(t) + \mu^0 \\
y(t) &= C_m x_m(t)
\end{align*}
\]

where \( x_m(t) = [i_{sd}(t) \ i_{SQ}(t) \ \omega_e(t)]^T \), \( u(t) = [u_{sd}(t) \ u_{SQ}(t)]^T \), and with the coefficient \( \kappa_t = \frac{Sz_p L_h}{J r_f J_m} \), the matrices \( A_m \) and \( B_m \) are defined as

\[
A_m = \begin{bmatrix}
-\frac{1}{\tau_r} & \frac{\omega_e^0}{\tau_e} + \frac{2 L_h \psi_{rd}^0}{\tau_e} & -\frac{k_r}{\tau_e \tau_f} & \frac{L_h \psi_{rd}^0}{\tau_e} - \frac{L_h \psi_{rd}^0}{\tau_e} \psi_{rd}^0 & \frac{d \psi_{rd}^0}{dt} - \frac{k_r}{\tau_e \tau_f} \psi_{rd}^0 - \frac{f_d}{J_m} \\
0 & \frac{k_r}{\tau_e \tau_f} & \frac{1}{\tau_e} & 0 & 0 \\
0 & \frac{L_h}{\tau_e} & Z_p \kappa_t \psi_{rd}^0 & \frac{Z_p \kappa_t \psi_{rd}^0}{\tau_e} & 0
\end{bmatrix}
\]

\[
B_m = \begin{bmatrix}
\frac{1}{\tau_r \tau_f} \\
0 & \frac{1}{\tau_r \tau_f} \\
0 & 0 \\
0 & 0
\end{bmatrix}; C_m = \begin{bmatrix} 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

The constant bias vector \( \mu^0 \), consisting of the steady-state parameters, is given by,

\[
\mu^0 = \begin{bmatrix}
-\omega_e^0 \psi_{rd}^0 + \frac{L_h}{\tau_e} \psi_{rd}^0 - \frac{L_h}{\tau_e} \frac{1}{\psi_{rd}^0} \psi_{rd}^0 \\
-\omega_e^0 \psi_{rd}^0 + \frac{k_r}{\tau_e \tau_f} \psi_{rd}^0 \\
0 & 0 & 0
\end{bmatrix} = \begin{bmatrix}
\psi_{rd}^0 \\
0 \\
0
\end{bmatrix} = \begin{bmatrix}
\frac{\psi_{rd}^0}{L_h} \\
0 \\
0
\end{bmatrix}
\]

Inside the model, the operating conditions of parameters \( \omega_e^0, \psi_{rd}^0, \psi_{sd}^0 \) and \( \psi_{SQ}^0 \) need to be pre-defined, since \( \psi_{rd}^0 \) is equal to the set-point value, \( \psi_{sd}^0 \) is derived based on model equation (2.23) where \( \frac{d \psi_{rd}(t)}{dt} = 0 \) in the steady-state.

\[
\psi_{sd}^0 = \frac{\psi_{rd}^0}{L_h}
\]

the \( \psi_{sd}^0 \) value definition is relatively complicated, which is derived based on the mechanical equation (2.30) where \( \frac{d \omega_e(t)}{dt} = 0 \).

\[
\psi_{sd}^0 = \frac{(f_d / J_m) \omega_e^0 + T_L}{k_t \psi_{rd}^0}
\]

Furthermore, the selection of the pre-defined operating conditions and the covered range of these operating conditions should be considered before the control design. The decision is made by obtaining the closed-loop eigenvalues from

\[
eig\{A(t) - BK_{mpc}^0\}
Chapter 3. Centralized Model Predictive Control

Figure 3.9: Determinant closed-loop eigenvalues of $K_{mpc}^0$ with $A(t)$

where $A(t)$ is the augmented system matrix with time-varying $\omega_r(t)$ and $K_{mpc}$ is the feedback gain of the MPC controller computed based on a pre-defined $\omega_0^r$.

The impact of the time varying system matrix $A(t)$ will reflect on the location of closed-loop eigenvalues, where the time-varying parameter $\omega_r(t)$ is chosen from 100rpm to 2000rpm with an increment of 100rpm, $\omega_e(t) = \omega_r(t) \times Z_p \times \frac{2\pi}{60}$ rads/s. Meanwhile, the feedback predictive controller gain $K_{mpc}$ is derived based on linearised model with $\omega_0^r = 1400$rpm and the rest parameters are defined to be identical to Table 3.2. The eigenvalues are plotted within one complex plane as shown in Figure 3.9, note that only the determinant poles are displayed.

Inside of the graph, the blue crosses represent the eigenvalues of the system matrix $A(t)$ with $\omega_0^r = 1400$rpm. Inside the low speed range of the dynamic model, the closed-loop control system tends to be unstable, with the prescribed degree of stability at $\beta = 20$, the closed-loop eigenvalues across the $-\beta$ line for the mathematical model with $\omega_r(t)$ less than 400 rpm. Finally, the closed-loop system is unstable when $\omega_r(t) = 100$ rpm.

For the higher speed region, the imaginary part of the closed-loop eigenvalues increases progressively, which will lead to large overshoot and longer response time as well as the oscillation in the closed-loop response.

Therefore, the operating conditions are chosen as low-speed operation ($\omega_l^e$), median speed operation ($\omega_m^e$) and high speed operation ($\omega_h^e$). With the three operating conditions selected and steady-state parameters defined, three linear time invariant models are derived for this situation. Since the electrical rotor speed $\omega_e(t)$ is the key parameter that leads to the time variation for the model, this parameter, which is measurable in real-time feedback, will be used as the identifier for the operating conditions of the induction motor. In addition, it is worthwhile to note that the variables in the linear time-invariant model (3.35) are the actual physical variables as their steady-state values have not been subtracted. However, the model is only valid at the operating conditions
defined by the set of steady-state parameters, \( \omega_s^0, \omega_e^0, i_{sq}^0, i_{sd}^0, \psi_{rd}^0 \).

### 3.3.2 Revised Approach of Constraints Implementation

In the case of induction motor control, all constraints under consideration are input variable constraints, which are the stator voltages \( u_{sd} \) and \( u_{sq} \). Assuming that the DC-bus voltage is supplied with \( V_{dc} \) Volt, and with the modulation limitation, the manipulated variables are constrained by the following relation:

\[
\sqrt{u_{sd}^2 + u_{sq}^2} \leq \frac{V_{dc}}{\sqrt{3}}
\] (3.36)

This is a quadratic constraint with respect to the input variables. The constrained predictive control problem becomes quadratic optimization subject to quadratic constraints. The solution to this nonlinear constrained optimization problem demands a substantial amount of on-line computational power, and the nonlinear optimizer also complicates the real-time implementation of the gain scheduled predictive control algorithm. The proposed solution is to approximate the quadratic constraint (3.36) using eight linear approximate constraints so that the constrained predictive control problem can be solved using a Quadratic Programming (QP) procedure for on-line implementation.

Figure 3.10 shows the constraint equation (3.36), which is the area of a circle with the radius of \( \frac{V_{dc}}{\sqrt{3}} \). In order to obtain the linear approximation, the circular area is approximated using the area of an octagon as shown in Figure 3.10. For notational simplicity, \( u_{sd} \) and \( u_{sq} \) are denoted as \( f \) and \( g \), and a unit circle is chosen for the initial analysis. On Figure 3.10, the eight pairs of the coordinates are marked in anti-clock wise manner, and similarly marked are the eight straight lines. The values of the coordinates are given in Table 3.3.

<table>
<thead>
<tr>
<th>( k )</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f_k )</td>
<td>1</td>
<td>-\frac{\sqrt{2}}{2}</td>
<td>0</td>
<td>-1</td>
<td>-\frac{\sqrt{2}}{2}</td>
<td>0</td>
<td>-\frac{\sqrt{2}}{2}</td>
<td>0</td>
</tr>
<tr>
<td>( g_k )</td>
<td>0</td>
<td>1</td>
<td>\frac{\sqrt{2}}{2}</td>
<td>0</td>
<td>-\frac{\sqrt{2}}{2}</td>
<td>-1</td>
<td>-\frac{\sqrt{2}}{2}</td>
<td>-1</td>
</tr>
</tbody>
</table>

Table 3.3: The values of the coordinates
There are four inequalities associated with the upper part of the circle, and four reversed inequalities associated with the lower part of the circle. For \( k = 1, 2, 3, 4 \), the inequalities for the upper part of the circle are expressed as

\[
g - \frac{g_k - g_{k+1}}{f_k - f_{k+1}} f \leq -\frac{g_k - g_{k+1}}{f_k - f_{k+1}} f_k + g_k \tag{3.37}
\]

For \( k = 5, 6, 7 \), the inequalities for the lower part of the circle are expressed as

\[
g - \frac{g_k - g_{k+1}}{f_k - f_{k+1}} f \geq -\frac{g_k - g_{k+1}}{f_k - f_{k+1}} f_k + g_k \tag{3.38}
\]

For \( k = 8 \), the inequality is

\[
g - \frac{g_8 - g_1}{f_8 - f_1} f \geq -\frac{g_8 - g_1}{f_8 - f_1} f_8 + g_8 \tag{3.39}
\]

The linear inequalities can be written in general forms as

\[
\alpha_k f + \beta_k g \leq \gamma_k \tag{3.40}
\]

or

\[
\alpha_k f + \beta_k g \geq \gamma_k \tag{3.41}
\]

where \( \alpha_k = -\frac{g_k - g_{k+1}}{f_k - f_{k+1}} \), \( \beta_k = 1 \) and \( \gamma_k = -\frac{g_k - g_{k+1}}{f_k - f_{k+1}} f_k + g_k \) for \( k = 1, 2, 3, 4, 5, 6, 7 \). For \( k = 8 \), \( \alpha_k = -\frac{g_8 - g_1}{f_8 - f_1} \), \( \beta_k = 1 \) and \( \gamma_k = -\frac{g_8 - g_1}{f_8 - f_1} f_8 + g_8 \). Note that the parameters \( \alpha_k \) and \( \beta_k \) are independent of the radius of the circle, however, the parameter \( \gamma_k \) is
proportional to the radius, which is in the case of induction motor control, $\frac{V_d}{\sqrt{3}}$. Thus, all $\gamma_k$, $k = 1, 2, \ldots, 8$ will multiply the radius to obtain their actual values for the specific application.

For the induction motor control problem, $f = u_{sd}(t)$ and $g = u_{sq}(t)$. The next task is to reformulate the inequalities with the Laguerre coefficient vector $\eta$ so that the linear inequalities become the linear inequality constraints in the design of predictive control.

Taking the example of first four inequalities, by imposing the constraints on the first sample of the control signals, at the sampling instant $t_i$, the inequalities are

$$ \alpha_k u_{sd}(t_i) + \beta_k u_{sq}(t_i) \leq \gamma_k $$  \hspace{1cm} (3.42)

for $k = 1, 2, 3, 4$. Since the control variables are related to the Laguerre coefficient vector through

$$ u_{sd}(t_i) = u_{sd}(t_i - \Delta t) + L_1(0)^T \eta_1 \Delta t $$ \hspace{1cm} (3.43)

$$ u_{sq}(t_i) = u_{sq}(t_i - \Delta t) + L_2(0)^T \eta_2 \Delta t $$ \hspace{1cm} (3.44)

where $L_1(0)^T = \sqrt{2p_1} \begin{bmatrix} 1 & 1 & \ldots & 1 \end{bmatrix}$ and $L_2(0)^T = \sqrt{2p_2} \begin{bmatrix} 1 & 1 & \ldots & 1 \end{bmatrix}$; $\eta_1$ and $\eta_2$ are the Laguerre coefficient vectors for $\dot{u}_{sd}$ and $\dot{u}_{sq}$ respectively. By substituting (3.43) and (3.44) into (3.42), the following inequality expression is obtained:

$$ \alpha_k L_1(0)^T \Delta t \eta_1 + \beta_k L_2(0)^T \Delta t \eta_2 \leq \gamma_k - \alpha_k u_{sd}(t_i - \Delta t) - \beta_k u_{sq}(t_i - \Delta t) $$ \hspace{1cm} (3.45)

which is, in vector form,

$$ \begin{bmatrix} \alpha_k L_1(0)^T \Delta t & \beta_k L_2(0)^T \Delta t \end{bmatrix} \begin{bmatrix} \eta_1 \\ \eta_2 \end{bmatrix} \leq \gamma_k - \begin{bmatrix} \alpha_k & \beta_k \end{bmatrix} \begin{bmatrix} u_{sd}(t_i - \Delta t) \\ u_{sq}(t_i - \Delta t) \end{bmatrix} $$ \hspace{1cm} (3.46)

Similarly, for $k = 5, 6, 7, 8$, the kth inequality is expressed as

$$ \begin{bmatrix} \alpha_k L_1(0)^T \Delta t & \beta_k L_2(0)^T \Delta t \end{bmatrix} \begin{bmatrix} \eta_1 \\ \eta_2 \end{bmatrix} \geq \gamma_k - \begin{bmatrix} \alpha_k & \beta_k \end{bmatrix} \begin{bmatrix} u_{sd}(t_i - \Delta t) \\ u_{sq}(t_i - \Delta t) \end{bmatrix} $$ \hspace{1cm} (3.47)

which is equivalent to

$$ -\begin{bmatrix} \alpha_k L_1(0)^T \Delta t & \beta_k L_2(0)^T \Delta t \end{bmatrix} \begin{bmatrix} \eta_1 \\ \eta_2 \end{bmatrix} \leq -\gamma_k + \begin{bmatrix} \alpha_k & \beta_k \end{bmatrix} \begin{bmatrix} u_{sd}(t_i - \Delta t) \\ u_{sq}(t_i - \Delta t) \end{bmatrix} $$ \hspace{1cm} (3.48)

By combining the inequalities represented by (3.46) and those by (3.48), and writing them in a vector form, the linear inequality constraints for the design of model predictive
control are obtained:

\[
\begin{pmatrix}
\alpha_1 & \beta_1 \\
\alpha_2 & \beta_2 \\
\alpha_3 & \beta_3 \\
\alpha_4 & \beta_4 \\
-\alpha_5 & -\beta_5 \\
-\alpha_6 & -\beta_6 \\
-\alpha_7 & -\beta_7 \\
-\alpha_8 & -\beta_8
\end{pmatrix}
\begin{bmatrix}
L_1(0)^T & O_2 \\
O_1 & L_2(0)^T
\end{bmatrix}
\Delta t\eta \leq
\begin{bmatrix}
\gamma_1 \\
\gamma_2 \\
\gamma_3 \\
\gamma_4 \\
-\gamma_5 \\
-\gamma_6 \\
-\gamma_7 \\
-\gamma_8
\end{bmatrix}
+ \begin{bmatrix}
-\alpha_1 & -\beta_1 \\
-\alpha_2 & -\beta_2 \\
-\alpha_3 & -\beta_3 \\
-\alpha_4 & -\beta_4 \\
\alpha_5 & \beta_5 \\
\alpha_6 & \beta_6 \\
\alpha_7 & \beta_7 \\
\alpha_8 & \beta_8
\end{bmatrix}
\begin{bmatrix}
u_{sd}(t_i - \Delta t) \\
u_{sd}(t_i - \Delta t)
\end{bmatrix}
\tag{3.49}
\]

With the constraints formulated, the continuous-time predictive control problem is expressed as minimizing the cost function \( J \) \((3.26)\) subject to the set of linear inequality constraints (see \((3.49)\)) in real-time, which is solved using a quadratic programming algorithm \((96)\). Moreover, the on-line computational algorithms via finding the active constraints proposed in \((19, 97)\) have been tested in many real-time applications to achieve reliable and fast solutions of the quadratic programming problem, as demonstrated in this thesis.

### 3.3.3 Gain-Scheduled Predictive Controller

Since the plant is a nonlinear system, its linearized model is dependent on its operating conditions. It is apparent that for a linear model predictive controller to work in a wide range of operating conditions a gain scheduled predictive controller is needed, where several linear time-invariant model predictive controllers will be designed for the operating conditions. This is the actual design of the gain scheduled controller that will have the capability to interpret the linear predictive controllers and the mechanisms to ensure the continuity of the control signals.

One approach used in the design of a gain scheduled control system is to assign a set of weighting parameters with values between 0 and 1 that will correspond to each operating conditions of the nonlinear system. Here, the parameters \( \lambda^l \), \( \lambda^m \) and \( \lambda^h \) are used as the weights for low speed, median speed and high speed operations. In the literature \([48]\), there are two widely used approaches to calculate the weighting parameters. The first approach, also the simplest approach, is to assign their values according to the set-point signal of the system. For instance, when the set-point velocity signal \( \omega^0 \) is at the low speed where \( \omega^0 = \omega^l \), \( \lambda^l = 1 \), \( \lambda^m = 0 \) and \( \lambda^h = 0 \); when the desired velocity is at the median speed \( \omega^m = \omega^m \), \( \lambda^l = 0 \), \( \lambda^m = 1 \) and \( \lambda^h = 0 \); when the desired velocity is at the high speed where \( \omega^h = \omega^h \), \( \lambda^l = 0 \), \( \lambda^m = 0 \) and \( \lambda^h = 1 \). This approach has taken into consideration of the changes of plant dynamics due to set-point changes, however, it does...
not consider the possibility that the disturbances could cause the significant changes in plant dynamics. Hence, with this simple approach, closed-loop instability could occur if severe disturbances were encountered in plant operation.

The more general approach is to compute the weighting parameters $\lambda^l$, $\lambda^m$ and $\lambda^h$ according to the actual measurement of velocity $\omega_e$. In order to avoid random triggering of the model changes in the presence of noises and transient responses, a band is formed around the desired speed. By assigning a constant $\delta$ to the tolerance, the weighting constants are defined as

$\begin{align*}
-\delta + \omega^l_e &\leq \omega_e \leq \omega^l_e + \delta & \lambda^l = 1; \lambda^m = 0; \lambda^h = 0 \\
-\delta + \omega^m_e &\leq \omega_e \leq \omega^m_e + \delta & \lambda^l = 0; \lambda^m = 1; \lambda^h = 0 \\
-\delta + \omega^h_e &\leq \omega_e \leq \omega^h_e + \delta & \lambda^l = 0; \lambda^m = 0; \lambda^h = 1
\end{align*}$

Outside the band of the desired speed, neither linear models can accurately describe the dynamic system. A traditional way is to use a combination of these two models from the close regions. For instance, assuming that the actual operating condition is between the band of the desired median speed and that of the desired high speed ($\omega^m_e + \delta \leq \omega_e(t) < \omega^h_e - \delta$), by defining $\lambda^h (0 \leq \lambda^h \leq 1)$ as a function of $\omega_e(t)$, $\lambda^h(t)$ is calculated using the linear interpretation of the two boundaries between the median and high speeds given by

$$\lambda^h(t) = \frac{\omega_e(t) - \omega^m_e - \delta}{\omega^h_e - \omega^m_e - 2\delta} \; \text{(3.50)}$$

The weighting parameter $\lambda^m$ follows as $\lambda^m = 1 - \lambda^h (0 \leq \lambda^m \leq 1)$, and $\lambda^l = 0$ for this region. Similarly, for $\omega^l_e + \delta \leq \omega_e(t) < \omega^m_e - \delta$,

$$\lambda^m(t) = \frac{\omega_e(t) - \omega^l_e - \delta}{\omega^m_e - \omega^l_e - 2\delta} \; \text{(3.51)}$$

and $\lambda^l(t) = 1 - \lambda^m(t)$, $\lambda^h = 0$.

Figure 3.11 illustrates the weighting parameters that have been used to represent the operating regions of the induction motor.
Figure 3.11: Weighting parameters

The weighting parameters will be used in the gain scheduled predictive control law to ensure correct selection of the linear model and bumpless transfer from one predictive controller to another.

Assume that there are three linearized models obtained for the three operating conditions with their system matrices denoted by \((A_l, B_l, C_l)\), \((A_m, B_m, C_m)\), and \((A_h, B_h, C_h)\), respectively. Note that the previous cost function \((3.26)\) is based on a linearized model from a single operating condition. The cost function for the gain scheduled model predictive control is a combination of cost functions generated for different operating conditions. By using the weighting parameters \(\lambda_l\), \(\lambda_m\) and \(\lambda_h\), the cost function is chosen as

\[
J = \lambda_l (\eta^T \Omega_l \eta + 2 \eta^T \Psi_l \hat{x}(t_i)) + \lambda_m (\eta^T \Omega_m \eta + 2 \eta^T \Psi_m \hat{x}(t_i)) + \lambda_h (\eta^T \Omega_h \eta + 2 \eta^T \Psi_h \hat{x}(t_i))
\]

where \(\hat{x}(t_i) = x(t_i) - x^0\), \(\Omega_l\), \(\Omega_m\), \(\Omega_h\), \(\Psi_l\), \(\Psi_m\) and \(\Psi_h\) are the predictive control parameter matrices, computed on the basis of the model parameters for low, median and high speed of the induction motor. If the reference signals to the system change, then the components in \(x^0\) that correspond to the reference signals will change accordingly. By similar definition of operational constraints, the Laguerre parameter vector \(\eta\) is found by minimizing the cost function \(J\) \((3.52)\) subject to approximated operational constraints \((3.49)\). With the optimal Laguerre coefficients vector, the control signal for the gain scheduled predictive controller is realized at sample time \(t_i\) via

\[
u(t_i) = u(t_{i-1}) + L(0)^T \eta \Delta t\]

The control signal is ensured not to have a sudden jump effect when the operating condition changes. This is because the computation of the control signal using \((3.53)\) is
based on the past sample of the control signal and the derivative of the current control, and when \( \Delta t \to 0, u(t_i) = u(t_{i-1}) \).

When an observer is used to estimate the state variable vector \( x(t) \), it is designed for each operational condition and implemented using the linear interpretation in the same manner as the predictive controller.

Assume that the observer gains are \( K^l_{ob}, K^m_{ob} \) and \( K^h_{ob} \) that are designed using \((A^l, C)\), \((A^m, C)\), and \((A^h, C)\), respectively. The three observer equations are

\[
\frac{d\hat{x}(t)}{dt}^i = A^i\dot{x}(t) + B^i\dot{u}(t) + K^l_{ob}(y(t) - C\hat{x}(t)) \quad (3.54)
\]

\[
\frac{d\hat{x}(t)}{dt}^m = A^m\dot{x}(t) + B^m\dot{u}(t) + K^m_{ob}(y(t) - C\hat{x}(t)) \quad (3.55)
\]

\[
\frac{d\hat{x}(t)}{dt}^h = A^h\dot{x}(t) + B^h\dot{u}(t) + K^h_{ob}(y(t) - C\hat{x}(t)) \quad (3.56)
\]

At any given time \( t \), there is only one state vector \( \hat{x}(t) \), however different are the models. Thus, we assign the common state vector \( \hat{x}(t) \) as the weighted outcomes of the estimated states from \( (3.54)-(3.56) \). Similar to the expression of the cost function, the gain scheduled estimated state vector \( \hat{x}(t) \) is written as

\[
\frac{d\hat{x}(t)}{dt} = \lambda^l(A^l\dot{x}(t) + B^l\dot{u}(t) + K^l_{ob}(y(t) - C\hat{x}(t))) + \lambda^m(A^m\dot{x}(t) + B^m\dot{u}(t) + K^m_{ob}(y(t) - C\hat{x}(t))) + \lambda^h(A^h\dot{x}(t) + B^h\dot{u}(t) + K^h_{ob}(y(t) - C\hat{x}(t)))
\]

\[
(3.57)
\]

When the differential equation is discretized using first order forward difference approximation, it is written for implementation as

\[
\hat{x}(t_i + \Delta t) = \hat{x}(t_i) + (\lambda^l A^l + \lambda^m A^m + \lambda^h A^h)\hat{x}(t_i)\Delta t + (\lambda^l B^l + \lambda^m B^m + \lambda^h B^h)\dot{u}(t_i)\Delta t + (\lambda^l K^l_{ob} + \lambda^m K^m_{ob} + \lambda^h K^h_{ob})(y(t_i) - C\hat{x}(t_i))\Delta t
\]

\[
(3.58)
\]

This implementation of observer also ensures that the observer states will not sudden jump when the operating condition changes, as when \( \Delta t \to 0, \hat{x}(t_i + \Delta t) = \hat{x}(t_i) \).

### 3.3.4 Simulation and Experimental Results

To evaluate the proposed algorithms, one simulation and two experimental results are presented in this section, thus the speed control of wide speed range and proposed on-line constraints implementation are both evaluated.

There are three linear models used in the gain scheduled model predictive control system. The operating condition for low speed operation is selected as \( \omega^l = 200 \) rpm; for median speed operation is \( \omega^m = 700 \) rpm; and for high speed operation is \( \omega^h = 1400 \) rpm.
In order to simplify the controller tuning procedure and also to improve the numerical condition of the Hessian matrix of the predictive controller, the procedure for prescribed degree of stability with exponential data weighting outlined in [19] is deployed to position the closed-loop poles of the predictive controller at the left-hand side of $-\beta$ line ($\beta \geq 0$).

For all three predictive controllers, the weighting matrices $R = I$ ($I$ being the identity matrix) and $Q = C^TC$ where $C$ is the output matrix for the augmented model. The sampling interval is 200$\mu$s, a constant load torque is applied at 0.5 Nm and the gain of
Luenberger Observer is 1.3. The remaining parameters used in the implementation of the gain scheduled control system are listed in Table 3.4.

![Graphs showing experimental results for speed control, flux control, control signals, and current measurement.](image)

**Figure 3.13:** Experimental result of speed control using Gain-Scheduled MPC. Key: line(1) Actual feedback measurement; line (2) Set-point signal

**Speed control evaluation.** The reference signal of velocity is kept identical to the example case in Figure 3.8, in order to have a comparison between single model designed MPC controller and gain scheduled predictive controller. The simulation results are presented in Figure 3.12, while the experimental results are presented in Figure 3.13. Both sets of results achieve the set-points following with zero steady-state error, the control performance is significantly improved comparing to Figure 3.8. Note that, the load torque is a constant value in the simulation, while the load is realized using a coupled DC servo motor in the experiment, thus the load torque is altered directly proportional to the motor shaft velocity. It explains why the overshoot at $t = 4s$ in Figure 3.12(a) is vanished in the experimental result Figure 3.13(a), moreover, the overshoesss size is significantly larger in the simulation results shown in Figure 3.12(c) than the experimental results shown in Figure 3.13(c). Another difference is the noise level, especially in the current measurement. Overall, the Gain-Scheduled MPC is evaluated according to its control objective, which is the predictive control of speed at different operating conditions.
The reference signal of velocity is defined at three different set-points, which are altered to the predefined operating conditions of models. The feedback control performance is dependent on the modelling error and closed-loop controller gain; if the modelling error between the reference and the operating condition is large, then the feedback controller gain is limited to a small value or otherwise unstable according to the open-loop model dynamics, which is the reason which the $\beta$ value could be defined larger in the Gain-Scheduled MPC case.

![Figure 3.14: Experimental result of speed control using Gain-Scheduled MPC with constraints. Key: line(1) Feedback measurement without constraints; line (2) Set-point signal; line(3) Constraints case feedback](image)

**Constraints evaluation.** In the previous case, the constraints are not active because of the sufficiently large capacity of the power supply. Thus, the $V_{dc}$ value is reduced to demonstrate the situation where a smaller power supply is used and the nonlinear constraint becomes active in the transient response to a large reference signal change. The nonlinear constraint,

$$\sqrt{u_{sd}^2 + u_{sq}^2} \leq 140V$$

is imposed in the implementation.

The speed reference signal makes a large step change from 300 rpm to 1600 rpm at time $t = 4$ second (see Figure 3.14(a) for velocity and Figure 3.14(b) for flux). The
large reference step change in the velocity causes the control signals to increase (see Figure 3.14(c)) and as a result the quantity $\sqrt{u_{sd}^2 + u_{sq}^2}$ suddenly increases to about 190 (see Figure 3.14(d)). When the nonlinear constraint is imposed in the operation, control signals $u_{sd}(t)$ and $u_{sq}(t)$ are found such that the quantity $\sqrt{u_{sd}^2 + u_{sq}^2} \leq 140$. It is seen from Figure 3.14(d) that the constraint becomes active during the transient period and the quantity $\sqrt{u_{sd}^2 + u_{sq}^2}$ converges to a steady-state value afterwards. Figure 3.14 compares the closed-loop response under the constrained control with the response obtained without constraints. It seems that imposing the nonlinear constraint has changed the characteristics of the closed-loop dynamics in the sense that the response are less oscillatory for this case.

### 3.4 Summary

This Chapter has presented the centralized continuous-time model predictive control application of induction motor drive with experimental validations. In the continuous-time design, the linearized continuous-time models are directly used in the design stage, and at the later stage, the control signals are discretized for digital implementation. This perhaps offers some advantages at the applications of the induction motor drive control where the time constants are significantly smaller than those from the mechanical systems and choice of sampling interval requires compromise between the larger and small time constants. With the continuous-time design, it is possible to use a dual sampling rate in the implementation stage, for instance, a much faster sampling rate for the current measurement signals and a slower sampling rate for the velocity measurement signal. The following chapter will employ this strategy for the cascaded control structure. Another advantage of using the continuous-time predictive control design is its simple extension to Gain-Scheduled Model Predictive Control. When using the traditional continuous-time model predictive control, the non-linear physical models are linearized at several operating conditions and a family of linear model predictive controllers are designed. The Gain-Scheduled MPC is shown to automatically interpret these linear model predictive controllers. The experimental results obtained from gain scheduled predictive control of induction motor has shown its efficacy in controlling the AC drives. The continuous-time model predictive controller is conceptually more complex than its discrete-time counterpart because Laguerre functions are used in the description of the control trajectories. However, once this is overcome, the actual computational algorithms of the predictive controllers are available in MATLAB for the design and implementation (see [19]).
Chapter 4

Cascaded Model Predictive Control

4.1 Introduction

In this chapter, the continuous-time MPC control of induction motor in cascaded structure is discussed. Additionally, both speed control and position control applications are designed and implemented.

The continuous-time model predictive control algorithm follows the previous chapter, thus the control design will not be repeated in this chapter. The cascaded structure is based on the indirect Field Oriented Control (FOC) technique from literature [84]. The rotor flux $\psi_{rd}$ is controlled using the open loop model, thus the Luenberger observer is not used, in order to reduce the impact on the control performance from the observer. However, the position information of the $dq$ coordinates is still needed for Park-Clarke transformation and the slip estimation method is introduced in Section 4.2 to obtain $\theta_s(t)$. Then, Section 4.3 presents the velocity control using two MPC controllers in a cascaded structure. Moreover, the position control of induction motor is discussed in Section 4.4 using the same control structure.

4.2 Slip Estimation

The propose of slip estimation is to determine the position angle of $d$-axis of the $dq$ coordinates, in order to transform the space vectors from the fixed reference frame to the rotating frame using Park Transformation and vice versa. From the literature, there are different types of observer, such as Luenberger, Sliding mode and Kalman Filter,
are developed to observe the rotor flux vector, that contains the information about both magnitude and position angle. However, these observers are normally applied in direct vector control structure, which has closed-loop control of the rotor flux, thus the magnitude of rotor flux is required in the feedback. For indirect field oriented control, only position angle is needed, following method is introduced for the position angle \( \theta_s \) estimation.

Based on model equation (2.24), the rotational angular velocity of the \( dq \) coordinates \( \omega_s \) is calculated based on the rotor speed and stator currents.

\[
\omega_s(t) = \omega_e(t) + \frac{L_h i_{sq}(t)}{\tau_r \psi_{rd}(t)}
\]

From model equation (2.23), the relationship between the stator current \( i_{sd} \) and rotor flux \( \psi_{rd} \) is described using first order transfer function as follows

\[
\frac{\Psi_{rd}(s)}{I_{sd}(s)} = \frac{L_h}{1 + \tau_r s}
\]

During steady-state (ie. \( s = 0 \)), \( \psi_{rd}^{ss} = L_h \times i_{sd}^{ss} \), thus the following approximation is obtained

\[
\omega_s(t) = \omega_e(t) + \frac{1}{\tau_r} \frac{i_{sd}(t)}{i_{sq}(t)}
\]  

(4.1)

Since the closed-loop current control will conduct even faster dynamic response, the values of \( i_{sd}(t) \) and \( i_{sq}(t) \) in above equation could be substituted by their reference signals to the current controllers. As a results, the unnecessary noise and system dynamics of the actual current feedback will be neglected. The slip estimation equation will lead to

\[
\dot{\theta}_s(t) = \theta_e(t) + \frac{1}{\tau_r} \int_{0}^{t} \frac{i_{sq}^{*}(\tau)}{i_{sd}^{*}(\tau)} d\tau
\]  

(4.2)

where \( \theta_e(t) \) is the electrical rotor position angle equal to \( Z_p \times \theta_r(t) \).

Discretization of (4.2) gives the numerical solution of \( \dot{\theta}_s(t) \) that can be implemented in digital environment control system:

\[
\hat{\theta}_s(t_i) = \theta_e(t_i) + \frac{1}{\tau_r} \sum_{k=0}^{M-1} \frac{i_{sq}^{*}(t_k)}{i_{sd}^{*}(t_k)} \Delta t
\]

where \( t_i = t_0 + (M - 1) \times \Delta t \) and \( M \) is the number of samples from time \( t_0 \) to the time \( t_i \).
4.3 Cascaded MPC Speed Control

In this section, the speed control is achieved by using two predictive controllers in cascaded structure, as a result the electrical and mechanical dynamic model are separated for inner-loop and outer-loop control, respectively. The motivation is because that the inner-loop electrical model has much faster dynamics comparing to the outer-loop, that is $\tau'_r << \frac{L_m}{j_m}$. Thus, the control signal from outer-loop controller $i_{sq}$ will act as the reference signal $i_{sq}^*$ for inner-loop control system. If the inner closed-loop control could converge to its desired steady-state much faster than the outer-loop closed-loop dynamics, the inner-loop dynamics will not significantly affect the outer-loop control. In other words, the bandwidth of the outer-loop dynamics has to be sufficiently larger than the inner-loop, in order to obtain the stable control performance.

The control design, that contains the inner-loop current control and the outer-loop speed control, has the block diagram as shown in Figure 4.1. There are two continuous-time model predictive controllers for controlling currents and motor velocity, respectively. The rotor flux is considered under open-loop control by using the model equation (2.23), in order to determine the set-point signal for $d$-axis current $i_{sd}$,

$$I_{sd}(s) = \frac{1 + \tau r_s}{L_h} \Psi_{rd}(s)$$

thus, the implementation at arbitrary time $t_i$ is obtained as

$$i_{sd}^*(t_i) = \frac{\tau_r \psi_{rd}(t_i) - \psi_{rd}(t_i - \Delta t)}{L_h} + \frac{\psi_{rd}(t_i)}{L_h}$$
Furthermore, the slip estimation is implemented based on the reference signals for inner-loop control and the real-time position of the motor shaft. The inner-loop and out-loop control designs are discussed separately in this section.

4.3.1 Inner-loop Current Control

The design of the inner-loop current predictive controller is summarized. Firstly, the continuous-time state-space model is determined based on the differential equations of current model (2.21) and (2.22) from Chapter 2:

\[
\begin{align*}
\frac{di_{sd}(t)}{dt} &= -\frac{1}{\tau_s} i_{sd}(t) + \omega_s i_{sq}(t) + \frac{k_r}{r_s r_r} \psi_{rd}(t) + \frac{1}{r_s r_r} u_{sd}(t) \\
\frac{di_{sq}(t)}{dt} &= -\omega_s i_{sd}(t) - \frac{1}{\tau_s} i_{sq}(t) - \frac{k_r}{r_s r_r} \omega_e(t) \psi_{rd}(t) + \frac{1}{r_s r_r} u_{sq}(t)
\end{align*}
\]

Note that the time constant of the current model dynamics \(\tau_s\) has value of 0.006 sec, where the time constants of the flux and mechanical model equations (2.23) and (2.30) are \(\tau_r = 0.077\) sec and \(\frac{L_m}{J_d} = 2.2609\) sec, respectively. The two-input-two-output current model is coupled and symmetrical, the closed-loop of current control will have much larger bandwidth, hence the influence to the outer-loop control dynamics could be neglected.

By defining the two inputs \((u_{sd}(t), u_{sq}(t))\) and two outputs \((i_{sd}(t), i_{sq}(t))\), the continuous-time state space model is obtained as

\[
\begin{bmatrix}
i_{sd}(t) \\
i_{sq}(t)
\end{bmatrix} = \begin{bmatrix}
-\frac{1}{\tau_s} & \omega_s \\
-\omega_s & -\frac{1}{\tau_s}
\end{bmatrix} \begin{bmatrix}
i_{sd}(t) \\
i_{sq}(t)
\end{bmatrix} + \begin{bmatrix}
\frac{1}{r_s r_r} & 0 \\
0 & \frac{1}{r_s r_r}
\end{bmatrix} \begin{bmatrix}
u_{sd}(t) \\
u_{sq}(t)
\end{bmatrix} + \begin{bmatrix}
k_r & 0 \\
0 & k_r\omega_e(t)\psi_{rd}(t)
\end{bmatrix} \begin{bmatrix}
\psi_{rd}(t) \\
\psi_{rd}(t)
\end{bmatrix}
\]

where the synchronous speed \(\omega_s(t)\) is found as the coupling parameter of the system matrix.

Note that, the last term of the state space model contains the variables of the rotor flux \(\psi_{rd}(t)\) and the electrical rotor speed \(\omega_e(t)\), which are controlled by the outer-loop controllers. If the entire cascaded velocity control system is stable, the disturbance terms will converge to constant values in the steady-state. Hence, the predictive controller by using augmented model could completely reject the disturbance vector, which will not be included in the controller design process.

Another issue before the control design is the time varying term \(\omega_s(t)\) in (4.5). In the previous chapter, the gain-scheduled method was introduced to generate a set of linear predictive controllers. In this chapter, the inner-loop current predictive controller is designed to have a high gain feedback control system, which is robust in the presence of parameter variation. In the design, the parameter \(\omega_s\) is chosen to be a constant,
approximate equal to the slip value at the starting point, namely, $\omega_s = 12.36$. To evaluate the closed-loop robustness in the presence of parameter uncertainty, the closed-loop eigenvalues of MPC current control system are examined by varying $\omega_s$ between $(-2000, 2000)$.

![Figure 4.2: Closed-loop eigenvalues of MPC current control](image)

**Figure 4.2:** Closed-loop eigenvalues of MPC current control

Figure 4.2 presents the closed-loop eigenvalues of the inner-loop MPC current control, where the closed-loop eigenvalues are derived from

$$eig\{A(\omega_s(t)) - B \ast K_{mpc}\}$$

The design process of the continuous-time MPC is identical to Chapter 3, to avoid repetition, the MPC design will not be discussed, the control gain $K_{mpc}$ is computed based on the linearized model with $\omega_r = 0$, $i_{sd} = 1.0526$ A and $i_{sq} = 0.6$ A, the controller tuning parameters have: $N = 6$, $p = 350$ for the Laguerre function, $p$ value is large because of large closed-loop poles; the weighting matrices $Q = C^T C$ and $R = I$; prediction horizon $T_p = 0.03$ to avoid the long time off-line computational time; the prescribed degree of stability is defined as $\alpha = 1.2$ and $\beta = 200$. Then,

$$K_{mpc} = L(0)^T \Omega^{-1} \Psi$$

Additionally, the system matrix $A(\omega_s(t))$ is varying in term of the synchronous speed $\omega_s(t)$. Then, Figure 4.2(a) demonstrates the closed-loop eigenvalues trajectory with positive rotating direction, while Figure 4.2(b) presents the opposite direction. The result shows that the closed-loop system is numerically stable within the range of $\omega_s(t) = \pm 2000$, but the trajectory shows that if the amplitude of $\omega_s$ keep increasing, it will lead to the large imaginary values of the closed-loop eigenvalues, which cause the oscillations in feedback response and the system become sensitive to the internal time delay. The simulation results following will illustrate the influence of the different values of $\omega_s$. 
The MPC current controller has been implemented and examined exclusively, to ensure
that the closed-loop control dynamics of current control is accomplished before the outer-
loop design. The controller setting is identical to the previous case in the computation
of the MPC gain $K_{mpc}$.

In this section, the simulation time is 3 second, and the reference value of $i_{sd}$ is chosen
to be 1.0526 that corresponds to flux $\psi_{rd} = 0.6$. The reference value of $i_{sq}$ is chosen to
0.6 A at first, then at $t = 1.5$ second, it will step down to $-1$ A.

Figure 4.3: Simulation results of inner-loop MPC control. Key: line(1) Actual feed-
back; line (2) set-point signal.

Figure 4.4: Experimental results of inner-loop MPC control. Key: line(1) Actual
feedback; line (2) set-point signal.

Current control evaluation. The simulation results are shown in Figure 4.3 and
the experimental results are illustrated in Figure 4.4. In both results, the feedback
measurements are following their reference signals, where the experimental results are
more noisy due to the current sensor noise and the inverter switching noise with the
carrier frequency of 2kHz. Figure 4.3(b) shows that both of control signals are within
the constraints, but in Figure 4.4(b), the constraints of control signal $u_{sd}$ is active due
to the noise level.

Furthermore, the response time is significantly reduced (approximately within 0.01 sec),
which is essential for outer-loop control design. However, there are several problems with this current control algorithm, such as the noise level and the oscillations caused by the harmonics. The issues will be further analysed and compared with other methods in the later chapters.

### 4.3.2 Outer-loop Speed Control

After accomplishing the inner-loop current control, the design of the outer-loop speed control will be straightforward. Since the system is linearised using the cascaded structure, the speed control design is only dependent on the mechanical model equation (2.30),

\[
\frac{\omega_r(t)}{dt} = -\frac{f_d}{J_m} \omega_r(t) + \frac{3Z_p L_h}{2L_r J_m} \psi_r d(t) i_{sq}(t) - T_L
\]

then, the bilinear term is approximated as

\[
\psi_r d(t) i_{sq}(t) = -\psi_0^rd i_0^{sq} + \psi_r d i_{sq}(t) + \psi_r d(t) i_0^{sq}
\]

The MPC design process is similar to the inner-loop above, the augmented model of the outer-loop is derived by taking the derivative of equation, leads to

\[
\frac{d^2 \omega_r(t)}{dt^2} = -\frac{f_d}{J_m} \frac{d\omega_r(t)}{dt} + \frac{3Z_p L_h}{2L_r J_m} (\psi_r d(t) \frac{di_{sq}(t)}{dt} + i_0^{sq} \frac{d\psi_r d(t)}{dt} - \frac{d\psi_0^{rd} i_0^{sq}}{dt}) - \frac{dT_L}{dt}
\]

(4.6)

where \( \frac{d\psi_0^{rd} i_0^{sq}}{dt} = 0 \) for certain, and the rotor flux \( \psi_r d(t) \) is the defined as constant as \( \psi_0^{rd} \) during the operation, which is controlled by the \( d \)-axis current \( i_{ad} \), thus \( i_0^{sq} \frac{d\psi_r d(t)}{dt} = 0 \) at steady-state of the inner-loop, also the load torque is assumed constant during steady-state operation \( \frac{dT_L}{dt} = 0 \). Thus, the velocity model is a Single-Input-Single-Output (SISO) system, where the output signal is the motor velocity \( \omega_r(t) \) and the control input is the torque-dependent current \( i_{sq}(t) \) which is directly passed to the inner-loop current controller as the reference signal \( i_{sq}^* \).

To this end, the augmented model of the outer-loop is

\[
\begin{bmatrix}
\dot{\omega}_r(t) \\
\dot{\omega}_r(t)
\end{bmatrix} =
\begin{bmatrix}
\frac{-f_d}{J_m} & 0 \\
1 & 0
\end{bmatrix}
\begin{bmatrix}
\omega_r(t) \\
\dot{\omega}_r(t)
\end{bmatrix} +
\begin{bmatrix}
\frac{3Z_p L_h}{2L_r J_m} \psi_r d(t) i_{sq}(t) \\
0
\end{bmatrix}
\frac{di_{sq}(t)}{dt}
\]

\[
\omega_r(t) =
\begin{bmatrix}
0 & 1
\end{bmatrix}
\begin{bmatrix}
\dot{\omega}_r(t) \\
\dot{\omega}_r(t)
\end{bmatrix}
\]

Then, the unconstrained MPC controller is designed based on the same procedure in Chapter 3, the off-line computation generates the matrices of \( \Omega \) and \( \Psi \), then the optimal control signal is obtained,

\[
\dot{u}(t) = -L(0)^T \Omega^{-1} \Psi
\]
The constraints are imposed using Hildreth’s Quadratic Programming Procedure which could be referred to [19] for continuous-time MPC. The constraints implemented for outer-loop MPC controller, which is one of the advantages of using cascaded structure, so that the constraints on stator current $i_{sq}$ is embedded inside the control system. The inequality is defined as follows

$$-3A \leq i_{sq}(t) \leq 3A$$

where the limits are chosen based on the maximum instantaneous current value that the machine could accept during the transient period. Since the stator current $i_{sq}$ affects the generated electrical torque, which directly determine the acceleration rate of motor, so that the constrained value could be designed as a tuning parameter that would influence the speed control performance in applications.

Since the outer-loop MPC is based on a Single-Input-Single-Output (SISO) system, the tuning of the controller gain is straightforward, the controller parameters for simulation are defined in Table 4.1.

<table>
<thead>
<tr>
<th>$\bar{v}_{rd}$</th>
<th>$N$</th>
<th>$p$</th>
<th>$T_p$</th>
<th>$\alpha$</th>
<th>$\beta$</th>
<th>$Q$</th>
<th>$R$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.6</td>
<td>6</td>
<td>30</td>
<td>0.05</td>
<td>1.2</td>
<td>10</td>
<td>$C^2C$</td>
<td>$I$</td>
</tr>
</tbody>
</table>

Table 4.1: Controller parameters of speed MPC controller

Note that the closed-loop bandwidth of the inner-loop system is chosen at least 10 times larger than that of the outer-loop control system. In this section, by applying the exponential data weighting, the closed-loop eigenvalues of inner-loop system is much larger than the outer-loop, where $\beta = 200$ for inner-loop and $\beta = 10$ for outer-loop. Furthermore, the multi-rate sampling approach is applied for cascaded control structure, where the sampling time of the outer-loop speed control is set 5 times of the inner-loop controller, due to the slow dynamics of the mechanical model. There are several benefits to applying multi-rate sampling, such as reduced computational burden and filtered high frequency noise from the encoder.

**Speed control evaluation.** The reference of the speed control is set as a square wave with amplitude of ±1000 rpm and period of 4 sec, so that the capacity of proposed algorithm controlling all speed range is examined. To have a better comparison, the simulation and experimental results are both presented in Figure 4.5. The speed control has feedback tracking without any steady-state errors and overshoots, since the simulation is applying a constant load where the experiment has a coupled DC motor, which has larger torque directly proportional to the shaft speed, there are small ripples at negative speed response in Figure 4.5(a).

The constraints are both active for Figure 4.5(c) and 4.5(d), where the noise level of the experimental result for current control is higher due to the slow switching frequency
of the inverter. Thus, there are current noise exceeding the constraints since the constraints become active and the set-point signals to the inner-loop current controller are not the actual feedback. The control signals in the inner-loop system, shown in Figure 4.5(e), did not reach their limits as the outer-loop constraints become active. However, the constraints for the inner-loop control system become active in Figure 4.5(f) due to the noise peaks.
4.4 Cascaded MPC Position Control

In this section, an extended application of cascaded MPC is included, which is the position control of the induction motor drive. Normally, the position control is applied for servo motors in industries, but the induction motor has difficulties of controlling at the low speed range or zero speed. In industrial applications, such issue is generally solved in mechanical manner, such as using gear box or brakes.

The proposed position control is achieved by two MPC controllers in cascaded form, as illustrated in Figure 4.6, the outer-loop MPC controller is now controlling the position variable $\theta_r(t)$ but still generating the reference signal $i_{sq}$ for the inner-loop control, which is identical as introduced in Section 4.3. However, after obtaining the results of the predictive position, high frequency noise is generated in the control signal $i_{sq}$, which is caused by the second order derivative of the controlled output $\theta_r(t)$. Thus, an approach of using Non-minimal state-space model is introduced to embed a second order low-pass filter inside the predictive controller.

4.4.1 Predictive Position Control

The dynamic model between the motor shaft position angle $\theta_r$ and the motor speed $\omega_r$ is straightforward,

$$\frac{d\theta_r(t)}{dt} = \omega_r(t)$$

which results in a second order model for position control.

Based on the model equations (2.30) and (4.7), the continuous-time state-space model
for the outer-loop MPC design is derived as:

\[
\begin{align*}
\dot{x}_m(t) &= A_m x_m(t) + B_m u(t) + \mu(t) \\
y(t) &= C_m x_m(t)
\end{align*}
\] (4.8)

where

\[
A_m = \begin{bmatrix} -f_{Lm}^2 & 0 \\ 1 & 0 \end{bmatrix} ; B_m = \begin{bmatrix} \frac{3Z_p L_h \psi_0}{2L_r J_m i_s q(t)} \\ 0 \end{bmatrix}
\]

and

\[
C_m = \begin{bmatrix} 0 & 1 \end{bmatrix} ; \mu(t) = \begin{bmatrix} \gamma \\ 0 \end{bmatrix}
\]

The state vector is \( x_m(t) = [\omega_r(t) \theta_r(t)]^T \), output variable is \( y(t) = \theta_r(t) \), control signal is \( u(t) = i_{sq}(t) \) and the disturbance term is \( \gamma = -i_{sq}^0 \psi_{rd}^0 + \frac{3Z_p L_h \psi_0}{2L_r J_m i_{sq}(t)} - T_L \). As discussed in previous sections, the disturbance term is not included in the control design due to the augmented model. Thus, the state-space model (4.8) is still Single-Input-Single-Output but higher degree with two state variables.

The operational constraints are necessary in many applications of predictive control for safety reasons. In the position control, the constraints on \( i_{sq} \) are implemented as the reference signal of the inner-loop control.

The stator current components in \( dq \)-frame include \( i_{sd} \) and \( i_{sq} \), since \( i_{sd} \) is controlled to a constant value from the set-point of the rotor flux \( \psi_{rd} \). The constraints are imposed on the torque-component current \( i_{sq} \), which is the control signal from the outer-loop MPC. From previous sections of MPC design, the optimized solution \( \dot{u}(t) \) of the MPC is computed with the help of Laguerre functions. The inequality constraints are formulated as

\[
i_{sq}^{\min} - i_{sq}(t_i - \Delta t) \leq C_u \eta \leq i_{sq}^{\max} - i_{sq}(t_i - \Delta t)
\]

where \( C_u = L(0)^T \Delta t + L(t_i)^T A_p^{-T} - L(0)^T A_p^{-T} \), and \( A_p \) is a lower triangular matrix from the Laguerre functions.

**Experimental Results.** The experiment results are obtained using the test-bed in Appendix B, where the terminal of the coupled DC motor is connected with a high current power supply. Note that unlike the synchronous motor, such as PMS motor, the induction motor does not have a zero position due to the magnetic field, thus the zero position is normally defined as the initial position when the control system starts to operate, that is \( \theta_r(0) = 0 \).

The experiment is to evaluate the capacities of the proposed control method, in order to assess its control performance, such as the response time and the ability to reject the load torque at the steady-state position.

The controller parameters of the outer-loop position MPC is defined in Table 4.2, where
the closed-loop bandwidth is set wider than the speed control case, since the higher degree of stability is required in this situation. Additionally, The inner-loop MPC current control is identical to Section 4.3.

In the experiment, the flux is maintained constant and the reference signal of the rotor position is a square wave with magnitude of $\pm \pi$ rad. The constraints of the current reference $i_{sq}^*$ is defined as $\pm 3$ A. At the time around 1 second, the power supply of the coupled DC motor is switched on for load torque, which has an approximated value of $T_L = 0.5$ Nm.

The experimental results obtained are shown in Figure 4.7, where 4.7(a) presents the controlled output. The error signal is shown in Figure 4.7(b) for illustrating accuracy of the tracking error. Additionally, the disturbance rejection is shown in both figures and the torque dependent current $i_{sq}$ is shown to increase in Figure 4.7(c). Note that, the constraints are not active for inner-loop control signals as seen in Figure 4.7(d).
Furthermore, there is noise contained in the set-point signal of $i_{sq}$, which is caused by the computation of the feedback state variable $\dot{\theta}_r(t)$ in the augmented model. This noise is not affecting the outer-loop control performance, but it would induce the current noise and harmonics in current control loop, such issues could be resolved by using a non-minimal state-space model in the proposed control method, which is introduced in the following section.

### 4.4.2 Non-Minimal State-Space MPC Control

In the application of continuous-time MPC design, when the system has a high order model, the estimation of the augmented model state variable feedback signals becomes an issue due to the induced noise from the differentiation approximation. Wang [98] has done the comparison between the Non-minimal State-Space Model (NMSS) and observer-based approaches for such issue. In this section, one uses the NMSS approach to deal with the outer-loop second order continuous-time model.

Firstly, the linearized Laplace transfer function model of outer-loop control is presented as:

$$s^2\theta_r(s) + \frac{f_d}{J_m} s\theta_r(s) = \kappa_t \psi_{rd}^0 I_{sq}(s) \quad (4.9)$$

Assume that the filter to be used has the transfer function as follows

$$F(s) = \frac{t_0}{s^2 + t_1 s + t_0} \quad (4.10)$$

In order to complete the NMSS model, the filter with an integrator $sF(s)$ is embedded into the model equation (4.9). Then, the system model equation becomes

$$s^2\theta_r(s)sF(s) + \frac{f_d}{J_m} s\theta_r(s)sF(s) = \kappa_t \psi_{rd}^0 I_{sq}(s)sF(s) \quad (4.11)$$

The filtered variables are redefined as $\theta_f(s) = F(s)\theta_r(s)$ and $I_{sqf}(s) = F(s)I_{sq}(s)$. Then, the inverse Laplace transform of the model equation (4.11) is presented

$$\theta_f^{(3)}(t) = -\frac{f_d}{J_m} \dot{\theta}_f(t) + \kappa_t \psi_{rd}^0 \dot{i}_{sqf}(t) \quad (4.12)$$

where $\theta_f^{(n)}$ denotes the $n$-th order derivative of $\theta_f$.

Another model equation of NMSS is derived from the inverse Laplace transform of $s I_{sqf}(s) = s F(s) I_{sq}(s)$. By substituting the equation (4.10), the continuous-time differential equation is obtained

$$\dot{i}_{sqf}^{(3)}(t) = -t_1 \dot{i}_{sqf}(t) - t_0 \dot{i}_{sqf}(t) + t_0 i_{sq}(t) \quad (4.13)$$
Again, the inverse Laplace transform of \(s\theta_f(s) = sF(s)\theta_r(s)\) is expressed as,

\[
t_0\dot{\theta}_r(t) = \theta_f^{(3)}(t) + t_1\ddot{\theta}_f(t) + t_0\dot{\theta}_f(t)
\]  

(4.14)

The last model equation is obtained by substituting the equation (4.12) into equation (4.14).

\[
t_0\dot{\theta}_r(t) = (t_1 - \frac{f_d}{J_m})\ddot{\theta}_f(t) + t_0\dot{\theta}_f(t) + \kappa t\psi_0^0\dot{i}_{sq}(t)
\]  

(4.15)

The NMSS model is derived based on the differential equations (4.12), (4.13) and (4.15). By defining the state variable \(x_n(t) = [\ddot{\theta}_f(t) \dot{\theta}_f(t) \dot{i}_{sq}(t) \dot{i}_{sq}(t) \dot{\theta}_r(t)]^T\), the continuous-time non-minimal state space model is shown as follows

\[
\dot{x}_n(t) = A_n x_n(t) + B_n \dot{i}_{sq}(t) \quad (4.16)
\]

\[
\theta_r(t) = C_n x_n(t) \quad (4.17)
\]

where

\[
A_n = \begin{bmatrix}
-f_d/J_m & 0 & 0 & \kappa_t \psi_0^0 & 0 \\
1 & 0 & 0 & 0 & 0 \\
0 & 0 & -t_1 & -t_0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
\frac{t_1-f_d/J_m}{t_0} & \frac{t_0}{t_0} & 0 & \kappa_t \psi_0^0 & 0 \\
\end{bmatrix} \ ; \ B_n = \begin{bmatrix}
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
\end{bmatrix}
\]

and

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

The above model has already embedded an integrator, similar to the augmented model, which is ready for the MPC design as discussed in previous section. By comparing with the previous augmented model, the NMSS model has a fifth order structure, which is two order higher, that is brought by the embedded second order filter.

In the implementation of the proposed algorithm, the feedback of the state-variable needs to be estimated for the MPC design. By defining the feedback state variables as

\[
X^u(t) = \begin{bmatrix}
\dot{i}_{sq}(t) \\
\dot{i}_{sq}(t) \\
\end{bmatrix} \ ; \ X^a(t) = \begin{bmatrix}
i_{sq}(t) \\
i_{sq}(t) \\
\end{bmatrix}
\]

From the transfer function of the filter (4.10), the estimation of the derivative of the \(X^u(t)\) and \(X^a(t)\) are derived

\[
\dot{X}^u(t) = A_F X^u(t) + B_F \dot{\theta}_r(t)
\]

\[
\dot{X}^u(t) = A_F X^u(t) + B_F \dot{i}_{sq}(t)
\]
where

\[ A_F = \begin{bmatrix} -t_1 & t_0 \\ 1 & 0 \end{bmatrix}; B_F = \begin{bmatrix} t_0 \\ 0 \end{bmatrix} \]

Then, by applying the first order approximation of derivative \( \frac{dx(t)}{dt} = \frac{x(t_i) - x(t_i - \Delta t)}{\Delta t} \), the estimation of state variable is derived based on the previous sample information and the derivative of the actual measurement feedback.

\[
X^y(t_i) = X^y(t_i - \Delta t) + \Delta t(A_F X^y(t_i - \Delta t) + B_F \dot{\theta}_r(t_i))
\]

\[
X^u(t_i) = X^u(t_i - \Delta t) + \Delta t(A_F X^u(t_i - \Delta t) + B_F i_{sq}(t_i))
\]

where the derivative of the actual measurement is computed by the first order approximation as well.

\[ \omega_{nF} = 300 \text{ and } \xi = 0.707, \text{ hence, } t_0 = \omega_{nF}^2 = 90000 \text{ and } t_1 = 2\xi\omega_{nF} = 424.2. \]

The experimental results are shown in Figure 4.8 using the identical reference signals.

**Experimental Results.** The experiment set-up in this situation is identical to the previous one for comparison, the tuning parameters of the proposed controller remained the same as before in Table 4.2, except the parameters of the embedded filter defined as: \( \omega_{nF} = 300 \) and \( \xi = 0.707 \), hence, \( t_0 = \omega_{nF}^2 = 90000 \) and \( t_1 = 2\xi\omega_{nF} = 424.2 \). The experimental results are shown in Figure 4.8 using the identical reference signals.
and the load condition. Firstly, the noise of the current reference signal $i_{sq}^*$ has been reduced as shown in Figure 4.8(c), which validates the embedded low-pass filter of the NMSS model. However, by comparing the error signal in Figure 4.8(b) with the previous case (see Figure 4.7(b)), the embedded filter has an influence on the control performance, since the dynamics of the previous control system are significantly fast. Therefore, the proposed NMSS model could filter out the noise of the control signal, but also will introduce the extra dynamics into the control system.

### 4.5 Summary

In this chapter, the cascaded continuous-time model predictive control has been presented based on indirect FOC technique. Since the time constants of electrical systems are significantly smaller than those from the mechanical systems, the cascaded continuous-time predictive controllers could separately control the electrical and mechanical system with different sampling rates. Thanks to the cascaded structure, the applications of induction motor drive control, such as current control, velocity control and position control, could be achieved by using the continuous-time predictive controllers. The experimental results obtained for respective applications evaluate the advantages of the proposed control structure.

By using the identical control algorithm from the previous chapter, the modified control structure could bring various advantages, such as the current control realization, the additional control constraints implementation, the dual sampling rate and the simplified linearization procedure. The further analysis and comparisons could be found in Chapter 7.
Chapter 5

Finite Control Set - Model Predictive Control

5.1 Background Study

Up to this point, from the study of the induction motor drive control, the main control issue lies on the current control loop, which contains the non-linearity and strong coupling. In this chapter, a recent control method, named Finite Control Set (FCS), is derived for current control of induction motor. Generally, the implementation of current control is based on a three phase 2 Level-Voltage-Source-Inverter (2L-VSI), whose structure diagram is presented in Figure 5.1. Each leg of the inverter has two pairs of combination, which contains a IGBT switch and a free-wheeling diode, thus the middle point is connected to the operating induction motor. Since the induction motor is connected in star connection, instantaneously, each phase of the motor could be offered up to half of the total DC bus voltage ($V_{dc}/2$), for the simplification of analysis, it is sufficient to assume that the middle point between two DC sources is referred to the ground. Thus, all the voltages can be represented with respect to the ground. [53]

From Figure 5.1, there are two IGBT switches for each of the three legs. Within each leg of an inverter, only one switch is turned on (denoted by 1) while the other is off (denoted by 0) at any given time to prevent circuit short-cut. Thus, the switching states of the inverter can be identified by only considering the states of the three upper switches. With the states of three upper switches denoted as $S_i$ ($i = a, b, c$), the states of their corresponding lower switches can be represented by their negation $\bar{S}_i$ ($i = \bar{a}, \bar{b}, \bar{c}$). As a result, there are only eight possible switching states by turning on and off all the switches in the inverter.

Since the states of upper and lower switches within the same leg are complementary to
Chapter 5. Finite Control Set - Model Predictive Control

Figure 5.1: Topology of 2 Level-Voltage-Source-Inverter

each other, all eight switching states could be independently identified by the states of the three upper switches, as listed in Table 5.1. Among those, two switching states \( V_0 \) and \( V_7 \), which illustrate the case where either all the upper or all the lower switches are turned on, are called zero vector. In contrast, the other six states that forms a closed circuit, are called active vector. [54]

When the upper switch is on, that is \( S_i = 1 \) and \( \overline{S}_i = 0 \), the output of a phase leg is connected to the top rail of the supplies and thus \( v_i = \frac{V_{dc}}{2} \). Conversely, when the lower switch is on, the output is connected to the bottom rail of the supplies and hence \( v_i = -\frac{V_{dc}}{2} \). Corresponding to the switching states in Table 5.1, the resulting output voltage \( v_i \) are summarized in Table 5.2.

The transformation of the three-phase voltages to their components in \( \alpha\beta \) frame is achieved by the Clarke transformation,

\[
\begin{bmatrix}
u_{s\alpha} \\
u_{s\beta}
\end{bmatrix} = \frac{2}{3} \begin{bmatrix}
1 & -\frac{1}{2} & -\frac{1}{2} \\
0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2}
\end{bmatrix} \begin{bmatrix}
v_{an} \\
v_{bn} \\
v_{cn}
\end{bmatrix}
\] (5.1)
Thus, the $\alpha\beta$ representation of three-phase output voltages can be expressed in terms of the switching states of three upper leg switches,

$$
\begin{bmatrix}
u_a \\
v_b \\
v_c
\end{bmatrix} - \begin{bmatrix}
v_n \\
v_n \\
v_n
\end{bmatrix} = \frac{2}{3} V_{dc} \begin{bmatrix}
1 & -\frac{1}{2} & -\frac{1}{2} \\
\frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\
0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2}
\end{bmatrix}
\begin{bmatrix}
S_a \\
S_b \\
S_c
\end{bmatrix}
$$

(5.2)

where the property that the Clarke transformation of any constant vector leads to a zero vector has been utilized.

Let the matrix $U$ be defined by the switching states:

$$
U = \begin{bmatrix}
0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1
\end{bmatrix}
$$

(5.3)

and matrix $D$ be the Clarke transformation from the three phase voltage to $\alpha\beta$ frame:

$$
D = \begin{bmatrix}
1 & -\frac{1}{2} & -\frac{1}{2} \\
0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2}
\end{bmatrix}
$$

(5.4)

The multiplication of $D$ and $U$ matrices leads to

$$
DU = \begin{bmatrix}
0 & 1 & \frac{1}{2} & -\frac{1}{2} & -1 & -\frac{1}{2} & \frac{1}{2} & 0 \\
0 & 0 & \frac{\sqrt{3}}{2} & \frac{\sqrt{3}}{2} & 0 & -\frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} & 0
\end{bmatrix}
$$

(5.5)

With the transformation matrix (5.5), the operational constraints due to the voltage source inverter are expressed in the $\alpha\beta$ frame as the equality constraints, which are characterized by the values in the matrix

$$
\begin{bmatrix}
0 & 1 & \frac{1}{2} & -\frac{1}{2} & -1 & -\frac{1}{2} & \frac{1}{2} & 0 \\
0 & 0 & \frac{\sqrt{3}}{2} & \frac{\sqrt{3}}{2} & 0 & -\frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} & 0
\end{bmatrix} \frac{2}{3} V_{dc}
$$

(5.6)

where $V_{dc}$ is the voltage for the DC power supply. More precisely, the control variable $u_{sa}$ will only take the values defined by the first row of the matrix (5.6) while the control variable $u_{sb}$ will only take the values by the second row, with further constraints that their values must form the exactly paired relations as in (5.6). These constraints result
in the situation where the control movements are restricted to a finite control set of parameters. For instance, with the DC bus voltage \( V_{dc} \) given, the \( u_{sa} \) and \( u_{sb} \) voltage values are chosen among the following set of parameters in pairs:

\[
\begin{align*}
\bar{u}_{sa}^0 &= 0; & u_{sa}^1 &= V_{dc}; & u_{sa}^2 &= \frac{1}{2}V_{dc}; & u_{sa}^3 &= -\frac{1}{2}V_{dc}; \\
\bar{u}_{sb}^0 &= 0; & u_{sb}^1 &= 0; & u_{sb}^2 &= \frac{\sqrt{3}}{2}V_{dc}; & u_{sb}^3 &= -\frac{\sqrt{3}}{2}V_{dc}; \\

u_{sa}^4 &= -V_{dc}; & u_{sa}^5 &= -\frac{1}{2}V_{dc}; & u_{sa}^6 &= \frac{1}{2}V_{dc}; & u_{sa}^7 &= 0; \\
u_{sb}^4 &= 0; & u_{sb}^5 &= -\frac{\sqrt{3}}{2}V_{dc}; & u_{sb}^6 &= \frac{\sqrt{3}}{2}V_{dc}; & u_{sb}^7 &= 0.
\end{align*}
\]

The super-scripts of \( u_{sa} \) and \( u_{sb} \) correspond to the indices of the IGBT’s switching states. Once one of the indices is identified, the control action is determined and implemented via the VSI inverter. Although the pair \( u_{sa}^0, u_{sb}^0 \) are identical to \( u_{sa}^7, u_{sb}^7 \), what action should the inverter take is different in the sense that one corresponds to all off-states while the other to all on-states. To avoid excessive switching actions from the inverter, when \( u_{sa} = 0 \) and \( u_{sb} = 0 \), the action that the inverter takes will depend on the inverter’s previous action. For instance, the previous action is two states on and one state off, then inverter’s current action corresponding to \( u_{sa} = 0 \) and \( u_{sb} = 0 \) should be all states-on.

The constraints on the control variables in \( dq \) frame are functions of synchronous angle \( \theta_s \) due to the deployment of Park Transform. Namely, the operational constraints in the \( dq \) frame due to the VSI operations are also expressed in equality constraints with the following form:

\[
\begin{bmatrix}
\cos \theta_s & \sin \theta_s \\
-\sin \theta_s & \cos \theta_s
\end{bmatrix}
\begin{bmatrix}
0 & 1 & \frac{1}{2} & -\frac{1}{2} & -1 & -\frac{1}{2} & \frac{1}{2} & 0 \\
0 & 0 & \frac{\sqrt{3}}{2} & \frac{\sqrt{3}}{2} & 0 & -\frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} & 0
\end{bmatrix}
\frac{2}{3}V_{dc}
\]

(5.7)

where \( \theta_s \) is the electrical synchronous angle.

This basically says that with a given synchronous angle \( \theta_s \), there are only seven pairs of \( u_{sd} \) and \( u_{sq} \) values that can be exactly realized by the VSI inverter. In the \( \alpha\beta \) frame the constraints are the finite set of constant parameters defined by (5.6) once the DC bus voltage \( V_{dc} \) is given, however, in the \( dq \) frame, this set of constant parameters becomes functions of the synchronous angle \( \theta_s \).

It is emphasized that with a given \( \theta_s \) value and a sampling time \( t \), in the \( dq \) frame, the set of \( u_{sd} \) and \( u_{sq} \) values are constant. For instance, when \( \theta_s = \theta_s^0 \), the \( u_{sd} \) and \( u_{sq} \) are chosen among the following parameters:

\[
\begin{align*}
\bar{u}_{sd}^0 &= 0; & \bar{u}_{sd}^1 &= \frac{2}{3}V_{dc}\cos\theta_s^0; & \bar{u}_{sd}^2 &= \frac{2}{3}V_{dc}\cos(\theta_s^0 - \frac{2\pi}{3}); & \bar{u}_{sd}^3 &= \frac{2}{3}V_{dc}\cos(\theta_s^0 - \frac{4\pi}{3}); \\
u_{sd}^0 &= 0; & u_{sd}^1 &= -\frac{2}{3}V_{dc}\sin\theta_s^0; & u_{sd}^2 &= -\frac{2}{3}V_{dc}\sin(\theta_s^0 - \frac{2\pi}{3}); & u_{sd}^3 &= -\frac{2}{3}V_{dc}\sin(\theta_s^0 - \frac{4\pi}{3}); \\
\bar{u}_{sd}^4 &= -\frac{2}{3}V_{dc}\cos\theta_s^0; & \bar{u}_{sd}^5 &= \frac{2}{3}V_{dc}\cos(\theta_s^0 - \frac{2\pi}{3}); & \bar{u}_{sd}^6 &= -\frac{2}{3}V_{dc}\cos(\theta_s^0 - \frac{4\pi}{3}); & \bar{u}_{sd}^7 &= 0; \\
u_{sd}^4 &= 0; & u_{sd}^5 &= \frac{2}{3}V_{dc}\sin(\theta_s^0 - \frac{2\pi}{3}); & u_{sd}^6 &= \frac{2}{3}V_{dc}\sin(\theta_s^0 - \frac{4\pi}{3}); & u_{sd}^7 &= 0.
\end{align*}
\]
5.2 Study of Original FCS-MPC

The original Finite Control Set proposed the implementation procedure as follows: [66]

1. At sampling time $t_i$, the measured currents $i_{sd}(t_i)$, $i_{sq}(t_i)$ and the seven pairs of the control signal candidates $u^k_{sd}$ and $u^k_{sq}$ ($k = 0, 1, \ldots 7$) are available. Furthermore, the measurement of electrical motor speed $\omega_e(t_i)$ and the estimation of synchronous speed $\omega_s(t_i)$ and rotor flux $\psi_{rd}(t_i)$ are available as well.

2. Within a for loop of $i = 0$ to 7, the one-step-ahead prediction will firstly be computed as

\[
\begin{align*}
    i_{sd}^i(t_i + \Delta t) &= i_{sd}(t_i) + \Delta t \left( -\frac{1}{\tau_{\sigma}} i_{sd}(t_i) + \omega_s i_{sq}(t_i) + \frac{k_r}{\tau_{\sigma} \tau_{\sigma} \tau_r} \psi_{rd}(t_i) + \frac{1}{\tau_{\sigma} \tau_{\sigma}} u^i_{sd}(t_i) \right) \\
    i_{sq}^i(t_i + \Delta t) &= i_{sq}(t_i) + \Delta t \left( -\omega_s i_{sd}(t_i) - \frac{1}{\tau_{\sigma}} i_{sq}(t_i) - \frac{k_r}{\tau_{\sigma} \tau_{\sigma} \tau_r} \omega_e(t_i) \psi_{rd}(t_i) + \frac{1}{\tau_{\sigma} \tau_{\sigma}} u^i_{sq}(t_i) \right)
\end{align*}
\]

3. For every loop cycle $i$, the objective function $J$ is computed based on the squares of the error between the predictive values and the reference signals $i^*_{sd}(t_i)$ and $i^*_{sq}(t_i)$:

\[
J = (i^*_{sd}(t_i) - i_{sd}^i(t_i + \Delta t))^2 + (i^*_{sq}(t_i) - i_{sq}^i(t_i + \Delta t))^2
\]

4. After the for loop computation, the index with minimum value of the objective function $J$ is found at $j$, thus the optimal control solution is computed as $u^j_{sd}$ and $u^j_{sq}$.

5. As the sampling time proceeds to $t_i + \Delta t$, the procedure will start again from the first step.

The algorithm of FCS is straightforward and the implementation is sufficiently easy, which is one significant advantage of the Finite Control Set-MPC. Based on the control algorithm from above, the simulation result is obtained as shown in Figure 5.2 at the sampling time of 20\(\mu\)s.

The Finite Control Set is found as an efficient way for current control in different power electronics applications, however, there are several weaknesses for such control method, which are still popular topics in the research field, such as steady-state error, switching frequency distribution, constraints implementation and robustness with modelling errors. In this thesis, the research question according to the steady-state error problem is investigated.

The original cost function of FCS is defined as sum of the squared errors between the
desired and predicted signals:

\[ J = (i_{sd}^*(t_i) - i_{sd}(t_i + \Delta t))^2 + (i_{sq}^*(t_i) - i_{sq}(t_i + \Delta t))^2 \]  (5.8)

where \(i_{sd}(t_i + \Delta t)\) and \(i_{sq}(t_i + \Delta t)\) present the one-step-ahead prediction of the stator currents \(i_{sd}(t_i)\) and \(i_{sq}(t_i)\), respectively.

From chapter 2, the differential equations of the current model in \(dq\) frame is presented, which contains the inputs of \(u_{sd}, u_{sq}\) and the outputs of \(i_{sd}\) and \(i_{sq}\).

\[
\frac{di_{sd}(t)}{dt} = -\frac{1}{\tau_{\sigma}} i_{sd}(t) + \omega_s(t)i_{sq}(t) + \frac{k_r}{r_{\sigma} \tau_r \tau_{\sigma}} \psi_{rd}(t) + \frac{1}{r_{\sigma} \tau_{\sigma}} u_{sd}(t) \]  (5.9)

\[
\frac{di_{sq}(t)}{dt} = -\omega_s(t)i_{sd}(t) - \frac{1}{r_{\sigma}} i_{sq}(t) - \frac{k_r}{r_{\sigma} \tau_r} \omega_s(t) \psi_{rd}(t) + \frac{1}{r_{\sigma} \tau_{\sigma}} u_{sq}(t) \]  (5.10)

By assuming the sampling time of \(\Delta t\), at arbitrary time \(t_i\), the derivatives are approximated as \(\frac{di_{sd}(t)}{dt} \approx \frac{i_{sd}(t_i + \Delta t) - i_{sd}(t_i)}{\Delta t}\) and \(\frac{di_{sq}(t)}{dt} \approx \frac{i_{sq}(t_i + \Delta t) - i_{sq}(t_i)}{\Delta t}\). Then, the discretized differential equations become the difference equations:

\[
i_{sd}(t_i + \Delta t) = i_{sd}(t_i) + \Delta t\left(-\frac{1}{\tau_{\sigma}} i_{sd}(t_i) + \omega_s(t_i)i_{sq}(t_i) + \frac{k_r}{r_{\sigma} \tau_r \tau_{\sigma}} \psi_{rd}(t_i) + \frac{1}{r_{\sigma} \tau_{\sigma}} u_{sd}(t_i)\right) \]  (5.11)
\begin{align}
i_{sq}(t_i + \Delta t) &= i_{sq}(t_i) + \Delta t\left(-\omega_s i_{sd}(t_i) - \frac{1}{\tau_s} i_{sq}(t_i) - \frac{k_r}{r_s \tau_d} \omega_e(t_i) \psi_{rd}(t_i) + \frac{1}{r_s \tau_d} u_{sq}(t_i)\right) \tag{5.12} \\
\end{align}

After substituting the above prediction equations into equation (5.8), the objective function \( J \) will contain the variables that are measured at the sampling time \( t_i \) and the manipulated variables \( u_{sd}(t_i) \) and \( u_{sq}(t_i) \), as

\begin{align}
J &= (i_{sd}^*(t_i) - i_{sd}(t_i)) - \Delta t\left(-\frac{1}{\tau_s} i_{sd}(t_i) + \omega_s i_{sq}(t_i) + \frac{k_r}{r_s \tau_d} \psi_{rd}(t_i) + \frac{1}{r_s \tau_d} u_{sd}(t_i)\right)^2 \\
&\quad + \left(i_{sq}(t_i) - i_{sq}(t_i) - \Delta t(-\omega_s i_{sd}(t_i) - \frac{1}{r_s} i_{sq}(t_i) - \frac{k_r}{r_s \tau_d} \omega_e(t_i) \psi_{rd}(t_i) + \frac{1}{r_s \tau_d} u_{sq}(t_i))\right)^2 \\
&\quad \tag{5.13}
\end{align}

Since at the sampling instant \( t_i \), from previous section, there are seven pairs of \( u_{sd}(t_i) \) and \( u_{sq}(t_i) \) available as candidates, the next step in the finite control set controller design is to find the pair of manipulated variables that will minimize the objective function \( J \) (5.13). For this purpose, the seven values of the objective function \( J \) are calculated with respect to the candidate pairs of \( u_{sd}(t_i) \) and \( u_{sq}(t_i) \) and denoted as \( J^0, J^1, J^2, \ldots, J^7 \). A simple search function is used to find the minimal value of \( J^k \) and its associated index. Once this index is found, the control signal at time \( t_i \) to the VSI is determined through Table 5.1 and the corresponding voltage is obtained through 5.2. However, in order to reduce unnecessary switchings, if the index is found to be 0, then the previous states of the VS inverter are required to determine whether the index 0 or 8 should be used in the control action.

When the sampling time progresses to \( t_i + \Delta t \), the new measurements of \( i_{sd}(t_i + \Delta t) \), \( i_{sq}(t_i + \Delta t) \) currents, velocity measurement \( \omega_e(t_i + \Delta t) \) and the estimation of synchronous speed \( \omega_s(t_i + \Delta t) \) are obtained, the seven new pairs of candidates \( u_{sd}(t_i + \Delta t) \) and \( u_{sq}(t_i + \Delta t) \) are computed due to the new synchronous angle \( \theta_s(t_i + \Delta t) \). With all the variables in the objective function (5.13) being updated, minimization is performed to find the new minimal value of \( J^m \) and its index \( m \) at sampling time \( t_i + \Delta t \), leading to the control signal for the voltage source inverter.

The essence of the finite set control method is based on the receding horizon control principle, which uses one-step-ahead prediction and on-line optimization to solve the constrained optimal control problem. The closed-loop feedback mechanism is generated when using the updated \( i_{sd}(t_i) \) and \( i_{sq}(t_i) \) current measurements in the prediction.

For convenience of programming, the difference equations (5.11) and (5.12) are expressed in matrix and vector forms:

\begin{equation}
\begin{bmatrix}
i_{sd}(t_i + \Delta t) \\
i_{sq}(t_i + \Delta t)
\end{bmatrix} = (I + \Delta t A_m(t_i)) \begin{bmatrix}
i_{sd}(t_i) \\
i_{sq}(t_i)
\end{bmatrix} + \Delta t B_m \begin{bmatrix}
u_{sd}(t_i) \\
u_{sq}(t_i)
\end{bmatrix} + \gamma_D(t_i) \Delta t \tag{5.14}
\end{equation}
where $I$ is the identity matrix with dimension of $2 \times 2$ and the system matrices $A_m(t_i)$ and $B_m$ are defined as

$$A_m(t_i) = \begin{bmatrix} -\frac{1}{r_{\sigma}} & \omega_s(t_i) \\ -\omega_s(t_i) & -\frac{1}{r_{\sigma}} \end{bmatrix}; B_m = \begin{bmatrix} \frac{1}{r_{\sigma} i_{\sigma}} & 0 \\ 0 & \frac{1}{r_{\sigma} i_{\sigma}} \end{bmatrix}$$

as well as, the disturbance vector $\gamma_D(t_i)$ is expressed as

$$\gamma_D(t_i) = \begin{bmatrix} \frac{k_v}{r_{\sigma} i_{\sigma} T_{\sigma}} \psi_{rd}(t_i) \\ -\frac{k_v}{r_{\sigma} i_{\sigma} T_{\sigma}} \omega_e(t_i) \psi_{rd}(t_i) \end{bmatrix}$$

### 5.3 Revised Finite Control Set Method

#### 5.3.1 Analysis of Finite Control Set Method

In order to analyze the closed-loop performance via feedback control, the objective function $J$ is re-written in vector form:

$$J = \begin{bmatrix} i^*_{sd}(t_i) - i_{sd}(t_i + \Delta t) \\ i^*_{sq}(t_i) - i_{sq}(t_i + \Delta t) \end{bmatrix} \begin{bmatrix} i^*_{sd}(t_i) - i_{sd}(t_i + \Delta t) \\ i^*_{sq}(t_i) - i_{sq}(t_i + \Delta t) \end{bmatrix}$$

Moreover, the difference equations (5.11) and (5.12) are expressed in matrix and vector forms given by (5.14). For notational simplicity, let the vector $[f_d(t_i) f_q(t_i)]^T$ be defined as

$$\begin{bmatrix} f_d(t_i) \\ f_q(t_i) \end{bmatrix} = \begin{bmatrix} \dot{i}_{sd}(t_i) \\ \dot{i}_{sq}(t_i) \end{bmatrix} - (I + \Delta t A_m(t_i)) \begin{bmatrix} i_{sd}(t_i) \\ i_{sq}(t_i) \end{bmatrix} - \gamma_D(t_i) \Delta t$$

Then it can be verified by combining (5.16) with (5.14) that the objective function (5.15) has the compact expression:

$$J = \left(\begin{bmatrix} f_d(t_i) \\ f_q(t_i) \end{bmatrix} - \Delta t B_m \begin{bmatrix} u_{sd}(t_i) \\ u_{sq}(t_i) \end{bmatrix}\right) \left(\begin{bmatrix} f_d(t_i) \\ f_q(t_i) \end{bmatrix} - \Delta t B_m \begin{bmatrix} u_{sd}(t_i) \\ u_{sq}(t_i) \end{bmatrix}\right)$$

which is in the quadratic objective function form:

$$J = \begin{bmatrix} f_d(t_i) & f_q(t_i) \end{bmatrix} \begin{bmatrix} f_d(t_i) \\ f_q(t_i) \end{bmatrix} - 2 \begin{bmatrix} u_{sd}(t_i) & u_{sq}(t_i) \end{bmatrix} \Delta t B_m^T \begin{bmatrix} f_d(t_i) \\ f_q(t_i) \end{bmatrix}$$

$$+ \begin{bmatrix} u_{sd}(t_i) & u_{sq}(t_i) \end{bmatrix} \Delta t^2 B_m^T B_m \begin{bmatrix} u_{sd}(t_i) \\ u_{sq}(t_i) \end{bmatrix}$$

As a result, the system becomes least squares minimization problem. By defining the vector $u_{dq}(t_i) = [u_{sd}(t_i) u_{sq}(t_i)]^T$, the first derivative of the objective function $J$ respect
to control signal vector $u_{dq}$ is derived as

$$\frac{\partial J}{\partial u_{dq}(t_i)} = -2\Delta t B^T_m \begin{bmatrix} f_d(t_i) \\ f_q(t_i) \end{bmatrix} + 2\Delta t^2 B^T_m B_m \begin{bmatrix} u_{sd}(t_i) \\ u_{sq}(t_i) \end{bmatrix}$$  \tag{5.19}$$

The necessary condition of the minimum $J$ is obtained as $\frac{\partial J}{\partial u_{dq}(t_i)} = 0$. Note that $r_{\sigma}\tau'_{\sigma} > 0$, due to the physical model of induction motor, thus the matrix $B^T_m B_m$ is positive definite, given by

$$B^T_m B_m = \begin{bmatrix} \frac{1}{(r_{\sigma}\tau'_{\sigma})^2} & 0 \\ 0 & \frac{1}{(r_{\sigma}\tau'_{\sigma})^2} \end{bmatrix}$$

By letting the derivative equation (5.19) equal to zero, the control signals that give the minimum objective function are:

$$\begin{bmatrix} u_{sd}(t_i) \\ u_{sq}(t_i) \end{bmatrix} = \frac{1}{\Delta t} \begin{bmatrix} r_{\sigma}\tau'_{\sigma} & 0 \\ 0 & r_{\sigma}\tau'_{\sigma} \end{bmatrix} \begin{bmatrix} f_d(t_i) \\ f_q(t_i) \end{bmatrix}$$  \tag{5.20}$$

Then, by substituting the auxiliary vector (5.16) back into (5.20), the optimal control solution is obtained as

$$\begin{bmatrix} u_{sd}(t_i) \\ u_{sq}(t_i) \end{bmatrix} = \begin{bmatrix} \frac{r_{\sigma}\tau'_{\sigma}}{\Delta t} & 0 \\ 0 & \frac{r_{\sigma}\tau'_{\sigma}}{\Delta t} \end{bmatrix} \begin{bmatrix} i_{sd}(t_i) \\ i_{sq}(t_i) \end{bmatrix} - (I + \Delta t A_m(t_i)) \begin{bmatrix} i_{sd}(t_i) \\ i_{sq}(t_i) \end{bmatrix} - \gamma_D(t_i)\Delta t$$  \tag{5.21}$$

Without considering the restriction of the control signals, this is the optimal solution of the predictive control system with one-step-ahead prediction. Because the actual electrical velocity $\omega_e(t_i)$ is used in the computation of the prediction, the control law is linear time-varying.

There is an alternative way to find the minimum of the objective function via the technique of completing squares. This completing squares approach will lead to a different method to evaluate the objective function for finding the control signals among the candidate variables.

By adding and subtracting the following term into the quadratic objective function (5.18)

$$\begin{bmatrix} f_d(t_i) \\ f_q(t_i) \end{bmatrix} \Delta t B_m (\Delta t^2 B^T_m B_m)^{-1} B^T_m \Delta t \begin{bmatrix} f_d(t_i) \\ f_q(t_i) \end{bmatrix}$$
the value of the objective function $J$ remains unchanged, which leads to form a completed squares:

$$J_0 = \left[ \begin{array}{cc} u_{sd}(t_i) & u_{sq}(t_i) \\ \end{array} \right] \Delta t^2 B_m^T B_m \left[ \begin{array}{c} u_{sd}(t_i) \\ u_{sq}(t_i) \end{array} \right] - 2 \left[ \begin{array}{c} u_{sd}(t_i) \\ u_{sq}(t_i) \end{array} \right] \Delta t B_m^T f_d(t_i)$$

$$+ \left[ \begin{array}{c} f_d(t_i) \\ f_q(t_i) \end{array} \right] \Delta t B_m (\Delta t^2 B_m^T B_m)^{-1} B_m^T \Delta t \left[ \begin{array}{c} f_d(t_i) \\ f_q(t_i) \end{array} \right]$$

which is

$$J_0 = \left( \begin{array}{c} u_{sd}(t_i) \\ u_{sq}(t_i) \end{array} \right) - (\Delta t^2 B_m^T B_m)^{-1} B_m^T \Delta t \left[ \begin{array}{c} f_d(t_i) \\ f_q(t_i) \end{array} \right] \Delta t^2 B_m^T B_m$$

$$\times \left( \begin{array}{c} u_{sd}(t_i) \\ u_{sq}(t_i) \end{array} \right) - (\Delta t^2 B_m^T B_m)^{-1} B_m^T \Delta t \left[ \begin{array}{c} f_d(t_i) \\ f_q(t_i) \end{array} \right] \left( \begin{array}{c} f_d(t_i) \\ f_q(t_i) \end{array} \right)$$

Thus, the objective function $J$ becomes two parts

$$J = J_0 + J_{\text{min}}$$

where $J_{\text{min}}$ is

$$J_{\text{min}} = - \left[ \begin{array}{c} f_d(t_i) \\ f_q(t_i) \end{array} \right] B_m (B_m^T B_m)^{-1} B_m^T \left[ \begin{array}{c} f_d(t_i) \\ f_q(t_i) \end{array} \right] \left[ \begin{array}{c} f_d(t_i) \\ f_q(t_i) \end{array} \right]$$

Since the weighting matrix $\Delta t^2 B_m^T B_m$ in $J_0$ (5.22) is positive definite and $J_{\text{min}}$ above is independent of the control variables $u_{sd}(t_i)$ and $u_{sq}(t_i)$, then the minimum of the objective function $J$ is achieved if $J_0$ is minimized, which is the completed square become zero in (5.22), thus the control variables $u_{sd}(t_i)$ and $u_{sq}(t_i)$ are chosen as

$$\left[ \begin{array}{c} u_{sd}(t_i) \\ u_{sq}(t_i) \end{array} \right] = (\Delta t^2 B_m^T B_m)^{-1} B_m^T \Delta t \left[ \begin{array}{c} f_d(t_i) \\ f_q(t_i) \end{array} \right]$$

Note that, the optimal solution obtained in (5.23) via completing the squares is identical to that obtained previously in (5.20).

Furthermore, with the completing squares approach, the constant term $J_{\text{min}}$ is easily examined via

$$J_{\text{min}} = \left[ \begin{array}{c} f_d(t_i) \\ f_q(t_i) \end{array} \right] (I - B_m (B_m^T B_m)^{-1} B_m^T) \left[ \begin{array}{c} f_d(t_i) \\ f_q(t_i) \end{array} \right]$$
Since
\[
B_m = \begin{bmatrix}
\frac{1}{r_στ_σ} & 0 \\
0 & \frac{1}{r_στ_σ}
\end{bmatrix}
\]
so that, the matrix \(I - B_m(B_m^T B_m)^{-1}B_m^T\) is a zero matrix, which leads to \(J_{\text{min}} = 0\), hence \(J = J_0\). Therefore, it can be summarized that the sum of squares error between the predicted and reference signals is zero if the control signals are chosen according to (5.20) or (5.23).

### 5.3.2 Feedback Controller Gain Design

From previous section, the feedback control gain in the one-step-ahead predictive control system at sampling instant \(t_i\) is
\[
K_{fcs}(t_i) = \begin{bmatrix}
\frac{r_στ_σ'}{Δt} & 0 \\
0 & \frac{r_στ_σ'}{Δt}
\end{bmatrix} (I + ΔtA_m(t_i)) \tag{5.24}
\]
which is derived from (5.21), that leads to
\[
\begin{bmatrix}
u_{sd}(t_i) \\
u_{sq}(t_i)
\end{bmatrix} = K_{fcs}(t_i)\begin{bmatrix}
i_{sd}(t_i) \\
i_{sq}(t_i)
\end{bmatrix} - \begin{bmatrix}
k_rψ_{rd}(t_i) \\
-k_rω_e(t_i)ψ_{rd}(t_i)
\end{bmatrix} \tag{5.25}
\]
where the last term represents the disturbance term, in this case, it is subtracted for nonlinear compensation. For the feedback controller gain \(K_{fcs}\) in (5.24), which will increase as the sampling interval \(Δt\) decreases. Simply as \(Δt \to 0\), then \(K_{fcs} \to \infty\). Thus, with a sufficiently small value of \(Δt\), the controller gain could be approximated by
\[
K_{fcs}(t_i) \approx \begin{bmatrix}
\frac{r_στ_σ'}{Δt} & 0 \\
0 & \frac{r_στ_σ'}{Δt}
\end{bmatrix} \tag{5.26}
\]
The internal closed-loop stability of the one-step-ahead predictive control system has been determined by substituting the feedback control signal (5.25) into (5.14), where the reference signals are considered to be 0 in the original control law (5.21), thus the closed-loop system has the following form at steady-state:
\[
\begin{bmatrix}
i_{sd}(t_i + Δt) \\
i_{sq}(t_i + Δt)
\end{bmatrix} = \begin{bmatrix}
0 & 0 \\
0 & 0
\end{bmatrix} \begin{bmatrix}
i_{sd}(t_i) \\
i_{sq}(t_i)
\end{bmatrix}
\]
Thus, the closed-loop eigenvalues of the system (5.3.2) are at the origin of the complex plane, as a result, at arbitrary sampling time \(t_i\), the closed-loop eigenvalues at zero guarantees its stability for a discrete time system. However, in order to ensure the internal closed-loop stability for \(0 \leq t < \infty\), an additional condition on the slow variation
of the system matrix is required because the system is time varying and the control law is time varying as well. This is based on the classical work by Desoer in the area of linear system theory [99] which stated that for a linear time varying system, it is stable if it has all eigenvalues lying strictly inside the unit circle and if it is slowly time-varying. A question arises from the fact that in this design, the closed-loop system matrix is zero in (5.3.2), where the time variation of the control system comes from? The answer to the question lies in the derivation of (5.3.2) where the assumption that all the electrical parameters in the induction motor are exactly values is implicitly used. Under this assumption, the control law will result in a cancellation of the dynamics. Therefore, in reality with some degree of parameter mismatch between the model and the induction motor, there would not be the perfect cancellation, leading to time-varying nature of the system matrix. Hence, the slow time-variation of the controller gain is needed as part of the closed-loop stability condition. However, if the sampling interval $\Delta t$ is sufficiently small, then the controller gain is close to a constant gain matrix (see (5.26)).

5.3.3 Constrained Optimal Control Design

In this section, the constraints are deployed for the one-step-ahead prediction of the current control system, since there are only seven sets of candidates of control signals $u_{sd}(t_i)$ and $u_{sq}(t_i)$ for the implementation of the control law, the optimal control signals computed from (5.21) are not necessarily equal to any one of the seven pair values. Thus, a search procedure was needed in determination of the actual control signals $u_{sd}(t_i)$ and $u_{sq}(t_i)$ among the candidates.

From previous section, the optimal control design without constraints was proposed, the solution that could minimize the objective function is given by (5.23), which virtually leads to the zero value of the objective function $J$. Now, the optimal control signals are denoted by

$$\begin{bmatrix} u_{sd}(t_i)^{opt} \\ u_{sq}(t_i)^{opt} \end{bmatrix} = (\Delta t^2 B_m^T B_m)^{-1} B_m^T \begin{bmatrix} f_d(t_i) \\ f_q(t_i) \end{bmatrix}$$

(5.27)

Then, the objective function for the constrained control problem leads to

$$J = \left( \begin{bmatrix} u_{sd}(t_i) \\ u_{sq}(t_i) \end{bmatrix} - \begin{bmatrix} u_{sd}(t_i)^{opt} \\ u_{sq}(t_i)^{opt} \end{bmatrix} \right)^T (\Delta t^2 B_m^T B_m) \left( \begin{bmatrix} u_{sd}(t_i) \\ u_{sq}(t_i) \end{bmatrix} - \begin{bmatrix} u_{sd}(t_i)^{opt} \\ u_{sq}(t_i)^{opt} \end{bmatrix} \right)$$

(5.28)

where $J = J_0$ because $J_{min} = 0$. Since the weighting matrix $\Delta t^2 B_m^T B_m$ is

$$\Delta t^2 B_m^T B_m = \begin{bmatrix} \frac{\Delta t^2}{(r \sigma \tau_d)^2} & 0 \\ 0 & \frac{\Delta t^2}{(r \sigma \tau_d)^2} \end{bmatrix}$$
the objective function $J$ can also be written as

$$J = \frac{\Delta t^2}{(r_\sigma \tau'_\sigma)^2} (u_{sd}(t_i) - u_{sd}(t_i)^{opt})^2 + \frac{\Delta t^2}{(r_\sigma \tau'_\sigma)^2} (u_{sq}(t_i) - u_{sq}(t_i)^{opt})^2$$  \hspace{1cm} (5.29)

An immediate comment follows from (5.29). Note that the minimum value of the objective function when $u_{sd}(t_i) \neq u_{sd}(t_i)^{opt}$ and $u_{sq}(t_i) \neq u_{sq}(t_i)^{opt}$ is weighted by $\Delta t^2$, where $\Delta t$ is the sampling interval. It is obvious that in the finite control set method, the choice of sampling interval plays an important role as illustrated by simulation results. Further discussion of sampling interval will be introduced in the following section.

To seek the optimal solution that will minimize the objective function $J$ with the limited choices of $u_{sd}(t_i)$ and $u_{sq}(t_i)$, namely the seven pairs of $u_{sd}(t_i)$ and $u_{sq}(t_i)$, the seven values of the objective function $J$ (5.29) are calculated with respect to the candidate pairs of $u_{sd}(t_i)$ and $u_{sq}(t_i)$ and denoted as $J^0, J^1, J^2, \ldots, J^7$. A simple search function is used to find the minimal value of $J^m$ and its associated index.

There is a geometric interpretation for the minimization of the objective function (5.29) subject to the finite control set. Since the symmetrical structure of the induction motor (ie. $r_\sigma \tau'_\sigma = r_\sigma \tau'_\sigma$), the variations of the $J$ form a family of circles centered at $(u_{sd}(t_i)^{opt}, u_{sq}(t_i)^{opt})$. The optimal solution is the pair of $u_{sd}(t_i)^k$ and $u_{sq}(t_i)^k$ values that form a line to touch the circle in a shortest distance. This geometric interpretation is illustrated in Figure 5.3.

Although the original objective function (5.18) is identical to the objective function (5.29) after the analysis, the latter case offers an insight into the design problem, also more convenient in the computation of the control law. For the objective function (5.29), we can firstly calculate the feedback control gain $K_{fcs}$ and the optimal control signal without constraints. Then we evaluate the cost function with the actual seven pairs of
of voltage variables against the optimal solution. The pair that yields a smallest cost function is the solution of the control signal.

The control law is summarized as follows:

1. At sampling time \( t_i \), with the measured currents \( i_{sd}(t_i) \), \( i_{sq}(t_i) \), the reference currents \( i_{sd}(t_i)^* \) and \( i_{sq}(t_i)^* \), compute the optimal control signals \( u_{sd}(t_i)^{opt} \) and \( u_{sq}(t_i)^{opt} \) via

\[
\begin{bmatrix}
u_{sd}(t_i)^{opt} \\
u_{sq}(t_i)^{opt}
\end{bmatrix} = \begin{bmatrix}
\frac{r_{στ}}{Δt} & 0 \\
0 & \frac{r_{στ}}{Δt}
\end{bmatrix}
\begin{bmatrix}
\nu^*_sd(t_i) \\
\nu^*_sq(t_i)
\end{bmatrix}
-(I + ΔtA_m(t_i))
\begin{bmatrix}
\nu_{sd}(t_i) \\
\nu_{sq}(t_i)
\end{bmatrix}
-γ_D(t_i)Δt
\]

2. Compute the value of the objective function for \( k = 0, 1, 2, \ldots, 6 \)

\[
J^k = \frac{Δt^2}{(r_{στ})^2} (u_{sd}(t_i)^k - u_{sd}(t_i)^{opt})^2 + \frac{Δt^2}{(r_{στ})^2} (u_{sq}(t_i)^k - u_{sq}(t_i)^{opt})^2
\]

3. Find the minimum of the objective function \( J^k \) and its corresponding index number.

4. From this index number, construct the three phase voltage control signals.

The weighting factors on the errors of \( u_{sd} \) and \( u_{sq} \) are identical, thus it is sufficient to evaluate the objective function using

\[
J^k = (u_{sd}(t_i)^k - u_{sd}(t_i)^{opt})^2 + (u_{sq}(t_i)^k - u_{sq}(t_i)^{opt})^2
\]

However, the factors \( \frac{Δt^2}{(r_{στ})^2} \) and \( \frac{Δt^2}{(r_{στ})^2} \) could help resolving the scaling problem that may arise from large errors between these variables.

**Simulation Results.** The proposed method of revised FCS-MPC should be identical to the original FCS-MPC from the analysis before. Hence, another set of simulation results are obtained using the same settings in the simulation, the results are presented in Figure 5.4.

By comparing between the simulation results of both original FCS and revised FCS, the observation claims that they are identical to each other, which proves the validation of the proposed revised FCS method, which has the same capacity as the original FCS but the structure has been changed, so that, embedding an integrator become possible in the following section.

In order to perform the experimental result of revised FCS-MPC, the sampling time has been changed to 80μs, due to the limitation of the real-time computation of the xPC Target. The reference signal of \( i_{sq}^* \) is altered as well to a step pulse, the resistance at the coupled DC motor terminal has been reduced to 4 Ohm, in a way, the load torque
Experiment evaluation. Figure 5.5 presented the experimental results of the proposed method, the results of controlled outputs are obtained in Figure 5.5(a) and 5.5(b), the noise level of the measured currents are significantly large, due to several reasons, firstly, the sampling time is increased to 80µs, as we claimed previously, the FCS-MPC has the performance dependent on the sampling rate \( \Delta t \), another reason could be the mismatching between the ADC (Analogue-Digital-Converter) module and the PWM module inside the xPC Target implementation. Since the main objective in this section is focused on the steady-state error elimination, thus the obtained results are acceptable. Figure 5.5(c) presents the FFT (Fast Fourier Transform) analysis of the phase current, the switching frequency of the IGBT is obtained, unlike the PWM-based control with a centralized switching frequency distribution, FCS-MPC has its switching frequency spreading over a range of frequencies, which is another research issue of FCS but will not be discussed in this thesis. The motor speed is shown in Figure 5.5(d) to ensure that the velocity of motor is not too fast to cause the field saturation problem.
5.4 Revised FCS-MPC with Integral Action

As discussed previously, the finite control set method has one research question with the steady-state error. The reason of occurring steady-state error had been analyzed from the previous section, the design method adopted a high gain feedback control with constraints in which the feedback control gain is designed using least squares optimization and seven candidate pairs of the electrical voltage values, \( u_{sd}(t_i) \) and \( u_{sq}(t_i) \), in order to form the finite control set with constraints. However, the presented method does not have integral action, as a result, the steady-state errors are observed from the previous results. Similar to other linear control systems, the steady-state errors will be reduced if the feedback controller gain \( K_{fcs} \) is increased, in this case, the feedback control gain will be increased as the sampling interval \( \Delta t \) reduced as shown in (5.24).

In this section, the revised Finite Control Set analyzed previously is modified to include integral action in the controller. The design approach is to embed the integrator into the controller, which is the most widely used method in the applications. Similar to the Model Predictive Control discussed in previous chapters, the incremental model method is introduced to modify the existing schemes.
The essence of finite control set scheme is an optimal output feedback control with the gain matrix \( K_{fcs} \) where the optimal control signals \( u_{sd}(t_i) \) and \( u_{sq}(t_i) \) are expressed in the feedback control framework:

\[
\begin{bmatrix}
  u_{sd}(t_i)^{opt} \\
  u_{sq}(t_i)^{opt}
\end{bmatrix} = K_{fcs} \begin{bmatrix}
  i_{sd}(t_i) \\
  i_{sq}(t_i)
\end{bmatrix} - \begin{bmatrix}
  i_{sd}(t_i) \\
  i_{sq}(t_i)
\end{bmatrix}
\]

(5.30)

where the \( K_{fcs} \) is used for the reference signals as well as the measured current signals (a small modification from the original scheme), also the feedforward compensation is neglected in the simpler expression.

This finite control set controller has proportional control, but not includes integral action which is evident from (5.30). In order to generate integral action in the feedback controller, the integrated error signals between the current reference signals \( i_{sd}(t_i)^* \), \( i_{sq}(t_i)^* \) and \( i_{sd}(t_i) \), \( i_{sq}(t_i) \) will need to be included in the controller. Since the finite control set controller is a discrete time controller, discrete-time control system design is better suited. The operator for integrator in the discrete-time system is expressed as \( \frac{1}{1-q^{-1}} \) where \( q^{-1} \) is the backward shift operator defined as \( q^{-1}x(t_i) = x(t_i - \Delta t) \). In a similar expression to integral controller, an additional term that has the functionality of an integrator is added to the original finite control set scheme, leading to the new finite control set controller:

\[
\begin{bmatrix}
  u_{sd}(t_i)^{opt} \\
  u_{sq}(t_i)^{opt}
\end{bmatrix} = K_{fcs} \begin{bmatrix}
  \frac{k_I}{1-q^{-1}}(i_{sd}(t_i)^* - i_{sd}(t_i)) \\
  \frac{k_I}{1-q^{-1}}(i_{sq}(t_i)^* - i_{sq}(t_i))
\end{bmatrix} - K_{fcs} \begin{bmatrix}
  i_{sd}(t_i) \\
  i_{sq}(t_i)
\end{bmatrix}
\]

(5.31)

where \( k_I \) is the integral gain for the \( i_{sd} \) and \( i_{sq} \) currents where \( 0 < k_I \leq 1 \), the reason for using identical notation for both current is due to the identical time constant for both dynamics. The parameter \( k_I \) is the integral gain in the discrete integral controller.

Figure 5.6 shows the configuration of the new finite control set controller with integral action, where the control signals calculated are the \( u_{sd}^{opt} \) and \( u_{sq}^{opt} \).

In the presence of constraints, there are seven pairs of candidate variables for the \( u_{sd} \) and \( u_{sq} \) voltages. As before, upon obtaining the signals \( u_{sd}(t_i)^{opt} \) and \( u_{sq}(t_i)^{opt} \) at the sampling time \( t_i \), the actual control signals \( u_{sd}(t_i) \) and \( u_{sq}(t_i) \) are determined by computing the value of the objective function for \( k = 0, 1, 2, \ldots, 6 \)

\[
J^k = \frac{\Delta t^2}{(r_\sigma^2\tau_\sigma^2)^2}(u_{sd}(t_i)^k - u_{sd}(t_i)^{opt})^2 + \frac{\Delta t^2}{(r_\sigma^2\tau_\sigma^2)^2}(u_{sq}(t_i)^k - u_{sq}(t_i)^{opt})^2
\]

and finding the minimum of the objective function \( J^k \) and its corresponding index number. We may call this modified controller I-FCS.
5.4.1 Calculation of Integral Gain

The selection of the integral gain $k_I$ is discussed in this section. From Figure 5.6, there are two feedback loops for proposed control method, one is the inner-loop proportional control system which has the controller gain $K_{fcs}$ as discussed before, while the outer-loop controllers contain the integral action. So from this design topology, the closed-loop transfer function for the inner-loop system will depend on the outer-loop integral controller design. Recall that the inner-loop system has the discrete-time state space model (5.14):

$$
\begin{bmatrix}
i_{sd}(t_i + \Delta t) \\
i_{sq}(t_i + \Delta t)
\end{bmatrix} = (I + \Delta t A_m(t_i)) \begin{bmatrix} i_{sd}(t_i) \\
i_{sq}(t_i)
\end{bmatrix} + \Delta t B_m \begin{bmatrix} u_{sd}(t_i) \\
u_{sq}(t_i)
\end{bmatrix} + \gamma_D(t_i) \Delta t \quad (5.32)
$$

where the feedback control signals $u_{sd}(t_i)$ and $u_{sq}(t_i)$ are calculated using the feedback gain matrix $K_{fcs}$ as illustrated in the inner-loop control system:

$$
\begin{bmatrix}
u_{sd}(t_i) \\
u_{sq}(t_i)
\end{bmatrix} = K_{fcs} \begin{bmatrix} e_d(t_i) - i_{sd}(t_i) \\
e_q(t_i) - i_{sq}(t_i)
\end{bmatrix}
$$

where $e_d(t_i)$ and $e_q(t_i)$ are the reference signals to the inner-loop, as well as the manipulated signals for the outer-loop system. Then, by substituting the control signals above into the state space model (5.32), the closed-loop relationship between the $z$-transforms of the reference signals and the feedback signals is obtained as

$$
\begin{bmatrix}
I_{sd}(z) \\
I_{sq}(z)
\end{bmatrix} = (zI - (I + A_m(t_i)\Delta t) + \Delta t B_m K_{fcs})^{-1} \Delta t B_m K_{fcs} \begin{bmatrix} \tilde{I}_{sd}(z) \\
\tilde{I}_{sq}(z)
\end{bmatrix} \quad (5.33)
$$
where the feedback control gain $K_{fcs}$ is

$$K_{fcs}(t_i) = \begin{bmatrix} \frac{r_{xP}}{\Delta t} & 0 \\ 0 & \frac{r_{xqP}}{\Delta t} \end{bmatrix} (I + \Delta t A_m(t_i))$$

which can be easily verified that the matrix $(zI - (I + A_m(t_i)\Delta t) + \Delta t B_m K_{fcs})^{-1}$ has the diagonal form of

$$\begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{z} \end{bmatrix}$$

and the matrix $\Delta t B_m K_{fcs}$ becomes $I + \Delta t A_m(t_i)$. Thus the closed-loop transfer function (5.33) leads to

$$\begin{bmatrix} I_{sd}(z) \\ I_{sq}(z) \end{bmatrix} = \left( \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{z} \end{bmatrix} + \Delta t A_m(t_i) z^{-1} \right) \begin{bmatrix} \tilde{I}_{sd}(z) \\ \tilde{I}_{sq}(z) \end{bmatrix}$$

(5.34)

If the sampling interval $\Delta t$ is sufficiently small, the quantity of $\Delta t A_m(t_i) z^{-1}$ could be neglected. Thus the inner-loop dynamics are approximately expressed as

$$I_{sd}(z) \approx z^{-1}, \quad I_{sq}(z) \approx z^{-1}$$

Once the inner-loop system is understood, the design of the outer-loop integral controller becomes straightforward. By using $d$-axis current as example, the open-loop transfer function for the outer-loop system includes the integral controller $\frac{k_I}{1-z^{-1}}$ together with the time delay $z^{-1}$ from the inner closed-loop system. Hence, the outer closed-loop has the transfer function as

$$\frac{I_{sd}(z)}{I_{sd}(z)} = \frac{k_I z^{-1}}{1 - z^{-1} + k_I z^{-1}}$$

(5.35)

where the closed-loop pole is located at $1 - k_I$. By choosing a desired closed-loop pole as $0 \leq p_{cl} < 1$, the integral gain is determined as $k_I = 1 - p_{cl}$. The design for $q$-axis current is identical to (5.35).

The desired closed-loop pole $p_{cl}$ is a pole in the discrete-time system and this parameter is the design parameter selected by the user. We can choose it in a relative value to its continuous-time counterpart. For instance, if the continuous-time counterpart is $-a_{cl}$, the pole in discrete-time is $p_{cl} = e^{-a_{cl} \Delta t}$ where $\Delta t$ is the sampling interval. Thus, we can determine the value of $p_{cl}$ according to the desired response time in the continuous-time. For instance, if we wish the desired closed-loop current control system has a time constant of $100 \times 10^{-6}$ second ($a_{cl} = \frac{1}{100 \times 10^{-6}}$), then with a sampling interval $\Delta t = 10 \times 10^{-6}$ second, $p_{cl} = e^{-a_{cl} \Delta t} = e^{-0.1} = 0.9048$. By reducing the desired closed-loop constant to $50 \times 10^{-6}$ second, $p_{cl} = e^{-0.2} = 0.8187$. 


5.4.2 Derivation of I-FCS Controller without Constraints

The I-FCS control algorithm outlined in previous section is derived without constraints. Recall that the discretized model (5.32) of the induction motor current in \(dq\) coordinates is obtained as

\[
\begin{bmatrix}
    i_{sd}(t_i + \Delta t) \\
    i_{sq}(t_i + \Delta t)
\end{bmatrix} = (I + \Delta tA_m(t_i)) \begin{bmatrix}
    i_{sd}(t_i) \\
    i_{sq}(t_i)
\end{bmatrix} + \Delta tB_m \begin{bmatrix}
    u_{sd}(t_i) \\
    u_{sq}(t_i)
\end{bmatrix} + \gamma_D(t_i)\Delta t \quad (5.36)
\]

This approximation of the continuous-time differential equation model is still valid after the backward shift, which holds at the sampling time of \(t_i - \Delta t\), leads to

\[
\begin{bmatrix}
    i_{sd}(t_i) \\
    i_{sq}(t_i)
\end{bmatrix} = (I + \Delta tA_m(t_i - \Delta t)) \begin{bmatrix}
    i_{sd}(t_i - \Delta t) \\
    i_{sq}(t_i - \Delta t)
\end{bmatrix} + \Delta tB_m \begin{bmatrix}
    u_{sd}(t_i - \Delta t) \\
    u_{sq}(t_i - \Delta t)
\end{bmatrix} + \gamma_D(t_i - \Delta t)\Delta t \quad (5.37)
\]

Subtracting (5.37) from (5.36) leads to the difference model between two sampling instants:

\[
\begin{bmatrix}
    i_{sd}(t_i + \Delta t) - i_{sd}(t_i) \\
    i_{sq}(t_i + \Delta t) - i_{sq}(t_i)
\end{bmatrix} = (I + \Delta tA_m(t_i)) \begin{bmatrix}
    i_{sd}(t_i) - i_{sd}(t_i - \Delta t) \\
    i_{sq}(t_i) - i_{sq}(t_i - \Delta t)
\end{bmatrix} + \Delta tB_m \begin{bmatrix}
    u_{sd}(t_i) - u_{sd}(t_i - \Delta t) \\
    u_{sq}(t_i) - u_{sq}(t_i - \Delta t)
\end{bmatrix} + \Delta t(\gamma_D(t_i) - \gamma_D(t_i - \Delta t)) + (A_m(t_i) - A_m(t_i - \Delta t))\Delta t \begin{bmatrix}
    i_{sd}(t_i - \Delta t) \\
    i_{sq}(t_i - \Delta t)
\end{bmatrix} \quad (5.38)
\]

where in the process of derivation the following term is both added and subtracted to (5.38):

\[
A_m(t_i)\Delta t \begin{bmatrix}
    i_{sd}(t_i - \Delta t) \\
    i_{sq}(t_i - \Delta t)
\end{bmatrix}
\]

Thus the matrix \((A_m(t_i) - A_m(t_i - \Delta t))\Delta t\) contained in the final term of (5.38) is expressed as

\[
(A_m(t_i) - A_m(t_i - \Delta t))\Delta t = \begin{bmatrix}
    0 & a_{12} \\
    a_{21} & 0
\end{bmatrix}
\]

where \(a_{12} = \Delta t(\omega_s(t_i) - \omega_s(t_i - \Delta t))\) and \(a_{21} = -\Delta t(\omega_s(t_i) - \omega_s(t_i - \Delta t))\).

Since the quantity \(\Delta t(\omega_s(t_i) - \omega_s(t_i - \Delta t))\) is sufficiently small for a small sampling interval \(\Delta t\) in the induction motor drive application, the matrix \((A_m(t_i) - A_m(t_i - \Delta t))\Delta t\) is approximated by a zero matrix. Thus the final term of (5.38) is neglected, for the same reason, the compensation term \(\Delta t(\gamma_D(t_i) - \gamma_D(t_i - \Delta t))\) is neglected as well.
For notational simplicity, the following incremental variables are defined:

\[
\begin{align*}
\Delta i_{sd}(t_i + \Delta t) &= i_{sd}(t_i + \Delta t) - i_{sd}(t_i) \\
\Delta i_{sq}(t_i + \Delta t) &= i_{sq}(t_i + \Delta t) - i_{sq}(t_i) \\
\Delta i_{sd}(t_i) &= i_{sd}(t_i) - i_{sd}(t_i - \Delta t) \\
\Delta i_{sq}(t_i) &= i_{sq}(t_i) - i_{sq}(t_i - \Delta t) \\
\Delta u_{sd}(t_i) &= u_{sd}(t_i) - u_{sd}(t_i - \Delta t) \\
\Delta u_{sq}(t_i) &= u_{sq}(t_i) - u_{sq}(t_i - \Delta t)
\end{align*}
\]

With these incremental variables defined and the approximation taken, the incremental model of induction motor (5.38) becomes

\[
\begin{bmatrix}
\Delta i_{sd}(t_i + \Delta t) \\
\Delta i_{sq}(t_i + \Delta t)
\end{bmatrix} = (I + \Delta tA_m(t_i)) \begin{bmatrix}
\Delta i_{sd}(t_i) \\
\Delta i_{sq}(t_i)
\end{bmatrix} + \Delta tB_m \begin{bmatrix}
\Delta u_{sd}(t_i) \\
\Delta u_{sq}(t_i)
\end{bmatrix}
\]

(5.39)

In order to include the integral action into the controller, the weighted current errors \(e_d(t_i) = k_I(i_{sd}^*(t_i) - i_{sd}(t_i))\) and \(e_q(t_i) = k_I(i_{sq}^*(t_i) - i_{sq}(t_i))\) are chosen as the steady-states of \(\Delta i_{sd}(t_i)\) and \(\Delta i_{sq}(t_i)\), where \(0 < k_I < 1\). Subtracting the steady-states from the incremental model (5.39) leads to

\[
\begin{bmatrix}
\Delta i_{sd}(t_i + \Delta t) - e_d(t_i) \\
\Delta i_{sq}(t_i + \Delta t) - e_q(t_i)
\end{bmatrix} = (I + \Delta tA_m(t_i)) \begin{bmatrix}
\Delta i_{sd}(t_i) - e_d(t_i) \\
\Delta i_{sq}(t_i) - e_q(t_i)
\end{bmatrix} + \Delta tB_m \begin{bmatrix}
\Delta u_{sd}(t_i) \\
\Delta u_{sq}(t_i)
\end{bmatrix}
\]

(5.40)

Once again, the objective function \(J\) is defined for minimization as

\[
J = \begin{bmatrix}
\Delta i_{sd}(t_i + \Delta t) - e_d(t_i) \\
\Delta i_{sq}(t_i + \Delta t) - e_q(t_i)
\end{bmatrix} \begin{bmatrix}
\Delta i_{sd}(t_i + \Delta t) - e_d(t_i) \\
\Delta i_{sq}(t_i + \Delta t) - e_q(t_i)
\end{bmatrix}
\]

(5.41)

where the incremental current signals \(\Delta i_{sd}(t_i + \Delta t)\) and \(\Delta i_{sq}(t_i + \Delta t)\) are regulated to be as close as possible to \(e_d(t_i)\) and \(e_q(t_i)\).

Then, the following vector is defined for notational simplicity:

\[
\begin{bmatrix}
g_d(t_i) \\
g_q(t_i)
\end{bmatrix} = -(I + \Delta tA_m(t_i)) \begin{bmatrix}
\Delta i_{sd}(t_i) - e_d(t_i) \\
\Delta i_{sq}(t_i) - e_q(t_i)
\end{bmatrix}
\]

(5.42)

\[
\begin{bmatrix}
ge_d(t_i) \\
ge_q(t_i)
\end{bmatrix} = (I + \Delta tA_m(t_i)) \begin{bmatrix}
e_d(t_i) - \Delta i_{sd}(t_i) \\
e_q(t_i) - \Delta i_{sq}(t_i)
\end{bmatrix}
\]
Then by substituting (5.40) with the simplified notation into the objective function (5.41), it can be readily verified that the objective function has the expression

\[
J = \left( \begin{bmatrix} g_d(t_i) \\ g_q(t_i) \end{bmatrix} - \Delta t B_m \begin{bmatrix} \Delta u_{sd}(t_i) \\ \Delta u_{sq}(t_i) \end{bmatrix} \right)^T \begin{bmatrix} g_d(t_i) \\ g_q(t_i) \end{bmatrix} - \Delta t B_m \begin{bmatrix} \Delta u_{sd}(t_i) \\ \Delta u_{sq}(t_i) \end{bmatrix}
\]

which is

\[
J = \begin{bmatrix} g_d(t_i) \\ g_q(t_i) \end{bmatrix} g_d(t_i) - 2 \begin{bmatrix} \Delta u_{sd}(t_i) \\ \Delta u_{sq}(t_i) \end{bmatrix} \Delta t B_m \begin{bmatrix} g_d(t_i) \\ g_q(t_i) \end{bmatrix} + \begin{bmatrix} \Delta u_{sd}(t_i) \\ \Delta u_{sq}(t_i) \end{bmatrix} \Delta t^2 B_m^T B_m \begin{bmatrix} \Delta u_{sd}(t_i) \\ \Delta u_{sq}(t_i) \end{bmatrix}
\]

(5.43)

The objective function (5.43) is a quadratic function similar to the objective function (5.18) in previous section. By using the same least squares minimization procedure as outlined before, the optimal incremental control signals \(\Delta u_{sd}(t_i)\) and \(\Delta u_{sq}(t_i)\) are computed by minimizing the objective function (5.43)

\[
\begin{bmatrix} \Delta u_{sd}(t_i) \\ \Delta u_{sq}(t_i) \end{bmatrix} = (\Delta t^2 B_m^T B_m)^{-1} \Delta t B_m \begin{bmatrix} g_d(t_i) \\ g_q(t_i) \end{bmatrix}
\]

\[
= \frac{1}{\Delta t} \begin{bmatrix} r_\sigma \bar{\tau}_\sigma' \\ 0 \end{bmatrix} \begin{bmatrix} r_\sigma \bar{\tau}_\sigma \\ 0 \end{bmatrix} \begin{bmatrix} g_d(t_i) \\ g_q(t_i) \end{bmatrix}
\]

(5.44)

where the matrix \(B_m^T B_m\) is positive definite same as before, given by

\[
B_m^T B_m = \begin{bmatrix} \frac{1}{(r_\sigma \bar{\tau}_\sigma)^2} & 0 \\ 0 & \frac{1}{(r_\sigma \bar{\tau}_\sigma')^2} \end{bmatrix}
\]

By substituting the variables \(g_d(t_i)\) and \(g_q(t_i)\) (5.42) into the optimal solution (5.44), we obtain the expression of the incremental control signals

\[
\begin{bmatrix} \Delta u_{sd}(t_i) \\ \Delta u_{sq}(t_i) \end{bmatrix} = \frac{1}{\Delta t} \begin{bmatrix} r_\sigma \bar{\tau}_\sigma' \\ 0 \end{bmatrix} \begin{bmatrix} r_\sigma \bar{\tau}_\sigma \\ 0 \end{bmatrix} (I + \Delta t A_m(t_i)) \begin{bmatrix} e_d(t_i) - \Delta i_{sd}(t_i) \\ e_q(t_i) - \Delta i_{sq}(t_i) \end{bmatrix}
\]

\[
= K_{fcs} \begin{bmatrix} e_d(t_i) - \Delta i_{sd}(t_i) \\ e_q(t_i) - \Delta i_{sq}(t_i) \end{bmatrix}
\]

where the feedback control gain \(K_{fcs}\) is defined by

\[
K_{fcs} = \frac{1}{\Delta t} \begin{bmatrix} r_\sigma \bar{\tau}_\sigma' \\ 0 \end{bmatrix} \begin{bmatrix} r_\sigma \bar{\tau}_\sigma \\ 0 \end{bmatrix} (I + \Delta t A_m(t_i))
\]
which is identical to the case without integrator in previous section.

Recall that the weighted error signals are $e_{d}(t_i) = k_I(i^*_s(t_i) - i_{sd}(t_i))$ and $e_{q}(t_i) = k_I(i^*_s(t_i) - i_{sq}(t_i))$, which are substituted back into the optimal incremental control signal computation leading to

$$\begin{bmatrix}
\Delta u_{sd}(t_i) \\
\Delta u_{sq}(t_i)
\end{bmatrix} = K_{fcs} \left( \begin{bmatrix}
 k_I(i^*_s(t_i) - i_{sd}(t_i)) \\
 k_I(i^*_s(t_i) - i_{sq}(t_i))
\end{bmatrix} - \begin{bmatrix}
 \Delta i_{sd}(t_i) \\
 \Delta i_{sq}(t_i)
\end{bmatrix} \right)$$

Finally, as $q^{-1}$ denotes the shift operator, the incremental control signal $\Delta u_{sd}(t_i) = u_{sd}(t_i) - u_{sd}(t_i - \Delta t)$ is expressed as $(1 - q^{-1})u_{sd}(t_i)$, as well as $\Delta u_{sq}(t_i) = u_{sq}(t_i) - u_{sq}(t_i - \Delta t)$ is expressed as $(1 - q^{-1})u_{sq}(t_i)$, then, dividing (5.45) by the factor $1 - q^{-1}$ leads to

$$\begin{bmatrix}
u_{sd}(t_i) \\
u_{sq}(t_i)
\end{bmatrix} = K_{fcs} \left[ \begin{bmatrix}
 \frac{k_I}{1-q^{-1}}(i^*_s(t_i) - i_{sd}(t_i)) \\
 \frac{k_I}{1-q^{-1}}(i^*_s(t_i) - i_{sq}(t_i))
\end{bmatrix} - K_{fcs} \begin{bmatrix}
i_{sd}(t_i) \\
i_{sq}(t_i)
\end{bmatrix}\right]$$

where the incremental current signals $\Delta i_{sd}(t_i) = i_{sd}(t_i) - i_{sd}(t_i - \Delta t)$ and $\Delta i_{sq}(t_i) = i_{sq}(t_i) - i_{sq}(t_i - \Delta t)$ are replaced by $(1 - q^{-1})i_{sd}(t_i)$ and $(1 - q^{-1})i_{sq}(t_i)$, respectively. The optimal control signals $u_{sd}(t_i)$ and $u_{sq}(t_i)$ given by (5.45) are identical to the proposed control signals in (5.31). This completes the derivation of the optimal control signal with integral action without constraints.

An equivalent expression of the cascaded feedback system based on (5.45) is shown in Figure 5.6, which can be seen as moving the integrators in Figure 5.7 from outer-loops to the inner-loop.

**Figure 5.6:** Feedback current control using I-FCS.
5.4.3 Optimal Control Design with Constraints

To consider the constraints implementation within the optimal control design, the objective function \( J \) used in the previous section is modified for the completed square analysis.

By adding and subtracting the following term to the original objective function \( J \) given by (5.43), whose quantity remains unchanged.

\[
\begin{bmatrix}
g_d(t_i) & g_q(t_i)
g_d(t_i) & g_q(t_i)
\end{bmatrix} \Delta t B_m (\Delta t^2 B_m^T B_m)^{-1} B_m^T \Delta t \begin{bmatrix} g_d(t_i) \\
g_q(t_i)
\end{bmatrix}
\]

with this term added, the completed squares is formed as:

\[
J_0 = \left( \begin{bmatrix}
\Delta u_{sd}(t_i) \\
\Delta u_{sq}(t_i)
\end{bmatrix} - (\Delta t^2 B_m^T B_m)^{-1} B_m^T \Delta t \begin{bmatrix} g_d(t_i) \\
g_q(t_i)
\end{bmatrix} \right)^T \Delta t^2 B_m^T B_m \) (5.47)
\]

\[
\times \left( \begin{bmatrix}
\Delta u_{sd}(t_i) \\
\Delta u_{sq}(t_i)
\end{bmatrix} - (\Delta t^2 B_m^T B_m)^{-1} B_m^T \Delta t \begin{bmatrix} g_d(t_i) \\
g_q(t_i)
\end{bmatrix} \right)
\]

With the \( J_0 \) given by (5.48), the original objective function \( J \) becomes

\[
J = J_0 + J_{\text{min}}
\]

where \( J_{\text{min}} \) is

\[
J_{\text{min}} = \begin{bmatrix}
g_d(t_i) & g_q(t_i)
g_d(t_i) & g_q(t_i)
\end{bmatrix} (I - B_m (B_m^T B_m)^{-1} B_m^T) \begin{bmatrix} g_d(t_i) \\
g_q(t_i)
\end{bmatrix} = 0
\]

due to the diagonal matrix \( B_m \), which is

\[
B_m = \begin{bmatrix}
\frac{1}{r_o \sigma} & 0 \\
0 & \frac{1}{r_o \sigma}
\end{bmatrix}
\]

Therefore, the minimum of the original objective function \( J \) is achieved if \( J_0 \) is minimized with the optimal control signals:

\[
\begin{bmatrix}
\Delta u_{sd}(t_i) \\
\Delta u_{sq}(t_i)
\end{bmatrix} = (\Delta t^2 B_m^T B_m)^{-1} B_m^T \Delta t \begin{bmatrix} g_d(t_i) \\
g_q(t_i)
\end{bmatrix} \) (5.48)
\]

This solution is identical to the solution presented in (5.44) that has been derived without constraints.

By defining the unconstrained solution as the optimal solution denoted by \( \Delta u_{sd}(t_i)^{\text{opt}} \)
Thus, by calculating the actual incremental control signal \( s \) using the same past control objective function
\[
J = \left( \begin{array}{c}
\Delta u_{sd}(t_i)^{opt} \\
\Delta u_{sq}(t_i)^{opt}
\end{array} \right) = (\Delta t^2 B_m^T B_m)^{-1} B_m^T \Delta t
\begin{bmatrix}
g_d(t_i) \\
g_q(t_i)
\end{bmatrix}
\]
By substituting this expression into the original objective function, leads to
\[
J = \left( \begin{array}{c}
\Delta u_{sd}(t_i) \\
\Delta u_{sq}(t_i)
\end{array} \right) - \left( \begin{array}{c}
\Delta u_{sd}(t_i)^{opt} \\
\Delta u_{sq}(t_i)^{opt}
\end{array} \right) = \Delta u_{sd}(t_i) - \Delta u_{sq}(t_i)
\]
Also, by definition of the incremental control signals, the following relationship is true:
\[
\begin{bmatrix}
\Delta u_{sd}(t_i)^{opt} \\
\Delta u_{sq}(t_i)^{opt}
\end{bmatrix} = \begin{bmatrix}
u_{sd}(t_i)^{opt} \\
u_{sq}(t_i)^{opt}
\end{bmatrix} - \begin{bmatrix}
\Delta u_{sd}(t_i) \\
\Delta u_{sq}(t_i)
\end{bmatrix}
\]
Thus, by calculating the actual incremental control signals using the same past control signal states, that is,
\[
\begin{bmatrix}
\Delta u_{sd}(t_i) \\
\Delta u_{sq}(t_i)
\end{bmatrix} = \begin{bmatrix}
u_{sd}(t_i) \\
u_{sq}(t_i)
\end{bmatrix} - \begin{bmatrix}
\Delta u_{sd}(t_i) \\
\Delta u_{sq}(t_i)
\end{bmatrix}
\]
The objective function, which is used in the implementation of the I-FCS, is defined as
\[
J = \left( \begin{array}{c}
u_{sd}(t_i) \\
u_{sq}(t_i)
\end{array} \right) - \left( \begin{array}{c}
u_{sd}(t_i)^{opt} \\
u_{sq}(t_i)^{opt}
\end{array} \right) = \frac{\Delta t^2}{(r_m T)^2} (u_{sd}(t_i) - u_{sd}(t_i)^{opt})^2 + \frac{\Delta t^2}{(r_m T)^2} (u_{sq}(t_i) - u_{sq}(t_i)^{opt})^2
\]
In the presence of constraints, there are seven pairs of candidate variables for the \( v_d \) and \( v_q \) voltages. When having the integrators in the I-FCS controller, upon obtaining the signals \( v_{d}(t_i)^{opt} \) and \( v_{q}(t_i)^{opt} \) with integral action at the sampling time \( t_i \), the actual control signals \( v_{d}(t_i) \) and \( v_{q}(t_i) \) are determined by computing the value of the objective function for \( k = 0, 1, 2, \ldots, 6 \)
\[
J^k = \frac{\Delta t^2}{(r_m T)^2} (u_{sd}(t_i)^k - u_{sd}(t_i)^{opt})^2 + \frac{\Delta t^2}{(r_m T)^2} (u_{sq}(t_i)^k - u_{sq}(t_i)^{opt})^2
\]
The pair of constrained control signals \( u_{sd}(t_i)^k \) and \( u_{sq}(t_i)^k \) is found to minimize the objective function \( J^k \) subject to the index number \( k \).
5.4.4 Implementation of I-FCS Controller

The design of the Finite Control Set with integral action is shown in Figure 5.7 as the configuration of a cascaded feedback control system, which clearly indicated that integrators have been embedded in the outer-loop systems. The equivalent structure of FCS controller with integrators is shown in Figure 5.6, that is applied for the implementation of the I-FCS control algorithm, which is presented using the difference of the control signals with iterative computation. From Figure 5.6, we obtain:

\[
\begin{bmatrix}
\Delta u_{sd}(t_i)^{opt} \\
\Delta u_{sq}(t_i)^{opt}
\end{bmatrix} = K_{fcs} \begin{bmatrix}
k_f(i_{sd}^*(t_i) - i_{sd}(t_i)) \\
k_f(i_{sq}^*(t_i) - i_{sq}(t_i))
\end{bmatrix} - K_{fcs} \begin{bmatrix}
\Delta i_{sd}(t_i) \\
\Delta i_{sq}(t_i)
\end{bmatrix}
\]

where the following relationships are expressed:

\[
(1 - q^{-1})u_{sd}(t_i)^{opt} = u_{sd}(t_i)^{opt} - u_{sd}(t_i - \Delta t)^{opt} = \Delta u_{sd}(t_i)^{opt}
\]
\[
(1 - q^{-1})u_{sq}(t_i)^{opt} = u_{sq}(t_i)^{opt} - u_{sq}(t_i - \Delta t)^{opt} = \Delta u_{sq}(t_i)^{opt}
\]
\[
(1 - q^{-1})i_{sd}(t_i) = i_{sd}(t_i) - i_{sd}(t_i - \Delta t) = \Delta i_{sd}(t_i)
\]
\[
(1 - q^{-1})i_{sq}(t_i) = i_{sq}(t_i) - i_{sq}(t_i - \Delta t) = \Delta i_{sq}(t_i)
\]

Thus, the optimal control signals \(u_{sd}(t_i)^{opt}\) and \(u_{sq}(t_i)^{opt}\) are calculated based the past sample states of these signals together with their incremental values:

\[
\begin{bmatrix}
u_{sd}(t_i)^{opt} \\
u_{sq}(t_i)^{opt}
\end{bmatrix} = \begin{bmatrix}
u_{sd}(t_i) - \Delta t)^{opt} \\
u_{sq}(t_i) - \Delta t)^{opt}
\end{bmatrix} + \begin{bmatrix}
\Delta u_{sd}(t_i)^{opt} \\
\Delta u_{sq}(t_i)^{opt}
\end{bmatrix}
\]

With the optimal control signals defined, the objective function \(J\) is calculated with respect to the seven pairs of candidate voltage control variables.

The implementation procedure is discussed as follows:

1. At the sampling time \(t_i - \Delta t\), take the measurements of \(u_{sd}(t_i - \Delta t), u_{sq}(t_i - \Delta t), i_{sd}(t_i - \Delta t), i_{sq}(t_i - \Delta t)\). Initialize the optimal control signals at the sampling time \(t_i - \Delta t\) as \(u_{sd}(t_i - \Delta t)^{opt} = u_{sd}(t_i - \Delta t), u_{sq}(t_i - \Delta t)^{opt} = u_{sq}(t_i - \Delta t)\). Perform the following computation at the sampling time \(t_i\) with current measurements \(i_{sd}(t_i), i_{sq}(t_i)\) and their reference signals \(i_{sd}^*(t_i), i_{sq}^*(t_i)\).

2. Calculate the optimal incremental control signals using the following equation:

\[
\begin{bmatrix}
\Delta u_{sd}(t_i)^{opt} \\
\Delta u_{sq}(t_i)^{opt}
\end{bmatrix} = K_{fcs} \begin{bmatrix}
k_f(i_{sd}^*(t_i) - i_{sd}(t_i)) \\
k_f(i_{sq}^*(t_i) - i_{sq}(t_i))
\end{bmatrix} - K_{fcs} \begin{bmatrix}
\Delta i_{sd}(t_i) \\
\Delta i_{sq}(t_i)
\end{bmatrix}
\]
3. Calculate the optimal control signals using the past optimal control states:

\[
\begin{bmatrix}
    u_{sd}(t_i)^{opt} \\
    u_{sq}(t_i)^{opt}
\end{bmatrix}
= \begin{bmatrix}
    u_{sd}(t_i - \Delta t)^{opt} \\
    u_{sq}(t_i - \Delta t)^{opt}
\end{bmatrix} + \begin{bmatrix}
    \Delta u_{sd}(t_i)^{opt} \\
    \Delta u_{sq}(t_i)^{opt}
\end{bmatrix}
\]

4. Calculate the value of the objective function \( J \) with respect to the finite control set for \( k = 0, 1, 2, \ldots, 7 \)

\[
J^k = \frac{\Delta t^2}{(r_\sigma \tau_\sigma)^2} (u_{sd}(t_i)^k - u_{sd}(t_i)^{opt})^2 + \frac{\Delta t^2}{(r_\sigma \tau_\sigma)^2} (u_{sq}(t_i)^k - u_{sq}(t_i)^{opt})^2
\]

5. Find the minimum of \( J^k \) and its corresponding index, which leads to the control signals to be implemented.

6. Go to Step 2 in the computation as the sampling time progresses to \( t_i + \Delta t \).

### 5.4.5 Experimental Results and Discussions

![Experimental result of I-FCS in dq frame. Key: line(1) Actual feedback; line (2) set-point signal.](image-url)
The experimental results are obtained from the xPC Target-based induction motor control test-bed where MATLAB Simulink software is used for control algorithm implementation and the induction motor is coupled with a servo DC motor as its load. The supply voltage at DC-link is 520 V.

In the current control scheme, the sampling interval is $\Delta t = 80 \mu s$ that is the lowest limit restricted by the equipment. The closed-loop reference signals for the $d$-axis stator current is $i_{sd}^* = 0.877A$ and the $q$-axis stator current is $i_{sq}^* = 1.5A$. Current control experiments have been presented using the I-FCS predictive control scheme, to compare with the original FCS results presented in Figure 5.5.

**Current response.** Figure 5.8 presents the experimental results obtained from using the I-FCS predictive control scheme, where the integral gain is selected as $k_I = 0.15$. Comparing Figure 5.5(a)-5.5(b) with Figure 5.8(a)-5.8(b), it is seen that the noise levels of both $d$-axis and $q$-axis stator currents are about the same. However, calculations shown in Table 5.3 present that when using the I-FCS predictive controller, the mean values of the steady-state error have been significantly reduced comparing with when using the original FCS controller.

It is worthwhile to note that the steady-state response of the original FCS predictive control system is dependent on the selection of the system physical parameters. However, with the integral FCS predictive controller, this performance uncertainty in steady-state operation is removed.

**Frequency response of phase current.** Comparing Figure 5.5(c) with 5.8(c), there
Chapter 5. Finite Control Set - Model Predictive Control

Table 5.3: Comparison between original FCS and I-FCS

<table>
<thead>
<tr>
<th></th>
<th>mean of SSE $i_{sd}$</th>
<th>mean of SSE $i_{sq}$</th>
<th>Steady-state speed</th>
<th>Response time</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCS</td>
<td>0.0164A</td>
<td>0.0361A</td>
<td>520RPM</td>
<td>0.0005s</td>
</tr>
<tr>
<td>$k_I = 0.15$</td>
<td>-2.5397 × 10^{-3}A</td>
<td>-3.6636 × 10^{-4}A</td>
<td>550RPM</td>
<td>0.0011s</td>
</tr>
<tr>
<td>$k_I = 0.25$</td>
<td>2.8822 × 10^{-4}A</td>
<td>9.0139 × 10^{-6}A</td>
<td>550RPM</td>
<td>0.0007s</td>
</tr>
</tbody>
</table>

is no significant difference between the amplitude of the frequency response of the phase current. It is worthwhile to emphasis that because there is no pulse width modulation used in the implementation, which is different from the previous controllers implementation, the amplitude of frequency response is quite widely spread for both cases, instead of locating along the carrier frequency.

**Velocity response.** Although the steady-state errors when using the original FCS predictive controller are quite small for this induction motor with this set of physical parameters, the dynamic response of the velocity is affected. Figure 5.8(d) compares the open-loop responses of the speed of the motor with the original FCS predictive controller and the I-FCS predictive controller. Note that under the identical load condition of the coupled DC motor, the motor runs a bit faster during steady-state when using the I-FCS method. The reason for the difference could be the mean values of the currents produced by the induction motor are larger and closer to the reference signals, hence leading to faster steady-state speed. This could also imply that the motor runs more efficiently when using the I-FCS predictive controller which removes the steady-state current errors.

**Selection of the integral gain.** The integral gain $k_I$ is the performance parameter for the closed-loop current responses that affect the dynamic response of the currents. In general, the larger the gain is, the faster the current dynamic response will be. Figure 5.9 shows transient response of $i_{sq}$ during the step change from 0 to 1.5A, where three different values of integral gain are used. When $k_I = 0.05$, the response time is slowest as the closed-loop pole is closer to the unit circle. The fastest response is obtained with $k_I = 0.25$. Note that, the transient responses are very close between $k_I = 0.25$ and $k_I = 0.15$, due to the limitation of the IGBTs operation.
5.5 FCS-MPC in $\alpha\beta$-Coordinates

In the previous section, the I-FCS current control is designed in $dq$-coordinates, where the current reference signals are constant or piece-wise constant. Therefore, the FCS-MPC plus integral controller derived in the form of I-FCS are appropriate to regulate the currents in the rotational $dq$ frame.

In this section, the FCS-MPC of current in the $\alpha\beta$-frame (or the stationary frame) is investigated. Since the control system is directly designed based on the fixed $\alpha\beta$ frame, the observation of the rotating coordinates is unnecessary. However, the currents $i_{\alpha}(t)$ and $i_{\beta}(t)$ in the $\alpha\beta$-coordinates are sinusoidal functions, due to the linear transformation from the three phase currents $i_a(t)$, $i_b(t)$ and $i_c(t)$. As a result, the control signals $u_{\alpha}(t)$ and $u_{\beta}(t)$ are also sinusoidal. Therefore, the reference signals to the FCS-predictive control system requires sinusoidal signals as well, which is different from those in $dq$ frame.

From Chapter 2, the mathematical model of the induction motor in the stationary frame are decoupled and time-invariant with respect to the inputs and outputs. Thus, the original FCS controllers are analyzed in this section, however in order to track the sinusoidal reference signals without steady-state errors, the resonant characteristic is embedded inside the controller for the $\alpha\beta$ frame. Furthermore, the frequency of these sinusoidal signals are dependent to the synchronous velocity $\omega_s(t)$, hence the resonant controller is time-varying for such application.

5.5.1 Original FCS-MPC Current Control

The idea of original FCS-predictive control of current in $\alpha\beta$ coordinates is designed identically to the controller in $dq$ frame. Both of their objective functions are computing the sum of squares errors between the current reference signals ($i_{\alpha}^*, i_{\beta}^*$) and the one-step-ahead prediction of the current signals ($i_{\alpha}, i_{\beta}$), then at sampling time $t_i$, the optimal voltage control signals ($u_{\alpha}, u_{\beta}$) are obtained with respect to the minimum value of the objective function.

From Chapter 2, the dynamic model of induction motor currents are recalled as follows:

$$\frac{di_{\alpha}}{dt} = -\frac{1}{\tau_{\alpha}}i_{\alpha} + \frac{k_r}{r_{\sigma}\tau_{\sigma}r_r}\psi_{\tau \alpha}(t) + \frac{k_r}{r_{\sigma}\tau_{\sigma}r_r}\omega_e(t)\psi_{\tau \beta}(t) + \frac{1}{r_{\sigma}\tau_{\sigma}}u_{\alpha}(t) \quad (5.51)$$

$$\frac{di_{\beta}}{dt} = -\frac{1}{\tau_{\beta}}i_{\beta} - \frac{k_r}{r_{\sigma}\tau_{\sigma}r_r}\psi_{\tau \alpha}(t) + \frac{k_r}{r_{\sigma}\tau_{\sigma}r_r}\psi_{\tau \beta}(t) + \frac{1}{r_{\sigma}\tau_{\sigma}}u_{\beta}(t) \quad (5.52)$$

where the model has two single-input-single-output system with coupled flux disturbances, thus similar to the previous section for $dq$ frame, the objective function is selected
for calculating the control variables at arbitrary sampling time $t_i$.

$$J = (i_{sa}(t_i) - i_{sa}(t_i + \Delta t))^2 + (i_{s\beta}(t_i) - i_{s\beta}(t_i + \Delta t))^2$$

$$= \left( \begin{bmatrix} i_{sa}(t_i) \\ i_{s\beta}(t_i) \end{bmatrix} - \begin{bmatrix} i_{sa}(t_i + \Delta t) \\ i_{s\beta}(t_i + \Delta t) \end{bmatrix} \right)^T \left( \begin{bmatrix} i_{sa}(t_i) \\ i_{s\beta}(t_i) \end{bmatrix} - \begin{bmatrix} i_{sa}(t_i + \Delta t) \\ i_{s\beta}(t_i + \Delta t) \end{bmatrix} \right)$$

(5.53)

where $i_{sa}(t_i + \Delta t)$ and $i_{s\beta}(t_i + \Delta t)$ are one-step-ahead predictions of $i_{sa}(t_i)$ and $i_{s\beta}(t_i)$, respectively. The predictions are expressed in matrix and vector form by using first order approximation from the model equations (5.51) and (5.52)

$$\begin{bmatrix} i_{sa}(t_i + \Delta t) \\ i_{s\beta}(t_i + \Delta t) \end{bmatrix} = (I + \Delta t A_m) \begin{bmatrix} i_{sa}(t_i) \\ i_{s\beta}(t_i) \end{bmatrix} + \Delta t B_m \begin{bmatrix} u_{sa}(t_i) \\ u_{s\beta}(t_i) \end{bmatrix}$$

$$+ \Delta t \begin{bmatrix} \frac{k_r}{r \tau_r \tau_e} \psi_{r\alpha}(t_i) + \frac{k_r}{r \tau_r \tau_e} \omega_e(t_i) \psi_{r\beta}(t_i) \\ -\frac{k_r}{r \tau_r \tau_e} \omega_e(t_i) \psi_{r\alpha}(t_i) + \frac{k_r}{r \tau_r \tau_e} \psi_{r\beta}(t_i) \end{bmatrix}$$

(5.54)

where $I$ is a $2 \times 2$ identity matrix and the system matrices $A_m$ and $B_m$ are defined as

$$A_m = \begin{bmatrix} -\frac{1}{r} & 0 \\ 0 & -\frac{1}{r} \end{bmatrix}; \quad B_m = \begin{bmatrix} \frac{1}{r \tau_r \tau_e} & 0 \\ 0 & \frac{1}{r \tau_r \tau_e} \end{bmatrix}$$

By substituting the one-step-ahead prediction (5.54) into the objective function $J$ (5.53), leads to

$$J = \begin{bmatrix} f_\alpha(t_i) & f_\beta(t_i) \end{bmatrix} \begin{bmatrix} f_\alpha(t_i) \\ f_\beta(t_i) \end{bmatrix} - 2 \begin{bmatrix} u_{sa}(t_i) & u_{s\beta}(t_i) \end{bmatrix} \Delta t B_m^T \begin{bmatrix} u_{sa}(t_i) \\ u_{s\beta}(t_i) \end{bmatrix}$$

$$+ \begin{bmatrix} u_{sa}(t_i) & u_{s\beta}(t_i) \end{bmatrix} \Delta t^2 B_m^T B_m \begin{bmatrix} u_{sa}(t_i) \\ u_{s\beta}(t_i) \end{bmatrix}$$

(5.55)

where the auxiliary functions $f_\alpha(t_i)$ and $f_\beta(t_i)$ are defined as

$$\begin{bmatrix} f_\alpha(t_i) \\ f_\beta(t_i) \end{bmatrix} = \begin{bmatrix} i^*_{sa}(t_i) \\ i^*_{s\beta}(t_i) \end{bmatrix} - (I + \Delta t A_m) \begin{bmatrix} i_{sa}(t_i) \\ i_{s\beta}(t_i) \end{bmatrix}$$

$$- \Delta t \begin{bmatrix} \frac{k_r}{r \tau_r \tau_e} \psi_{r\alpha}(t_i) + \frac{k_r}{r \tau_r \tau_e} \omega_e(t_i) \psi_{r\beta}(t_i) \\ -\frac{k_r}{r \tau_r \tau_e} \omega_e(t_i) \psi_{r\alpha}(t_i) + \frac{k_r}{r \tau_r \tau_e} \psi_{r\beta}(t_i) \end{bmatrix}$$

(5.56)
Similar to the derivation steps outlined in \(dq\) frame section, the optimal control signals \(u_{s\alpha}(t_i)\) and \(u_{s\beta}(t_i)\) that minimizes the objective function \(J\) are computed as:

\[
\begin{bmatrix}
    u_{s\alpha}(t_i)_{opt} \\
    u_{s\beta}(t_i)_{opt}
\end{bmatrix} = (\Delta t^2 B_m^T B_m)^{-1} \Delta t B_m^T \begin{bmatrix}
    f_\alpha(t_i) \\
    f_\beta(t_i)
\end{bmatrix}
\]

\[
= \frac{1}{\Delta t} \begin{bmatrix}
    r_\sigma \tau'_\sigma & 0 \\
    0 & r_\sigma \tau'_\sigma
\end{bmatrix} \begin{bmatrix}
    f_\alpha(t_i) \\
    f_\beta(t_i)
\end{bmatrix}
\]

(5.57)

Note that in \(\alpha\beta\) reference frame, the system matrix \(A_m\) is diagonal and there is no interaction between current \(i_{s\alpha}\) and \(i_{s\beta}\). Thus the calculations of optimal control signals \(u_{s\alpha}^{opt}\) and \(u_{s\beta}^{opt}\) are scalar operation. Since the matrix \(I + A_m \Delta t\) is a diagonal having the form:

\[
I + A_m \Delta t = \begin{bmatrix}
    1 - \frac{\Delta t}{\tau_\sigma} & 0 \\
    0 & 1 - \frac{\Delta t}{\tau_\sigma}
\end{bmatrix}
\]

Hence (5.57) leads to

\[
u_{s\alpha}(t_i)_{opt} = \frac{r_\sigma \tau'_\sigma}{\Delta t} i_{s\alpha}^*(t_i) - \frac{r_\sigma \tau'_\sigma}{\Delta t} \frac{\Delta t}{\tau_\sigma} i_{s\alpha}(t_i) - \frac{k_r}{r_\sigma \tau_\sigma} \psi_{\alpha}(t_i) - \frac{k_r}{r_\sigma \tau'_\sigma} \omega_{\epsilon}(t_i) \psi_{\beta}(t_i)
\]

\[
u_{s\beta}(t_i)_{opt} = \frac{r_\sigma \tau'_\sigma}{\Delta t} i_{s\beta}^*(t_i) - \frac{r_\sigma \tau'_\sigma}{\Delta t} \frac{\Delta t}{\tau_\sigma} i_{s\beta}(t_i) + \frac{k_r}{r_\sigma \tau_\sigma} \omega_{\epsilon}(t_i) \psi_{\alpha}(t_i) - \frac{k_r}{r_\sigma \tau'_\sigma} \psi_{\beta}(t_i)
\]

where \(i_{s\alpha}^*(t_i)\) and \(i_{s\beta}^*(t_i)\) are current reference signals in the \(\alpha\beta\) reference frame, \(\omega_{\epsilon}(t_i)\), \(i_{s\alpha}(t_i)\) and \(i_{s\beta}(t_i)\) are from the measurement at sampling time \(t_i\), \(\psi_{\alpha}(t_i)\) and \(\psi_{\beta}(t_i)\) are estimated based on the model equations as

\[
\psi_{\alpha}(t_i) = \psi_{\alpha}(t_i - \Delta t) + \Delta t \left( \frac{L_h}{\tau_r} i_{s\alpha}(t_i - \Delta t) - \frac{1}{\tau_r} \psi_{\alpha}(t_i - \Delta t) - \omega_{\epsilon}(t_i - \Delta t) \psi_{\beta}(t_i - \Delta t) \right)
\]

\[
\psi_{\beta}(t_i) = \psi_{\beta}(t_i - \Delta t) + \Delta t \left( \frac{L_h}{\tau_r} i_{s\beta}(t_i - \Delta t) + \omega_{\epsilon}(t_i - \Delta t) \psi_{\alpha}(t_i - \Delta t) - \frac{1}{\tau_r} \psi_{\beta}(t_i - \Delta t) \right)
\]

It is clearly seen that predictive controller uses a high gain proportional feedback control with a feedforward compensation. Furthermore, the feedback control gain is dependent on the sampling interval of the current control system with the value

\[
k_{fcs}^\alpha = k_{fcs}^\beta = \frac{r_\sigma \tau'_\sigma}{\Delta t} \left( 1 - \frac{\Delta t}{\tau_\sigma} \right)
\]

(5.58)

In order to ensure a negative feedback in the current control, the quantity \(1 - \frac{\Delta t}{\tau_\sigma} < 1\), that is \(\Delta t < \frac{\tau_\sigma}{2}\), the sampling time has to be smaller than the current model time constant. A similar approach to the FCS predictive control in \(dq\) frame is adopted to the FCS problem in the \(\alpha\beta\) coordinates. Following the same procedure as outlined in previous section of this chapter, the objective function \(J\) (5.55) is expressed in term of the optimal
control signals in \( \alpha \beta \) frame as voltage signals in the \( \alpha \beta \) reference frame as

\[
J = \left( \begin{array}{c}
u_{\alpha\alpha}(t_i) \\
u_{\alpha\beta}(t_i) \\
u_{\beta\alpha}(t_i) \\
u_{\beta\beta}(t_i)
\end{array} \right) - \left( \begin{array}{c}
u_{\alpha\alpha}(t_i)^{opt} \\
u_{\alpha\beta}(t_i)^{opt} \\
u_{\beta\alpha}(t_i)^{opt} \\
u_{\beta\beta}(t_i)^{opt}
\end{array} \right) \right) \left( \begin{array}{c}
u_{\alpha\alpha}(t_i) \\
u_{\alpha\beta}(t_i) \\
u_{\beta\alpha}(t_i) \\
u_{\beta\beta}(t_i)
\end{array} \right) - \left( \begin{array}{c}
u_{\alpha\alpha}(t_i)^{opt} \\
u_{\alpha\beta}(t_i)^{opt} \\
u_{\beta\alpha}(t_i)^{opt} \\
u_{\beta\beta}(t_i)^{opt}
\end{array} \right) \right)^T \left( \Delta t^2 B_m^T B_m \right) \left( \begin{array}{c}
u_{\alpha\alpha}(t_i) \\
u_{\alpha\beta}(t_i) \\
u_{\beta\alpha}(t_i) \\
u_{\beta\beta}(t_i)
\end{array} \right) - \left( \begin{array}{c}
u_{\alpha\alpha}(t_i)^{opt} \\
u_{\alpha\beta}(t_i)^{opt} \\
u_{\beta\alpha}(t_i)^{opt} \\
u_{\beta\beta}(t_i)^{opt}
\end{array} \right)
\]

\[
\frac{\Delta t^2}{r_\alpha^2} (\nu_{\alpha\alpha}(t_i) - \nu_{\alpha\alpha}(t_i)^{opt})^2 + \frac{\Delta t^2}{r_\beta^2} (\nu_{\beta\beta}(t_i) - \nu_{\beta\beta}(t_i)^{opt})^2 = \frac{\Delta t^2}{(r_\alpha r_\beta)^2} (\nu_{\alpha\beta}(t_i) - \nu_{\beta\alpha}(t_i)^{opt})^2 (5.59)
\]

With both objective function and the optimal control signals defined, the next step in the FCS predictive control is to find the control signal \( u_{\alpha\alpha}(t_i) \) and \( u_{\alpha\beta}(t_i) \) that will minimize the objective function subject to the limited number of choices of voltage variables. In the \( \alpha \beta \) reference frame, there are seven pairs of candidate voltage values, which are also time-invariant. Their exact values are characterized by the values listed below:

\[
\begin{bmatrix} 0 & 1 & \frac{1}{2} & -\frac{1}{2} & 1 & -\frac{1}{2} & \frac{1}{2} \\ 0 & 0 & \frac{\sqrt{3}}{2} & \frac{\sqrt{3}}{2} & 0 & -\frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \frac{2}{3} V_{dc} \tag{5.60}
\]

In FCS current control proposed in the \( \alpha \beta \) reference frame, the seven pairs of \( u_{\alpha\alpha} \) and \( u_{\alpha\beta} \) values from (5.60) are used to evaluate the objective function (5.59). The pair of \( u_{\alpha\alpha} \) and \( u_{\alpha\beta} \) that has yielded a minimum of the objective function \( J \) will be chosen as the FCS current control signals in the \( \alpha \beta \) reference frame.

As we may recall, in the \( dq \) reference frame, there are also seven candidate pairs that the \( u_{sd}(t_i) \) and \( u_{sq}(t_i) \) are permitted to take. However, the candidate variables are sinusoidal functions with respect to the electrical angle \( \theta_s(t) \). Because the candidate variables in the \( \alpha \beta \) reference frame are constants, in addition to the optimal voltage variables being scalars, the real-time computational load for the control law is less than that in the \( dq \) reference frame.

It is worthwhile to emphasize that because the reference signals in the \( \alpha \beta \) frame are sinusoidal signals, the error signals \( i_{\alpha\alpha}(t)^* - i_{\alpha\alpha}(t) \) and \( i_{\alpha\beta}(t)^* - i_{\alpha\beta}(t) \) will not converge to zero as \( t \to \infty \). However, as sampling interval \( \Delta t \) reduces, the \( |i_{\alpha\alpha}(t)^* - i_{\alpha\alpha}(t)| \) and \( |i_{\alpha\beta}(t)^* - i_{\alpha\beta}(t)| \) will reduce as the feedback controller gain \( \frac{r_\alpha^2}{\Delta t} \) increases.

Figure 5.10 shows the configuration of the feedback control system to generate optimal control signals \( u_{\alpha\alpha}(t)^{opt} \) and \( u_{\alpha\beta}(t)^{opt} \).

The reference signals to the FCS current control in the \( \alpha \beta \) reference signals are sinusoidal signals in which their frequency is determined by the synchronous velocity of the induction motor \( \omega_s(t) \). In the applications, the desired operational performance in a closed-loop current control is specified via the desired values of \( i_{*sd} \) and \( i_{*sq} \) currents. For instance, the desired value for \( i_{*sd} \) is chosen related to the rotor flux as \( \frac{\psi_{rd}}{L_r} \) and the \( i_{*sq} \) current is related to electrical torque in demand. The reference signals to the \( i_{*sd} \) and \( i_{*sq} \) currents are transparent to the applications and easy to choose. For these reasons,
the reference signals to $i_{s\alpha}^*$ and $i_{s\beta}^*$ currents are calculated using the Park Transform:

$$
\begin{bmatrix}
  i_{s\alpha}^*(t) \\
  i_{s\beta}^*(t)
\end{bmatrix} = 
\begin{bmatrix}
  \cos \theta_s(t) & -\sin \theta_s(t) \\
  \sin \theta_s(t) & \cos \theta_s(t)
\end{bmatrix}
\begin{bmatrix}
  i_{sd}^*(t) \\
  i_{sq}^*(t)
\end{bmatrix}
$$

(5.61)

### 5.5.2 Resonant FCS-MPC Current Control Design

The experimental results of original FCS predictive control in $\alpha\beta$ frame presents the steady-state errors in the $i_{sd}$ and $i_{sq}$ currents. In order to reduce the steady-state errors in $dq$ currents, the feedback errors of $i_{s\alpha}^*(t) - i_{s\alpha}(t)$ and $i_{s\beta}^*(t) - i_{s\beta}(t)$ in the current control system need to be eliminated.

It is known from the internal model control principle that for the feedback control system to track a periodic signal, the signal generator needs to be embedded in the controller. Thus, the generator of a sinusoidal signal is embedded in the feedback control system, due to the reference current signals are sinusoidal signals, as discussed before. As a result, the output current signals $i_{s\alpha}(t)$ and $i_{s\beta}(t)$ would track their reference signals without steady-state errors. In short, the resonant controller has a polynomial factor of $1 - 2\cos(\omega_d(t))z^{-1} + z^{-2}$, which is $(1 - e^{j\omega_d(t)}z^{-1})(1 - e^{-j\omega_d(t)}z^{-1})$, where $\omega_d(t)$ is the discrete frequency of the sinusoidal reference signal. In other words, there is a pair of complex poles contained in the controller, where the locations of these poles are at...
$e^{\pm j\omega_d(t)}$ on the complex plane. Furthermore, since the frequency $\omega_d(t)$ of the sinusoidal signals is time-varying depending on the synchronous system $\omega_s(t)$, thus the controller parameters are needed to compute on-line based on the estimation of $\omega_s(t)$.

The resonant FCS current controller is proposed to have the feedback structure as illustrated in Figure 5.11. In the proposed control system structure, the feedback controllers $k_{\alpha_fcs}^\alpha$ and $k_{\beta_fcs}^\beta$ derived from the one-step-ahead prediction and optimization shown in (5.58) are used in the inner-loops for fast dynamic response, while two resonant controllers are used in the outer-loops to provide further compensations for the tracking errors between the reference and feedback current signals in the $\alpha\beta$ reference frame.

The frequency $\omega_d(t)$ is the discrete frequency, which has the unit of radian per sample interval. If the sinusoidal signal has a period of $T$ and the sampling interval is $\Delta t$, the number of samples within this period $T$ is $N_T = \frac{T}{\Delta t}$. Thus the discrete frequency $\omega_d$ is calculated as

$$\omega_d = \frac{2\pi}{N_T} = \frac{2\pi \Delta t}{T}$$

For example, if the synchronous velocity of the induction motor is $\omega_s = 200$ rad/s, and the sampling interval is $\Delta t = 20\mu s$, then the discrete frequency is

$$\omega_d = \omega_s \times \Delta t = 0.004 \text{ rad/sample}$$
The frequency parameter $\omega_d$ is time-varying when the synchronous velocity changes. However, when the sampling interval $\Delta t$ is small, this variation has a small effect on the locations of the complex poles in the controller. For instance, suppose that the induction motor velocity varies from 100 to 10000 rpm, then the synchronous velocity approximately varies from 20 to 2000 rad/s, when $\Delta t = 20 \mu s$, the corresponding $\omega_d$ to $\omega_r = 100$ rpm is approximately $0.0004$ rad/sample and $\omega_r = 10000$ rpm is $0.04$ rad/sample. The controller poles for the former case are approximately $1$ and the latter case approximately $0.9992 \pm j0.04$.

The design of the outer-loop resonant controllers are analyzed for $\alpha\beta$ frame. Firstly, from the previous section for $dq$ frame case, the inner-loop system has the closed-loop transfer function of $z^{-1}$, the same procedure is applied here to obtain the same result in $\alpha\beta$ coordinates, the derivation will not be shown to prevent the repeating work.

Thus, Figure 5.12 illustrates the outer-loop system for controlling current $i_{sa}$ with the resonant controller and the inner-loop approximated by the transfer function $z^{-1}$. Then, the closed-loop system from the reference signal $i_{sa}^*$ to $i_{sa}$ is described by the $z$-transfer function

$$T(z) = \frac{k_1 z^{-1} + k_2 z^{-2}}{1 - 2 \cos \omega_d z^{-1} + z^{-2} + k_1 z^{-1} + k_2 z^{-2}}$$

(5.62)

This is a second order discrete-time system with two closed-loop poles. Thus, the two coefficients from the resonant controller, $k_1$ and $k_2$, can be uniquely determined by using the technique of pole-assignment controller design. Assume that the desired closed-loops are identical, denoted as $0 \leq \lambda < 1$, the desired closed-loop polynomial for the discrete system is given by

$$(1 - \lambda z^{-1})^2 = 1 - 2 \lambda z^{-1} + \lambda^2 z^{-2}$$

By comparing the desired closed-loop polynomial with the actual closed-loop polynomial given by the denominator of (5.62), the following equalities are obtained:

$$k_1 - 2 \cos \omega_d = -2 \lambda$$

$$k_2 + 1 = \lambda^2$$
These equalities lead to the solutions for the gains of the resonant controller where

\[ k_1 = 2 \cos \omega_d - 2 \lambda \]  
\[ k_2 = \lambda^2 - 1 \]

The performance tuning parameter for the resonant FCS controller is the location of the pair of desired discrete closed-loop poles \( 0 \leq \lambda < 1 \). The parameter is chosen in the design to reflect the closed-loop bandwidth of the feedback control system, depending on the quality of the current model and noise level in the system. A smaller \( \lambda \) corresponds to faster closed-loop response for the resonant FCS control system, which on the other hand, it may cause noise amplification and the resulted closed-loop system less robust.

If one wishes to use the closed-loop performance specification in continuous-time that closely corresponds to the underlying physical system, then the desired closed-loop polynomial is chosen as \( s^2 + 2 \xi w_n s + w_n^2 \). For \( \xi = 0.707 \) (or other damping coefficient less than one), the pair of continuous-time complex poles are \( s_{1,2} = -\xi w_n \pm j w_n \sqrt{1 - \xi^2} \).

With a sampling interval \( \Delta t \), the pair of poles are converted from continuous-time to discrete-time via the following relationships:

\[ z_1 = e^{-\xi w_n \Delta t + j w_n \sqrt{1 - \xi^2} \Delta t} \]
\[ z_2 = e^{-\xi w_n \Delta t - j w_n \sqrt{1 - \xi^2} \Delta t} \]

The desired closed-loop polynomial in discrete-time, but having direct relation to the underlying continuous-time performance, becomes,

\[
(1 - e^{-\xi w_n \Delta t + j w_n \sqrt{1 - \xi^2} \Delta t} z^{-1})(1 - e^{-\xi w_n \Delta t - j w_n \sqrt{1 - \xi^2} \Delta t} z^{-1}) = 1 - 2e^{-\xi w_n \Delta t} \cos(w_n \sqrt{1 - \xi^2} \Delta t) z^{-1} + e^{-2\xi w_n \Delta t} z^{-2}
\]

(5.65)

When the desired closed-loop poles are selected this way, the coefficients of the resonant controller are found by comparing the desired closed-loop polynomial (5.65) with the actual closed-loop polynomial given by the denominator in (5.62):

\[ k_1 = 2 \cos \omega_d - 2 e^{-\xi w_n \Delta t} \cos(w_n \sqrt{1 - \xi^2} \Delta t) \]  
\[ k_2 = e^{-2\xi w_n \Delta t} - 1 \]

(5.66)  
(5.67)

where \( w_n \) is the desired bandwidth for the closed-loop current control system specified in the continuous-time.
5.5.3 Derivation of Resonant Controller

The control system shown in Figure 5.11 is the resonant control system without the constraints of using the finite control set in the \( \alpha \beta \) frame. The resonant control algorithm for using the finite control set is an extension of the unconstrained control case. Thus, the algorithm is firstly summarized, followed by its derivation.

**Algorithm 1** Derivation of resonant controller gain

The resonant FCS control signals in the \( \alpha \beta \) reference frame at sampling time \( t_i \), \( u_{s\alpha}(t_i) \) and \( u_{s\beta}(t_i) \), are found by finding the minimum of the objective function \( J \) with respect to the index \( k \),

\[
J = \frac{\Delta t^2}{(r_\sigma \tau_{\sigma}')^2}(u_{s\alpha}(t_i)^k - u_{s\alpha}(t_i)^{opt})^2 + \frac{\Delta f^2}{(r_\sigma \tau_{\sigma}')^2}(u_{s\beta}(t_i)^k - u_{s\beta}(t_i)^{opt})^2
\]

(5.68)

where the values of \( u_{s\alpha}(t_i)^k \) and \( u_{s\beta}(t_i)^k \) (\( k = 0, 1, 2, \ldots, 6 \)) are given by the finite control set,

\[
\begin{bmatrix}
0 & 1 & \frac{1}{2} & -\frac{1}{2} & -1 & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\
0 & 0 & \sqrt{3} & -\sqrt{3} & 0 & -\sqrt{3} & \sqrt{3} & 0
\end{bmatrix}
\frac{2}{3} V_{dc}
\]

and the signals \( u_{s\alpha}(t_i)^{opt} \) and \( u_{s\beta}(t_i)^{opt} \) are computed iteratively using the following equations:

\[
u_{s\alpha}(t_i)^{opt} = 2 \cos \omega_d u_{s\alpha}(t_i - \Delta t)^{opt} - u_{s\alpha}(t_i - 2\Delta t)^{opt} + u_{s\alpha}^{opt}(t_i)
\]

(5.69)

\[
u_{s\alpha}^{opt}(t_i) = k_{fcs}^\alpha [k_1(i_{s\alpha}(t_i)^* - i_{s\alpha}(t_i)) + k_2(i_{s\alpha}(t_i - \Delta t)^* - i_{s\alpha}(t_i - \Delta t))
- (i_{s\alpha}(t_i) - 2 \cos \omega_d i_{s\alpha}(t_i - \Delta t) + i_{s\alpha}(t_i - 2 \Delta t))]
\]

(5.70)

\[
u_{s\beta}(t_i)^{opt} = 2 \cos \omega_d u_{s\beta}(t_i - \Delta t)^{opt} - u_{s\beta}(t_i - 2\Delta t)^{opt} + u_{s\beta}^{opt}(t_i)
\]

(5.71)

\[
u_{s\beta}^{opt}(t_i) = k_{fcs}^\beta [k_1(i_{s\beta}(t_i)^* - i_{s\beta}(t_i)) + k_2(i_{s\beta}(t_i - \Delta t)^* - i_{s\beta}(t_i - \Delta t))
- (i_{s\beta}(t_i) - 2 \cos \omega_d i_{s\beta}(t_i - \Delta t) + i_{s\beta}(t_i - 2 \Delta t))]
\]

(5.72)

The feedback control gains used in the computation are defined as:

\[
k_{fcs}^\alpha = k_{fcs}^\beta = \frac{r_\sigma \tau_{\sigma}'}{\Delta t} (1 - \frac{\Delta t}{\tau_{\sigma}})
\]

(5.73)

\[
k_1 = 2 \cos \omega_d - 2 \lambda
\]

(5.74)

\[
k_2 = \lambda^2 - 1
\]

(5.75)

\( 0 \leq \lambda < 1 \) is the desired closed-loop pole location for the current control system.

To derive how the control signals are chosen in the presence of constraints, the predictive control solution is resorted. Consider the difference equations that describe the currents.
Chapter 5. Finite Control Set - Model Predictive Control

\(i_{sa}\) and \(i_{sb}\) at the sampling time \(t_i\):

\[
i_{sa}(t_i + \Delta t) = (1 - \frac{\Delta t}{r_\sigma}) i_{sa}(t_i) + \frac{\Delta t}{r_\sigma \tau_\sigma} u_{sa}(t_i) + \Delta t(-\frac{k_r}{r_\sigma \tau_\sigma} \omega_c(t_i) \psi_{\alpha\alpha}(t_i) + \frac{k_r}{r_\sigma \tau_\sigma} \omega_c(t_i) \psi_{\alpha\beta}(t_i)) \tag{5.76}
\]

\[
i_{sb}(t_i + \Delta t) = (1 - \frac{\Delta t}{r_\sigma}) i_{sb}(t_i) + \frac{\Delta t}{r_\sigma \tau_\sigma} u_{sb}(t_i) + \Delta t(-\frac{k_r}{r_\sigma \tau_\sigma} \omega_c(t_i) \psi_{\beta\alpha}(t_i) + \frac{k_r}{r_\sigma \tau_\sigma} \omega_c(t_i) \psi_{\beta\beta}(t_i)) \tag{5.77}
\]

Because there is no interaction between the variables in the \(\alpha\beta\) coordinates, for simplicity, the prediction for \(i_{sa}\) is considered only, then the results will be naturally extended for the variable \(i_{sb}\). Define the operator \(D(z^{-1})\) as

\[
D(z^{-1}) = 1 - 2cos\omega_d z^{-1} + z^{-2}
\]

with \(z^{-1}\) as the backward shift operator \(z^{-1} f(t_i) = f(t_i - \Delta t)\). By applying the operator \(D(z^{-1})\) to both sides of (5.76), leads to

\[
i_{sa}(t_i + \Delta t)^s = (1 - \frac{\Delta t}{r_\sigma}) i_{sa}(t_i)^s + \frac{\Delta t}{r_\sigma \tau_\sigma} u_{sa}(t_i)^s \tag{5.78}
\]

where

\[
i_{sa}(t_i + \Delta t)^s = D(z^{-1}) i_{sa}(t_i + \Delta t) \tag{5.79}
\]

\[
i_{sa}(t_i)^s = D(z^{-1}) i_{sa}(t_i) \tag{5.80}
\]

\[
u_{sa}(t_i)^s = D(z^{-1}) u_{sa}(t_i) \tag{5.81}
\]

Note that the rotor flux variables \(\psi_{\alpha\alpha}(t_i)\) and \(\psi_{\beta\beta}(t_i)\) are both sinusoidal signals, when the operator \(D(z^{-1})\) is applied, these terms of (5.76) will vanish.

To include the resonant action into the controller, the weighted current errors \(e_\alpha(t_i) = k_1 (i_{sa}(t_i)^s - i_{sa}(t_i)) + k_2 (i_{sa}(t_i - \Delta t)^s - i_{sa}(t_i - \Delta t))\) is chosen as the steady-state of the \(i_{sa}(t_i)^s\), where \(k_1\) and \(k_2\) are given by (5.75) and (5.75). The steady-state of \(u_{sa}(t_i)^s\) is chosen to be zero. Subtracting the steady-state from the model (5.78) gives:

\[
i_{sa}(t_i + \Delta t)^s - e_\alpha(t_i) = (1 - \frac{\Delta t}{r_\sigma \tau_\sigma}) (i_{sa}(t_i)^s - e_\alpha(t_i)) + \frac{\Delta t}{r_\sigma \tau_\sigma} u_{sa}(t_i)^s \tag{5.82}
\]

After applying the same procedure to the \(\beta\)-axis current, we obtain the formulation for the \(i_{sb}\) variable as

\[
i_{sb}(t_i + \Delta t)^s - e_\beta(t_i) = (1 - \frac{\Delta t}{r_\sigma \tau_\sigma}) (i_{sb}(t_i)^s - e_\beta(t_i)) + \frac{\Delta t}{r_\sigma \tau_\sigma} u_{sb}(t_i)^s \tag{5.83}
\]
where \( e_\beta(t_i) = k_1(i_{s\beta}(t_i)^* - i_{s\beta}(t_i)) + k_2(i_{s\beta}(t_i - \Delta t)^* - i_{s\beta}(t_i - \Delta t)) \). The control objective is to minimize the error function \( J \), where

\[
J = \left[ \begin{array}{cc}
  i_{sa}(t_i + \Delta t)^s - e_\alpha(t_i) & i_{sb}(t_i + \Delta t)^s - e_\beta(t_i)
\end{array} \right] \left[ \begin{array}{c}
i_{sa}(t_i + \Delta t)^s - e_\alpha(t_i) \\
i_{sb}(t_i + \Delta t)^s - e_\beta(t_i)
\end{array} \right]
\]

(5.84)

which is to regulate the filtered current signals \( i_{sa}(t_i + \Delta t)^s \), \( i_{sb}(t_i + \Delta t)^s \) to be as close as possible to \( e_\alpha(t_i) \) and \( e_\beta(t_i) \).

By substituting (5.82) and (5.83) into (5.84), and following the same derivation procedure as outlined previously, the optimal solutions of \( u_{sa}(t_i)^s \) and \( u_{sb}(t_i)^s \) that will minimize the objective function (5.84) are obtained as

\[
\begin{align*}
u_{sa}(t_i)^{opt} &= k_{fcs}^{\alpha}(e_\alpha(t_i) - i_{sa}(t_i)^s) \\
u_{sb}(t_i)^{opt} &= k_{fcs}^{\beta}(e_\beta(t_i) - i_{sb}(t_i)^s)
\end{align*}
\]

where the feedback controller gains are defined as

\[
k_{fcs}^{\alpha} = k_{fcs}^{\beta} = \frac{r_\sigma \tau_\sigma'}{\Delta t(1 - \frac{\Delta t}{\tau_\sigma})}
\]

Furthermore, the objective function \( J \) can be expressed via completing squares as

\[
J = \frac{\Delta t^2}{(r_\sigma \tau_\sigma')^2}(u_{sa}(t_i)^s - u_{sa}(t_i)^{opt})^2 + \frac{\Delta t^2}{(r_\sigma \tau_\sigma')^2}(u_{sb}(t_i)^s - u_{sb}(t_i)^{opt})^2
\]

(5.85)

Note that \( u_{sa}(t_i)^s \), \( u_{sa}(t_i)^{opt} \), \( u_{sb}(t_i)^s \), \( u_{sb}(t_i)^{opt} \) are filtered voltage control variables.

Thus, based on the definition of the filtered control signals, the following relationships are obtained:

\[
\begin{align*}
u_{sa}(t_i)^{opt} &= \frac{u_{sa}(t_i)^{opt}}{1 - 2\cos\omega_d z^{-1} + z^{-2}} \\
u_{sb}(t_i)^{opt} &= \frac{u_{sb}(t_i)^{opt}}{1 - 2\cos\omega_d z^{-1} + z^{-2}}
\end{align*}
\]

which leads to the expressions of \( u_{sa}(t_i)^{opt} \) and \( u_{sb}(t_i)^{opt} \) in an iterative manner:

\[
\begin{align*}
u_{sa}(t_i)^{opt} &= u_{sa}(t_i)^{opt} - 2\cos\omega_d u_{sa}(t_i - \Delta t)^{opt} + u_{sa}(t_i - 2\Delta t)^{opt} \\
u_{sb}(t_i)^{opt} &= u_{sb}(t_i)^{opt} - 2\cos\omega_d u_{sb}(t_i - \Delta t)^{opt} + u_{sb}(t_i - 2\Delta t)^{opt}
\end{align*}
\]

(5.86)  

(5.87)
By calculating the actual filtered control signals using the same past optimal control signal states, it leads to

\[
\begin{align*}
    u_{s\alpha}(t_i)^s &= u_{s\alpha}(t_i) - 2\cos\omega_d u_{s\alpha}(t_i - \Delta t)^{opt} + u_{s\alpha}(t_i - 2\Delta t)^{opt} \quad (5.88) \\
    u_{s\beta}(t_i)^s &= u_{s\beta}(t_i) - 2\cos\omega_d u_{s\beta}(t_i - \Delta t)^{opt} + u_{s\beta}(t_i - 2\Delta t)^{opt} \quad (5.89)
\end{align*}
\]

By substituting the filtered variables (5.86)-(5.89) into the objective function (5.85), it becomes

\[
J = \frac{\Delta t^2}{\sigma r^2} (u_{s\alpha}(t_i) - u_{s\alpha}(t_i)^{opt})^2 + \frac{\Delta t^2}{\sigma r^2} (u_{s\beta}(t_i) - u_{s\beta}(t_i)^{opt})^2 \quad (5.90)
\]

which is identical to the one used in the Algorithm 1. This completes the derivation of the resonant FCS control algorithm. It is emphasized that the resonant FCS control Algorithm 1 used the past optimal control signal states \((u_{s\alpha}(t_i - \Delta t)^{opt}, u_{s\alpha}(t_i - 2\Delta t)^{opt})\) together with the present \((u_{s\alpha}(t_i))\) to predict the filtered \(u_{s\alpha}(t_i)^s\). If the past implemented control signal states \((u_{s\alpha}(t_i - \Delta t), u_{s\alpha}(t_i - 2\Delta t))\) were used in the prediction, then it could result in accumulated errors from the finite control set and lead to steady-state errors in the resonant FCS control system.

5.5.4 Experimental Evaluations

In this section, two sets of experimental evaluation results are presented. One is the original FCS predictive control of the induction motor drive while the other is the resonant FCS predictive control. Both evaluations use the same motor parameters and the same sampling interval \(\Delta t = 80\mu s\). With the given motor parameters and sampling interval, the proportional controller gains are calculated as

\[
k_{fcs}^\alpha = k_{fcs}^\beta = 1310.3
\]

The reference signal to the \(d\)-axis stator current \(i_{sd}^*\) is 0.8772A and the reference signal to the \(q\)-axis stator current \(i_{sq}^*\) is 0A for an initial period and a step change to 1.5A. The load disturbance is dependent to the real-time motor shaft velocity.

<table>
<thead>
<tr>
<th></th>
<th>mean of SSE (i_{s\alpha})</th>
<th>mean of SSE (i_{s\beta})</th>
<th>Steady-state speed</th>
<th>Response time</th>
</tr>
</thead>
<tbody>
<tr>
<td>FCS</td>
<td>-0.0015A</td>
<td>-0.0013A</td>
<td>522RPM</td>
<td>0.0003s</td>
</tr>
<tr>
<td>FCS-Res</td>
<td>(-1.2358 \times 10^{-4})A</td>
<td>(-2.2802 \times 10^{-4})A</td>
<td>548RPM</td>
<td>0.0006s</td>
</tr>
</tbody>
</table>

Table 5.4: Comparison of proposed methods in \(\alpha/\beta\) frame.

**Results from the original FCS predictive current control.** The experimental results of the original FCS predictive control in \(\alpha/\beta\) reference frame are shown in Figure
5.13(a)-5.13(c). In Figure 5.13(a), the stator currents, $i_{s\alpha}$ and $i_{s\beta}$, are shown to track the sinusoidal reference signals and these signals are converted to their corresponding current signals in $dq$ reference frame as seen in Figure 5.13(b). The FFT analysis of the phase current is performed in Figure 5.13(c). With respect to this set of physical parameters and the selection of sampling interval $\Delta t$, the tracking errors in $\alpha\beta$ coordinates are very small, not obvious from reading of the graphs. Instead, the calculation
results are presented in Table 5.4. One may conclude that at this test environment, the original FCS predictive current control system produces a relatively small steady-state error. However, further tests need to be conducted when the physical parameters are varying and a larger load is added, because with parameter uncertainties, the closed-loop performance will be affected.

![Comparison of open-loop response of the motor speed.](image)

**Figure 5.15:** Comparison of open-loop response of the motor speed.

**Results from the resonant FCS predictive current control.** The experimental results of the resonant FCS predictive control system are shown in Figure 5.14(a)-5.14(c). In Figure 5.14(a), it is seen that the stator currents in $\alpha\beta$ coordinates track their sinusoidal current reference signals, and in Figure 5.14(b), the corresponding current signals are shown in $dq$ frame. Figure 5.14(c) shows the magnitude of the frequency response of the phase current measurement. All three figures confirm that the resonant FCS current control system provides a satisfactory closed-loop performance. The mean errors of the steady-state response are calculated and shown in Table 5.4, to compare with the original FCS predictive current control, the reduction of the steady-state errors is obtained. Moreover, by comparing the open-loop velocity responses as shown in Figure 5.15, it is seen that the two current control schemes result in different motor velocity responses, particularly in the steady-state. For the original FCS current controller, the steady-state speed is about 522 RPM and for the resonant FCS current controller, the steady-state speed is about 548 RPM, which is a higher speed given all other physical conditions being identical. This could mean that by regulating the steady-state current responses closer to the desired reference signals, the induction motor has an improved efficiency in operation.
5.6 Summary

In this chapter, the Finite Control Set predictive current control of the induction motor drive has been discussed in both $dq$ and $\alpha\beta$ reference frames. The revised approaches are proposed and implemented to eliminate the steady-state errors, which appear in the original FCS current responses.

5.6.1 Summary of $dq$-frame FCS-MPC

In the former part of this chapter, the I-FCS predictive current control system was proposed in order to vanish the steady-state error, which appears in the original FCS-MPC control system.

In the traditional FCS-MPC, the objective function is generally chosen with sum of absolute errors for $dq$ stator currents. Obviously, this kind of objective functions is conceptually simple, however, it does not readily yield an analytical optimal solution. Instead, this chapter has seen that with a sum of squared errors, the objective function for the traditional FCS predictive controller has a simple analytical solution, leading to a closed-loop feedback control system that is equivalent to an optimal deadbeat control system.

The original FCS predictive controllers do not contain integrator in the design. One of the key problems is the steady-state performance of the current control systems that is not guaranteed error free in its mean value. By considering the dynamics of the original FCS predictive control system as unit sample delayed system, an integral control term is embedded to the FCS-MPC via a cascaded structure. Thus, the closed-loop pole is selected by defining the integral gain in a simplicity fashion where $k_I = 1 - p_{cl}$, where $p_{cl}$ is the desired discrete-time closed-loop pole.

Therefore, the design parameter of the I-FCS predictive control systems are the sampling interval $\Delta t$ and the location of the closed-loop pole. In general, the smaller $\Delta t$ value will optimal the control performance, but the trade-off of a smaller sampling interval $\Delta t$ leads to larger computational load and more power losses due the faster switching of the voltage inverter. On the other hand, the selection of integral gain is straightforward. A larger integral gain will lead to a faster closed-loop current response. Normally, in the given experiment test-bed, the integral gain is recommended to be between 0.05 to 0.25.

5.6.2 Summary of $\alpha\beta$-frame FCS-MPC

The resonant FCS predictive current control in $\alpha\beta$ coordinates has been presented with the design and implementation. It is shown that the traditional FCS predictive current
controller that deploys a cost function using sum of squared errors is essentially a high gain proportional controller in the absence of constraints. Thus, in the steady-state operation, the current outputs can not entirely follow the sinusoidal current reference signals.

By taking a similar approach as proposed for $dq$ reference frame, a resonant controller is designed in a cascaded structure to track the sinusoidal reference signal without steady-state errors. In the design of the resonate controller, the original FCS predictive control system is modelled by unit sample of delay. Since the $\alpha\beta$ model of induction motor has no interactions between $\alpha$ and $\beta$ variables, the FCS controller is decoupled into two SISO controllers, leading to a reduced computational load and a simpler controller structure. Furthermore, the Clarke transformation that converts the variables in the $dq$ frame to the $\alpha\beta$ frame is performed to generate the sinusoidal reference signals for the current control, however, the nonlinear transformation occurs outside the current feedback loop. The closed-loop performance of the resonant FCS predictive control system is also dependent to the choice of the sampling rate, similar to the previous $dq$ frame case. In general, a smaller $\Delta t$ will lead to a smaller current variations. The selection of the desired closed-loop poles for the resonate controller follows the specification on the desired closed-loop current responses.
Chapter 6

PID Control Strategy

6.1 Introduction

The main purpose of this chapter is to demonstrate the PID-based induction motor control, in order to compare with the proposed methods in the next chapter. Almost all induction motor drives in current industrial applications are controlled by PID controllers. The controller structures, such as Field Oriented Control (FOC), are already defined in the commercial drives. However, the controller parameters tuning can be investigated for performance improvement. From chapter 2, the mathematical model of an induction motor has more than one input and output signal in addition to being nonlinear systems. Since PID controller is designed for Single-Input-Single-Output (SISO) and Linear-Time Invariant (LTI) system, the architecture of the feedback control system should be specially considered depending on the actual applications. Because the FOC control structure is applied in this thesis, the number of PID controllers will be determined based on the control application.

In this chapter, the torque control, speed control and position control of induction motor are analysed, respectively, for each control application, the design, implementation and tuning procedures are discussed in its respective section.

6.2 Torque Control

Torque control, in the other word, current control of AC machine drive has many methods in the literatures, such as FOC, DTC and recently FCS, for the induction motor control, Direct Torque Control (DTC) is commonly used in industrial application. This thesis is focus on the vector control method, thus the torque control is achieved by using PI-based Indirect FOC technique in this section.
In Figure 6.1, the layout of torque control in this section is illustrated, the rotor flux is controlled by open-loop due to indirect vector control structure. The strategy to control the electromagnetic torque of an induction motor is based on PI control of the stator currents $i_{sd}(t)$ and $i_{sq}(t)$. The first step in the control system design is to determine the reference signals to the control loops.

**Reference signal to stator current $i_{sq}^\ast$.** Given a desired electromagnetic torque $T_e^\ast$ and a desired rotor flux amplitude $\psi_{rd}^\ast$, the desired stator current in the $q$-axis is calculated as

$$
  i_{sq}^\ast = \frac{2 L_r}{3 L_h Z_p} \frac{T_e^\ast}{\psi_{rd}^\ast}
$$

(6.1)
on the basis of the relationship given in (6.1). The desired stator current $i_{sq}^\ast$ is then used as the reference signal to PI control of the stator current.

**Reference signal to stator current $i_{sd}^\ast$.** From the model equation (2.23), the dynamic response of $\psi_{rd}$ from $i_{sd}$ is a first order system with time constant $\tau_r$. The differential equation (2.23) can be expressed in terms of the set-point values $\psi_{rd}^\ast$ and $i_{sd}^\ast$:

$$
  i_{sd}^\ast(t) = \frac{1}{L_h} \psi_{rd}^\ast(t) + \frac{\tau_r}{L_h} \frac{d\psi_{rd}^\ast(t)}{dt}
$$

(6.2)

In general, the set-point for rotor flux $\psi_{rd}^\ast$ is defined as constant. Therefore, the derivative of the reference rotor flux signal, $\frac{d\psi_{rd}^\ast(t)}{dt}$, is taken to be zero. The reference signal to $i_{sd}$ current is determine via the steady-state relationship:

$$
  i_{sd}^\ast = \frac{1}{L_h} \psi_{rd}^\ast
$$
where $L_h$ is the mutual machine inductance. However, if field-weakening is involved, the reference signal to rotor flux $\psi_{rd}$ may change. If the trajectory the $\psi^*_{rd}$ is chosen to be a combination of constant signals, interpreted with ramp signals between the transient periods, then the reference signal $i^*_{sd}$ is calculated using (6.2).

### 6.2.1 Linearization

The linearization in this chapter is achieved using feed-forward manipulation, in order to decouple the non-linear cross-coupling terms inside current model equations (2.21) and (2.22), the main idea is to firstly define the auxiliary variables $\hat{u}_{sd}$ and $\hat{u}_{sq}$ as follows

$$
\frac{1}{r_\sigma \tau_{\sigma}'} \hat{u}_{sd} = \omega_s i_{sq} + \frac{k_r}{r_\sigma \tau_{\sigma}'} \psi_{rd} + \frac{1}{r_\sigma \tau_{\sigma}'} u_{sd} \quad (6.3)
$$

$$
\frac{1}{r_\sigma \tau_{\sigma}'} \hat{u}_{sq} = -\omega_s i_{sd} - \frac{k_r}{r_\sigma \tau_{\sigma}'} \omega_e \psi_{rd} + \frac{1}{r_\sigma \tau_{\sigma}'} u_{sq} \quad (6.4)
$$

By substituting these equations into the original current model equations (2.21) and (2.22), two first-order linear model equations are obtained

$$
\frac{d}{dt} i_{sd} = -\frac{1}{\tau_{\sigma}'} i_{sd} + \frac{1}{r_\sigma \tau_{\sigma}'} \hat{u}_{sd} \quad (6.5)
$$

$$
\frac{d}{dt} i_{sq} = -\frac{1}{\tau_{\sigma}'} i_{sq} + \frac{1}{r_\sigma \tau_{\sigma}'} \hat{u}_{sq} \quad (6.6)
$$

Based on above equations (6.5) and (6.6), two PI controllers are designed for the stator current control by manipulating the auxiliary stator voltage in $dq$ frame. Therefore, the evaluation of the auxiliary variables are implemented in terms of PI controller parameters: proportional gain $K_c$ and integral time constant $\tau_I$.

$$
\hat{u}_{sd} = K_c^d (i^*_{sd}(t) - i_{sd}(t)) + \frac{K_c^d}{\tau_I} \int_0^t (i^*_{sd}(\tau) - i_{sd}(\tau)) d\tau \quad (6.7)
$$

$$
\hat{u}_{sq} = K_c^q (i^*_{sq}(t) - i_{sq}(t)) + \frac{K_c^q}{\tau_I} \int_0^t (i^*_{sq}(\tau) - i_{sq}(\tau)) d\tau \quad (6.8)
$$

where the super-scripts $(.)^d$ and $(.)^q$ of controller parameters are presented according to its own axis, respectively. By substituting equations (6.3) and (6.4) into above implementation, the actual control signals $u_{sd}$ and $u_{sq}$ are computed as follows

$$
u_{sd} = K_c^d (i^*_{sd}(t) - i_{sd}(t)) + \frac{K_c^d}{\tau_I} \int_0^t (i^*_d(\tau) - i_{sd}(\tau)) d\tau - r_\sigma \tau_{\sigma}' \omega_s(t) i_{sq}(t) - \frac{k_r}{\tau_r} \psi_{rd}(t)
$$

$$
u_{sq} = K_c^q (i^*_{sq}(t) - i_{sq}(t)) + \frac{K_c^q}{\tau_I} \int_0^t (i^*_q(\tau) - i_{sq}(\tau)) d\tau + r_\sigma \tau_{\sigma}' \omega_s(t) i_{sd}(t) + k_r \omega_e(t) \psi_{rd}(t)$$
where the nonlinear compensation terms are implemented using the feedback measurements or estimation based on actual measurement.

### 6.2.2 PI Controllers Design

The PI controllers are designed based on the transfer functions for the electrical system, which are obtained based on equations (6.5) and (6.6)

\[
\frac{I_{sd}(s)}{\hat{V}_{sd}(s)} = \frac{1}{r_{\sigma}\tau'_{\sigma}} \left( s + \frac{1}{\tau'_{\sigma}} \right)
\]

(6.9)

\[
\frac{I_{sq}(s)}{\hat{V}_{sq}(s)} = \frac{1}{r_{\sigma}\tau'_{\sigma}} \left( s + \frac{1}{\tau'_{\sigma}} \right)
\]

(6.10)

where \(I_{sd}(s)\) and \(I_{sq}(s)\) are the Laplace transforms of \(dq\) stator currents, \(\hat{V}_{sd}(s)\) and \(\hat{V}_{sq}(s)\) present the Laplace transforms of the intermittent voltage variables.

In the controller design, the proportional gain \(K_{dc}\) (or \(K_{qc}\)) and the integral time constant \(\tau_{di}\) (or \(\tau_{qi}\)) are determined using pole-assignment method. Since the identical plant transfer functions are obtained above, theoretically the controller parameters designed should be identical as well, as a result \(K_{ic}\) represents the value of \(K_{dc}\) and \(K_{qi}\), \(\tau_{ii}\) represents the value of \(\tau_{di}\) and \(\tau_{qi}\), the super-script (\(\cdot\)\(^{'i}\)) denotes the inner-loop current control to be distinct from the outer-loop controller parameters.

![Figure 6.2: Closed-loop control of \(i_{sd}\) stator current](image)

The transfer function of the PI controllers for inner current loop has the following form:

\[
C(s) = K_{ci}^i\left(1 + \frac{1}{\tau_{is}^i}s\right)
\]

(6.11)

The actual closed-loop characteristic polynomial is the denominator of the closed-loop transfer function that contains the first order system transfer function (6.9) or (6.10) and the PI controller (6.11), where the closed-loop system, with \(d\)-axis as example, is shown in Figure 6.2.

\[
D(s) = s(s + \frac{1}{\tau'_{\sigma}}) + \frac{1}{r_{\sigma}\tau'_{\sigma}} \frac{K_{ci}^i}{\tau_{fi}^i} (\tau_{is}^i s + 1)
\]

(6.12)

Since the pole-assignment strategy is applied, the desired closed-loop performance polynomial is specified by choosing a damping coefficient \(\xi\) (= 0.707 typically) and a natural
frequency $\omega_n$. With these choices, the desired closed-loop characteristic polynomial becomes

$$A_{i}^{cl}(s) = s^2 + 2\xi \omega_n s + \omega_n^2$$  \hspace{1cm} (6.13)

By equating equation (6.13) with (6.12), $A_{i}^{cl}(s) = D(s)$, and comparing their coefficients, the PI controller parameters for the inner-loop current control are calculated as:

$$K_i = 2\xi \omega_n r \tau' - r$$  \hspace{1cm} (6.14)

$$\tau_I = \frac{2\xi \omega_n r \tau' - r}{\omega_n^2 r \tau' - \tau}$$  \hspace{1cm} (6.15)

In the design, the damping coefficient $\xi$ is selected to be 0.707 and the natural frequency $\omega_n$ is selected to determine the desired closed-loop response speed, which also corresponds to the desired bandwidth of the closed-loop system, the inner-loop closed-loop poles are located at $-\xi \omega_n \pm \omega_n j\sqrt{1-\xi^2}$. In this thesis, $\omega_n$ is chosen relative to the bandwidth of the open-loop system, which is $\frac{1}{\tau'}$, thus a normalized parameter $0 \leq \gamma_i \leq 1$ is proposed to determine the desired closed-loop bandwidth as

$$\omega_n = \frac{1}{1-\gamma_i} \frac{1}{\tau'}$$  \hspace{1cm} (6.16)

where $\gamma_i$ is often selected to be close to 1, from 0.7 to 0.9, to give a satisfactory performance, more details will be discussed in later tuning section.

### 6.2.3 Implementation

This section contains the feedforward manipulation, operational constraints design, discretization of current controllers and results presentation.

From the previous section, the nonlinear compensation functions for the current model are shown as

$$f_d(t) = -r \tau'_w \omega_s(t)i_{dq}(t) - \frac{k}{\tau} \psi_{rd}(t)$$  \hspace{1cm} (6.17)

$$f_q(t) = r \tau'_w \omega_s(t)i_{sd}(t) + k \omega_c(t) \psi_{rd}(t)$$  \hspace{1cm} (6.18)

where $f_d(t)$ and $f_q(t)$ present the feedforward functions for $d$-axis and $q$-axis current control, respectively. Note that these compensators require the rotor flux $\psi_{rd}(t)$ and the synchronous angular velocity $\omega_s(t)$, which are not measurable directly by sensors, thus estimation schemes are needed in order to access the relevant information. The estimation of $\omega_s(t)$ is achieved using the slip estimation as shown in equation (4.11), however directly applying the equation will not lead to an accurate estimate of $\omega_s(t)$ due to the switching and measurement noise appeared on current feedbacks. Instead,
Chapter 6. PID Control Strategy

the reference signals $i^*_{sd}(t)$ and $i^*_{sq}(t)$ are used to obtain following estimation of $\omega_s(t)$ at the sampling time $t_i$

$$\dot{\omega}_s(t_i) = \omega_e(t_i) + \frac{i^*_{sq}(t_i)}{r_r i^*_{sd}(t_i)}$$

where integrators are used to control the currents, at the steady-state, $i_{sd}(t) = i^*_{sd}(t)$ and $i_{sq}(t) = i^*_{sq}(t)$. Based on the similar reason, the rotor flux $\psi_{rd}(t)$ in the feedforward function is chosen to be the set-point signal $\psi^*_{rd}(t)$, thus the implementations of nonlinear compensation for $d$-axis and $q$-axis are shown in Figure 6.3 and 6.4, respectively.

![Figure 6.3: Nonlinear compensation for d-axis current controller](image)

![Figure 6.4: Nonlinear compensation for q-axis current controller](image)

The operational constraints of the control signals $u_{sd}$ and $u_{sq}$ are designed due to inverter operation, where the voltages for the induction motor drive are limited by the voltages of power supply. Suppose that the DC-bus voltage is supplied with $V_{dc}$ Volt, with the modulation limitation, the control signal voltages are restricted due to the functionality of the inverter with the following relationship

$$\sqrt{u_{sd}^2 + u_{sq}^2} \leq \frac{V_{dc}}{\sqrt{3}}$$

which is a quadratic function of both $u_{sd}$ and $u_{sq}$, for the PI control design, the linear approximation is proposed to obtained the operational limits of $u_{sd}$ and $u_{sq}$. The method is to approximate the circular area with a rectangular area, which is defining a parameter $0 \leq \varepsilon \leq 1$, thus the limit values of $u_{sd}$ and $u_{sq}$ are set to

$$u_{sd}^{max} = \varepsilon \frac{V_{dc}}{\sqrt{3}}; \quad u_{sq}^{max} = \sqrt{1 - \varepsilon^2} \frac{V_{dc}}{\sqrt{3}}$$
where for instance, if \( \varepsilon = 0.6 \), then \( u_{sd}^{\text{max}} = 0.6 \frac{V_{dc}}{\sqrt{3}} \) and \( u_{sq}^{\text{max}} = 0.8 \frac{V_{dc}}{\sqrt{3}} \). Thus, the constraints implementation is specified as

\[
-u_{sd}^{\text{max}} \leq u_{sd}(t) \leq u_{sd}^{\text{max}} \\
-u_{sq}^{\text{max}} \leq u_{sq}(t) \leq u_{sq}^{\text{max}}
\]

The advantage of proposed method is that the constant constraints are obtained once the DC power supply voltage \( V_{dc} \) is determined, the constant constraints are simple to implement in a real-time control system. However, the shortness is its conservativeness due to the rectangular area being much smaller than the entire circular area.

The discretization of current controllers describes the implementation of the continuous-time controllers in digital environment. From previous section, the control signal computation of PI controller in general expression is defined as

\[
u(t) = K_i^c e(t) + \frac{K_i^c}{\tau_i^c} \int_0^t e(\tau) d\tau + f(t) \quad (6.19)
\]

where \( e(t) \) is the difference between the set-point and the actual feedback, \( f(t) \) presents the feedforward manipulation. When at the sampling instant \( t_i \), the above control signal is discretized as

\[
u(t_i) = K_i^c e(t_i) + \frac{K_i^c}{\tau_i^c} \sum_{k=0}^{M-1} e(t_k) \Delta t + f(t_i) \quad (6.20)
\]

where \( M \) is the number of samples. However, the control signal computed in (6.20) is meant for the deviation variable, which is not the actual control signal manipulated for the input to the physical system, a bias term, which is denoted as \( u_{ss} \), corresponds to the steady-state of the control signal, which needs to be considered in the implementation as shown

\[
u_{act}(t_i) = u(t_i) + u_{ss}
\]

where the steady-state information of the control signal is required before the controller design, which is one of the main drawbacks for this so-called position form implementation. Furthermore, another drawback is when the integrator wind-up situation is reached, the implementation of the anti-windup mechanism is not as straightforward as the velocity form implementation, which is introduced below.

The derivative of the control signal equation (6.19) is expressed for the velocity form implementation

\[
\frac{du(t)}{dt} = K_i^c \frac{de(t)}{dt} + \frac{K_i^c}{\tau_i^c} e(t) + \frac{f(t)}{dt} \quad (6.21)
\]
By applying the first-order approximation \( \frac{dx(t)}{dt} \approx \frac{x(t_i) - x(t_i - \Delta t)}{\Delta t} \), the approximation of above equation lead to

\[
 u(t_i) = u(t_i - \Delta t) + K_i^i (e(t_i) - e(t_i - \Delta t)) + \frac{K_i^i}{\tau_i^i} e(t_i) \Delta t + f(t_i) - f(t_i - \Delta t)
\]

By adding the bias term \( u_{ss} \) to both side of the above equation, the computational equation of the actual control signal using velocity form is obtained

\[
 u_{act}(t_i) = u_{act}(t_i - \Delta t) + K_i^i (e(t_i) - e(t_i - \Delta t)) + \frac{K_i^i}{\tau_i^i} e(t_i) \Delta t + f(t_i) - f(t_i - \Delta t)
\]

where only the initial information of the variables \( (u_{act}(0), e(0), f(0)) \) are required, then the present sample of the control signal is computed based on the previous sample information.

From the previous sections, the operational constraints are designed and the anti-windup mechanisms are mentioned when the saturation reached for a PI controller. This section demonstrates the implementation of saturation with anti-windup in velocity form, which is much straightforward. Assuming that the actual control variable is limited by \( U_{min} \) and \( U_{max} \). Namely, the actual signal must satisfy the following constraints:

\[
 U_{min} \leq u_{act}(t) \leq U_{max}
\]

When the saturation limit is violated, the derivative of the control signal is supposed to be zero, i.e. \( \frac{du_{act}(t)}{dt} = 0 \), leading to \( u_{act}(t_i) = u_{act}(t_i - \Delta t) \). Based on the implementation equation (6.22), in order to stop the integration, when the actual control signal reaches the limit, the following computation is expressed

\[
 u_{act}(t_i) = U_{min} \quad \text{if} \quad u_{act}(t_i) < U_{min}
\]

\[
 u_{act}(t_i) = U_{max} \quad \text{if} \quad u_{act}(t_i) > U_{max}
\]

When the sample time \( t_i \) moves one step forward, the \( u_{act}(t_i - \Delta t) \) carries the information of saturation at the previous sample time and the control signal computation is automatically informed of the saturation. Both requirements in an anti-windup mechanism are satisfied.

**Simulation evaluation.** From the previous implementation, the simulation result is obtained based on following setup: 520 V at the DC-link bus voltage supply, PWM carrier switching frequency is 2kHz, control system sampling interval is 100\( \mu \)s, the Simulink SimPower model is sampled at 10\( \mu \)s. The reference signal of the rotor flux \( \psi_{rd}^* = 0.6 \text{Wb} \) within the rated speed region, the set-point of torque dependant current \( i_{sq} \) is defined with a step change, The simulation time duration is set at 1 second.
6.2.4 Tuning of Controllers

The tuning process is proposed to seek better control performance by studying the control design, this task is often performed against the actual systems, because there are modelling errors, noises, parameter variations, and other undesired factors involved inside the model.

In order to demonstrate the frequency response of the disturbance and noise, the sensitivity transfer functions are introduced. Figure 6.6 presents the basic structure of one degree of freedom SISO control system. Based on the relationships, the following sensitivity functions are defined:

\[
S(s) = \frac{1}{1 + G(s)C(s)}
\]

\[
T(s) = \frac{G(s)C(s)}{1 + G(s)C(s)}
\]

\[
S_i(s) = \frac{G(s)}{1 + G(s)C(s)}
\]

\[
S_u(s) = \frac{C(s)}{1 + G(s)C(s)}
\]

**Figure 6.5:** Simulation results of torque control using PI controllers. Key: line(1) Actual feedback measurement; line (2) Set-point signal

(a) Set-point following of stator currents

(b) Control signals

**Figure 6.6:** One-degree of freedom control system structure
where $G(s)$ presents the plant transfer function as (6.9) or (6.10) and $C(s)$ is the transfer function of the PI controllers as (6.11). From above transfer functions, $T(s)$ represents the effect of both reference signal and measurement noise on the output. Thus when a wide closed-loop bandwidth is designed, the response of the set-point tracking is faster but the measurement noise will be amplified as well, which is the trade-off in control design.

From the transfer functions above, the effects of both input and output disturbances are determined by the sensitivity function $S(j\omega)$, where $s = j\omega$. In order to minimize the disturbance effect at certain frequency, the magnitude of sensitivity function $|S(j\omega)|$ should be as small as possible with respect to the given frequency.

$$|Y_d(j\omega)| = |S(j\omega)(D_o(j\omega) + G(j\omega)D_i(j\omega))|$$

At the same time, the measurement noise attenuation is influenced by the complementary sensitivity function $T(j\omega)$, where the output is shown as

$$|Y_m(j\omega)| = |T(j\omega)D_m(j\omega)|$$

Similarly, the measurement noise effect is attenuated by minimizing the magnitude of complementary sensitivity $|T(j\omega)|$ with the given frequency.

However, the relationship between the sensitivity and complementary sensitivity is constrained with given frequency level by

$$S(j\omega) + T(j\omega) = 1$$

which indicates the trade-off to make only one of them small over the same frequency bonds. Because the disturbances and the measurement noise are normally existing in control system, where both of them have respective frequency region, disturbance term $|D_o(j\omega) + G(j\omega)D_i(j\omega)|$ has frequency contents concentrated in low frequency region, whereas the measurement noise $|D_m(j\omega)|$ is generally in high frequency region.

In general, the sensitivity function is chosen at $S(j\omega) \approx 0$ at the low frequency band, and the complementary sensitivity $T(j\omega) \approx 0$ at the high frequency region.

Bode plot provides frequency response of a LTI SISO plant. By presenting the magnitude of the frequency response gain, the bandwidth of the examined transfer function could be obtained. Figure 6.7 presents the Bode plots of the both sensitivity $S(s)$ and complementary sensitivity $T(s)$ functions. As discussed previously, the sensitivity transfer function frequency response is determining the response of input and output disturbances, the magnitude of 1 will amplify the disturbances at the high frequency region, as shown in Figure 6.7(a), higher $\gamma_i$ value will shift the response to the right,
thus the system could reject higher frequency disturbances. On the other hand, Figure 6.7(b) shows the frequency response of the complementary sensitivity function, which is identical to the closed-loop transfer function. Therefore, the obtained cut-off frequencies presents the bandwidth of the respective closed-loop system, obviously, high $\gamma_i$ value will increase the bandwidth, at the same time, it will amplify more frequency region inside the control system.

Nyquist plot is another analysis tool of the control system design, similarly, it is a parametric plot of the frequency response. In this situation, the Nyquist plot is applied for analysis of the current loop control with different tuning parameters. The open-loop frequency response of the control system is illustrated based on $G(s)C(s)$. The Nyquist plot, as shown in Figure 6.8, present the system stability with different controller parameters, from $\gamma = 0.5$ to $\gamma = 0.95$, which meant that the closed-loop bandwidth is designed as from 2 to 20 times of the open-loop system dynamics. The result is analyzed based on the gain margin and phase margin, which are all far away from the critical stability point ($-1, 0$) due to the stability of the induction current model. Since the frequency response plots of the given tuning parameters are fairly close to each other, the change of $\gamma_i$ value does not have a significant influence on the closed-loop stability, due to the system is already sufficiently stable. Thus, the degree of the stability could be analysed by using the Root-Locus tool as follows.

Root-Locus analysis presents the poles of a LTI system using a graphical method, which could demonstrate how the system poles would change while a certain system parameter changes. In this case, the Root-Lucas analysis is obtained with respect to the change of the tuning parameter $\gamma_i$ in Figure 6.9. There are two scenarios presented here, firstly,
Chapter 6. *PID Control Strategy*

**Figure 6.8:** Nyquist plot of current control loop. Key: line(1) $\gamma_i = 0.5$; line(2) $\gamma_i = 0.8$; line(3) $\gamma_i = 0.9$; line(4) $\gamma_i = 0.95$

**Figure 6.9:** Root-loci for PI control of current system ($0.3 \leq \gamma_i \leq 0.98$)

The Root-Lucas of the closed-loop current control is shown in Figure 6.9(a), which is purely analysed based on the dynamic model, thus the conjugate complex poles increase in both real and imaginary axis when the tuning parameter $\gamma_i$ increases, the raise of the complex poles is progressive due to the $\gamma_i$ determines the multiple of the open-loop poles. However, Figure 6.9(b) demonstrates the Root-Lucas analysis when there is a unit time delay inside the closed-loop system, which is realistic since there are many factors that would cause the time delay in the real implementation of induction motor control, such as inverter switching and controller computation. The unit delay increases the order
of the closed-loop system, thus three closed-loop poles are plotted on the graph, the $\gamma_i$ value increases from 0.3 to 0.98 with increment of 0.02, from the observation in Figure 6.9(b), the closed-loop system becomes unstable when $\gamma_i = 0.96$, which is the critical point at the system with a unit time delay. Therefore, the value $\gamma_i$ should be defined smaller than 0.96 in the real-time implementation.

### 6.2.5 Experimental Results

After the tuning analysis, the experimental results are obtained from the test-bed to evaluate the tuning discussion. Figure 6.10 demonstrates the experimental results with respect to different values of tuning parameter $\gamma_i$. The experiment setups are detailed in Appendix B.

In order to compare the results with respect to the tuning parameter $\gamma_i$, there are three values defined and presented in Figure 6.10. The selected values of $\gamma_i$ contain two extreme cases (ie. 0.3 and 0.95) and one desired value of $\gamma_i = 0.8$. Figure 6.10(a), 6.10(c) and 6.10(e) illustrate the control results of both $d$-axis and $q$-axis stator currents. On the other hand, Figure 6.10(b), 6.10(d) and 6.10(f) present the manipulated variables of the closed-loop control system.

For $\gamma_i = 0.3$, the closed-loop bandwidth is small, hence the transient response is slow and oscillating at the step change of $i_{sq}^*$ when $t = 1$s. However, the smaller bandwidth results the lower noise level of the control signals $u_{sd}$ and $u_{sq}$. Since the feedback gain is small, the amplification of noise is reduced as well. For $\gamma_i = 0.95$, the closed-loop control is seen as marginally stable, based on the discussion of the Root-Lucas analysis. The experimental results are noisy and undesired with steady-state error, due to the constraints of the control system. However, the control performance will become worse if the constraints were absent, as shown in Figure 6.10(d). After tuning analysis procedure and trials experiments, the desired experimental results of PI-based current control are obtained in Figure 6.10(e) and 6.10(f). In this case, the control performance is acceptable, measured outputs are following the set-points and the response time is fast, additionally, the noise level amplification on the control signals is reasonable due to the feedback controller gain and absence of the current sensor filer.

As claimed previously, the current control is significant for the induction motor drive control, once the current control is accomplished, the speed and position control systems are just another control loop outside the current control. Thus, in this section, the current control design and implementation are discussed with the tuning parameter analysis, at the end, the desired experimental results are obtained and ready for the following sections.
Chapter 6. PID Control Strategy

6.3 Speed Control

The Speed control of induction motor, using traditional PI controllers, is established in the cascaded structure. By using indirect FOC technique, one more outer-loop PI controller is designed for the speed control, based on the mechanical model equation (2.30), the manipulated signal from the velocity PI controller is the reference signal of $i_{sq}^*$ for the inner-loop control. The traditional indirect FOC speed control has the structure as shown in Figure 6.11, in this section, the speed control PI controller is
designed and implemented, moreover, an alternative control architecture is introduced, where a proportional controller is used to replace the PI controller for the \( i_{sq} \) current control.

\[ \begin{align*}
\frac{d\omega_r(t)}{dt} &= -\frac{f_d}{J_m}\omega_r(t) + \frac{3Z_pL_h}{2L_rJ_m}\psi_{rd}(t)i_{sq}(t) - T_L(t) \\
\end{align*} \]

where the load torque \( T_L(t) \) is treated as input disturbance, which will not be included in the control design, then the bilinear term \( \psi_{rd}(t)i_{sq}(t) \) is linearized as \( \psi_{rd}^*i_{sq}^*(t) \), due to the rotor flux \( \psi_{rd}(t) \) is controlled by the inner-loop control dynamics, which much faster compared to outer-loop dynamics (at least 10 times), since \( i_{sq}(t) \) in outer-loop model is also the set-point signal \( i_{sq}^*(t) \) for inner-loop control system, in order to achieve the cascaded control objective, thus the dynamic model becomes

\[ \begin{align*}
\frac{d\omega_r(t)}{dt} &= -\frac{f_d}{J_m}\omega_r(t) + \frac{3Z_pL_h}{2L_rJ_m}\psi_{rd}^*i_{sq}(t) \\
\end{align*} \]

which has the transfer function as follows using Laplace transform.

\[ \frac{\Omega_r(s)}{I_{sq}(s)} = \frac{\kappa_t\psi_{rd}^*}{s + \frac{f_d}{J_m}} \]

(6.23)
where \( \kappa = \frac{3}{2} \frac{Z_p L_n}{L_r J_m} \). From previous torque control section, note that (6.14) and (6.15):
\[
K_c^i = 2 \xi \omega_n \tau_d' - r_d \quad \text{and} \quad \tau_d^i = \frac{2 \xi \omega_n \tau_d' - r_d}{\omega_d^2 r_d'},
\]
which derives the closed-loop transfer function of the inner-loop PI control
\[
\frac{I_s(s)}{I_s^*(s)} = \frac{(2 \xi \omega_n - \frac{1}{\tau_d'}) s + \omega_n^2}{s^2 + 2 \xi \omega_n s + \omega_n^2} \quad (6.24)
\]
where \( I_s \) presents the stator current for both \( I_{sd} \) and \( I_{sq} \), due to the identical plant transfer function for both axes. The inner-loop steady-state gain is 1 by letting \( s = 0 \).

By substituting (6.24) into (6.23), the transfer function between the current reference \( I_{sg}^*(s) \) and the motor velocity \( \Omega_r(s) \) is obtained as
\[
\frac{\Omega_r(s)}{I_{sg}^*(s)} = \frac{\kappa_t \psi_{rd}^* (2 \xi \omega_n - \frac{1}{\tau_d'}) s + \omega_n^2}{s + \frac{f_d}{J_m} s^2 + 2 \xi \omega_n s + \omega_n^2} \quad (6.25)
\]
Because a first order LTI model is needed for PI controller design using pole-assignment method, the above transfer function is approximated, note that the inner-loop closed-loop bandwidth \( \omega_n \) is selected in the design, in order to obtain the approximation, \( \omega_n >> \frac{f_d}{J_m} \) is chosen, where \( \omega_n \) selection is large enough to neglect the dynamics from the inner-loop current control, thus the first order model for PI design is obtained
\[
\frac{\Omega_r(s)}{I_{sg}^*(s)} \approx \frac{\kappa_t \psi_{rd}^*}{s + \frac{f_d}{J_m}}
\]
Due to the approximation, the design of the outer-loop bandwidth is limited by the inner-loop dynamics, which meant that the approximation error will become significant if the outer-loop bandwidth is defined close or larger than the inner-loop bandwidth. Accordingly, the parameter design of velocity PI controller is based on the model dynamics using pole-assignment method
\[
K_{c}^o = \frac{2 \xi \omega_n^o - \frac{f_d}{J_m}}{\kappa_t \psi_{rd}^*} \quad (6.26)
\]
\[
\tau_{I}^o = \frac{2 \xi \omega_n^o - \frac{f_d}{J_m}}{\omega_n^2} \quad (6.27)
\]
where \( K_{c}^o \) and \( \tau_{I}^o \) are the outer-loop PI controller parameters, \( \omega_n^o \) represents the desired outer-loop bandwidth, super-script \((.)^o\) is denoted distinct from inner-loop. The outer-loop bandwidth is defined similarly according to outer-loop model time constant
\[
\omega_n^o = \frac{1}{1 - \gamma_\omega J_m}
\]
where \( \gamma_\omega \) is the tuning parameter for velocity PI controller.

The implementation of velocity PI controller is using velocity form similar to the previous
current PI controller. Thus the first order approximation of the derivatives according to the time instance \( t_i \) is shown as

\[
\frac{i_{sq}^*(t_i) - i_{sq}^*(t_i - \Delta t)}{\Delta t} = K_c^o \frac{e^o(t_i) - e^o(t_i - \Delta t)}{\Delta t} + K_c^o \frac{\tau_I^o}{\tau_I^o} (e^o(t_i))
\]

(6.28)

where \( e^o(t) = \omega^o(t) - \omega_r(t) \) and \( \Delta t \) is the sampling interval. Based on the velocity form implementation, by adding and subtracting the same steady-state value for each pair, the equation remains unchanged. Thus the actual physical variables are defined \( \tilde{i}_{sq}(t_i) = i_{sq}^*(t_i) + i_{sq_{ss}} \), \( \tilde{\omega}_r^*(t_i) = \omega_r^*(t_i) + \omega_{r_{ss}} \), and \( \tilde{\omega}_r(t_i) = \omega_r^*(t_i) + \omega_{r_{ss}} \). Then, the above equation is derived based on the actual physical variables for implementation

\[
\tilde{i}_{sq}^*(t_i) = \tilde{i}_{sq}(t_i - \Delta t) + K_c^o (\tilde{\omega}_r^*(t_i) - \tilde{\omega}_r^*(t_i - \Delta t) - \tilde{\omega}_r(t_i)) + \frac{K_c^o \Delta t}{\tau_I^o} (\tilde{\omega}_r^*(t_i) - \tilde{\omega}_r(t_i))
\]

(6.29)

If the set-point signal of \( \omega_r(t) \) is constant during steady state, \( \tilde{\omega}_r^*(t_i) - \tilde{\omega}_r^*(t_i - \Delta t) = 0 \), an alternative implementation is derived, which has an effect of reducing overshoot in the closed-loop response.

\[
\tilde{i}_{sq}^*(t_i) = \tilde{i}_{sq}(t_i - \Delta t) - K_c^o (\tilde{\omega}_r^*(t_i) - \tilde{\omega}_r(t_i - \Delta t)) + \frac{K_c^o \Delta t}{\tau_I^o} (\tilde{\omega}_r^*(t_i) - \tilde{\omega}_r(t_i))
\]

The over current protection of AC machine is essential in industrial application, which is not only for the safety reason, also in some cases the application requires the maximum torque output generated in speed start-up procedure. The implementation of manipulated signal limit in PI controller was introduced in previous torque control section, here the implementation of current limit \( i_{sq}(t) \) is discussed, if the inner-loop feedback error is defined as: \( e_{isq}(t) = i_{sq}^*(t) - i_{sq}(t) \), the constraints imposed on the reference current \( i_{sq}^*(t) \) is shown as

\[
-I_{sq}^{max} + e_{isq}(t) \leq i_{sq}^*(t) \leq I_{sq}^{max} + e_{isq}(t)
\]

(6.30)

where \( I_{sq}^{max} \) denotes the maximum current allowed. When integrator is applied in the inner-loop controller, then in steady state,

\[
\lim_{t \to \infty} e_{isq}(t) = \lim_{t \to \infty} i_{sq}^*(t) - i_{sq}(t) = 0
\]

However, a different answer will be obtained in the following section with a proportional controller used for \( i_{sq} \) control.

**Experimental evaluation.** The experimental results of speed control with PI plus PI control architecture is presented with the given setup, where the DC link bus voltage supply is 520 V, PWM carrier switching frequency is 2kHz, the sampling interval \( \Delta t \) is 100\( \mu \)s for inner-loop current control and 500\( \mu \)s for outer-loop speed control. The
reference signal of the rotor flux $\psi_{rd}^* = 0.5\text{Wb}$ within the rated speed region, the set-point of motor speed is defined with a step change, from standstill to half of the rated speed to the rated speed. The low-pass filter of the encoder measurement is set with the bandwidth of 150rad/s. The experiment time duration is set at 4 second. The tuning parameter of the inner-loop current control is $\gamma_i = 0.8$.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure6.12.png}
\caption{Experimental results of the PI+PI Speed control with tuning. Key: line(1) $\gamma_\omega = 0.9$; line(2) $\gamma_\omega = 0.95$; line(3) $\gamma_\omega = 0.98$; line(4) set-point signal.}
\end{figure}

In Figure 6.12, the experimental results of speed control are obtained by using three different PI controller tuning parameter $\gamma_\omega$, the step responses of three cases demonstrate the response time and control performance. Apparently, $\gamma_\omega = 0.9$ has the slowest control response while the quickest response with $\gamma_\omega = 0.98$. The plots are twisted during the second step change at $t = 2s$, due to the feature of the load condition, since the coupled DC motor is connected to a power resistor, the load torque will alter when the rotor shaft speed changes. The peaks of the speed measurement are obtained as the low-pass filter of the encoder has a high cut-off frequency, which could not filter out all the measurement noise, but in order to minimize the influence on the control performance, that is the price has to be paid.

From the tuning procedure previously, the tuning parameter of $\gamma_\omega = 0.98$ had been selected for the outer-loop speed PI controller. Thus, the rest of the experimental results are presented in Figure 6.13, which contains the current control performance and the control signals. The speed responses of $\gamma_\omega = 0.95$ and $\gamma_\omega = 0.98$ are fairly close to each other, since the constraint of the current reference $i_{sq}^*$ is active during the step change, as shown in Figure 6.13(a). In Figure 6.13(b), the control signals are excess the constrained limits at several points, due to the feed-forward compensation is added after the constraints check.
6.3.2 PI plus P Control Structure

The alternative architecture of speed control is illustrated in this section. Figure 6.14 shows the block diagram of the $q$-axis of the speed feedback control with inner-loop using proportional controller. The control design is proposed due to the final objective being the speed control, instead of current control, only one integrator is embedded in outer-loop PI controller to ensure zero steady-state error, thus the design of two controllers in $q$-axis, PI and P, is analysed together in order to achieve the control goal. Additionally, $d$-axis feedback control has only one controller due to indirect vector control, thus $i_{sd}$ controller remains with integrator to ensure zero steady-state error in $d$-axis. In cascaded control system, the design begins with the inner-loop control system, which will be obtained a steady-state error between the desired reference and the actual feedback, due to the proportional controller feature. A proper selection of the proportional gain $K_q^p$ is essential, since the expected steady-state error need to be considered in the design of the outer-loop PI controller. Thus, the closed-loop transfer function between the reference signal $I_{sq}^*(s)$ and $I_{sq}(s)$ is derived as

$$\frac{I_{sq}(s)}{I_{sq}^*(s)} = \frac{K_q^p}{s + \frac{1}{r_s \tau \sigma} + \frac{K_q^p}{r_s \tau \sigma}}$$  \hspace{1cm} (6.31)
where the closed-loop pole is simply determined at \(-\frac{1}{\tau_\sigma' - \frac{K_q^\ell}{r_\sigma' r_\sigma}}\), that implies a larger value of \(K_q^\ell\) leads to faster response of the inner-loop current control. Furthermore, the steady-state gain of the current control-loop is calculated by letting \(s = 0\) in (6.31) as

\[
\alpha = \frac{\frac{K_q^\ell}{r_\sigma' r_\sigma}}{\frac{1}{\tau_\sigma'} + \frac{K_q^\ell}{r_\sigma' r_\sigma}} \quad (6.32)
\]

where \(\alpha\) becomes the tuning parameter, with its value chosen as 0 < \(\alpha\) < 1. By solving (6.32), the design of the proportional controller is obtained as

\[
K_q^\ell = \frac{\alpha}{1 - \alpha} r_\sigma
\]

By substituting the proportional gain equation above into the close-loop transfer function (6.31), which becomes

\[
\frac{I_{sq}(s)}{I_{qs}(s)} = \frac{\frac{\alpha}{1 - \alpha} \frac{1}{\tau_\sigma'}}{s + \frac{1}{1 - \alpha} \frac{K_q^\ell}{\tau_\sigma}}
\]

Therefore, the closed-loop pole is located at \(-\frac{1}{1 - \alpha} \frac{1}{\tau_\sigma'}\), where the open-loop pole is at \(-\frac{1}{\tau_\sigma'}\), then the ratio between them is \(\frac{1}{\tau_\sigma'}\). For example, if \(\alpha = 0.9\), the controller gain is \(K_q^\ell = 9 \times r_\sigma\), the closed-loop pole is 10 times of the open-loop pole at \(-10 \frac{1}{\tau_\sigma'}\), the steady-state error will be

\[
\lim_{t \to \infty} \frac{i_{sq}^*(t) - i_{sq}(t)}{i_{sq}^*(t)} = 0.1
\]

The manipulated value computation of \(u_{sq}(t)\) is obtained using proportional controller and feedforward configurations

\[
u_{sq}(t) = K_q^\ell (i_{sq}^*(t) - i_{sq}(t)) + r_\sigma \tau_\sigma' \omega_s(t)i_{sd}(t) + k_r \omega_s(t) \psi_{rd}(t)
\]

The design of the PI speed controller is different for present architecture, due to the inner-loop steady-state error. By substituting the closed-loop transfer function of \(i_{sq}(t)\) (6.31) into the velocity model transfer function (6.23), then the transfer function between \(\Omega_r(s)\) and \(I_{sq}^*(s)\) is obtained

\[
\frac{\Omega_r(s)}{I_{sq}^*(s)} = \frac{\frac{\nu_{psi}}{\tau_{r/\sigma}}}{s + \frac{1}{J_m} \frac{1}{\tau_{r/\sigma}}} \frac{\frac{K_q^\ell}{r_\sigma' r_\sigma}}{s + \frac{1}{1 - \alpha} \frac{K_q^\ell}{\tau_\sigma' r_\sigma}}
\]

\[
= \frac{\frac{\nu_{psi}}{\tau_{r/\sigma}}}{s + \frac{1}{J_m} \frac{1}{\tau_{r/\sigma}}} \frac{\alpha}{1 - \alpha} \frac{1}{r_\sigma' \tau_\sigma'} \frac{1}{s + \frac{1}{1 - \alpha} \frac{K_q^\ell}{\tau_\sigma' r_\sigma}} \quad (6.33)
\]

Again, because the PI controller requires a first order LTI model, an approximation is made by ensuring the proportional feedback control gain \(K_q^\ell\) being large, as a result, the inner-loop dynamics is much faster than the outer-loop \((1 - \alpha) \tau_\sigma' >> \frac{1}{J_m} \frac{1}{\tau_{r/\sigma}}\). Thus, the
first order model is approximated for PI controller design

\[
\frac{\Omega_r(s)}{I_{sq}(s)} \approx \frac{\kappa_i \psi_{rd} \alpha}{s + \frac{f_d}{J_m}}
\]

Then, the PI controller parameters are calculated as

\[
K_c^o = \frac{2\xi \omega_n - \frac{f_d}{J_m}}{\kappa_i \psi_{rd} \alpha}
\]

\[
\tau_f^o = \frac{2\xi \omega_n - \frac{f_d}{J_m}}{\omega_n^2}
\]

where the closed-loop steady-state gain \( \alpha \) from the inner-loop control will affect the value of \( K_c^o \) for the outer-loop PI controller. More discussion on the natural frequency \( \omega_n \) with respect to closed-loop performance will be discussed in later section.

The implementation of the proportional controller of inner-loop is straightforward, and also the constraints implementation can be applied using a saturation in position form, due to no integrator in feedback control loop.

However, special consideration is made for constraints implementation of outer-loop controller, since the steady-state error of inner-loop control.

\[
\lim_{t \to \infty} e_{isq}(t) = \lim_{t \to \infty} i_{sq}^*(t) - i_{sq}(t) \neq 0
\]

**Figure 6.15:** Experimental results of the PI+P Speed control with tuning. Key:

- line(1) \( \gamma_\omega = 0.9 \)
- line(2) \( \gamma_\omega = 0.95 \)
- line(3) \( \gamma_\omega = 0.98 \)
- line(4) set-point signal

**Experimental evaluation.** The experiment of PI plus P speed control has been implemented using the identical settings as the previous case. Additionally, the tuning parameter of the proportional controller of \( i_{sq}(t) \) is defined as \( \alpha = 0.9 \) after the tuning
The proposed PI+P control architecture is accomplished by focusing on the final control objective, that is the speed control. Thus, the results obtained in Figure 6.15 have similar characteristics with the results in Figure 6.12, so that, the control structure is validated and could be used for some special applications, which does not require accurate current control at inner-loop.

The inner-loop current control results are shown in Figure 6.16, where the steady-state error can be observed in Figure 6.16(a) for $i_{sq}$ current control, since the tuning parameter of inner-loop proportional controller is defined as $\alpha = 0.9$, thus the steady-state error is not obvious, the current $i_{sd}$ is controlled using the PI controller same as before using tuning parameter of $\gamma_i = 0.9$. In Figure 6.16(b), the control signal of $u_{sd}$ should be identical to PI+PI case, while the control signal $u_{sq}$ is computed using the proportional controller gain $K_q$ multiplies the steady-state error of the $i_{sq}$ current control, thus, the larger proportional gain could reduce the steady-state error in this situation.

Figure 6.16: Experimental results of Speed control using PI+P controllers. Key: line(1) Actual feedback measurement; line (2) Set-point signal
6.4 Position Control

Traditionally, position control using vector control has been achieved by adding another PI feedback control loop outside and cascaded to the velocity control loop, in order to deal with the additional order, \( \theta_r(t) = \int_0^t \omega_r(\tau) d\tau \). Thus, the Laplace transfer function of the position model is

\[
\frac{\Theta_r(s)}{\Omega_r(s)} = \frac{1}{s}
\]

which is substituted into velocity model transfer function (6.23) leads to

\[
\frac{\Theta_r(s)}{I_{sq}(s)} = \frac{\kappa \psi_{rd}^*}{s(s + \frac{J_m}{L_r})}
\]

(6.34)

where \( \kappa = \frac{3 Z_L L_n}{2 L_r J_m} \), in this thesis, a PID controller with filter is designed to control this second order model. The layout of the control structure is shown in Figure 6.17. Note that the derivative controller plus a filter \((D + F)\) is placed at the feedback loop, instead of the error signal, in order to avoid the amplification of the derivative term. In this section, the PID controller of position control is designed and implemented with traditional PI current control for the inner-loop, then alternative structure with proportional control of inner-loop is introduced, at the end, the tuning and comparison between two architectures are analysed.

![Diagram](image-url)

**Figure 6.17:** Position control of induction motor using indirect vector control structure \((D + F\) presents the derivative term of controller with embedded filter)
6.4.1 PID plus PI Control Structure

In order to achieve the control goal, the position control model transfer function is firstly obtained by substituting closed-loop transfer function of current control (6.24) into the position model (6.34)

\[
\frac{\Theta_r(s)}{I_{sq}(s)} = \frac{\kappa \psi_{rd}^*}{s(s + \frac{f_d}{J_m})} \frac{(2\xi \omega_n - \frac{1}{\tau_c})s + \omega_n^2}{s^2 + 2\xi \omega_n s + \omega_n^2} \tag{6.35}
\]

Similarly, due to the closed-loop response time of inner-loop is much faster than outer-loop time constant \( \frac{f_d}{J_m} \), the approximation by neglecting the inner-loop dynamics is made to obtain the second order model for design

\[
\frac{\Theta_r(s)}{I_{sq}(s)} \approx \frac{\kappa \psi_{rd}^*}{s(s + \frac{f_d}{J_m})} = \frac{b}{s(s + a)}
\]

where \( a = \frac{f_d}{J_m} \) and \( b = \kappa \psi_{rd}^* \). An ideal PID controller has the transfer function as shown

\[
C(s) = K_c\left(1 + \frac{1}{\tau_1 s} + \tau_D s\right) = \frac{c_2 s^2 + c_1 s + c_0}{s}
\]

where \( K_c = c_1 \) is the proportional gain, \( \tau_1 = \frac{c_1}{c_0} \) is the integral time constant and \( \tau_D = \frac{c_2}{c_1} \) is the derivative gain.

Thus, the closed-loop transfer function of the position control

\[
\frac{\Theta_r(s)}{\Theta_r^*(s)} = \frac{\frac{b(c_2 s^2 + c_1 s + c_0)}{s^2(s + a) + b(c_2 s^2 + c_1 s + c_0)}}{\frac{b(c_2 s^2 + c_1 s + c_0)}{s^2(s + a) + b(c_2 s^2 + c_1 s + c_0)}} = \frac{b(c_2 s^2 + c_1 s + c_0)}{s^2(s + a) + b(c_2 s^2 + c_1 s + c_0)} \tag{6.36}
\]

where the closed-loop polynomial has third order, by applying pole-assignment technique, the desired closed-loop polynomial is designed for the performance specification.

In order to keep the property of the second order polynomial \( s^2 + 2\xi \omega_n s + \omega_n^2 \), another fast pole is determined to be \(-n \times \omega_n \ (n >> 1)\), which leads to the desired closed-loop polynomial as

\[
A_d^d = (s^2 + 2\xi \omega_n s + \omega_n^2)(s + n \times \omega_n) = s^3 + t_2 s^2 + t_1 s + t_0
\]

where \( t_2 = (2\xi + n)\omega_n \), \( t_1 = (2\xi + 1)\omega_n^2 \), \( t_0 = n\omega_n^3 \). Note that the pair of dominant desired closed-loop poles are still located at \(-\xi \omega_n \pm j \omega_n \sqrt{1 - \xi^2} \), where \( \xi = 0.707 \), thus the parameter \( \omega_n \) is approximately the bandwidth of desired closed-loop system as before.
The design process of the PID controller is straightforward, by letting the actual closed-loop polynomial equal to the desired closed-loop polynomial as

\[ s^2(s + a) + b(c_2s^2 + c_1s + c_0) = s^3 + t_2s^2 + t_1s + t_0 \]

The PID controller parameters are obtained based on the model dynamics and desired closed-loop specification:

\[
\begin{align*}
K_c &= \frac{(2\xi_n + 1)\omega_n^2}{b} \\
\tau_I &= \frac{2\xi_n + 1}{\omega_n} \\
\tau_D &= \frac{(2\xi + n)\omega_n - a}{(2\xi_n + 1)\omega_n^2}
\end{align*}
\]

where the design parameter \( \xi = 0.707 \) and \( \omega_n = \frac{1}{\sqrt{1 - \gamma^2}} \times a \), where \( 0 < \gamma < 1 \). Therefore, the computation of the control signal \( i_{sq}^*(t) \) is obtained as

\[
i_{sq}^*(t) = K_c(\theta_r^*(t) - \theta_r(t)) + \frac{K_c}{\tau_I} \int_0^t (\theta_r^*(\tau) - \theta_r(\tau))d\tau - K_c\tau_D\omega_r(t)
\]

where the derivative term is implemented in an alternative approach. Theoretically, the derivative term should be \( u_D(t) = K_c\tau_D\frac{d(\theta_r^*(t) - \theta_r(t))}{dt} \), since the derivative of a step change on reference signal \( \theta_r^*(t) \) could lead to a large overshoot of the control signal \( u_D(t) \). Thus the derivative term is only applied on the feedback loop as shown

\[
u_D(t) = -K_c\tau_D\frac{d(\theta_r(t))}{dt} = -K_c\tau_D\omega_r(t)
\]

The implementation of the PID position controller is completed using velocity form as well, note that the filter embedded with the derivative term is not included for design in order to reduce order of the closed-loop polynomial, however it is essential in the implementation due to the measurement noise of the velocity. Therefore, the Laplace transform of the control signal with embedded filter is obtained as

\[
I_{sq}^*(s) = K_c(\theta_r^*(s) - \theta_r(s)) + \frac{K_c}{\tau_I s} (\theta_r^*(s) - \theta_r(s)) - \frac{K_c\tau_D\omega_r(s)}{\tau_f s + 1}
\]

where \( \tau_f \) is the time constant of filter, chosen to be \( \tau_f = 0.1 \times \tau_D \).

If \( u_D^f(s) \) is denoted for the derivative control signal as \( u_D^f(s) = \frac{K_c\tau_D\omega_r(s)}{\tau_f s + 1} \), then the velocity form implementation of derivative term control signal at time instant \( t_i \) is obtained

\[
u_D^f(t_i) = \frac{\tau_f}{\tau_f + \Delta t} u_D^f(t_i - \Delta t) + \frac{K_c\tau_D\Delta t}{\tau_f + \Delta t} \omega_r(t_i)
\]
Overall, the control signal $i_{sq}^*(t_i)$ is computed from PID controller.

$$i_{sq}^*(t_i) = i_{sq}(t_i - \Delta t) + K_c(\theta_r^*(t_i) - \theta_r^*(t_i - \Delta t) - \theta_r(t_i) + \theta_r(t_i - \Delta t)) + \frac{K_c}{\tau_I}(\theta_r^*(t_i) - \theta_r(t_i)) - u_{D}^f(t_i) + u_{D}^f(t_i - \Delta t)$$

Alternatively, if the reference signal is constant during the steady-state, that is $\theta_r^*(t_i) - \theta_r^*(t_i - \Delta t) = 0$, then the control signal computation becomes:

$$i_{sq}^*(t_i) = i_{sq}(t_i - \Delta t) - K_c(\theta_r(t_i) - \theta_r(t_i - \Delta t)) + \frac{K_c}{\tau_I}(\theta_r^*(t_i) - \theta_r(t_i)) - u_{D}^f(t_i) + u_{D}^f(t_i - \Delta t)$$

The constraints implementation of the PID controller is identical to the previous velocity case as (6.30).

---

**Figure 6.18:** Comparison of position control using PID+PI. Key: line (1) $\gamma_\theta = 0.95$; line (2) $\gamma_\theta = 0.98$; line (3) $\gamma_\theta = 0.99$; line (4) Set-point signal.

---

**Figure 6.19:** Experimental of inner-loop control using $\gamma_\theta = 0.99$. Key: line (1) Actual measurement; line (2) Set-point signal.

---

**Experimental evaluation.** The experimental results of position control are shown in Figure 6.18 using the PID+PI control structure. The power supply of the coupled DC motor is turned on around 1s to provide the load torque as a disturbance. The
constraints of the current reference is defined at ±3A, the sampling time of inner-loop
current is 100µs while the outer-loop is 500µs.
Since the position control requires the large bandwidth, the defined value of γθ is suffi-
ciently large. Note that, when γθ = 0.95, the control system could not completely reject
the disturbance effect of the load change, as shown in Figure 6.18(a), the results of other
two cases have satisfying control performance, no steady-state errors and disturbance
rejected completely. Moreover, the error signal of θ∗r − θr is presented in Figure 6.18(b),
by taking a closer view of the error signal, the difference between the results of γθ = 0.98
and γθ = 0.99 becomes obvious, especially after the load torque is switched on. There-
fore, the best control performance is obtained when the tuning parameter γθ = 0.99 is
defined.
However, the limitation of high feedback gain as defined γθ = 0.99 is noise raise on
the reference signal i∗sq for inner-loop, which is the similar to the cascaded MPC case
in Chapter 4. In PID controller situation, a low-pass filter is normally embedded with
the derivative term of the controller. The results shown in Figure 6.19(a) are obtained
after the filter time constant changes to τf = 0.5τD, which still observing the noise on
the current reference signal i∗sq, but if the filter is too 'heavy', the control performance
will be affected, which is another trade-off in the control design process. Figure 6.19(b)
prets the control signals results, which does not active the constraints during the
experiment operation.

### 6.4.2 PID plus P Control Structure

![Diagram](Figure 6.20: Cascade feedback control of angular position of induction motor)

The alternative structure is replacing the inner-loop PI control on q-axis with a propor-
tional controller, in order to achieve the final control objective, that is position control.
The block diagram of presented architecture is shown in Figure 6.20. The design of the
proportional controller is introduced in the previous section.
Here the influence on the outer-loop PID controller from the modification is discussed,
firstly, the open-loop transfer function is approximated with the inner-loop steady-state
gain $\alpha$

$$\frac{\Theta_r(s)}{I_{sq}(s)} \approx \frac{\kappa_t \psi_r^{*} \alpha}{s(s + \frac{f_d}{J_m})}$$

Then, the model parameters $a = \frac{f_d}{J_m}$ remain the same, whereas $b = \kappa_t \psi_r^{*} \alpha$ is altered for the controller parameters design.

Since the design and implementation procedures of the inner-loop proportional control are identical to the PI+P case previously, the detail will not be discussed to avoid the repetition.

Figure 6.21: Comparison of position control using PID+PI. Key: line(1) $\gamma_\theta = 0.95$; line (2) $\gamma_\theta = 0.98$; line (3) $\gamma_\theta = 0.99$; line (4) Set-point signal.

In order to provide a reasonable comparison with the previous section, the experimental results of the position control are obtained using identical settings as the PID+PI case. Additional information is the tuning parameter of the inner-loop current control of $q$-axis, which is defined as $\alpha = 0.9$.

Figure 6.21 presents the comparison of the position control performance using different controller parameter value $\gamma_\theta$, the step responses are similar to the observation from Figure 6.18. However, note that the control response of $\gamma_\theta = 0.99$ in Figure 6.21(a) has even slower response time than $\gamma_\theta = 0.98$ results, moreover, the results of these two cases have been compared in Figure 6.21(b), where the result of $\gamma_\theta = 0.99$ has better performance during the steady-state operation under load condition and the disturbance rejection process. The explanation of this observation could be found from the inner-loop current control results.

In Figure 6.22, the current results of inner-loop control are presented with respect to different $\gamma_\theta$ values. The current control of $\gamma_\theta = 0.98$ in Figure 6.22(a) has reasonable results, the $q$-axis current is controlled with very small steady-state error. On the other hand, the results of $\gamma_\theta = 0.99$ is considered as undesired, as shown in Figure 6.22(b), specially during the response transient, the feedback system amplifies the noise level, so that the large peaks are observed in the current reference signal $i_{sq}^{*}$. Note that, the
reference signals of $i_{sq}^*$ in both sets of results exceed the constraints, which is caused by the one sample difference during the computation of the real-time constraints. Therefore, the limitation of the inner-loop proportional controller is observed, due to the amplification of the noise level at the high feedback gain situation.

### 6.5 Summary

This chapter has presented the design, tuning and implementation of the PID control strategy of the induction motor drive. The results are discussed and summarized as follows.

**Current control.** In the current control system design, the linearization is first introduced to eliminate the cross terms and the input disturbance. Hence, the PI controller is designed based on the first-order SISO plant. The anti-windup mechanism is included in the implementation using the velocity form. Then, the sensitivity functions are introduced, in order to discuss the frequency responses by using different tuning analysis. In general, high gain controller is preferable for two main reasons. One is the pulse width modulation errors are modelled as input disturbance, and to reduce the effect of this disturbance, the amplitude of the input sensitivity function needs to be small at the low and medium frequency regions. The other reason of a high gain current controller is essential for outer-loop velocity and position control, with the cascaded control structure, the design is based on the outer-loop dynamic model while ignoring the inner-loop feedback dynamics. In short, if a PI controller is used for the current control and the bandwidth of the current control system is specified as $\omega_m = \frac{1}{1-\gamma_i}a$ where $-a$ is the open-loop pole of the current system. Moreover, $\gamma_i$ is tuned to obtain 10 to 20 times of the open-loop pole location.

**Velocity and position control.** Both of the velocity and position controllers are in
the outer-loop of the control system structure. PI controller is used for velocity and PID controller is used for position control because of the requirement of steady-state performance for load disturbance rejection and reference following. The major concern for the outer-loop is the robustness of the closed-loop system against unmodelled dynamics neglected from the inner-loop system and the modelling errors of the mechanical part of the system. For the controller tuning, the parameters $\gamma_\omega$ and $\gamma_\theta$ are recommended to a smaller closed-loop bandwidth, which will lead to a slower closed-loop response, but will have a larger tolerance to the unmodelled dynamics from the inner-loop system and from the potential mismatch of the inertial parameter in the mechanical system.

**Choice between P current control and PI current control.** By comparing two controller, the proportional current controller is simpler in its structure, but limited applications in current control. Proportional current controller could only be used when there is a PI or PID controller cascaded connected in its outer-loop. However, the simpler controller structure when using P controller does not necessarily provide the advantages over slightly more complicated structure when using PI controller, since the inner-loop PI controller has a better performance in disturbance rejection.
Chapter 7

Comparison and Discussion of Control Algorithms using Experimental Results

7.1 Introduction

The main purpose of this chapter is to compare the proposed control methods of induction motor in this thesis. The categories of the demonstrated control strategies could be summarised to: (1) Current control, which includes the continuous-time MPC, the Finite Control Set-MPC and PI control methods; (2) Velocity control, which contains the continuous-time Gain-Scheduled MPC, cascaded continuous-time MPC and PI-based FOC methods; (3) Position control, which includes the cascaded MPC with Non-minimal state-space model and the PID control methods. In this chapter, the control performances are compared and analysed according to the control system structures in each category. Moreover, the robustness analysis is illustrated for the proposed controllers to demonstrate the capacity of the control methods in the presence of physical parameter mismatches.

7.2 Current Control

The current control methods, that have been introduced in this thesis, consist of three types: continuous-time MPC, FCS-MPC and PI controller. In this section, the experimental results of these control methods are compared and discussed, in addition to robustness analysis. Thus, the proposed current control methods are investigated for
Figure 7.1: Comparison of current control methods. Key: line (1) Actual measurement; line (2) Set-point signal.
Chapter 7. Comparison and Discussion of Control Algorithms using Experimental Results

their strengths and weaknesses.

7.2.1 Comparison Analysis

<table>
<thead>
<tr>
<th></th>
<th>MPC</th>
<th>FCS</th>
<th>PI</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>6</td>
<td>450</td>
<td>0.1F</td>
</tr>
<tr>
<td>p</td>
<td>Q</td>
<td>R_L</td>
<td>T_p</td>
</tr>
<tr>
<td>α</td>
<td>β</td>
<td>Δt</td>
<td>k_I</td>
</tr>
<tr>
<td>γ_i</td>
<td>Δt</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 7.1: Controller parameters of current controllers.

**Experiment setup.** The experimental results are compared based on the validated motor parameters from Appendix A. The controller parameters and experiment settings are displayed in Table 7.1. Note that, the controllers are tuned to obtained similar closed-loop eigenvalues or poles locations, in order to provide a fair comparison. The reference signals are selected at rated operation condition for all three cases: \( i_{sd}^* = 1.0526A \) and \( i_{sq}^* = 1.5A \), where \( i_{sd}^* \) is calculated using \( \psi_{rd}^*/L_h \) and the set-point value of the rotor flux is \( \psi_{rd} = 0.6Wb \). The load condition is identical for all three experiments, in which the coupled DC motor is connected with 3.5 Ohm power resistance at its terminal. The switching frequency for the employed PWM is 2kHz.

**Control responses results.** Figure 7.1 presents the experimental result of the current control comparisons, where the first row shows the step response of \( i_{sd} \) current and the second row is for \( i_{sq} \) current control. Obviously, the set-point following is achieved for all control methods. The high noise level is due to several reasons, such as slow switching frequency and possible mismatching between sampling instances of ADC and PWM. The noise amplitude is larger in the methods with PWM implementation (ie. MPC and PI), due to the PWM frequency is defined at 2kHz, which is relatively small value since the limitation of the xPC Target hardware.

<table>
<thead>
<tr>
<th></th>
<th>mean of SSE ( i_{sd} )</th>
<th>mean of SSE ( i_{sq} )</th>
<th>Response time</th>
<th>Steady-state speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPC</td>
<td>0.0018A</td>
<td>3.3327 \times 10^{-3}A</td>
<td>0.0052s</td>
<td>700RPM</td>
</tr>
<tr>
<td>FCS</td>
<td>-2.5854 \times 10^{-4}A</td>
<td>1.3642 \times 10^{-3}A</td>
<td>0.0011s</td>
<td>706RPM</td>
</tr>
<tr>
<td>PI</td>
<td>0.0317A</td>
<td>-1.0411 \times 10^{-4}A</td>
<td>0.0027s</td>
<td>676RPM</td>
</tr>
</tbody>
</table>

Table 7.2: Comparison results of current controllers.

**Numerical results.** Since the details of the experimental results are difficult to observe from the graphs, the numerical details are shown in Table 7.2 for analysis. The Steady-State Errors (SSE) of both \( dq \) axis current are presented, which are calculated based on the mean values of the difference between the reference signal and the measurement (ie. \( r(t) - y(t) \)) at every sampling instance. The SSE values are reasonable to prove the perfect set-point following capacity of the control methods, additionally, the \( i_{sd} \) SSE
of MPC and PI have relatively large values, due to the noise level and the oscillations during the transient. The variance value of the noise level during the steady-state are calculated by:

$$\sigma = \frac{\sum ((i_{sd} - i_{sd}^*)^2 + (i_{sq} - i_{sq}^*)^2)}{M}$$

where $M$ is the number of sample used for calculation. Hence, the variances are obtained as:

$$\sigma_{MPC} = 0.1373; \quad \sigma_{FCS} = 0.0719; \quad \sigma_{PI} = 0.1310$$

The response times indicate the length of duration for the feedback current to firstly reach the reference signal, since it is difficult to observe the response time from Figure 7.1, the numerical results are shown here to provide the comparison. The response time of FCS method is the quickest as expected, because the FCS technique is derived being equivalent to a deadbeat control with integral action. The PI control method has the middle value of response time according to its closed-loop pole locations, which are $-593.04 \pm j593.21$. On the other hand, the closed-loop eigenvalues of the MPC method are $-432.23 \pm j18.54$ and $-599.96 \pm j0.0025$, which are similar to PI controller, but higher order and heavier computational burden.

Moreover, the steady-state speed is the measurement of the motor shaft speed during the current control procedure, which could indicate the energy efficiency in a way if the mechanical load is identical. Figure 7.2 presents the speed responses of the current control methods on the common graph, together with the number in Table 7.2, the FCS control method has relatively high energy efficiency, that is the FCS current method could generate more mechanical power using the same current commands. Additionally, the MPC method could also provide similar steady-state speed with FCS, whereas the PI control method has obviously slower steady-state speed during the current control.

**Figure 7.2:** Speed Measurement for current control comparison. Key: line(1) MPC; line(2) FCS; line(3) PI.

**Discussion.** So far, the FCS method appears to be the finest current control method based on comparison, but as discussed previously, there are still some research issues in
this field. One issue is the switching frequency, since the FCS predictive controller generates the switching state of the IGBTs directly, the switching frequency is unexpected and could be varying during the operation, as shown in Figure 7.3(a), the switching frequency is spread over the range from 2kHz to 6kHz. On the other hand, both of PI and MPC current control methods are implemented based on the pulse width modulation. Hence the FFT analysis, which was presented in Figure 7.3(b), shows the frequency distribution of the phase current for both PI and MPC current controllers. For the traditionally PWM based control methods, the switching frequency is centralized according to the carrier frequency of PWM, which has the benefit when the low-pass filter is employed for current measurement, in most of industry controllers, the PWM carrier frequency could be defined more than 20kHz by using DSP based realization, where the PWM based control methods have significant advantages to apply low-pass filter in current feedback. Furthermore, the switching frequency issue of FCS has been addressed by different approaches in the literature.

### 7.2.2 Robustness Analysis

The robustness analysis is generally used by control engineers to examine the control response when there is parameter mismatching in the designed model. In this section, the three current controllers are tested with respect to the change of model parameter \( L_h \), which is the mutual machine inductance in the air gap, since the value of \( L_h \) is found to be the important physical parameter in the current dynamic model. The experimental setups and the controller tuning parameters remained the same as previous section, but since the reference signal \( i_{sd}^* \) is calculated from the reference of the rotor flux based on the relationship of \( i_{sd}^* = \psi_{rd}^*/L_h \), the change of \( L_h \) value could alter the set-point signal of the \( d \)-axis current.

**Smaller \( L_h \) value.** The current controllers are designed based on the modified mutual
Figure 7.4: Comparison of current control methods with 0.5$L_h$. Key: line (1) Actual measurement; line (2) Set-point signal.
Chapter 7. Comparison and Discussion of Control Algorithms using Experimental Results

The experimental results of the current control methods are shown in Figure 7.4, the organization of the figures is identical to the previous case. From the observation of the results, the \( i_{sd} \) current control has steady-state error for both MPC and PI methods, as shown in Figure 7.4(a) and 7.4(c) respectively. However, the FCS method in Figure 7.4(b) could control the \( i_{sd} \) without any steady-state error, the numerical comparison is also presented in Table 7.3. Note that, the control of \( i_{sq} \) current has been achieved for all three methods. The control signals of MPC and PI current controllers are presented in Figure 7.5, where the control signal \( u_{sd} \) reaches the constraints during the steady-state. Thus, the control of \( d \)-axis current could not achieve the set-point following. The noise level in Figure 7.4(e) is significantly smaller comparing to the other two cases in Figure 7.4(d) and 7.4(f), additionally, the noise level is also less than the previous case shown in Figure 7.1(e), due to change of the parameter value \( L_h \). The steady-state error values of \( i_{sq} \) are presented in Table 7.3, the relatively large SSE value of MPC method is due to the centralized control design.

The response times of all three methods are identical to the previous section, which proves that the controller bandwidth remains unchanged. The speed measurement are also recorded for an comparison, as shown in Figure 7.6. Since the different results of the current control responses, the operational speed has significant different values. The steady-state speed of FCS method remains the largest due to the zero steady-state error.
Chapter 7. Comparison and Discussion of Control Algorithms using Experimental Results

Figure 7.6: Speed Measurement for current control comparison with $0.5L_h$. Key: line(1) MPC; line(2) FCS; line(3) PI.

of the current control set-point following. The other two control methods have obviously lower speed levels as the current control of $i_{sd}$ is ineffective, in addition, the measured speed is higher for MPC comparing to the PI controller, due to the smaller SSE value in Table 7.3.

**Larger $L_h$ value.** The current control methods are tested with the mutual inductance of $2L_h$, which means that the actual mutual inductance is half of the model parameter value used in the control design. Similarly, the reference value of $i_{sd}^*$ has been changed to $i_{sd}^* = 0.5263A$.

<table>
<thead>
<tr>
<th></th>
<th>mean of SSE $i_{sd}$</th>
<th>mean of SSE $i_{sq}$</th>
<th>Response time</th>
<th>Steady-state speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPC</td>
<td>$8.3908 \times 10^{-4}A$</td>
<td>$-7.5011 \times 10^{-5}A$</td>
<td>0.0051s</td>
<td>475RPM</td>
</tr>
<tr>
<td>FCS</td>
<td>$1.4591 \times 10^{-4}A$</td>
<td>$-1.7928 \times 10^{-5}A$</td>
<td>0.001s</td>
<td>477RPM</td>
</tr>
<tr>
<td>PI</td>
<td>$7.0544 \times 10^{-5}A$</td>
<td>$-6.1706 \times 10^{-5}A$</td>
<td>0.0021s</td>
<td>477RPM</td>
</tr>
</tbody>
</table>

Table 7.4: Comparison results of current controllers with $2L_h$.

The experimental results are presented in Figure 7.7 for current control methods with $2L_h$ in the control design. Apparently, the set-point following task is achieved for all three control methods, in addition, the noise level is similar to the case which is designed based on the original value of $L_h$. From Table 7.4, the steady-state errors are significantly small for all three control methods in both $dq$ axis, which proves that the current controllers are still functional when the model parameter $L_h$ is mismatched at two times of the actual value. The response times are unchanged due to the closed-loop dynamics are kept constant. Furthermore, the steady-state speed measurements in this case are overlapping each other, as shown in Figure 7.8, because the reference signals are identical and the feedbacks are indeed following the reference, the generated electromagnetic torques are identical for all control methods, thus the similar speed responses are obtained.
Figure 7.7: Comparison of current control methods with $2L_h$. Key: line (1) Actual measurement; line (2) Set-point signal.
7.3 Velocity Control

The proposed speed control methods in the previous chapters include the centralized continuous-time MPC using Gain-Scheduling technique and the cascaded continuous-time MPC. Both of these methods will be compared with the PI-based speed control system as discussed in Chapter 6. In addition, the FCS predictive current control will be employed to provide a common inner-loop dynamics, in order to examine the outer-loop speed controllers, MPC and PI respectively. Furthermore, the robustness analysis is also investigated according to the motor parameter mismatching situation. In this case, the mechanical inertia $J_m$ of the motor shaft is selected as the mismatched parameter, since it has significant influence to the velocity control dynamics, besides, it could be altered when the attachment of motor shaft changes for different applications.

7.3.1 GS-MPC & PI-based FOC

In this subsection, the Gain-Scheduled MPC speed control system, which has been illustrated in Chapter 3, is compared with the traditional PI-based FOC speed control method. The control structures are different, one is centralized controller, the other is cascaded control. Moreover, the control techniques are also different: Direct FOC for GS-MPC and Indirect FOC for PI-based system, it is well known that Direct FOC has the weakness to handle the zero or low speed range control of the induction motor, thus the step reference signal is selected to provide fair comparison. In addition, the closed-loop bandwidths of both control systems are tuned similarly for the same purpose.

**Experiment and controller setting.** The experimental test-bed is set identical for both control systems. The coupled DC motor terminal is connected to 30 Ohm power resistor to provide reasonable load torque, the low-pass filter of the encoder feedback has
Chapter 7. Comparison and Discussion of Control Algorithms using Experimental Results

| GS-MPC Eigenvalues 1       | −169.08 ± j34.69; −11.90; −20.74 ± j3.59; −20 |
| GS-MPC Eigenvalues 3       | −154.00 ± j132.21; −52.20; −19.16 ± j0.30; −20 |
| GS-MPC Eigenvalues 3       | −146.07 ± j284.47; −68.93; −19.56 ± j0.21; −20 |
| PI Current loop           | −169.44 ± j169.49                                 |
| PI Velocity loop          | −19.5445 ± j19.5504                               |

Table 7.5: Closed-loop eigenvalues of GS-MPC and PI.

the cut-off bandwidth of 120 rad/s, the sampling interval is defined at 200µs, due to the heavy computational burden of the GS-MPC, the Luenberger observer is defined with gain of 1.3, the rest of experimental settings are following the instruction in Appendix B.

The controller parameters of the GS-MPC are defined following the values in Table 3.4, while the tuning parameters of the PI controllers are: $\gamma_i = 0.3$ and $\gamma_\omega = 0.982$. Therefore, the closed-loop eigenvalues are presented in Table 7.5. Because the multi-model is applied for GS-MPC, the control system dynamics are different, in addition, the tuning parameters could be selected separately as well to deliver better control performance.

For GS-MPC eigenvalues, the first pair of the complex poles with larger magnitude is the closed-loop eigenvalues of the current model, where the second pair is for speed control, the other two real numbers are the dynamics of the embedded integrators for the current and speed models, respectively. The closed-loop poles of the PI controllers are simply obtained using $-\xi \omega_n \pm \omega_n j \sqrt{1 - \xi^2}$. The experiment has duration of 10s, the speed reference signal is defined with three levels: 400, 700 and 1600 RPM and the set-point of the rotor flux is $\psi_{rd} = 0.6$ Wb.

**Speed control comparison.** Figure 7.9 presents the comparison as proposed, the velocity control results are plotted on common graph in Figure 7.9(a) for both methods, besides, the error signals are shown in Figure 7.9(b) to provide closer view of the speed control responses. The speed response of the PI control system has faster rising time during the step response, but some overshoot and oscillations occurred from the PI control performance. On the other hand, the GS-MPC method has more smooth transient response and less noise during the steady-state operation. The current measurements of GS-MPC and PI are shown in Figure 7.9(c) and 7.9(d) respectively. The GS-MPC is a centralized controller which only has two controlled outputs: rotor flux and motor speed, thus the stator currents are not under feedback control, Figure 7.9(c) presents the measurement of the stator current, to compare with the controlled current performance by PI controllers in Figure 7.9(d). Since the GS-MPC method directly control the rotor flux instead of $i_{sd}$, the $i_{sd}$ measurement of GS-MPC is not kept constant as the PI controller. Lack of current control dynamics is one of the weaknesses of the GS-MPC centralized control system. Furthermore, the control signals, which are the stator voltages, are shown in Figure 7.9(e) and 7.9(f). The observation of control signals is the
Chapter 7. Comparison and Discussion of Control Algorithms using Experimental Results

Comparison and Discussion of Control Algorithms using Experimental Results

Figure 7.9: Speed control comparison between GS-MPC and PI. Key: blue line: GS-MPC; black line: PI FOC; red dash: Set-point signal.

Discussion. After the comparison between the proposed GS-MPC and the traditional PI control method, the following conclusions are obtained: the Gain-Scheduled MPC could achieve the speed control control objective under the load condition; GS-MPC

noise level difference, where the noise level at control signals of GS-MPC is significantly less than the PI controllers, due to the state observer of the continuous-time MPC, which not only estimates the state feedbacks but also acts as a low-pass filter. Moreover, the revised approach of the modulation limit could allow more efficient control inputs for GS-MPC method.
could provide better control performance, based on the multiple tuning parameters for different operating conditions; however, the computational burden of the GS-MPC is much heavier than the PI controller, as well as the off-line computation; moreover, the current control lacking is another weakness of the GS-MPC method.

**Robustness analysis.** The robustness analysis of the GS-MPC is accomplished by designing the control system based on two different inertia values: $0.5J_m$ and $2J_m$. From the control point of view, the modification of inertia value will cause the mismatching of the model dynamics, since the time constant is obtained by $\tau = f_d/J_m$, however, the

![Robustness analysis of GS-MPC](image)

**Figure 7.10:** Robustness analysis of GS-MPC. Key: line(1): set-point signal; line(2): actual measurement.
inertia value changes when the shaft attachment changes in the applications. Therefore, the robustness analysis according to $J_m$ is reasonable to examine the proposed control method.

Figure 7.10 presents the experimental results of the GS-MPC, which is designed based on different values of $J_m$. On the left hand side, the GS-MPC is using the mismatching parameter $0.5J_m$ as the inertia value for the controller design, On the right hand side, the inertia value is defined as $2J_m$. By comparing to the original case in Figure 7.9(a), the overshoots during transients are observed, where the transient response in Figure 7.10(b) is slower and smoother comparing to the original case, it is because the mismatching parameter of $J_m$ could affect the closed-loop MPC gain values, hence influence the closed-loop control performance. However, the MPC controller could control both situations with stable and non-error steady-state operation. In addition, the stator currents responses, which are shown in Figure 7.10(c) and 7.10(d), the similar results are presented according to the respective velocity response, due to the $J_m$ parameter is only included in the mechanical model equation. The control signals of both results have better illustration of the closed-loop bandwidth, by comparing the control actions during the step changes in Figure 7.10(e) and 7.10(f), the control actions $u_{sd}$ and $u_{sq}$ are obviously faster for $2J_m$ case than the other, where the larger overshoots are obtained for $2J_m$ situation, however, the appearance in speed control results shows that the $0.5J_m$ case has larger overshoot in speed control performance, since the GS-MPC method is a centralized controller, which contains both electrical and mechanical models. Moreover, due to the different time constants in both models, the control actions could cause opposite performance in the speed responses.
7.3.2 Cascaded MPC & PI-based FOC

For controlling the induction motor drive, the cascaded structure is proven to be sufficient, due to the large difference of the electrical and mechanical system dynamics. In this section, the proposed cascaded MPC method in Chapter 4 will be compared with the PI-based FOC method, both of the methods are applied with the Indirect FOC technique, expect that the PI-based FOC contains three controllers while the cascaded MPC system has two controllers, since the inner-loop MPC is controlling both \( d \)-axis and \( q \)-axis stator currents. Besides, the low speed range control could be operated as the benefit of the Indirect FOC technique. In order to demonstrate the capacities of the control methods, different shapes of the reference signals are employed, which includes: the step changes with both directions, the ramp signals and zero speed operation. After that, the robustness of the cascaded MPC is analysed with step change set-point.

<table>
<thead>
<tr>
<th></th>
<th>Current MPC Eigenvalues</th>
<th>Speed MPC Eigenvalues</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(-332.24 \pm j14.83); (-499.95 \pm j0.0025)</td>
<td>(-16.7756 \pm j10.5273)</td>
</tr>
<tr>
<td>PI Current loop</td>
<td>(-338.88 \pm j338.98)</td>
<td>(-16.4585 \pm j16.4635)</td>
</tr>
<tr>
<td>PI Velocity loop</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 7.6: Closed-loop eigenvalues of cascaded MPC and PI.

Experiment and controller settings. To provide the reasonable comparisons, the MPC controllers and PI controllers are designed and tuned to have similar bandwidths and realized based on the identical test-bed setup. Thus, the load condition is identical to the previous section, the cut-off bandwidth of the encoder filter is defined at 150 rad/s, due to the faster closed-loop bandwidth of the speed control. The sampling interval is 100\( \mu \)s for inner-loop and 500\( \mu \)s for outer-loop, thus the computational load is reduced in this case, then the rest of experimental settings are identical to the definition in Appendix B.

The controller parameters are tuned to obtain the closed-loop eigenvalues in Table 7.6. For cascaded MPC tuning, the exponential weighting parameters are defined as \( \alpha = 1.2 \) and \( \beta = 250 \) for inner-loop and \( \alpha = 1.2 \) and \( \beta = 5 \) for outer-loop control, since the other tuning parameters do not provide significant influence to the closed-loop control, they are defined following the definition in Chapter 4. On the other hand, the PI controller of the inner-loop control is tuned with \( \gamma_i = 0.65 \) and the outer-loop controller has \( \gamma_\omega = 0.981 \), in order to provide the similar closed-loop pole locations.

Step reference signal. In the first scenario, the step change reference with both directions is defined for both control systems. During the 12s experiment operation, the reference signal are divided into six constant operational references with each duration of 2s, the set-point values are defined in order of: 700 RPM, 1400 RPM, 700 RPM, −700
Figure 7.11 presents the experimental results comparison. Note that, the peaks in the speed feedback are caused by the encoder measurement, since the low-pass filter is now set with higher cut-off frequency in order to reduce its influence on the feedback control performance, hence the high frequency peaks appear in the feedback signals. In Figure 7.11(a), the speed control responses of both schemes are obtained almost overlapping to each other. In order to provide a closer view of the speed control performance, the error signals between reference and feedback ($\omega^*_r - \omega_r$) are presented in Figure...
7.11(b). From the comparison of the experimental results, the speed response PI control system has faster response time than the cascaded MPC during transient, at the same time, transient overshoots are also obtained from PI control response, especially when the reference changes from positive to negative direction at $t = 6$ sec, obviously, the overshoot is obtained for PI control result, whereas the cascaded MPC system provides the better performance with smooth and no overshoot in the speed response. After that, the current control results are compared between Figure 7.11(c) and Figure 7.11(d), the control signals are compared based on Figure 7.11(e) and Figure 7.11(f). Similar control performances are observed for both control schemes, expect that the noise level in cascaded MPC results is slightly larger than the PI control, due to the extra closed-loop eigenvalues generated by the augmented model.

Therefore, the conclusion could be made that the speed control performance is primarily determined by the outer-loop speed controller. The closed-loop dynamics of the current control does not provide significant impact on the speed control, due to the cascaded control structure. In order to provide more fair comparison between the outer-loop speed controllers, the FCS predictive current controller is applied for both schemes in the next subsection.

Ramp reference signal. The second scenario is based on the ramp reference signal. In general, the ramp signal and zero speed operation are employed to examine the proposed control method for induction motor drive. In this scenario, the reference signals have duration of 12 sec, firstly, the speed rises from standstill to the rated speed within 4 sec, after 2 sec of constant speed operation at rated speed, the speed will drop in ramp to zero within another 4 sec, then will stay at zero speed for 2 sec. The controllers design and tuning parameters remained the same as in Table 7.6.

The experimental results of ramp reference control are shown in Figure 7.12, which includes the responses of the ramp acceleration and deceleration procedure. Figure 7.12(a) presents comparison of the speed responses during the ramp and zero-speed operations for both control systems, obviously both feedback control systems are stable and seen to achieve the set-point following control objective. However, the error signals of these speed responses, which are shown in Figure 7.12(b), suggest that the operational errors existed during the ramp change, since the ramp change has transfer function of $\frac{1}{s^2}$, which is second order reference signal, the integral action inside the control loop is not enough to completely remove all the steady-state errors during the ramp response control. Moreover, the ramp response error of cascaded MPC is observed to be larger than the error of PI control. Similar to the previous scenario, the inner-loop current control performance and the control signals have similar results as shown in Figure 7.12(c) to 7.12(f). In addition, the noise level of the inner-loop MPC control is larger than the PI control system.
Chapter 7. Comparison and Discussion of Control Algorithms using Experimental Results

Robustness analysis. In this section, the robustness analysis of the cascaded MPC scheme is illustrated with the step response with the mismatching of motor parameter $J_m$. The controller tuning parameters of the MPCs are kept unchanged from the previous section, as well as the inner-loop electrical model, but the outer-loop mechanical model is altered according to the change of $J_m$ value, which will lead to the different values of the controller feedback gain. Two set of experimental results are obtained with respect to the inertia values of $0.5J_m$ and $2J_m$. The outer-loop MPC has the exponential data weighting defined as $\alpha = 1.2$ and $\beta = 5$, which is defined same as before, but the
closed-loop eigenvalues are different: (1) $0.5J_m$ case: $s = -21.2121 \pm j15.5188$; (2) $2J_m$ case: $s = -13.7413 \pm j6.9107$. Obviously, the larger value of $J_m$ used for the design will generate the smaller bandwidth for the closed-loop control, since the controller would expect a larger inertia at the motor shaft.

The experimental results of the robustness examinations are illustrated in Figure 7.13 where the results according to $0.5J_m$ are shown on the left hand side and the results of $2J_m$ are presented on the right hand side. Overall, the cascaded MPC method could still achieve the control objectives with the significant parameter mismatching in the design.
model. The step response of speed control based on $0.5J_m$ is obtained to be faster and with overshoot comparing to the case of $2J_m$. This is because the closed-loop bandwidth of the $0.5J_m$ is wider, thus the closed-loop control performance obtained is aggressive and tends to become unstable according to the robust stability theory. Note that, the oscillations obtained at the low speed level are caused by the random current bias from the sensor measurements, which is amplified in the $0.5J_m$ case due to the larger value of controller feedback gain.

In conclusion, the proposed cascaded MPC method performs better for induction motor drive control. By comparing with the traditional PI control, the cascaded MPC could provide the capacities of the PI control method, in addition, the model predictive control also have benefits on MIMO system control and optimal control design with constraints.

7.3.3 MPC+FCS & PI+FCS

In order to obtain the comparison of the outer-loop speed controllers, the FCS predictive current control system is employed in this case to provide identical inner-loop current control dynamics. Thus, the combinations of MPC plus FCS and PI plus FCS are developed to seek for better speed control performance. In this subsection, the reference signals are defined identical to the previous case for comparison, both step and ramp set-point signals are implemented to obtain the experimental results.

The controller parameters remain unchanged, thus the closed-loop bandwidths of the outer-loop velocity control are following the values in Table 7.6. However, the sampling time has been altered for both inner and outer loop system, since the FCS method requires fast sampling rate to deliver fine performance, the sampling interval $\Delta t$ is defined as $80\mu s$ for current control and $400\mu s$ for outer-loop velocity control.

**Step reference signal.** The experimental results of step reference are presented in Figure 7.14. The difference between the speed control responses is more comparable than the previous subsection. Based on the observation from Figure 7.14(b), the overshoots occurred from the transient of the step change response of PI speed controller, whereas the model predictive controller has similar response time but less overshoot. The resultant difference between the two speed responses is small due to the normal operation with fine modelling. It is expected that the performance could be significantly improved for MPC controller in a larger scale system or critical operation condition.

The current control performances, which are displayed in Figure 7.14(c) and 7.14(d), have identical closed-loop response due to the same controller employed. Note that the noise level is significantly smaller than the current control results in Figure 7.11, because the sampling time and switching frequency are different. In addition, the control signals $i_{sq}^*$ generated from the outer-loop speed controllers have different shapes during
the transient responses, where the set-point $i_{sq}^*$ from the model predictive controller has several oscillations in order to deliver the smooth speed control performance.

**Ramp reference signal.** The ramp reference includes the acceleration, braking and zero speed procedures. Figure 7.15 illustrates the experimental results of proposed task. The results could also be compared with the previous case shown in Figure 7.12. First of all, the error signals are non-zero during the ramp operation, the reason had been discussed previously, the similar error values are obtained for both MPC and PI speed controllers by comparing with previous Figure 7.12(b), except that the performances have less noise on both signals, since the FCS predictive current control system delivers a better control performance. In addition, the spikes are observed at the current measurement feedback, since the hardware issue, which the sampling instance between the ADC module and the switching command are not synchronized.
Figure 7.15: Speed control comparison between MPC+FCS and PI+FCS using ramp reference. Key: line(1): actual measurement; line(2): set-point signal.

7.4 Position Control

In this thesis, the position control section is the extended research from the speed control schemes. The position control methods presented in this thesis are summarized as cascaded MPC and PID. In this section, the comparison between these two methods is illustrated based on the experimental results. Since the derivative of the position measurement is normally noisy, the filter is generally embedded in the control system. Hence, the MPC with Non-Minimal State-Space method is applied to compare with the PID with filter controller. Moreover, the robustness analysis of the MPC position controller is also included in this section.

**Experiment and controller settings.** The test-bed setups are implemented identically for both control systems. The coupled DC servo motor is supplied by high current source to provide the load torque during the position control. The sampling interval are defined as: 100µs for inner-loop and 500µs for outer-loop. Then, the carrier frequency is 2kHz for the PWM implementation in both current control methods. Moreover, other settings are following Appendix B.
Chapter 7. **Comparison and Discussion of Control Algorithms using Experimental Results**

<table>
<thead>
<tr>
<th>Controller Type</th>
<th>Eigenvalues</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current MPC</td>
<td>$-399.99; -399.99; -232.26; -232.2643$</td>
</tr>
<tr>
<td>Position MPC</td>
<td>$-353.50 \pm j353.61; -41.48 \pm j2.33; -36.51$</td>
</tr>
<tr>
<td>PI Current loop</td>
<td>$-296.52 \pm j296.61$</td>
</tr>
<tr>
<td>PID Position loop</td>
<td>$-34.75 \pm j34.76$</td>
</tr>
</tbody>
</table>

**Table 7.7:** Closed-loop eigenvalues of cascaded MPC and PID.

The closed-loop eigenvalues of the model predictive controller are displayed together with the closed-loop poles of the PID plus PI controllers in Table 7.7. The eigenvalues of the MPC position controller have five values, where the first pair of the complex poles is generated by dynamics of the embedded second order low-pass filter using NMSS, the second pair of complex pole is the closed-loop bandwidth of the predictive position controller, at the end, the last real eigenvalue is obtained from the augmented model. Since the eigenvalues of the embedded low-pass filter of NMSS are significantly larger than the position controller bandwidth, the dynamics of the filter is generally neglected. On the other hand, the closed-loop poles of the PID plus PI method are simply calculated from $-\xi \omega_n \pm j\omega_n \sqrt{1 - \xi^2}$. In order to obtain the closed-loop eigenvalues in Table 7.7, the tuning parameters of the predictive controller are defined as: $\alpha = 1.2$, $\beta = 200$ for inner-loop and $\beta = 20$ for outer-loop; the second order low-pass filter of the NMSS has bandwidth at 500 rad/s. For PI current controller, the tuning parameter is defined at $\gamma_i = 0.6$, where $\gamma_\theta = 0.991$ for the outer-loop PID controller, in addition, the filter of the derivative controller has time constant of $\tau_f = 0.1\tau_D$. Furthermore, the constraints of manipulated variables from the outer-loop controllers are implemented at $-3 \leq i_{sq}^* \leq 3$ A.

**Comparison of experimental results.** The set-point of the motor position is square wave signal. The duration of the operation is 6s. The motor shaft remains zero at first for 2s, then rises to $\pi$ rad and hold for another 2s, at the end, drops back to zero position. In addition, the load torque of 0.5 Nm in opposite direction will be manually turned on around 1 sec after the experiment start, in such a way that the load disturbance rejection and position control under load condition could be evaluated for both control methods.

The experimental results of both methods are presented in Figure 7.16, where the position control feedbacks are plotted on common graph, as shown in Figure 7.16(a), which illustrates that both position methods have achieved the disturbance rejection and complete set-point following. The error signals of $\theta^*_r(t) - \theta_r(t)$ are shown in Figure 7.16(b) to deliver closer view of the control performance, hence the difference is observed that the MPC position control has slower and less overshoot than the PID controller during the transients of step changes. Moreover, larger ripples are observed during the disturbance rejection process around $t = 1$ sec. The position control performance is
distinct because the MPC position controller has the low-pass filter embedded in the entire optimal control design. In contrast, the PID controller only has the filter built in the derivative control action, thus the closed-loop dynamics of the MPC are slower than the PID controller. Nevertheless, the results during the steady-state operation are similar for both control methods. The current control and control signals are displayed as well, where the MPC results are shown on the left hand side, and the PID results are presented on the right hand side. There are still noises contained in the current set-point $i^*_q$ for both control methods. However, the noise level is significantly smaller for
Chapter 7. Comparison and Discussion of Control Algorithms using Experimental Results

Figure 7.17: Robustness analysis of MPC position control. Key: line (1) Actual measurement; line (2) Set-point signal.
the MPC position controller comparing to the PID controller, due to the benefit of the filter used in the NMSS method. The difference is also detected on the control signals as shown in Figure 7.16(e) and 7.16(f), where the control signals of model predictive controller has much smaller ripple than the PID control signals during the step change responses.

**Robustness analysis.** The robustness analysis has been performed for the MPC position control with NMSS method. The experimental results are obtained for three different $J_m$ values: $0.5J_m$, $J_m$ and $2J_m$. Since the motor parameter has been altered, the closed-loop eigenvalues of the outer-loop position controller will be changed. Here, the first pair of complex poles is determined by the embedded low-pass filter, hence they will remain unchanged. The other three eigenvalues are calculated based on respective $J_m$ value: for $0.5J_m$, $-41.23 \pm 2.15$ and $-36.57$; for $J_m$, $-41.48 \pm 2.33$ and $-36.51$; for $2J_m$, $-41.57 \pm 2.37$ and $-36.53$.

![Figure 7.18: Error signal of MPC position control robustness analysis. Key: line(1) $2J_m$; line (2) $J_m$; line (3) $0.5J_m$.](image)

The experimental results of the robustness analysis are shown in Figure 7.17, where the first column presents the results of $0.5J_m$, the middle column is for $J_m$ and the last column is for $2J_m$. The step reference is employed with two steps: $\pi$ rads for first 3s then $2\pi$ rads for another 3s, the load torque disturbance is turned on around 1s same as before. From the first row of the results in Figure 7.17, the control objective and disturbance rejection are achieved for all cases. However, the difference could be observed from the control signals $i_{sq}^*$ generated from the outer-loop MPC, the noise level is the smallest for $0.5J_m$ case, but the control system is unstable for $2J_m$ case before the load torque turned on, note that the closed-loop control system become stable after the load torque turned on, which means that the controller designed based on $2J_m$ is marginally stable which is sensitivity to the input disturbance in the system, in addition, the noise level of $i_{sq}^*$ is the largest for $2J_m$ case even during the stable operation.

In order to obtain comparison of the position control performance, the error signals of all three cases are plotted in Figure 7.18. The oscillation of the $2J_m$ case on the
current control performance could be obtained with the zoom-in view, the oscillations with small amplitude on the position feedback continues until the load switched on. The larger size overshoot is observed from the 0.5$J_m$ results, as well as the disturbance rejection transient. Overall, the MPC position controller is observed to be sensitive with respect to the modelling error, due to the wide bandwidth and the higher order dynamics.

### 7.5 Summary

This chapter compares the proposed control schemes with the traditional PID-based control design by obtaining the experimental results of controlling an induction motor drive. The robustness of the arranged controllers is also analysed with mismatched the machine parameters.

**Current control.** For current control applications, the three types of the current controllers are implemented and compared with respect to their stator current responses, open-loop speed responses and the FFT analysis. Based on the observed findings, it is seen that the Finite Control Set could deliver the best current control performance, as the steady-state errors had been completed eliminated by using the proposed approaches. However, the lack of modulation in the implementation will lead to a spread distribution of the frequency response, unlike the centralized frequency response by using pulse width modulation. The issues of these findings include the induced harmonics level and switching losses, as well as the difficulty to filter the unnecessary switching frequency. The robustness analysis of these three current controllers suggests that the FCS predictive control method has the best capacity to handle the unmodelled situations.

**Velocity control.** The speed control systems, which include the centralized GS-MPC, cascaded MPC and the PI-based FOC methods, are extensively compared and discussed via different reference signals. The GS-MPC is observed to deliver better speed control performance than the PI controllers, due to high degree of freedom for controller tuning and revised constraints implementation using Quadratic Programming. However, the complexity of the controller requires heavy computational load and the lack of current control will lead potential issues, such as harmonics. The robustness of the GS-MPC method is evaluated using different values of the mechanical inertia, the controller could precisely achieve the control objective even designed with modelling error.

The cascaded MPC speed control was compared with the PI control via step change, ramp and zero reference signals. The similar experimental results are observed for both control methods. To obtain reasonable comparison between speed controllers, the inner-loop current control system is replaced by the FCS predictive control. Thus, the more obvious difference had been found, that the continuous-time MPC speed controller
Chapter 7. *Comparison and Discussion of Control Algorithms using Experimental Results*

delivered smoother and less overshoot control performance under identical closed-loop bandwidth. The robustness analysis of the cascaded MPC method also suggest that it could achieve the control objective when the system model is incorrect to some degree. The cascaded control structure is found more suitable for induction motor speed control, since the significant difference between the electrical and mechanical model dynamics.

**Position control.** The position control application of the machine drive is generally extended from the velocity control. In this thesis, the position controllers are designed by modifying the speed controller, instead of adding another control loop. The main challenge of the position control is noise induced from the derivative of the rotor shaft angle. For both cascaded MPC and PID plus PI control methods, a low-pass filter is embedded for their control design, especially for continuous-time model predictive control, the Non-Minimal state space model is applied. The comparison of experimental results for both methods detects that the performance of position response for MPC has slower and smaller oscillation than PID controller by tuning with identical closed-loop bandwidth.
Chapter 8

Conclusion

8.1 Summary

In this thesis, the model predictive control of the three phase induction motor drive was studied. Particularly, the continuous-time model predictive control was designed using both centralized and cascaded structure, and the recent finite control set predictive control was also investigated. It can be concluded that the model predictive control concept can achieve the control objective as well as deliver high performance in the motor drive control applications, especially when the non-linearity issue is resolved for the mathematical model. The experimental results obtained from the proposed control methods are compared with the traditional PID control in Chapter 7, moreover, the robustness of these controllers are also analysed and concluded. Thus, the brief summaries of each contribution will be revealed in this chapter.

The following summarizes the major conclusions from Chapter 3-4 regarding continuous-time model predictive control of the induction motor drive:

- Linearization of the dynamic model is essential for the continuous-time MPC design, particularly for the centralized controller structure.

- In Chapter 3, the design of continuous-time MPC based on LTI model was thoroughly studied. For MPC design of the full order non-linear induction motor model, the Gain-Scheduled method was introduced with respect to different operating conditions. The implementation of the non-linear constraints is developed from the benefit of the on-line Quadratic Programming procedure. The analysis and experimental results showed that the proposed control system could handle the wide range speed control precisely with disturbance rejection, but the absence of current control could introduce uncertainties to the control system. The increased
complexity of high order dynamics limit might restrain the closed-loop bandwidth because of the uncertainty and expected modelling errors.

- In Chapter 4, the cascaded continuous-time MPC for current, velocity and position control application of induction motor drive was studied. The stability analysis of the inner-loop stator current dynamics suggested that the non-linearity could be eliminated by using the high gain feedback control design. Since the mechanical model was controlled separately in the outer-loop, the MPC design for the velocity and position controllers were studied, respectively.

- The cascaded control structure could offer more flexibility and better performance since the different closed-loop bandwidth were designed for respective model dynamics.

The contributions from Chapter 5 regarding Finite Control Set predictive control are summarized as follows:

- The original FCS predictive control was studied and revised in the feedback control manner.

- The revised FCS method was analysed, the feedback controller gain was designed for the constrained optimal control.

- The integral action was developed with the revised FCS method for vanishing the steady-state error. The integral gain was calculated regarding to the closed-loop bandwidth. The I-FCS controller was derived with constrained optimal control design. The implementation of the proposed controller was illustrated and evaluated via the experimental results.

- The FCS method in $\alpha\beta$ coordinates was also studied. The resonant controller was designed for eliminating the steady-state errors in the sinusoidal signals of the $\alpha\beta$ stator currents.

- The experiment evaluation suggests that the proposed method of FCS could successfully solve the research issue of the steady-state errors, which appeared in the original FCS method. The control systems were proven to be very robust against the motor parameter variations.
8.2 Future Research

The following potential topics are considered appropriate for the future research:

- The proposed control designs are extensible to other power electronics applications.
- The control algorithms could be applied to specific applications which generally require high control performance, such as wind energy generation, electric vehicle and CNC machines.
- Improve the FCS method with centralized switching frequency and optimized weighting factor.
- Study the induction motor control connected to multi-level inverter or buck-to-buck converter.
Appendix A

Motor Parameters Identification & Validation

In this thesis, the parameters of induction motor dynamic model are crucial for vector control methods, but the machine could accidentally contain uncertain parameters from production. The induction motor used in test-bed for the experimental operation is a medium-sized standard motor from SEW-Eurodrive, as shown in figure A.1, an encoder is included with shaft-mount. In this section, several tests are examined for parameter identification, then the identified parameters are validated by using MATLAB Simulink.

Figure A.1: Induction motor
A.1 Parameter Identification

The parameter identification of induction motor model has routine tests, which are found from literatures ([100],[101],[102]). There are three tests running for electrical model parameters: DC-test, No-load test and Rotor-block test, which are analysed as follows, then the first-order step response is test for obtaining the mechanical model parameters.

**DC-test**: By connecting two phases of the induction motor to a DC-voltage source, DC current flows through these two phase in series due to the Y connection of the stator winding, the impedance of the inductance of the stator winding is neglected because of the constant voltage supply. Therefore, the stator resistance is calculated by Ohm’s Law, ie. \( R_s = 11.2 \) Ohm.

**No-load test**: The purpose of no-load test attempts to make the rotor side of the equivalent circuit open-circuit, by operating the induction motor at its rated velocity, the speed difference between flux rotating and the rotor shaft rotating is minimized, so the slip approaches to zero, then the rotor load becomes \( \frac{R_r}{s} = \infty \). In this situation, the phase voltage and current is obtained for the stator resistance \( R_s \), stator leakage inductance \( L_{ls} \) and mutual machine inductance \( L_h \) as shown in Figure A.2. By measuring three components: phase voltage \( V_{ph} \), phase current \( I_0 \) and supplied power \( P \), then, after calculation, the mutual machine inductance is obtained in this test, \( L_h = 0.57 \) H.

![Figure A.2: Equivalent circuit of no-load test](image)

**Rotor-block test**: By preventing rotor moving when the power is supplied, the slip value is defined as one, ie \( s = 1 \), a short-circuit is formed at the rotor side of the equivalent circuit, as shown in figure A.3. Now keep increasing the phase voltage until the rated current is reached, again by measuring the power, voltage and current, the rotor resistance, stator and rotor leakage inductance are calculated. ie. \( R_r = 8.3 \) Ohm, \( L_{ls} = 0.0455 \) H and \( L_{lr} = 0.068 \) H.

**Mechanical parameter identification.** The mechanical model equation (2.30) is applied in this section, by defining zero load condition, the Laplace transform of the
Appendix A. Motor Parameters Identification & Validation

Figure A.3: Equivalent circuit of rotor-block test

The equation becomes:

\[(J_m s + f_d) \Omega_r(s) = \frac{3Z_p L_h}{2L_r} I_{sq}(s) \Psi_{rd}(s)\]

A first-order transfer function is obtained if the flux \(\Psi_{rd}(s)\) is controlled at constant value \(\psi_{rd}^0\) during steady state. Then, the transfer function of mechanical model is shown as follows

\[
\frac{\Omega_r(s)}{I_{sq}(s)} = \frac{K_{ss}}{\tau_M s + 1}
\]

where \(K_{ss} = \frac{3Z_p L_h \psi_{rd}^0}{2L_r J_m}\) and \(\tau_M = \frac{J_m}{f_d}\). By obtaining the step response of the velocity result, the values of steady-state gain \(K_{ss}\) and the time constant \(\tau_M\) are estimated, thus the inertia constant \(J_m\) and friction coefficient \(f_d\) are calculated as shown:

\[
J_m = \frac{3Z_p L_h \psi_{rd}^0}{2L_r K_{ss}}, \quad f_d = \frac{J_m}{\tau_M}
\]

The step response of velocity model is implemented using the inner-loop current control via two PI controllers, which is discussed in Appendix B. The experiment is implemented with rotor flux reference \(\psi_{rd}^* = 0.2\) Wb, the step change of the current \(i_{sq}^*\) is from 0.2 to 0.3A. The experimental results of the speed response and the current control are obtained and the calculation of the steady-state gain and time constant is processed as

\[
K_{ss} = \frac{\Delta \omega_r}{\Delta i_{sq}} \approx \frac{10.31}{0.1} = 103.1, \quad \tau_M \approx 2.2609
\]

Therefore, the mechanical model parameters is identified as the inertia constant \(J_m = 0.0052\) kg \(\cdot\) m\(^2\) and the friction coefficient \(f_d = 0.0023\) N \(\cdot\) m \(\cdot\) s.

A.2 Parameter Validation

The purpose of parameter validation is to determine the accuracy of identified parameter values, which contain certain error due to the measurement noise and reading errors. In this section, the motor parameters, which include both electrical and mechanical
parameters, are validated using Matlab Simulink SimPower toolbox as shown in Figure A.4, the procedure is defining the SimPower induction motor model with the previous identified parameters, then the motor is supplied at the rated condition, at the result, the line current and shaft velocity are observed to be close enough to the rated values on the nameplate of the given induction motor as shown in Table A.1.

![Motor parameter validation in Matlab Simulink](image)

**Figure A.4:** Motor parameter validation in Matlab Simulink

<table>
<thead>
<tr>
<th>Manufacture</th>
<th>SEW-EURODRIVE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>3-phase Inverter duty VPWM</td>
</tr>
<tr>
<td>Rated Power</td>
<td>750 Watts</td>
</tr>
<tr>
<td>Power Factor</td>
<td>0.79</td>
</tr>
<tr>
<td>Supply Frequency</td>
<td>50 Hz</td>
</tr>
<tr>
<td>Number of pole pairs</td>
<td>2</td>
</tr>
<tr>
<td>Rated Speed</td>
<td>1435 rpm</td>
</tr>
<tr>
<td>Rated Current</td>
<td>1.75 A</td>
</tr>
<tr>
<td>Rated Voltage</td>
<td>415 V</td>
</tr>
<tr>
<td>Connection</td>
<td>Y (star connection)</td>
</tr>
</tbody>
</table>

**Table A.1:** Name plant information of the experiment induction motor
Appendix B

Simulation & Experiment Setup

B.1 Simulation Setup

The simulators included in this thesis have been implemented using the MATLAB Simulink Toolbox. In this appendix, the detail of simulation implementation without the controller is introduced extensively, thus the open-loop operation of an induction motor will be build up in simulation.

Firstly, the induction motor model is simulated by using the Asynchronous Machine block in the SimPowerSystems Toolbox as shown in Figure B.1(a), then, the parameter definition of the block is presented in Figure B.1(b).

![Asynchronous Machine SI Units](image)

(a) Induction Motor Simulation Model  
(b) Parameter Setting of IM model

**Figure B.1:** Set-up of induction motor model block

The simulation of the induction motor open loop control is illustrated in Figure B.2. The model inputs are the stator voltages $u_{sd}$ and $u_{sq}$ and outputs are the stator currents and
mechanical measurements, instead of the estimation the position angle of \( dq \) coordinates, the field position is directly defined using a periodic triangular wave. After that, the general PWM block from Simulink is applied to generate the switching states for the 3-phase inverter, the semiconductors inside the inverter will transfer the DC bus voltage into the controlled two level and three phase voltage for the induction motor operation. Furthermore, the current measurement is achieved by measuring two-phase currents \( i_a \) and \( i_c \), then the third can be calculated using KCL (ie. \( i_b = 0 - i_a - i_c \)). The mechanical measurements, such as rotor speed and position, are directly provided by the Induction motor block as shown in Figure B.1(a).

In this thesis, the SimPower components in the simulator are generally sampled faster than the control system blocks. The sampling time is defined as 10\( \mu \)s, and the control system is sampled at its respective rate depending on the controller design. To ensure that the simulation set-up is closed to the actual experiment, the 3-phase inverter is defined to IGBT/Diodes and bridge arms number of 3. The switching frequency in PWM is defined as 2kHz for both simulation and experimental cases.

![Figure B.2: Open loop Simulator in Matlab Simulink](image)

**B.2 Experiment Setup**

The experiments in this thesis are implemented using MATLAB Simulink and realized using xPC Target, thus the experimental software is simply transformed from the simulation model. An illustration of the entire experimental test-bed is shown in Figure B.3, the detail of each component is introduced as follows.

*Controller:* The controller algorithm design and editing is accomplished using MATLAB Simulink in the Host Computer, which is connected to the xPC Target via a crossover cable. The xPC Target is another computer, which contains two extra boards: National
Appendix B. Simulation & Experiment Setup

Figure B.3: Test-bed of the induction motor control experiment

Instrument PCI-6024E and QUAD04, the former board is for control computation and the later one is for encoder data gathering. Once the control program is designed, the Host computer will compile and upload the program to the Target PC, which will operate the control experiment in real time mode.

Power Supply: The power supply is divided into two aspect: low voltage and high voltage. The former one is for the IGBTs of the Inverter at the low voltage side, which is defined at 24 V. Then, the high voltage supply, hence the DC-link bus voltage, is set at 520 V. Thus, the induction motor is supplied at its rated voltage of 415 V after the modulation.

Inverter: The inverter used in this test-bed is a three-phase two-level voltage-source inverter (2L-VSI). The inverter module has two layers, the first layer is a drive board, which reads the control commands from the controller to control the IGBTs, the second layer contains the high voltage side of the IGBTs, which has the dc-link bus capacitors and the protections, the IGBT module is attached to a piece of heat-sink.

Mechanical Load: As shown in Figure B.3, on the left hand wide of the test bench, a DC motor is coupled with the induction motor. In order to provide the safe load torque to the induction motor shaft, there are two connections at the terminal of DC motor. The first connection is illustrated in Figure B.3, where the DC motor terminal is connected to a high current power supply, in order to provide the desired load torque, this situation is applied when the position control system is evaluated. The other connection is designed for speed control, a power resistor, which has properties of low resistance
and high current capacity, will be connected at the DC motor terminal, so that during operation of the speed control system, there is current flow through the power resistors due to the re-generation mode of DC motor. Thus, the DC motor will provide the load torque at the opposite direction of the induction motor shaft. Moreover, the load torque will be increased when the induction motor speed is defined faster.

**Induction Motor and Sensors:** The induction motor used in this test-bed is from SEW-EURODRIVE, the data sheet of motor type DRE80M4 can be easily found in the their web-site. The motor information was introduced in Table A.1. On the other hand, the sensor components are encoder and current sensors. The encoder is built on the induction motor shaft, which is an incremental encoder with resolution of 1024. The two current sensors, for two phase currents measurement, are LTS-6-NP from REM company, which can handle the current measurement of ±6 A. Furthermore, the power supply of both encoder and current sensor are connected from the control board, in order to ensure the common ground.
Appendix C

Luenberger Observer

Since the Direct Field Oriented Control structure was implemented in this thesis, all the vectors in the control design are represented in the direct-quadrature (dq) coordinates. Therefore, an appropriate observer is used to estimate the position angle of the dq coordinates. In this appendix, the Luenberger observer is introduced and implemented referring to [13], the equations of the Luenberger observer is described as:

\[
\begin{align*}
\frac{d}{dt}\hat{x} &= \bar{A}\hat{x} + \bar{B}\bar{u} + \bar{K}(\bar{y} - \hat{y}) \\
\hat{y} &= \bar{C}\hat{x}
\end{align*}
\]

where \( \hat{x} \) is the estimated state, \( \bar{y} \) is the measured output, \( \hat{y} \) is the estimated output, \( \bar{A} \) is the state matrix, \( \bar{B} \) is the input matrix, \( \bar{K} \) is the observer gain matrix, \( \bar{C} \) is the output matrix.

Because the observer is to estimate the position angle of the d-q coordination, therefore, the design is based on the vectors in the stator fixed coordination (αβ). The state-space model of induction motor is characterized as follows:

\[
\frac{d}{dt}\begin{bmatrix} \bar{i}_s \\ \bar{\Psi}_r \end{bmatrix} = \begin{bmatrix} \bar{A}_{11} & \bar{A}_{12} \\ \bar{A}_{21} & \bar{A}_{22} \end{bmatrix} \begin{bmatrix} \bar{i}_s \\ \bar{\Psi}_r \end{bmatrix} + \begin{bmatrix} \bar{B}_1 \\ 0 \end{bmatrix} \bar{u}_s
\]

\[
\bar{i}_s = \bar{C}\bar{x} = \begin{bmatrix} I & 0 \end{bmatrix} \begin{bmatrix} \bar{i}_s \\ \bar{\Psi}_r \end{bmatrix}
\]

\[
\bar{A}_{11} = -\begin{bmatrix} \frac{R_s}{\sigma L_s} + \frac{1-\sigma}{\sigma \tau_r} & 0 \\ 0 & \frac{R_s}{\sigma L_s} + \frac{1-\sigma}{\sigma \tau_r} \end{bmatrix} = a_{r11}\bar{I}
\]

\[
\bar{A}_{12} = \frac{L_h}{\sigma L_s L_r} \begin{bmatrix} \frac{1}{\tau_r} & \omega_r \\ -\omega_r & \frac{1}{\tau_r} \end{bmatrix} = a_{r12}\bar{I} + a_{i12}\bar{J}
\]
Appendix C. Luenberger Observer

\[ \bar{A}_{21} = \begin{bmatrix} \frac{L_h}{\tau_r} & 0 \\ 0 & \frac{L_h}{\tau_r} \end{bmatrix} = a_{r21} \bar{I} \]

\[ \bar{A}_{22} = \begin{bmatrix} -\frac{1}{\tau_r} & -\omega_r \\ \omega_r & -\frac{1}{\tau_r} \end{bmatrix} = a_{r22} \bar{I} + a_{i22} \bar{J}; \]

\[ \bar{B} = \begin{bmatrix} \frac{1}{\sigma L_s} & 0 \\ 0 & \frac{1}{\sigma L_s} \end{bmatrix}, \bar{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \bar{J} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \]

The observer determines the feedback by the stator fixed coordination current error between the actual measurement and the calculated estimation. The observer gain matrix is designed according to the eigenvalues of the model:

\[ \det(s \bar{I} - \bar{A} + \bar{K} \bar{C}^T) \]

by using the pole assignment method with a constant \( k \) referring to \([103]\), the tuning parameter \( k \) defines the location of the eigenvalues, so that, the observer gain matrix is defined as:

\[ \bar{K} = \begin{bmatrix} g_1 & -g_2 \\ g_2 & g_1 \\ g_3 & -g_4 \\ g_4 & g_3 \end{bmatrix} \]

where \( g_1 = (k - 1)(a_{r11} + a_{r22}), g_2 = (k - 1)a_{i22}, g_3 = (k^2 - 1)(ca_{r11} + a_{r21}) - c(k - 1)(a_{r11} + a_{r22}), g_4 = -c(k - 1)a_{i22}, c = \frac{\sigma L_s L_u}{L_h} \).

Since the observer model is time varying depending on the rotational rotor speed. Refer to \([104]\), the estimation of the rotational rotor speed \( \omega_r \) is possible based on the concept of the Lyapunov’s stability theorem. Finally, the following equation is analyzed for rotor speed estimation:

\[ \hat{\omega}_r = K_p(e_{isa \Psi_{r,3}} - e_{isa \Psi_{r,1}}) + K_I \int (e_{isa \Psi_{r,3}} - e_{isa \Psi_{r,1}}) dt \]

By choosing high values for the tuning parameters \( K_p \) and \( K_I \), the speed adaption algorithm will converge fast in the power electronic environment.
Bibliography


time controller for PWM inverter-fed induction motor drives. In *Fifth European

without rotational transducers. *IEEE Transactions on Industry Applications*, 28

1993.

[14] J. Holtz. Sensorless control of induction machines - with or without signal injec-

motor drives. In *Third International Conference on Power Electronics and


Predictive control in power electronics and drives. *IEEE Transactions on Industrial


[21] C.R. Cutler and Ramaker B.L. Dynamic matrix control - a computer control


1972.
application to a batch reactor process. In *AIChE Annual Meeting*, San Francisco,
1984.

saturation and plant uncertainty part 1. robustness. *International Journal of

class of constrained discrete-time systems: Stability and moving-horizon approxi-

[28] V. Nevistic and J.A. Primbs. Finite receding horizon linear quadratic control: A
1997.

[29] A. Bemporad, L. Chisci, and E. Mosca. On the stabilizing property of the zero

Based Process Control*, volume 293 of *NATO ASI Series*, pages 367–396. Springer


1993.

[33] H. Chen and F. Allgower. A quasi-infinite horizon nonlinear model predictive

[34] S.L. de Oliveira Kothare and M. Morari. Contractive model predictive control for


Zurich, 1997.


