Information Driven Control Design: A Case for PMSM Control

Han Zhang

Follow this and additional works at: https://engagedscholarship.csuohio.edu/etdarchive
Part of the Electrical and Computer Engineering Commons
How does access to this work benefit you? Let us know!

Recommended Citation
Zhang, Han, "Information Driven Control Design: A Case for PMSM Control" (2017). ETD Archive. 1026.
https://engagedscholarship.csuohio.edu/etdarchive/1026

This Dissertation is brought to you for free and open access by EngagedScholarship@CSU. It has been accepted for inclusion in ETD Archive by an authorized administrator of EngagedScholarship@CSU. For more information, please contact library.es@csuohio.edu.
INFORMATION DRIVEN CONTROL DESIGN:
A CASE FOR PMSM CONTROL

HAN ZHANG

Bachelor of Engineering in Mechanical Engineering and Automation
Tsinghua University
June 2010

Master of Science in Electrical Engineering
Cleveland State University
December 2011

submitted in partial fulfillment of the requirements for the degree

DOCTOR OF ENGINEERING

at

CLEVELAND STATE UNIVERSITY
DECEMBER 2017
We hereby approve this dissertation for
Han Zhang
Candidate for the Doctor of Engineering degree for the Department of
Electrical Engineering and Computer Science
And
CLEVELAND STATE UNIVERSITY’S
College of Graduate Studies by

Dissertation Committee Chairperson, Dr. Zhiqiang Gao

Department and Date

Dissertation Committee Member, Dr. Ana Stankovic

Department and Date

Dissertation Committee Member, Dr. Lili Dong

Department and Date

Dissertation Committee Member, Dr. Hanz Richter

Department and Date

Dissertation Committee Member, Dr. Sally Shao

Department and Date

Dissertation Committee Member, Dr. Hui Tan

Department and Date

December 7th, 2017
Student’s Date of Defense
ACKNOWLEDGEMENTS

First, I would like to give the special thanks to my advisor, Dr. Zhiqiang Gao, for his unconditional support on both my academic development and personal life. During the past seven years, his immense knowledge and passion for research have inspired me greatly. I would not accomplish the current understanding of control system design and finish this dissertation without his guidance, stimulating discussions, continuous encouragement and patience. I’m also truly grateful to Dr. Ana Stankovic, Dr. Lili Dong, Dr. Hanz Richter, Dr. Sally Shao and Dr. Hui Tan for serving on my committee, sharing their precious time and providing helpful suggestions.

In addition, I want to thank my labmates Qing Zheng, Gang Tian, Shen Zhao, Qinling Zheng, Jason Tatsumi, Yu Hu, Xuaoxu Li and Lei Wang for their friendship and support. Moreover, I want to thank my collaborators on various research projects for their insights and suggestions, including but not limited to Jack Zeller, Hui Xie, Gregg Calhoun, Jack Redilla, Tao Cai, Bin Zhu, Lei Wang, Wencao Xue, Kang Song and Sen Chen. Thanks also go to senior design students Vlad Shtin, David Sadey and Shubo Zhang for the opportunity to share a great experience. I will not forget my friends Xin Hui, Jingjing Wang, Mingjie Tang, Yang Zhao, Zao Liu and Miao Wang for their help in my personal life. Special thanks go to my internship supervisors Dr. Hui Tan from Texas Instruments for his inspiring discussion and sharing industrial problems and the leading solutions with me.

My deepest gratitude goes to my beloved wife Zeyuan Zhang for her support, encouragement, tolerance and unwavering love. Thanks to my lovely son James Zhang, my parents, my parents in law and other family members. It is their help and expectation that supports me to go through this challenging journey and finally finish the program.
INFORMATION DRIVEN CONTROL DESIGN:  
A CASE FOR PMSM CONTROL

HAN ZHANG

ABSTRACT

The key problem in control system design was the selection and processing of information. The first part was to collect some system dynamics offline or online in a cost-effective manner and use them in the controller design effectively. Next was to minimize the phase lag in the feedback loop to ensure best performance and stability. A systematic information-driven design strategy was discussed. A few key problems in permanent magnet synchronous motor control were taken in a case study: the current loop and decoupling, velocity loop with position feedback and position estimation at low speed. An active disturbance rejection based integrated current loop control solution was presented. Some implementation problems were also discussed: restructuring of active disturbance rejection control for implementation, scaling of extended state observer in fixed-point implementation and observer-based parameter estimation. The proposed methods were tested in simulation and hardware experiments.
# TABLE OF CONTENTS

ABSTRACT ......................................................................................................................................................................................... v  

LIST OF TABLES .................................................................................................................................................................................. x  

LIST OF FIGURES ................................................................................................................................................................................ xi  

I INTRODUCTION ............................................................................................................................................................................. 1  

1.1 An Information-Driven Design Strategy ................................................................................................................................. 2  

1.2 Research Topics in PMSM Control ............................................................................................................................................. 6  

1.2.1 Background on PMSM ............................................................................................................................................................ 6  

1.2.2 Existing Control Framework of PMSM - Field Oriented Control ........................................................................................ 9  

1.2.3 Velocity control with position feedback .............................................................................................................................. 13  

1.2.4 Rotor Position Estimation ...................................................................................................................................................... 14  

1.3 Dissertation Organization ............................................................................................................................................................ 14  

II RESEARCH PROBLEMS ................................................................................................................................................................. 16  

2.1 Practical Implementation Issues .................................................................................................................................................. 16  

2.2 PMSM Control Issues ................................................................................................................................................................. 18  

2.2.1 Current Regulation with FOC .............................................................................................................................................. 18  

2.2.2 Velocity Control with Position Feedback ........................................................................................................................... 19  

2.2.3 Rotor Position Estimation ...................................................................................................................................................... 19  

2.3 Literature Review .......................................................................................................................................................................... 20  

2.3.1 A Historical View of ADR ................................................................................................................................................. 21  

2.3.2 Practical Implementation Issues ......................................................................................................................................... 22  

2.3.3 PMSM Control Design Methods ..................................................................................................................................... 24  

2.3.4 Rotor Position Estimation in PMSM ................................................................................................................................ 27  

III PRACTICAL IMPLEMENTATION OF ADRC ................................................................................................................................. 29
3.1 Parameter Estimation .................................................................................... 30
3.2 Incorporating Offline Information in ESO Design ........................................ 35
3.3 Error-Based ADRC ........................................................................................ 37
  3.3.1 First-Order System ................................................................................... 39
  3.3.2 Second-Order System ............................................................................. 41
  3.3.3 Simulation Verification ............................................................................ 44
3.4 Approximate Differentiation with TD ........................................................... 45
3.5 A New Profile Generator Based on the Modified Tracking Differentiator .... 48
3.6 Scaling for the Fixed-Point Implementation .................................................. 49

IV FREQUENCY CHARACTERISTICS .................................................................. 52
  4.1 Phase and Delay Margin of Generic ADRC .................................................. 53
    4.1.1 First-Order System ................................................................................ 53
    4.1.2 Second-Order System ......................................................................... 55
    4.1.3 Higher Order System ......................................................................... 58
  4.2 Phase and Delay Margin of the Custom Designed ADRC ............................ 62
  4.3 Simulation Verification of the Custom Designed ADRC ............................... 66

V CURRENT AND VELOCITY REGULATION DESIGN ....................................... 70
  5.1 PMSM Characteristics .............................................................................. 71
  5.2 Current Loop Design .............................................................................. 73
    5.2.1 Keep all Dynamics in Total Disturbance .............................................. 74
    5.2.2 Remove Some Known Dynamics from Total Disturbance ................... 78
    5.2.3 Remove all Known Dynamics from Total Disturbance .................... 82
  5.3 Velocity Loop Design ............................................................................. 85
  5.4 Simulation Verification ............................................................................. 88
    5.4.1 Current loop Verification .................................................................... 88
5.4.2 Velocity Loop Verification .......................................................... 92
5.5 Hardware Validation ........................................................................ 94

VI POSITION ESTIMATION AT LOW SPEED ........................................... 97
6.1 The Fundamental Idea of Back-EMF Observer Based Sensorless Control .... 97
6.2 The Proposed ESO Based Sensorless Estimator ...................................... 98
6.3 Simulation Verification ........................................................................ 100
6.4 The Analysis of Sliding Mode Observer with Small Error ...................... 101

VII CONCLUSIONS AND FUTURE WORK ........................................... 104
7.1 Conclusions ..................................................................................... 104
7.2 Future Work ..................................................................................... 105

BIBLIOGRAPHY ..................................................................................... 106

APPENDICES ....................................................................................... 121
A. Design and Build of Hardware Test Bed ............................................. 121
A.1 Power Subsystem ............................................................................. 124
A.1.1 Power Entry ............................................................................... 124
A.1.2 Power Inverter .......................................................................... 126
A.1.3 Power Switch and Driver ............................................................. 126
A.2 Control Subsystem .......................................................................... 129
A.2.1 The MCU Controller ................................................................. 129
A.2.2 Current Feedback ...................................................................... 129
A.2.3 Position Feedback ..................................................................... 130
A.2.4 Command and Data Acquisition System .................................... 132
A.2.5 Power Supply ............................................................................ 133
A.2.6 Motor and Load ........................................................................ 135
A.3 MCU Software Architecture ............................................................ 136
A.3.1. CPU 1 Boot Sequence ................................................................. 137
A.3.2. Software Flow of CPU 1 Main Function ..................................... 138
A.3.3. Software Flow of Main Hwi ...................................................... 140
A.3.4. Software Flow of Torque Swi .................................................... 140
B. Schematics of Motor Control Test Bed .............................................. 142
B.1. MCU and Sensor Board Schematics .............................................. 142
B.2. High Voltage Board Schematics .................................................... 155
B.3. Power Supply Board Schematics ................................................... 160
LIST OF TABLES

Table 1 Parameters of Motor in Simulation................................................................. 88
Table 2 Parameters of Motor in Hardware Experiment................................................. 95
LIST OF FIGURES

Figure 1  Stator Current Space Vector and Its Component in (a,b,c) ................................................. 10
Figure 2  Stator Current Space Vector and Its Components in the Stationary Reference Frame .......................................................... 11
Figure 3  Stator Current Space Vector and Its Component in (α, β) in the d,q Rotating Reference Frame ........................................................................................................ 12
Figure 4  Basic Scheme of FOC for AC Motor ...................................................................................... 13
Figure 5  Block Diagram of a Estimation .............................................................................................. 31
Figure 6  a Estimation Result ................................................................................................................ 31
Figure 7  a Estimation Result with Different b Value ........................................................................ 32
Figure 8  Block Diagram of b Estimation .............................................................................................. 34
Figure 9  b Estimation Result ................................................................................................................ 35
Figure 10  Block Diagram of Output vs. Error Based ADRC .............................................................. 44
Figure 11  Output vs. Error Based ADRC ............................................................................................ 45
Figure 12  TD vs. Linear Approximation ............................................................................................. 47
Figure 13  TD vs. Linear Approximation With Noise ......................................................................... 47
Figure 14  Block Diagram of Reformulated System ............................................................................. 54
Figure 15  Performance of the Conventional and Custom Designed ADRC ....................................... 69
Figure 16  ADRC vs. PI plus Decoupling ............................................................................................ 90
Figure 17  ADRC vs. PI plus Decoupling Zoom-In ............................................................................ 90
Figure 18  ADRC vs. PI plus Decoupling with Parameter Uncertainty ............................................... 91
Figure 19  ADRC vs. PI plus Decoupling with Parameter Uncertainty Zoom-In ................................ 92
Figure 20  Velocity Control with Load Disturbance .......................................................................... 93
Figure 21  Velocity Control with Load Disturbance Zoom In ............................................................ 94
Figure 22  Hardware Experiment Result ............................................................................................ 96
Figure 23  Simulation Result of ESO Based Position Estimation ...................................................... 100
Figure 24  Simulation Result of ESO Based Position Estimation ......................... 101
Figure 25  Block Diagram of the Enhanced Sliding Mode Observer ..................... 102
Figure 26  Picture of Motor Drive Rack ............................................................. 122
Figure 27  Picture of the Power Inverter and Controller Tray ............................. 123
Figure 28  Block Diagram of Power Supply and Protection Module .................... 125
Figure 29  Photo of Power Supply and Protection Panel ..................................... 125
Figure 30  Schematic of the Voltage-Sensing System ........................................... 126
Figure 31  Schematic of PM50RL1A060 IGBT IPM ............................................. 127
Figure 32  Schematic of IPM Interface Circuit .................................................... 128
Figure 33  Schematic of the Current Sensor ....................................................... 130
Figure 34  Schematic of the current signal conditioning system ......................... 130
Figure 35  Schematic of the Sin/Cos Encoder Feedback ..................................... 131
Figure 36  Schematic of the RS 485 Feedback .................................................... 132
Figure 37  Schematic of ±15 V in Power Supply Version 1 ............................... 133
Figure 38  Schematic of Power Supply Power Entry .......................................... 134
Figure 39  Schematic of 5 V Power in Supply Version 2 ..................................... 134
Figure 40  Schematic of ±15 V in Power Supply Version 2 ............................... 135
Figure 41  Picture of Hardware Test Bed ............................................................ 136
Figure 42  Boot Sequence of CPU 1 ................................................................. 138
Figure 43  Software Flow of CPU 1 Main Function ............................................ 139
Figure 44  Software Flow of Main Hwi .............................................................. 140
Figure 45  Software Flow of Torque Swi ......................................................... 141
Chapter I

INTRODUCTION

Cybernetics was conceived with both control and communication in mind by Norbert Wiener in his seminal book of 1948 [1], according to which animals and machines share the same mechanisms of control and information collection, transfer and processing. Six years later, Hsue-Shen Tsien provided the engineering framework in his book, Engineering Cybernetics [2].

Classical control theory focuses on the problem of stability of feedback loop, with the tools borrowed from the telecommunication theories, such as Bode plot, Nyquist plot and root-locus. However, there are many unique problems in control engineering problems, such as how to select, acquire and handle the information needed to meet the final design target, which cannot be fully resolved with the existing feedback theory.

The modern control theory, on the other hand, was developed primarily by mathematicians with the assumption that the detailed mathematical model of the target system was given [3]. Many advanced design and optimization methods were then developed rigorously based on the mathematical model. Although some techniques were developed afterward to deal with the uncertainties in system dynamics, such as robust control in the framework of $H_\infty$ control, the amount of model uncertainty to be
contended with is still very limited [4, 5]. In fact, in both the classical and modern control framework, the role of information was never fully articulated.

Inspired by active disturbance rejection control, an information-driven control system design strategy was explored and experimented in this dissertation. Unlike the traditional PID and the modern model-based design, the focus here is on how to select, measure, estimate and utilize the available information, both online and offline, to achieve the best result. A case study in permanent-magnet synchronous motor control is used as illustrations.

1.1 An Information-Driven Design Strategy

The core problem in control is to make the output of a target system to follow the reference, i.e. command, by manipulating its input. The target system usually has a variety of inescapable imperfections such as noise, transient delay, sensor and actuator limit and limited sampling rate. In addition, the reference must be chosen appropriately with respect to the physical limitations and costs.

Classical control theory focuses on the stability of feedback loops using the frequency response methods such as Bode plot, Nyquist plot and root-locus. Those tools were borrowed from communication engineers in the 1930s who were concerned with the stability of negative feedback amplifiers. However, instead of making the output of a pre-designed and fixed target system to follow a particular reference signal, the communication engineers’ problem is to make high gain amplifiers linear in a certain frequency range for arbitrary input. In others words, in control the engineers are concerned with how well the output follows the command; in communication, however, the concern is squarely focused on the linearity of the input and output relationship. Since the design goals are different, so are the theory and methods. Unfortunately, such difference went largely unnoticed and the theory of feedback by the communication
engineers was taken as a theory of control after the WWII, in what is known as classical control theory [6].

Modern control theory (MCT) was conceived in the late 1950s, mostly by mathematicians, with the dual focus of stability and optimality [7]. Premised on the assumption that accurate mathematical model of the target system is given, MCT thrived during the cold war as an academic discipline but failed to penetrate industry practice, which is still dominated by PID and trial-and-error tuning. The reason is the fundamental incompatibility between the MCT premise and the industry practice. Since most industry systems are complicated, nonlinear, time-variant and full of uncertainties, they defy the precise the mathematical description assumed by theoreticians [3]. As a result, the design methods advocated by the so-called advanced control theories were rarely implementable in industrial settings [8]. Although some methods, such as robust control and $H_\infty$ control, was developed to handle the model uncertainties, the tolerance was normally very limited [4, 5].

In addition, the Lyapunov stability analysis, the gold standard in MCT, was originally developed in astronomy [9] and was concerned with the motion of the planets as time goes to infinity, it paid little attention to the transient response of control system, which was the most important part of most control system designs. The design for optimality, on the other hand, is only optimal with regard to the given mathematical model, not the underlying physical system. For any particular system, a small uncertainty in the system dynamics could invalid both the stability and optimality claims in the MCT. It is therefore concluded that general solutions of industrial control lie beyond the scope of MCT. Instead of stability and optimality, perhaps the problem of control is really the problem of information.

In particular, there are two kinds of information used in control system: online and offline. The offline information includes known system dynamics, knowledge of the
reference signal and the known system characteristics, such as fixed system delays. This information can either be derived from the law of physics or gathered from the field knowledge. Data-driven methods were often used to estimate some system parameters.

Online information mainly comes from the reference signal, the control signal and the feedback signal. The most common online information is obtained via feedback, of which the biggest challenge is to deal with the phase lag since phase lag leads to instability and poor dynamic performance in the closed-loop system. The system bandwidth has to be carefully chosen as the tradeoff between performance and stability margin. Higher bandwidth often leads to better performance but smaller stability margin [7].

In practice, engineers found that one way to achieve higher bandwidth on critical and fast control loops is to use the cascade control loops, where fast-changing dynamics forms the inner closed loop, in addition to the output feedback loop. The reference signal of this inner loop is generated by the outer loop. Since the inner loop has very fast dynamics and less phase lag, fast response can be achieved with high bandwidth in the inner loop to improve system dynamics and make the output tracking easier to design [10, 11]. An alternative approach is to reduce the effective phase lag by leveraging feedforward and reference information in the control system [12]. This method can improve the transient response without increasing the system bandwidth.

Hsue-Shen Tsien pointed out that the solution to the complexity and uncertainty problem is to get information of system dynamics online with continuous sensing and measuring [2]. Another way of using online information was to revise the system knowledge used in controller continuously with the input and output data to the system. Since this is often done at a much slower pace compared to the main feedback loop, lower bandwidth could be used and larger phase lag could be tolerated.
Apart from direct measurement, online information can also be collected through estimation. MCT provides a great tool, the state observer, with which all states in an observable system can be estimated from the input and output information. Another example was the sensorless control of the three-phase motor, where the rotor position is estimated from the voltage and current information [13, 14]. Active disturbance rejection (ADR) is an example of information-driven design strategy, where the offline information estimation, online system knowledge update and use of online information in feedback and feedforward could all be addressed together organically with the cascade integral form and the extended state observer.

In the framework of ADR, control problems are reformulated as disturbance rejection problems, where all the model uncertainties, unknown dynamics and external disturbances as a total disturbance and made an extended state in the reformed system. The total disturbance can then be estimated online with a state observer, i.e., the extended state observer (ESO). With the help of ESO, the information of unknown system dynamics, system uncertainties and external disturbances can be extracted online form the input and output signal of the system as an equivalent input disturbance. The effect of which can then easily be compensated in the control signal.

In a second-order system, for example, the phase characteristics of ADR controller, the disturbance cancellation action leads the conventional feedback based correction by 180 degrees ideally, when the total disturbance spectrum is much lower than the observer bandwidth. In practice, however, since the observer bandwidth is limited by many factors, such as the sampling rate, system noise and quantization, this phase advantage may be rather limited. The offline information on system dynamics can be used in the ESO design to improve its online signal processing performance with limited bandwidth.
In this dissertation, some practical topics for ADR based information-driven design will be discussed, regarding online and offline information selection, extraction and processing. The proposed solutions were then applied to some key problems in PMSM control as a case study.

1.2 Research Topics in PMSM Control

The ADR based information-driven design strategy is applied to some research topics in Permanent Magnet Synchronous Motor (PMSM) as a case study in this dissertation. First, the field oriented control (FOC) of PMSM current is discussed. The velocity control with position feedback is covered after that. Finally, the motor shaft position estimation is discussed.

1.2.1 Background on PMSM

An electric motor is a device that converts electrical energy into mechanical energy creating either a rotary torque or a linear force. There are two main categories of motors, direct current (DC) motors and alternating current (AC) motors. A DC motor has a constant magnetic field, which is generated by either permanent magnet or stator coil winding excited with DC, and an armature with two or more windings (poles). The current feed to the rotor is alternated with a commutator to maintain a constant torque direction when the winding is rotating with respect to the magnetic field. Nowadays, DC motors still dominate the market for low power and low voltage applications for its low cost and easy to drive. On the other hand, the current feed into an AC motor alternates direction by itself, which eliminates the need for a commutator. AC motors are widely used in higher voltage, torque or power applications, for its simplicity in both assembly and maintenance [15].
AC motors can then be divided into two main types: synchronous motors and asynchronous motors. The synchronous motor is an AC machine where the electric rotation of the shaft is synchronized with the rotation of the electromagnetic field in the stator caused by the rotation of supply current through the coil during steady-state operation. On the other hand, asynchronous motors relay on the rotor current induced by the speed difference between the rotor and stator field to generate torque [15].

In the early ages, synchronous motors are usually electromagnetically excited, that is, two electrical inputs need to be fed to the motor: one AC to the stator and one DC to the rotor. Modern designs, such as PMSMs, permanent magnets (PM), which is mounted on the surface or injected into the rotor magnetic core material, are often used to create a constant rotor magnetic field with a constant magnitude.

Although the first PM excitation motor system can be traced to as early as 19th century [16], most electric motors were using electromagnetic excitation due to the poor quality of the hard magnetic materials available back then. This fact was changed after the new magnet material Alnico was discovered in 1932. Comparing to other materials, Alnico has higher flux density, which can induce higher torque output and higher coercive force, helping the magnet resist demagnetization from armature reaction effect [15]. This made it possible to replace the electromagnetic excitation field with PM, yet only limited to little horsepower DC commutator machines.

Although AC induction motors dominated the industry in the 20th century, more and more brushless solutions with PM excitation have been used in servo systems with the help of the invention of high-performance rare-earth magnetic material. This new magnetic material had replaced the ferrite and Alnico magnets and improved the power mass ratio, dynamic performance and efficiency of PM machines tremendously [15].

Using PM brings the following benefits to motor design: first, efficiency is increased by eliminating the excitation field energy losses; second, higher torque and
power output can be packed into a small volume; third, better dynamic performance is obtained due to higher magnetic flux density in the air gap; and last but not least, the construction and maintenance cost is reduced [17].

Traditional synchronous motors are designed to work at a constant speed, the synchronous speed, determined by the number of poles in the motor and the AC line frequency. A conventional synchronous motor could not start itself and need to be driven to near synchronous speed by a separate torque. Same as a double-excited synchronous machine, PMSMs also need to be driven to near synchronous speed to lock into the synchronous mode [15].

Modern power electronics technology makes it possible to vary the frequency of stator current, thus the synchronous speed of a motor, from a value close to zero all the way to beyond its rated value. This allows the motor to run in synchronous mode throughout the entire operation period, by keeping the rotational speed of the stator magnetic field the same with the rotor magnetic field. Failed to run the motor in synchronous mode may result in non-optimal torque output efficiency or cause noise and vibration in the mechanical system. The rotor may even stop rotating and output zero torque in some situations known as “pull out” when not being driven properly [15].

In order to maximize the motor torque output, the angle between rotor and stator magnetic field should be maintained at 90 degrees. Therefore, the knowledge of the position of the rotor relative to stator magnetic field is essential in PMSM control [15]. Resolvers or high-resolution encoders with interfaces providing absolute position information such as Synchronous Serial Interface (SSI), Bi-directional Synchronous Serial Interface (BiSS), Hiperface and EnDat are used in high-end industrial solutions nowadays to obtain shaft angle [18]. Due to the high cost of these sensors and the increased risk of failure, “sensorless” solutions such as back electromotive force (back-EMF) observer and high frequency injection (HFI), where the rotor position is estimated
from the voltage and current information of stator windings with an observer, have been widely investigated in recent years [14, 19, 20]. Commercial solutions such as InstaSPIN-FOC from Texas Instruments have also been released [21, 22].

1.2.2 Existing Control Framework of PMSM - Field Oriented Control

In the early years, AC motors were only controlled by varying the frequency and magnitude of the AC voltage simultaneously by keeping a fixed magnitude to frequency ratio. No torque or current control was done like in DC motor control. To get better dynamic performance and efficiency, a more advanced technique, field-oriented control (FOC), also called vector control, was introduced in the early 1980s, with the help of the progress of digital signal processing (DSP) chips, and dominated the market since then. The use of FOC made it possible to control the current and torque of AC motors as was done in DC motors [15].

The separation between flux and torque control was embedded in the structure of traditional double excited DC machines. The current through the field excitation coil controls the flux and the current through the rotor coils controls the torque output. The stator and rotor magnetic fields are kept near perpendicular, by using the mechanical commutator and having multiple pairs of poles, which yields to nearly maximum torque output and power efficiency [15].

However, this is no longer the case in AC machines where the only source to control is the current through the stator windings, which have both torque producing and flux magnetizing components. FOC solves this problem by separating the two elements in the AC with vector projection technique and controlling the two current vectors individually. Be aware that the two current vectors have internal coupling in their dynamics although they are in orthogonal orientation.
In the PMSM situation, the field excitation current is usually driven to zero for surface-mounted PMSMs for maximum efficiency, since the rotor flux has already been provided by the PM. For interior permanent magnet synchronous motor (IPMSM), a non-zero field excitation current may be needed. From the knowledge of electromagnetic laws, we know that the maximum torque will be induced when the rotor and stator flux are orthogonal. Rotor position acquired from either position sensor or position observer is used to ensure this condition in FOC drives. For situations seeking high speed, some negative magnetizing current can be used to weaken the rotor field to lower the back-EMF constant, which allows torque output under higher speed [15].

Under the FOC framework, the three phase voltages and currents are described in space vector form. In term of current, the space vector representation is defined by

\[ \vec{I} = i_a + \alpha i_b + \alpha^2 i_c \]  

where \( i_a, i_b, \) and \( i_c \) are stator phase currents, \( \alpha = e^{\frac{2\pi}{3}} \) and \( \alpha^2 = e^{\frac{4\pi}{3}} \) represents the spatial operators as shown in Figure 1.

![Figure 1 Stator Current Space Vector and Its Component in (a,b,c)]
The space vector can then be represented in a reference frame with two orthogonal axes $\alpha$ and $\beta$ where $\alpha$ axis is in the same direction as $a$ axis as shown in Figure 2.

![Figure 2 Stator Current Space Vector and Its Components in the Stationary Reference Frame](image)

The projection, also called Clark transformation, could be represented by the following expression

\[
\begin{align*}
    i_\alpha &= i_a \\
    i_\beta &= \frac{1}{\sqrt{3}} i_a + \frac{2}{\sqrt{3}} i_b
\end{align*}
\]  

(2)

with current $i_\alpha$ and $i_\beta$ being the projected currents in $\alpha$ and $\beta$ axes.

Then comes the most important part in FOC, Park transformation, where $\alpha$ and $\beta$ currents are projected to a rotating orthogonal axis frame as shown in Figure 3 where $d$ axis, aligned with the rotor flux, stands for the rotor direct axis and $q$ axis represents the rotor quadrature axis. This transformation can be obtained from the following formulas

\[
\begin{align*}
    i_d &= i_\alpha \cos \theta + i_\beta \sin \theta \\
    i_q &= -i_\alpha \sin \theta + i_\beta \cos \theta
\end{align*}
\]

(3)
where $\theta$ is the relative position of the rotor flux with respect to $\alpha$ axis, the $d$ axis current $i_d$ is the magnetization component of the stator current and $q$ axis current $i_q$ is the torque generation component [15].

![Stator Current Space Vector and Its Component in (α, β) in the d,q Rotating Reference Frame](image)

**Figure 3 Stator Current Space Vector and Its Component in (α, β) in the d,q Rotating Reference Frame**

In FOC control scheme, two out of the three phase currents are measured with sensors and sent through the Clark and Park transformations to get the $d$ and $q$ axis currents. Two current regulators are used to drive the two currents to their desired values by manipulating the $d$ and $q$ axis voltages respectively. The two voltages then go through the inverse version of Park and Clark transformations and the three-phase voltage commands are generated. Then space vector PWM signals are generated accordingly [15]. A basic schematic of this control strategy is shown in Figure 4.
1.2.3 Velocity control with position feedback

Velocity control is the most common problem in traction control. Since the position information is required in three-phase motor drives, position feedback is likely already available. Unlike most analog based speed sensor, position sensor does not need calibration and are immune to most electromagnetic interference, which is common in most industrial environments. As a result, most motion control systems use position feedback to close the speed loop.

The most commonly used solution for velocity control with position feedback was to calculate the velocity with approximate differentiation methods and close the speed loop with it. However, since most effective approximate differentiation methods are second-order and the differentiator bandwidth is limited by the resolution of the position signal, this calculation introduces a large phase lag into the feedback loop.
1.2.4 Rotor Position Estimation

As is discussed in earlier chapters, the rotor position is required to implement vector control in a PMSM drive. Traditional solutions typically obtain this information with a position sensor such as encoder, resolver, or Hall effect sensor. However, it is possible to eliminate the need of a position transducer to reduce cost and increase reliability, since the shaft position information is embedded in the phase voltage and current signal [14, 19, 20].

Since the direction of the Back-EMF voltage direction was fixed in the rotating reference frame, the most common way to estimate the motor shaft position was to estimate the Back-EMF voltage angle from the voltage and current signal. As the Back-EMF voltage was proportional to the motor speed, this signal will be very small when the motor is running at low-speed. With the quantization error and signal noise in the real system, the position estimation at low speed will be very noisy.

1.3 Dissertation Organization

This dissertation is organized as follows. Chapter 1 gives an introduction to the information-driven design strategy we are going to discuss in this dissertation along with the research topics in PMSM control. The main topics discussed in this dissertation is stated in Chapter 2 along with some literature review. Chapter 3 presents the main results on the practical implementation issues. Chapter 4 discusses some frequency characteristics of ADRC with and without partial system information. A case study was given in the next two chapters with the PMSM control problems. The ADR based design for current loop and velocity loop control of PMSM was discussed in Chapter 5, along with some simulation verification and hardware experiment results of the proposed solutions. Then a solution for shaft position estimation of PMSM is given in Chapter 6. Finally, the concluding remarks and some future works are discussed in Chapter 7. A
three-phase motor control test bed, which contains a Texas Instruments C2000 series microcontroller-based controller, a communication system for data acquisition, the necessary protection and isolation circuits and the power electronics circuits of a motor drive, designed and built for this research is then given in the appendix.
Chapter II

RESEARCH PROBLEMS

We will first discuss some interesting topics in ADR based controller implementation and deployment regarding online and offline information extraction and processing that we will study in this dissertation. Then, we will discuss some open problems in PMSM current and velocity control under various conditions. Then some challenging issues in the shaft position estimation of PMSM will be discussed. Finally, a literature review of the topics is given in the last part of the chapter.

2.1 Practical Implementation Issues

In this section, we will discuss some practical implementation issues of ADR based information-driven controller.

Although ADR based controller design does not require a detailed mathematical model, some basic information is still necessary. In addition, knowing some offline information on system dynamics and use them in the controller design can greatly increase the performance of the controller and reduce the bandwidth requirement. So, estimating some system parameters under the ADR framework is important.
There are two main methods of using offline information in controller design under the ADR framework. The first way is to stick with the cascade integral model, wrap all known and unknown dynamics into the total disturbance and use the offline information to help the ESO in online information processing. The second way is to modify the system model used in ESO design with the known system dynamics leaving only unknown disturbance to be estimated.

Many times, we will need to deploy a controller in an existing industrial controller that does not give us full flexibility in programming. In particular, since the dominating PID controller is error-based, implementing ADR based controller in an error-based manner can make it easier to deploy. In addition, the error-based configuration gives us the ability to reform the ADR based controller into the commonly used two-pole-two-zero (2p2z) and three-pole-three-zero (3p3z) form, making it much easier to be programmed into existing controllers. In addition, when derivatives of the reference signal are not easily accessible, error-based implementation of ADRC can show its benefits, as the derivatives are estimated by the ESO as well.

The profile is a widely used technique in industrial controllers, with which the reference signal to the controller is reformed to be trackable by the target system. Feedforward is one of the most common ways of using offline system information and disturbance information. However, both techniques were rarely discussed in the academic field. Under the ADR framework, using profile for reference feedforward becomes a very straightforward task, without the need for detailed system model.

Some feedback signals, for example, velocity, were not normally directly measurable. As a result, calculating derivative of signals online is a common need. Since the noise amplification and phase lag introduced by the differentiator is always a tradeoff, a good nonlinear approximate differentiator is essential.
Since many industrial applications still use fixed point microcontrollers, fixed point implementing of ADRC affects the calculation accuracy of online information processing greatly. As a result, proper scaling of the ESO is the key in ADRC deployment on fixed point platform.

2.2 PMSM Control Issues

2.2.1 Current Regulation with FOC

In traditional DC-motor-based servo systems, a very tight current regulation is not required, since a well-designed speed or position loop could tolerate some small errors in torque output effectively. As a result, it is usually sufficient to use a PI controller with some feed-forward, which is the most commonly used industrial controller, in current loop design in DC motor drives. The current control quality in an AC motor under the FOC framework, on the other hand, will not only affect its torque output accuracy but also have a noticeable influence on other factors, such as its power efficiency [15].

As we discussed in Chapter 1.2, the two components embedded in the stator current, the magnetizing current, which is aligned with the $d$ axis, and the torque current, which is aligned with the $q$ axis, could be controlled separately in the FOC framework. This made it possible to inherit the control techniques of the separately excited DC motor, where the scheduling of magnetizing and torque current profile is well studied. In order to adopt those solutions, the $d$ axis current and $q$ axis current need to be regulated at their desired values throughout the motor operation.

Although the two current vectors are geometrically orthogonal, they are dynamically interdependent, i.e. changing one of the currents will affect the other, even under normal operating conditions. Although the PI controller is still dominating the current regulator design in most industrial PMSM drive solutions, they are not sufficient to regulate the currents since PI controller is not good at dealing with disturbances and
uncertainties. Furthermore, the rotor velocity and rotor flux density are also involved in the voltage-current dynamic. Those uncertainties made it difficult to apply traditional model-based decoupling methods.

2.2.2 Velocity Control with Position Feedback

As was discussed in Section 1.2.3, the most commonly used method for collecting velocity information online to close the speed loop was to calculate from position information with approximate differentiation. With the limitation of position sensor resolution, especially in lower-end applications, the differentiator bandwidth was greatly limited. This made the phase lag introduced by the second-order approximate differentiation problematic in the velocity loop design.

Under the ADR framework, the position information can be processed with an ESO to extract the velocity information and the total disturbance in the system. Doing so can eliminate one order in the approximate filter, which reduces the phase lag associated with it and eliminates a set of tuning parameters.

2.2.3 Rotor Position Estimation

As was discussed in Section 1.2.2, the position sensor can be eliminated from PMSM servo system with the many benefits such as lower cost, better reliability and smaller size. The rotor position information can be estimated online from the voltage and current information instead. The conventional methodology of, so-called, sensorless control is based on the fundamental model of a PMSM, by estimating the back-EMF elements, in the $\alpha-\beta$ reference frame, with some observer. This method has already been well established and a properly designed observer will be able to estimate the rotor
flux position, thus estimate the rotor position, in high- to medium-speed situations, since the rotor north pole is fixed on the \(d\) axis \([20, 23]\).

However, in many cases, the conventional methods have poor performance or even fail. In the industrial environment, there are measurement and switching noise in the voltage and current signal. As a result, some widely used observing methods, such as sliding mode observer (SMO), will result in noisy estimation result due to the amplification of the noise with its high gain. When the motor is in the low-speed region, the back-EMF signal becomes very small as it is proportionality to the electrical velocity of the motor. The signal often falls into the same level of magnitude as the noises, leaving poor signal-to-noise ratios (SNRs), which yields to inadequate shaft position estimation. When the motor is at a standstill, there is no back-EMF to be measured, making all back-EMF based methodologies not applicable. Although some saliency-based and high-frequency signal injection (HFSI) -based methods have been proposed to solve those problems, they mainly work for salient-pole machines, for example, IPMSMs \([19, 24]\). For nonsalient-pole machines, like surface mount PMSMs, the problem becomes more challenging, since the structure of the rotor is nearly symmetric.

### 2.3 Literature Review

In this section, we first give a historical view of ADR based control design framework. Then we will discuss some existing results on the ADRC implementation topics we discussed in Chapter 2.1. The existing results on current and motion control of PMSM under different conditions are given. We finish this chapter by introducing some existing solutions on position estimation of PMSM.
2.3.1  A Historical View of ADR

As was discussed earlier, the key idea of ADR based control design is to extract the total disturbance information online and cancel its effect in the control law. The idea of disturbance information extraction is not new at all. It can be traced back to the famous Chinese invention, the south-pointing chariot [25], where the information of the turn of the chariot was extracted from the speed difference between its two wheels. Thousands of years later, Jean-Victor Poncelet designed a speed governor for a steam engine with a similar idea, where the load variation information, i.e., the disturbance information, of the engine was extracted with the sensor and was used directly to form the control signal [9]. Since the load information has a $90^\circ$ phase lead compared to the velocity information, the total phase lag in the control loop was reduced. The theory behind the disturbance information extraction, however, was not established until 1939 when G. B. Shchipanov proposed the theory of invariance [26]. Unlike the traditional feedback control theory, the theory of invariance discusses how to make the output invariant to any system changes, including inputs, disturbances and uncertainties.

In the early applications, the disturbance was directly measured with separate sensors. As some disturbances are not directly measurable and adding sensors will affect system cost and reliability, extracting the disturbance information form estimation becomes necessary. Many different forms of disturbance observers were then presented to meet this need. Johnson first proposed the unknown input observer (UIO) in 1971 to estimate the unknown input disturbance of the system [27]. Some Japanese researchers, Umeno et al., then presented the disturbance observer (DOB) for external disturbance information extraction in the transfer function form [28]. Kwon and Chung then proposed the perturbation observer (POB) for perturbation information estimation in discrete form in 2002 [29]. All three observers above were designed to estimate only the external disturbance information.
In 1989, Han conceived the concept of total disturbance, where the unknown dynamics and system uncertainties are treated together with the external disturbance [3]. A unique disturbance estimator design, the extended state observer (ESO) was then presented by Han in 1995, where the total disturbance was formulated as an extended state and estimated with state observer [30]. ESO is a practical example of the continuous sensing and measuring suggested by Hsue-Shen Tsien. Once the total disturbance information is extracted, it can be used to cancel its effect in the control law. The active disturbance rejection control (ADRC) was then proposed by Han in 1998 with nonlinear ESO [31]. A linear version of ADRC controller and its parameterization was presented by Gao in 2003, making the controller and observer much easier to implement [32].

2.3.2 Practical Implementation Issues

As discussed in Chapter 2.1, there are many interesting topics about the practical implementation of ADR controller in industrial applications. Many types of research have been done on topics, such as parameter estimation, profile, feedforward. In addition, although ADRC does not require a detailed mathematical model, many research shows that using partial information in controller design can increase performance and reduce bandwidth requirement.

The profile is a technique that is widely used in industrial controllers, programmable logic controllers (PLC) and distributed control systems (DCS). The most basic function of the profile is to smooth out the reference signal with constraints such as changing velocity or acceleration, keeping the reference signal within the limitation of the physical system [33, 34]. Using a well-designed profile can also reduce shocking to the system and reduce energy cost. Some profiles can also provide the derivatives of the reference signal for the controller and feedforward design [35, 36]. One other common
use of the profile is in vibration systems, where a properly designed profile can reduce or eliminate the vibration of the system [34, 37, 38].

The simplest and most widely used method for reference in industrial control is look-up-tables, where the steady state control signal of each reference level was logged and put into a table by field engineer with manual or automatic method. This table was then used to generate the feedforward control signal [39]. This solution is effective and easy to implement since it does not require a mathematical model of the system. However, since only the steady state information is used to build the static table, it cannot help the transient as much in slower systems where a much larger control signal is needed for fast transient.

Another commonly used method of doing reference feedforward would be to use the inverse of the system model and was widely researched and used in various kind of systems. Due to the complexity of control targets, getting a deployable inverse of the model is often challenging. Scholars have proposed many solutions for inverse based control for linear and nonlinear systems [40-44]. Rigney, Butterworth, et al., have proposed their research on the inverse of nonminimum-phase systems [45, 46]. One limitation of inverse based feedforward is that they rely on a mathematical model, which is difficult and costly to get in many applications. Although research shows the inverse-based method can provide benefit on systems with uncertainties [47], they need to spend extra effort to deal with the uncertainties and disturbances.

Although the ADR based design strategy does not require a detailed mathematical model to design, using partial known system information could not only increase the performance, but also reduce the bandwidth requirement in many cases. There are two main categories of using system information in the ADR framework. One idea was to keep the cascade integral from and put all known information in the derivative of the disturbance term [48]. This solution does not require any variation in the control law.
design, leaving the separation between ESO and control law. The other idea was to separate the know dynamics from the disturbance term. This idea frees some extra design flexibility since the designer can choose the estimated or measured feedback signal in the control law for the know dynamics.

2.3.3 PMSM Control Design Methods

Since the speed and position control of PMSM have no significant differences to other motors that had been well developed, we focus on the academic results of torque control here. As the torque output of the motor is proportional to its torque current element, in particular, the quadrature current in FOC framework, under normal conditions. As a result, most researchers have focused on the current control of $d$-$q$ axes current. Many advanced control techniques had been applied to enhance the industrial leading PID controller, such as fuzzy logic [49], self-adaptive tuning [50, 51] and intelligent tuning [52] in recent years. Other advanced control methods have also been employed for high-quality current regulation. Bianchi applied time optimal control [53] while Lemmens and Bolognani used optimal design technique to set voltage and current saturation [54, 55].

Chou and Mohamed implemented robust control [56, 57], which could tolerate a certain amount of system model uncertainties, while Li used robust control with a disturbance observer [58] to deal with the external disturbances at the same time. However, the limitation embedded in robust control design method, which is only a small amount of system dynamics uncertainties could be tolerated, made it difficult to fit into the current control of PMSM under unbalanced and extreme conditions, where the system uncertainties are significantly larger.

Hassaine and Chang implemented sliding mode control (SMC) as the current regulator in the FOC framework [59, 60]; Hassan applied SMC in the direct torque
control of both surface mount PMSM and IPMSM [61, 62]; Hassan and Zhang combined SMC with adaptive control technique in their research [61, 63]; Repecho proposed an SMC solution [64] where the three phase-currents are individually controlled. A common issue among existing SMC solutions is that the control signals, such as the phase voltage, are prone to chattering, and the feedback signals, such as currents or torques, can be quite noisy.

Ortega introduced a passivity-based control (PBC) method for current regulation in FOC of voltage-fed and current-fed induction motors [65]. In addition, Li implemented interconnection and damping assignment (IDA) PBC as the current regulator in field weakening and FOC of a PMSM for maximum energy efficiency and wide speed range [66].

For the phase unbalance condition, Novak discussed UMP from uneven rotor magnetization [67] and Yu proposed incline of UMP caused by misalignment of the rotor [68]. Zhang presented a solution for compensating the load mass unbalance of a bearingless PMSM [69].

To achieve the maximum speed of PMSM, field weakening technique is used and many studies had been done [70, 71]. Since the torque-to-current relationship is no longer linear, studies on direct torque control had been taken [71, 72]. Bolognani used the motor dynamics information for model predictive control [73] and Xiaochun developed his control method based on d-q current cross-coupling effect [74].

On the maximum power output, the PWM overmodulation technique is implemented [75, 76], where the output voltage and current are no longer sinusoidal signals, and the conventional modeling and control technique is no longer sufficient. Studies had been done on how to model [77] and control current and torque of PMSM in the overmodulation region [78-81].
On high torque or high power output situations, the induced flux is often in the saturation region. Studies had been done on the saturation curves [82], of its mathematical model [83] and control solutions [83, 84]. Cheng proposed a torque feed-forward method based on the information of torque and current saturation curves [85, 86].

As discussed in the previous section, the PMSM servo has many nonlinearities and is hard to be modeled under extreme conditions, such as field weakening, overmodulation and flux saturation regions. However, under the active disturbance rejection framework, those nonlinearities and model uncertainties are treated as part of total disturbance together with external disturbances [87-89]. An extended state observer (ESO) is used to estimate the total disturbance as an equivalent input disturbance and compensate its effect in the control signal. It had been implemented successfully in many applications such as gasoline engine [90], MEMS gyroscope [91], chemical processes [92], hysteresis systems [93].

Several researchers have proposed their ADRC solutions for PMSM control. For speed and position control, Sun, Liu, et al., applied nonlinear ADRC [94-97], Wen applied nonlinear ADRC with fuzzy adaptive control [98] and Liu applied linear ADRC [99]. For the torque control, Fu and Wu used nonlinear ADRC on the current control loop [100, 101] and Zhang applied linear ADRC [102]. In addition, some direct speed and position control solutions without current regulator using nonlinear ADRC had been proposed [103, 104]. Sun proposed a direct speed solution with an ADRC speed controller and an ADRC speed observer [13].

However, those solutions are only applying the conventional ADRC formulation without employing the known system dynamics in their controller design. Interpreting some known system dynamics into the ADRC design will reduce the load of the ESO, which could achieve the same performance with much lower observer bandwidth requirement [91]. In addition, the two ESOs in the direct axis current regulator and the
quadrature axis current regulator could be designed as one observer, sharing their coupling information to reduce the load of ESO further.

The industry has also developed many new techniques in recent years. Many microcontroller manufacturers published their reference designs such as Texas Instruments, Freescale Semiconductor and STMicroelectronics. A motion control framework based on ADRC, the SpinTAC control technique, has been incorporated in the Texas Instruments InstaSPIN-Motion suite for their C2000 microcontroller line [22] and the NXP Kinetis Motor Suite ARM Cortex-M4 Kinetis MCU [105].

2.3.4 Rotor Position Estimation in PMSM

As we discussed in Section 2.2.3, more and more new designs use position estimator, which estimates the rotor position from the phase voltage and current signal, which has already been measured for current loop control. For lower-end applications, such as appliances like washing machines and air conditioners, using position estimation could eliminate the need for the expensive position sensor. For applications with functional safety requirement, such as electric vehicles, the estimated result can be used as a backup redundant to the position sensor.

The most commonly used principle for position estimation is to estimate the back-EMF in the $\alpha - \beta$ frame and calculate the angle with it. There are many solutions based on this methodology with observer techniques such as Luenberger observer [106, 107], sliding mode observer [20, 108], enhanced sliding mode observer (eSMO) [20, 23], Neural Network Observer [109], Frequency-Adaptive Disturbance Observer [110]. Although each solution has its special trick and has acceptable performance in the middle- to high-speed conditions, they perform unsatisfactorily in low-speed condition by nature, because the back-EMF signal is proportional to the motor speed and are very small at low speed. This method does not work for initial position estimation either.
Besides the most straightforward way of calculating the rotor position by doing arctangent, alternative methods such as phase-locked loop (PLL) [108, 111, 112] and model reference adaptive system (MARS) [113] provide various improvements and can extract the rotor speed information at the same time. These methods have some improvement in low-speed stability, but cannot fully address the back-EMF issue.

To have better low-speed performance and support initial position estimation at a standstill, many saliency-based methods have been developed to improve the position estimation performance. Those machine saliency-tracking methods could estimate the shaft position by using a high-frequency (HF) excitation, whose frequency is much higher than the motor base frequency. By measuring the response of those HF excitation signals, the rotor saliency information, thus the rotor position, can be extracted. The HF excitation can be injected with either a carrier signal injection, with either sinusoidal waveform or square waveform, or a pulse-width modulation (PWM) pattern modification [19, 24, 114]. These methods provide very good initial position estimation and low-speed performance, while suffers at higher speed. One other limitation is these methods only work with salient pole motors, i.e., interior permanent magnet synchronous motor (IPMSM).
Chapter III

PRACTICAL IMPLEMENTATION OF ADRC

In this chapter, some practical issues in implementation and deployment of ADRC in real-world control systems are discussed. In particular, we focus on solutions that are implementable on microcontrollers (MCUs) where the storage and computing power are limited. This includes the offline information extraction in term of key system parameters, as well as how such information can be used to make the online information extraction more efficient. Another solution of great practical significance is the formulation of the error-based ADRC, which seems straightforward mathematically but leads to a surprising discovery: ADRC is implementable in a traditional, error-based, transfer function form. Also discussed are improvements on two commonly used mechanisms in practical control system design, profile and feedforward, both are based on the offline information. In addition, it is shown how the nonlinear Tracking Differentiator (TD) can be used properly to extract the derivative information of a signal online. Finally, the scaling of ESO for fixed-point implementation is discussed for better online signal processing accuracy. The problems discussed in this chapter may seem relatively small to academic researchers, but they are of great practical significance, as will be shown below.
3.1 Parameter Estimation

As was discussed earlier, the offline information of system dynamics can either be derived from the law of physics or extracted from the online data. The data-driven estimation is especially useful in the industrial control environment since the systems under control are often complex and difficult to model. Here we will discuss a few methods to extract the offline information under the ADR framework using motion control system as an example.

A typical motion system can be written in the form of
\[ \dot{y} = -a \dot{y} + bu + d \]  \hspace{1cm} (4)
where the parametric uncertainty in \( a \) and \( b \), the unmodeled dynamics and the external disturbances are wrapped into the total disturbance term \( d \), to be estimated by ESO and canceled. The question is how to estimate \( a \) and \( b \) in an industrial setting where computation and memory are both very limited. Note that such information obtained offline can be used in the ESO design for improved performance.

First, for the motion system in (4), assume that an ESO has been designed so that its state vector \( Z = [z_1 \ z_2 \ z_3]^T \) estimates \( \begin{bmatrix} y & \dot{y} & -a \dot{y} + d \end{bmatrix}^T \) (as shown in Chapter 5.3). When there is little external torque disturbance, \( d \) is very small compared to \( ay \) and
\[
\begin{align*}
    z_2 &\approx \dot{y} \\
    z_3 &\approx -a \dot{y}
\end{align*}
\]  \hspace{1cm} (5)
As a result, the value of \( a \) can be calculated as
\[ a \approx -\frac{z_3}{z_2} \]  \hspace{1cm} (6)
when \( z_2 \neq 0 \). To test this method, some simulation verification was done. First, we picked three different target system with \( a \) valued at 1, 2 and 3, respectively, and \( b = 10 \). The \( b_0 \) value in the ESO was initially set to \( b_0 = b \). We can see that in all three cases the estimation result converges to the true \( a \) value, as shown in Figure 5 and Figure 6, where
the subsystem 2-4 are the same standard ADRC controller with third-order ESO and the PD controller (with $\omega_o = 16$ and $\omega_c = 4$).

![Block Diagram of a Estimation](image1)

**Figure 5** Block Diagram of a Estimation

![Estimation Result](image2)

**Figure 6** a Estimation Result
The estimation result with an inaccurate $b_o$ value tested by setting the real $b$ value 20% higher and lower than the real value. The simulation result is given in Figure 7. We can see that the estimation accuracy is dependent on the accuracy of the $b_o$ value used in the ESO design.

![a Estimation Result with Different b Value](image)

**Figure 7  a Estimation Result with Different b Value**

To address the above issue, a method based on the time constant is proposed as follows: let $\omega$ be the velocity and ignore the disturbance, the motion system of (4) can be rewritten as

$$\dot{\omega} = -a\omega + bu$$  \hspace{1cm} (7)

Assuming the input $u$ is a constant, the solution to this differential equation can be written as

$$\omega(t) = \frac{bu}{a} \left(1 - e^{-at}\right)$$  \hspace{1cm} (8)
When \( t \) goes to infinity, the steady-state value of the velocity is \( \omega(\infty) = \frac{bu}{a} \), which can be obtained with a simple open-loop test. During the transient period, at any time \( t = t_i \), the velocity can be measured as

\[
\omega(t_i) = k \frac{bu}{a}, k \in [0,1]
\]  

(9)

from which \( k \) can also be calculated. Then, from

\[
e^{-at_i} = 1 - k
\]

(10)

and

\[
-at_i = \ln(1-k)
\]

(11)

\( a \) could be estimated as

\[
a = -\frac{\ln(1-k)}{t_i}
\]

(12)

Note that, unlike the previous ESO-based estimation method, this time constant based method is independent of \( b \).

Next, we discuss two methods to estimate \( b \). For the motion system in (4), when there is little external torque disturbance, i.e. \( d \) is very small compared to \( ay \)

\[
bu \approx \dot{y} + ay
\]

(13)

When the motor starts from a standstill, the velocity of the motor, i.e., \( \dot{y} \) is nearly zero. So we have

\[
b \approx \frac{\dot{y}}{u}
\]

(14)

with \( u \) as a step function. Since the value of \( \dot{y} \) was not directly measurable, we need to use a third order approximate differentiation to estimate that. We then estimate the value of \( b \) as the peak value of (15) [115].
\[
\frac{1}{u} \frac{s^2}{(\tau s + 1)^3}
\]  

Note that in the digital implementation, the value of $\tau$ is limited by the sampling rate and resolution of the feedback signal. And this will make the result inaccurate when the value of $a$ is large. When an approximate of the value of $a$ is accessible, the accuracy can be improved by finding the peak value of

\[
\frac{1}{u} \frac{s^2 + as}{(\tau s + 1)^3}
\]  

instead.

The block diagram and result of simulation verification are given in Figure 8 and Figure 9 respectively. Two systems with the $a$ value of 10 and 50 were chosen, respectively. The compensated solution in (16) was tested with the $a$ value 10\% smaller than the real value to simulate a real-world environment.

![Block Diagram of $b$ Estimation](image-url)

**Figure 8  Block Diagram of $b$ Estimation**
We can see from Figure 9, as the \( a \) value increases from 10 to 50, the estimation result without compensation is off from about 7\% to as much as 20\%, whereas the compensated result all stayed within 5\% of accuracy.

### 3.2 Incorporating Offline Information in ESO Design

Although ADRC has shown the ability to control nonlinear, time-varying and uncertain processes in the absence of detailed knowledge of plant dynamics, it would be beneficial to leverage the existing offline information in system dynamics for enhanced performance. Specifically, utilizing the partial but available information of system dynamics could reduce the load of ESO, leading to better tracking and lower bandwidth requirement.

As was discussed earlier in Chapter 2.1, there are two main methods of using offline system information during ADR based controller design. The first method is to
stick with the cascade integral model, wrap all known and unknown dynamics into the total disturbance and use the offline information to help the ESO in online information processing. This method keeps the controller form unchanged and has better separation between controller and observer design, i.e., the controller law is independent of the ESO design. The second method is to incorporate the offline information into the system model used in ESO design, leaving only unknown dynamics and external disturbances in total disturbance to be estimated. This method gives an extra order of flexibility in the control law design, as is shown in the example below.

Consider the first-order system
\[ \dot{y} = -ay + bu + d \]  
(17)

For the first method, we can define the states as
\[ x_1 = y \]
\[ x_2 = f_1 = -ay + d \]  
(18)

Then the system in (17) can be rewritten as
\[ \dot{x}_1 = x_2 + bu \]
\[ \dot{x}_2 = -ax_1 + \hat{d} = -a(x_2 + bu) + h \]  
(19)

where \( h = \hat{d} \). The corresponding ESO is
\[ \dot{z}_1 = z_2 + bu + l_1(y - z_1) \]
\[ \dot{z}_2 = -az_2 - abu + l_2(y - z_1) \]  
(20)

and the corresponding control law is unchanged:
\[ u = \frac{u_0 - z_2}{b} \]  
(21)

which reduces the plant to
\[ \dot{y} = u_0 \]  
(22)

For the second method, assuming \( a \) in (17) is given, we can redefine the states as
Then (17) can be rewritten as
\[\begin{align*}
\dot{x}_1 &= y \\
\dot{x}_2 &= f_2 = \alpha
\end{align*}\]

The corresponding ESO is
\[\begin{align*}
\dot{z}_1 &= -az_1 + z_2 + bu + l_1(y - z_1) \\
\dot{z}_2 &= l_2(y - z_1)
\end{align*}\]

and the control law is modified as
\[u = \frac{u_0 - z_2 + az_1}{b}\] (26)
or
\[u = \frac{u_0 - z_2 + ay}{b}\] (27)
both will reduce the plant to
\[\dot{y} = u_0\] (28)

The difference is that there is an extra degree of freedom in the choice of the control law, (26) or (27). The former is less sensitive to measurement noise and has a smoother control signal, whereas the latter has less phase lag and better stability margin. In the existing literature, the design of (26) was adopted by default without any discussion. The purpose here is to provide an alternative for the users in seeking the common trade-off between performance on the one hand, and smoothness and stability margin on the other.

### 3.3 Error-Based ADRC

As was discussed earlier, most industrial controllers are error-driven, meaning that the input to the controller is the tracking error between the command and the plant...
output. For ease of use, an industrial controller is best represented in the form of the transfer function as the ratio of a control signal over the tracking error. A typical example is the PID, which is easily representable in a transfer function. Two pole or three pole transfer function blocks are also common in most industrial control user interface, such as those found in PLC (programmable logic controller) and DCS (distributed control system). It is therefore of interests to see if ADRC can be easily implemented in such function blocks. A positive answer will have significant implications in industrial applications.

In all existing forms of ADRC in the literature, a two-degree-of-freedom form is adopted where, in the inner loop, the ESO extracts the disturbance information and cancel it with a part of the control signal, leaving the controller to deal with the modified plant in the cascade integral form. The principle of ADRC, however, does not limit its implementation to this one form. In fact, it was well-known among early ADRC researchers in the CACT (Center for Advanced Control Technologies) that ESO can be implemented by replacing one of its input, $y$, with the tracking error, $e = r - y$. It is shown in this section, in doing so the ADRC algorithm can be realized in the form of error feedback commonly found in industrial control configurations. In addition, by estimating not only the total disturbance and various order of the derivatives of $e$, there is no need to generate separately, as is done currently in the literature, various order of the derivatives of the reference signal for the purpose of feedback and feedforward control. In other words, it is shown in this section that with error-based ADRC, the previous controller blocks such as of Profile Generator (TD), Feedforward Controller, Feedback Controller, and ESO can be combined into a single transfer function block in the standard industrial feedback control configuration.

For the sake of brevity, the error-based ADRC is derived below for the first and second-order plant. The solution is given in both continuous and discrete time form for
easy implementation. Simulation verification was performance to verify the performance of the error-based implementation.

### 3.3.1 First-Order System

Consider a generalized first-order system as

\[
\dot{y} = f + bu
\]  

(29)

where \( f \) represents the total disturbance. Define states as

\[
\begin{align*}
x_1 &= e = r - y \\
x_2 &= f_i
\end{align*}
\]  

(30)

where \( f_i = \dot{r} - \dot{f} \). Hence

\[
\begin{align*}
\dot{x}_1 &= \dot{r} - \dot{y} = \dot{r} - \dot{f} - \dot{f} u = \dot{r} - f - bu \\
\dot{x}_2 &= h
\end{align*}
\]  

(31)

and (29) can be represented as

\[
\begin{align*}
\dot{x}_1 &= \dot{r} - f - bu = x_2 - bu \\
\dot{x}_2 &= h
\end{align*}
\]  

(32)

where \( h = \dot{f} = \dot{r} - \dot{f} \). The corresponding ESO is

\[
\begin{align*}
\dot{z}_1 &= z_2 - bu + l_1 (e - z_1) \\
\dot{z}_2 &= l_2 (e - z_1)
\end{align*}
\]  

(33)

and the observer gain is chosen

\[
L = \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} = \begin{bmatrix} 2\omega_o \\ \omega_o^2 \end{bmatrix}
\]  

(34)

by selecting \( \omega_o \), i.e. the observer bandwidth. The control law

\[
u = \frac{u_0 + f_i}{b}
\]  

(35)

changes (29) to approximately

\[
\dot{y} = u_0
\]  

(36)
which is easily controlled with a proportional controller \( u_0 = K_p z_1 \). The control law can then be written as

\[
u = \frac{K_p z_1 + z_2}{b}
\]

where \( K_p = \omega_c \) sets the controller bandwidth at \( \omega_c \).

Note that since \( z_1 = \hat{r} - \hat{\dot{y}} \) and \( z_2 = \hat{r} - \hat{\dot{f}} \), equation (37) can be rewritten as

\[
u = \frac{K_p (\hat{r} - \hat{\dot{y}}) + \hat{r} - \hat{\dot{f}}}{b}
\]

which shows that this control law is the combination of feedback and feedforward control, as well as disturbance estimation and cancellation. This implementation makes it unnecessary to generate separately in Profile Generator the smooth signal \( r \) and \( \dot{r} \). Even if the given reference signal \( r \) is discontinuous, the corresponding \( \hat{r} \) and \( \dot{\hat{r}} \) are smooth because they are generated by the ESO with the bandwidth of \( \omega_o \). Therefore, in this formulation, the achievable bandwidth in the control loop is closely dependent on \( \omega_o \).

Perhaps the most important and surprising consequence of this formulation is that the corresponding ADRC can now be implemented in the transfer function form commonly seen in the industry, as shown below.

Substituting (37) into (33) we have

\[
\begin{align*}
\dot{z}_1 &= z_2 - (K_p z_1 + z_2) + l_1 (e - z_1) = (-K_p - l_1) z_1 + l_1 e \\
\dot{z}_2 &= l_2 (e - z_1) = -l_2 z_1 + l_2 e \\
\end{align*}
\]

or

\[
\dot{Z} = \begin{bmatrix} -K_p - l_1 & 0 \\ -l_2 & 0 \end{bmatrix} Z + \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} e
\]

where \( Z = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \). The transfer function from \( e \) to \( Z \) is

\[
TF = C (sI - A)^{-1} B
\]
where \( A = \begin{bmatrix} -K_p - I_1 & 0 \\ -I_2 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} I_1 \\ I_2 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 \end{bmatrix} \) and \( TF = \begin{bmatrix} TF_1 \\ TF_2 \end{bmatrix} = \begin{bmatrix} Z_1 \\ Z_2 \\ E \end{bmatrix} \). Then, equation (37) can be rewritten as

\[
\frac{U(s)}{E(s)} = \frac{\frac{K_p TF_1 + TF_2}{b}}{\frac{(K_p l_1 + l_2)s + l_2 K_p}{b(s + l_1 + K_p)s}}
\]

and its discrete form based on Tustin approximation is obtained by letting \( s = \frac{2z - 1}{Tz + 1} \) and

\[
\frac{U(z)}{E(z)} = \frac{T}{2} \frac{\bigg(2K_p l_1 + Tl_2 K_p + 2l_2\bigg) z^2 + 2Tl_2 K_p z - 2K_p l_1 + Tl_2 K_p - 2l_2}{\bigg(2 + l_1 T + K_p T\bigg) z - 2 + l_1 T + K_p T\bigg)(z - 1)b}
\]

### 3.3.2 Second-Order System

Consider the second-order system

\[
\dot{y} = f + bu
\]

where \( f \) represents the total disturbance. Define states as

\[
\begin{align*}
x_1 &= e = r - y \\
x_2 &= \dot{r} - \dot{y} = \dot{x}_1 \\
x_3 &= f
\end{align*}
\]

where \( f_1 = \dot{r} - f \). Then

\[
\dot{x}_2 = \dot{r} - \dot{y} = \dot{f} - (f + bu) = \dot{r} - f - bu = x_3 - bu
\]

and (44) can be rewritten as

\[
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \dot{r} - f - bu = x_3 - bu \\
\dot{x}_3 &= h
\end{align*}
\]

where \( h = \dot{f}_1 = \dot{r} - \dot{f} \). The ESO can then be designed as
\[
\begin{align*}
\dot{z}_1 &= z_2 + l_1 (e - z_1) \\
\dot{z}_2 &= z_3 - bu + l_2 (e - z_1) \\
\dot{z}_3 &= l_3 (e - z_1)
\end{align*}
\] (48)

where the observer gain can be parameterized as

\[
L = \begin{bmatrix}
l_1 \\
l_2 \\
l_3
\end{bmatrix} = \begin{bmatrix}
3\omega_o \\
3\omega_o^2 \\
\omega_o^3
\end{bmatrix}
\] (49)

by setting the observer bandwidth as \(\omega_o\). The control law

\[
u = \frac{u_0 + f_1}{b}
\] (50)

transforms (44) approximated to

\[
\dot{y} = u_0
\] (51)

which is easily controlled with a PD controller \(u_0 = K_p z_1 + K_d z_2\). The control law can then be given as

\[
u = \frac{K_p z_1 + K_d z_2 + z_3}{b}
\] (52)

Note here as \(z_1 = \hat{r} - \hat{y}\), \(z_2 = \hat{r} - \hat{y}\) and \(z_3 = \hat{r} - \hat{f}\), equation (52) can be rewritten as

\[
u = \frac{K_p (\hat{r} - \hat{y}) + K_d (\hat{r} - \hat{y}) + \hat{r} - \hat{f}}{b}
\] (53)

which equivalents to output based ADRC with reference feed-forward. Here the PD gains can be chosen as \(K_p = \omega_c^2\) and \(K_d = 2\omega_c\) where \(\omega_c\) is the control loop bandwidth.

Substituting (52) into (48) we have

\[
\begin{align*}
\dot{z}_1 &= z_2 + l_1 (e - z_1) = -l_1 z_1 + z_2 + l_1 e \\
\dot{z}_2 &= z_3 - (K_p z_1 + K_d z_2 + z_3) + l_2 (e - z_1) = (-K_p - l_2) z_1 - K_d z_2 + l_2 e \\
\dot{z}_3 &= l_3 (e - z_1) = -l_3 z_1 + l_3 e
\end{align*}
\] (54)
Rewritten (39) in the state space form we have

\[
\dot{Z} = \begin{bmatrix}
-l_1 & 1 & 0 \\
-l_2 - K_p & -K_d & 0 \\
-l_3 & 0 & 0
\end{bmatrix} Z + \begin{bmatrix}
l_1 \\
l_2 \\
l_3
\end{bmatrix} e
\]

where \( Z = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} \). Then the transfer function from \( e \) to \( Z \) is

\[
TF = C (sI - A)^{-1} B
\]

where \( A = \begin{bmatrix}
-l_1 & 1 & 0 \\
-l_2 - K_p & -K_d & 0 \\
-l_3 & 0 & 0
\end{bmatrix}, \quad B = \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \) and \( TF = \begin{bmatrix} TF_1 \\ E \\ TF_2 \\ Z_2 \\ TF_3 \\ Z_3 \end{bmatrix} = \begin{bmatrix} Z_1 \\ E \\ Z_2 \\ E \\ Z_3 \\ E \end{bmatrix} \).

Then (52) can be rewritten as

\[
\frac{U(s)}{E(s)} = \frac{K_p TF_1 + K_d TF_2 + TF_3}{b} = \frac{(K_p l_1 + K_d l_2 + l_3)s^2 + (K_p l_2 + K_d l_3)s + K_p l_3}{s [s^2 + (K_d + l_1)s + l_2 K_d + l_3 + K_p] b}
\]

and its discrete version based on the Tustin approximation can be obtained by letting

\[
s = \frac{2}{T} \frac{z - 1}{z + 1}
\]

\[
U(z) = \frac{1}{2} \left[ \begin{array}{c}
\left(4 l_3 + 4 l_1 K_p + 2 T K_p l_2 + 2 T l_1 K_d + T^2 l_1 K_p + 4 l_2 K_2 \right) z^2 \\
+ \left(-4 l_3 - 4 K_d l_2 + 2 T K_p l_2 + 3 T^2 l_1 K_p - 4 l_2 K_p + 2 T l_3 K_d \right) z^2 \\
+ \left(-2 T l_3 K_d - 4 l_3 - 4 K_d l_2 - 2 T K_p l_2 + 3 T^2 l_1 K_p - 4 l_1 K_p \right) z
\end{array} \right] T
\]

\[
E(z) = 1 \left[ \begin{array}{c}
\left(2 l_2 T + 2 K_d T + 4 + K_p T^2 + l_1 T^2 + l_2 K_d T^2 \right) z^2 \\
+ \left(-8 + 2 l_2 T^2 + 2 l_1 K_d T^2 + 2 K_p T^2 \right) z \\
-2 K_d T + 4 l_2 K_d T^2 + 4 + K_p T^2 + l_2 T^2 - 2 l_1 T
\end{array} \right] (z - 1)
\]

43
3.3.3 Simulation Verification

The proposed error-based ADRC was put to simulation test with a second-order plant. The comparison is made with the regular output-based ADRC, with and without the reference feed-forward. The system under control is chosen as \( \frac{1}{s^2 + 8s + 15} \) and all three ADRC controller parameters are set to \( b = 1 \), \( \omega_c = 40 \) and \( \omega_o = 160 \). The block diagram for MATLAB simulation is given in Figure 10 and the simulation result is given in Figure 11.

\[
\text{Figure 10 Block Diagram of Output vs. Error Based ADRC}
\]

At \( t = 0.6 \) second, a step input disturbance is introduced. We can see that the output-based ADRC with feed-forward has the best transient response, closely followed by the error-based implementation, and output-based ADRC without feed-forward has a much slower transient response. This verified the idea that error-based ADRC has similar
performance with output-based ADRC with feed-forward, without the need for the derivatives of the reference signal. For the disturbance rejection, it is shown that all three solutions have the same response.

![Output vs. Error Based ADRC](image)

**Figure 11  Output vs. Error Based ADRC**

### 3.4 Approximate Differentiation with TD

Extracting the derivative information of a signal online is often required in control system design. For example, many velocity control systems in the industry do not have direct velocity feedback. Instead, they use the position feedback to calculate the velocity signal. Since most sensors have noises, direct differentiation often result in an extremely noisy signal. This makes the use of approximate differentiator necessary. With the traditional linear approximate differentiators, a bandwidth compromise must be made between the noise level and the phase lag. That is, higher bandwidth leads to smaller phase lag but more noise.
To overcome the limitation of approximate linear differentiator, a nonlinear tracking differentiator (TD) is proposed by Han [115, 116]. The question to be investigated here is if such TD can offer any advantage in reducing the phase lag without increasing the noise level. We begin with the following difference equation of TD

\[ x_1[k + 1] = x_1[k] + x_2[k]T \]
\[ x_2[k + 1] = x_2[k] + f_{han}(x_1[k] - v, x_2[k], r, T)T \] (59)

where \( v \) is the input signal, \( x_1 \) tracks the input \( v \) and \( x_2 \) is its approximate derivative.

The function \( f_{han} \) is given as

\[
\begin{align*}
  d &= r h^2 \\
  a_0 &= hx_2 \\
  y &= x_1 + a_0 \\
  a_1 &= \sqrt{d(d + 8|y|)} \\
  a_2 &= a_0 + \text{sign}(y) \frac{a_1 - d}{2} \\
  a &= (a_0 + y) f_{sg}(y, d) + a_2 (1 - f_{sg}(y, d)) \\
  f_{han} &= -r \left( \frac{a}{d} \right) f_{sg}(a, d) - r \cdot \text{sign}(a) (1 - f_{sg}(a, d))
\end{align*}
\] (60)

where

\[
f_{sg}(x, d) = \frac{\text{sign}(x + d) - \text{sign}(x - d)}{2}
\] (61)

Here the parameter \( r \) is equivalent to the time constant \( \tau \) in linear differentiators when TD is operating in its linear region, which is defined by the parameter \( a \).

Although TD has been proposed almost two decades ago, its advantage over linear differentiators is still not clearly shown. So here we put TD to test in simulation in comparison with the traditional linear approximate differentiator of the form

\[
\frac{s}{(\tau s + 1)^2}
\] (62)
The simulation results are shown in Figure 12 without noise and in Figure 13 with noise. It can be seen that the TD with $r = 0.02$ has similar phase lag with the linear approximation with $\tau = 0.02$ when the derivative of input is near zero. The noise level of TD, on the other hand, is similar to the noise level of linear approximation with $\tau = 0.1$.

**Figure 12** TD vs. Linear Approximation

**Figure 13** TD vs. Linear Approximation With Noise
3.5 A New Profile Generator Based on the Modified Tracking Differentiator

Profile generation and feedforward control are the two most common ways of using offline information in industrial control. The former refers to the mechanism where the reference and its various derivatives are generated to reflect the desired response under the physical limitations; the latter is the means to overcome the limitation of the latency in feedback control for faster response. There are many physical limitations in most industrial processes and, critical to the success of control system, they must be accounted for in the design process.

Actuators, for example, are limited in the force they can provide and in its rate of change. Even though this has received scant notice in the textbooks, engineers have long discovered that the reference signal must be generated so that the corresponding control signal will remain within the operating range of the actuator. In motion control, such reference signal is called motion profile, which is generated to be smooth and to avoid actuator saturation. To meet the need for a smooth reference signal and its first-order derivative, Han proposed the tracking differentiator, or TD, through which the command is first filtered and differentiated [116]. With such TD produces a smooth reference signal and its derivative, it does not take into the consideration of the maximum acceleration. To this end, the original version of TD is modified for motion control as

\[ x_1[k+1] = x_1[k] + x_2[k]T \]

\[ x_2[k+1] = \begin{cases} 
 v_{\text{min}} & x_{20} \leq v_{\text{min}} \\
 x_{20} & v_{\text{min}} \leq x_{20} \leq v_{\text{max}} \\
 v_{\text{max}} & v_{\text{max}} \leq x_{20} \end{cases} \]

\[ x_{20} = x_2[k] + f_{\text{hom}}(x_1[k] - r, x_2[k], a_{\text{max}}, T)T \]

where \( T \) is the sampling period, \( r \) is the position command, \( x_1 \) is the position reference and \( x_2 \) is its derivative. This reference is generated subject to the motor speed and acceleration limits of \( [v_{\text{min}}, v_{\text{max}}] \) and \( [-a_{\text{max}}, a_{\text{max}}] \), respectively.
Note that, in the motion control industry, the speed and acceleration limits are built into the motion profile generator using lookup tables. Han’s TD does not have such feature but is extremely easy implement and to use. The above profile generator in (59) combines the advantages of both methods into a modified TD (MTD) and it is not limited to motion control.

Together with ESO, TD and MTD are different methods of obtaining various order of the derivatives of the reference signal for the purpose of controlling the cascade integral plant in ADRC. Using the ADRC design for the second-order plant as an example, the plant is first reduced to the cascade integral form of

$$\ddot{y} = u_0$$  \hspace{1cm} (64)

To make $y$ follow $r$, most textbook solutions are offered in the form of error feedback, but engineers know better: the feedback is always bandwidth limited due to the inherent phase lag in each component in the loop. A typical engineering solution to counter such limitation is the combined feedback-feedforward control $u_0 = \ddot{r} + K_p (r - y) + K_d (\dot{r} - \dot{y})$ where $\ddot{r}$ is the feedforward term and $K_p (r - y) + K_d (\dot{r} - \dot{y})$ is the feedback term. In doing so, the closed-loop system becomes

$$\ddot{y} = \ddot{r} + K_p (r - y) + K_d (\dot{r} - \dot{y})$$  \hspace{1cm} (65)

where $K_p$ and $K_d$ are gain parameters to be chosen for the desired transient response.

### 3.6 Scaling for the Fixed-Point Implementation

In the hardware implementation of ESO in the industrial control equipment, fixed point microprocessors are more common because of the cost and complexity. The computation is more accurate if the numbers in an equation are of the same order of magnitude. This is because the wider the numbers are apart from each other, the lower the calculation resolution will be.
Here we take a third order ESO as an example. Higher order ESO could be treated similarly. The observer gains of a third order ESO are of the order of

\[
L = \begin{bmatrix}
    O(\omega_o) \\
    O(\omega_o^2) \\
    O(\omega_o^3)
\end{bmatrix}
\]  

(66)

Since \(\omega_o\) is usually a large number, the gains are usually many order of magnitude apart.

The discrete ESO can be written as

\[
Z[k+1] = \begin{bmatrix}
1 & T & \frac{T^2}{2} \\
0 & 1 & T \\
0 & 0 & 1
\end{bmatrix} Z[k] + \begin{bmatrix}
\frac{bT^2}{2} \\
\frac{bT}{\omega_o} \\
0
\end{bmatrix} u[k] + L \left( y[k] - z_1[k] \right)
\]

(67)

where \(T\) is the sampling period and is usually a small number in the order of \(O\left(\frac{1}{\omega_o}\right)\).

The numerical property can be improved with the state transformation

\[
w_1 = z_1, w_2 = \frac{z_2}{\omega_o}, w_3 = \frac{z_3}{\omega_o^2}
\]

(68)

and the equation (67) becomes

\[
W[k+1] = \begin{bmatrix}
1 & T\omega_o & \frac{T^2\omega_o^2}{2} \\
0 & 1 & T\omega_o \\
0 & 0 & 1
\end{bmatrix} W[k] + \begin{bmatrix}
\frac{bT^2}{2} \\
\frac{bT}{\omega_o} \\
0
\end{bmatrix} u[k] + L' \left( y[k] - w_1[k] \right)
\]

(69)

where

\[
L' = \begin{bmatrix}
    l_1 \\
    \frac{l_2}{\omega_o} \\
    \frac{l_3}{\omega_o^2}
\end{bmatrix} = \begin{bmatrix}
    O(\omega_o) \\
    O(\omega_o) \\
    O(\omega_o)
\end{bmatrix}
\]

(70)
Note that with the proposed state transformation of (68), the ESO and its observer gain are all improved numerically where coefficients are of the same order of magnitude.

\[
\begin{bmatrix}
1 & T\omega_o & \frac{T^2\omega_o^2}{2} \\
0 & 1 & T\omega_o \\
0 & 0 & 1
\end{bmatrix}
= \begin{bmatrix}
1 & O(1) & O(1) \\
0 & 1 & O(1) \\
0 & 0 & 1
\end{bmatrix}
\] (71)

and

\[
\hat{X} = Z = \begin{bmatrix}
w_1 \\
w_2\omega_o \\
w_3\omega_o^2
\end{bmatrix}
\] (72)
Chapter IV

FREQUENCY CHARACTERISTICS

Phase lag is inherent in all components of feedback control system, such as sensor, actuator and the actual physical process itself, and it is the reason that the bandwidth of the feedback control system is always limited. In practice, engineers carefully find the trade-off between performance on the one hand which requires high bandwidth, and stability margins and noise sensitivity on the other hand which limit the achievable bandwidth. In the ADRC formulation, the total disturbance in a second-order system, for example, is $180^\circ$ ahead of the output deviation or tracking error it induces. Therefore its timely estimation and cancellation have an obvious phase lead advantage comparing to the control action based on the error-based feedback. Having such phase characteristics leads to distinct advantages on tracking and energy savings. However, the amount of phase lead is limited by the unavoidable phase lag in the ESO, which is a function of its bandwidth. It is therefore necessary to study the frequency domain characteristics of ADRC and explore, among other things, the minimum phase margin achievable with sufficient observer bandwidth. Furthermore, the bandwidth limitation is studied for different orders of ADRC subject to delay. Finally, the performance and effective bandwidth of ADRC with and without offline information are compared in a case study.
4.1 Phase and Delay Margin of Generic ADRC

In this section, frequency characteristics of conventional ADRC controlled systems of different orders is discussed. The transfer function form of the closed-loop system is derived, and the gain and phase margin are derived as a function of the controller and observer bandwidth.

4.1.1 First-Order System

Given a first-order system of the form

\[
G_p(s) = \frac{Y(s)}{U(s)} = \frac{b}{s + a}
\]  

(73)

The conventional ESO and control law design is given as

\[
\begin{bmatrix}
\dot{z}_1 \\
\dot{z}_2 \\
\end{bmatrix} = \begin{bmatrix}
0 & 1 \\
0 & 0 \\
\end{bmatrix} \begin{bmatrix}
z_1 \\
z_2 \\
\end{bmatrix} + \begin{bmatrix}
b \\
0 \\
\end{bmatrix} u + \begin{bmatrix}
2\omega_o^2 \\
\omega_o^2 \\
\end{bmatrix} (y - z_1)
\]  

(74)

and

\[
\frac{u}{b} = \frac{\omega_c (r - z_1) - z_2}{b}
\]  

(75)

respectively, where \( z_1 \) estimates the system output \( y \), \( z_2 \) estimates the total disturbance, \( \omega_c \) and \( \omega_o \) are the controller and observer bandwidth. To study the stability margin of the system, the equivalent closed-loop system is shown in Figure 14 in transfer function form where

\[
G_e(s) = \frac{1}{b} \frac{\omega_o (2\omega_c s + \omega_o s + \omega_c \omega_o)}{(s + 2\omega_o + \omega_c)s}
\]  

(76)

and

\[
H(s) = \frac{\omega_c (s^2 + 2\omega_o s + \omega_o^2)}{\omega_o (2\omega_c s + \omega_o s + \omega_c \omega_o)}
\]  

(77)
For simplicity, assume $\omega_c = \omega_o = w$ and let $s = j\omega$, then we have
\[
\left| \left( G_c(j\omega)G_p(j\omega) \right)^2 \right| = \frac{w^2(w^4 + 9w^2\omega^2)}{(9w^2 + \omega^2)(\omega^2(a^2 + \omega^2))} \tag{78}
\]

To calculate the phase margin and delay margin, let $\left| \left( G_c(j\omega)G_p(j\omega) \right)^2 \right| = 1$ and solve it for the real positive solution,
\[
\left( 9w^2 + \omega^2 \right)w^2(a^2 + \omega^2) - w^2(w^4 + 9w^2\omega^2) \\
= \omega^6 + (a^2 + 9w^2)\omega^4 + (9a^2w^2 - 9w^4)\omega^2 - w^6 \\
= 0
\tag{79}
\]

Equation (80) is not easy to solve in general, but in the case of $w \gg a$, or the closed-loop bandwidth is much higher than the open loop one, it can be simplified as
\[
\omega^6 + 9w^2\omega^4 - 9w^4\omega^2 - w^6 = 0
\tag{80}
\]
which indicates that $\omega$ and $w$ are of the same order of magnitude. We can then assume that $\omega = k \cdot w$ and equation (80) becomes
\[
\left( k^6 + 9k^4 - 9k^2 - 1 \right)w^6 = 0
\tag{81}
\]
Since $w \neq 0$, we have
\[
k^6 + 9k^4 - 9k^2 - 1 = 0
\tag{82}
\] and the only real positive solution of this equation is $k = 1$. This means that the phase and delay margin of the system can be calculated at the frequency of $\omega \approx w$, where
It can be seen that $\angle(a-j\omega) > -\frac{\pi}{2}$ and $\angle(8-6j) = -0.6435$. Then the phase margin can be calculated as

$$\text{PM}_1 = \pi + \angle(G_c(j\omega)G_p(j\omega)) = \pi - \frac{\pi}{2} - 0.6435 = 0.9273 \text{ rad} = 53.13^\circ$$

The delay margin is defined as the amount of transport delay the control system can tolerate before the closed-loop system become unstable. It can be determined as

$$DM \approx \frac{\text{PM}}{w} > \frac{0.9273}{w}$$

Therefore, it is concluded that for the first-order system with second-order ESO, the minimum phase margin is $\text{PM} > 53.13^\circ$ when the closed-loop bandwidth is much higher than the open loop one. In the presence of time delay $t_d$, the closed-loop stability is preserved only when $w < \frac{\text{PM}}{t_d}$ or

$$w < \frac{0.9273}{t_d}$$

This establishes the upper bound on the bandwidth in the presence of time delay for the conventional ADRC design with the first-order plant of equation (74). Note that the achievable bandwidth is severely limited when the time delay is large.

### 4.1.2 Second-Order System

For the second-order system

$$G_p(s) = \frac{Y(s)}{U(s)} = \frac{b}{s^2 + a_2s + a_0}$$
The conventional ESO is

\[
\begin{bmatrix}
  \dot{z}_1 \\
  \dot{z}_2 \\
  \dot{z}_3
\end{bmatrix} =
\begin{bmatrix}
  0 & 1 & 0 \\
  0 & 0 & 1 \\
  0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
  z_1 \\
  z_2 \\
  z_3
\end{bmatrix} +
\begin{bmatrix}
  0 \\
  b \\
  0
\end{bmatrix}
\begin{bmatrix}
  3\omega_0 \\
  3\omega_o^2 \\
  \omega_o^3
\end{bmatrix}(y - z_1)
\]

(88)

And the control law is

\[
\begin{align*}
u &= \frac{\omega_o^2(r - z_1) - 2\omega_c z_2 - z_3}{b}
\end{align*}
\]

(89)

respectively, where \( z_1 \) estimates the system output \( y \), \( z_2 \) the derivative of output, \( z_3 \) estimates the equivalent total input disturbance, \( \omega_c \) and \( \omega_o \) are the controller and observer bandwidth. To study the stability margin, the closed loop system is again transformed to the equivalent transfer function form in Figure 14 where

\[
G_c(s) = \frac{1}{b} \frac{\omega_o \left( 3s^2\omega_c^2 + 3s\omega_c\omega_o + 6s^2\omega_c\omega_o + \omega_o^2\omega_c^2 + \omega_o^2s^2 + 2\omega_o^2s\omega_c \right)}{s \left( 3\omega_o^2 + \omega_c^2 + s^2 + 2s\omega_c + 3\omega_o + 6\omega_c\omega_o \right)}
\]

(90)

and

\[
H(s) = \frac{\omega_o^2 \left( 3s\omega_o^2 + s^2 + 3s^2\omega_o + \omega_o^3 \right)}{\omega_o \left( 3s^2\omega_c^2 + 3s\omega_c\omega_o + 6s^2\omega_c\omega_o + \omega_o^2\omega_c^2 + \omega_o^2s^2 + 2\omega_o^2s\omega_c \right)}
\]

(91)

For simplicity, let \( \omega_c = \omega_o = w \) and we have

\[
\left| \left( G_c(j\omega)G_p(j\omega) \right)^2 \right| = \frac{w^2 \left( 100\omega^4w^4 + 5\omega^2w^6 + w^8 \right)}{\omega^2 \left( 100w^4 + 5\omega^2w^2 + \omega^4 \right) \left( \omega^4 - 2\omega^2a0 + a0^2 + a1^2\omega^2 \right)}
\]

(92)

Here the positive real solution of the equation \( \left| \left( G_c(j\omega)G_p(j\omega) \right)^2 \right| = 1 \) gives us the frequency where the phase margin and delay margin are determined, i.e.

\[
\omega^10 + \left( 5w^2 - 2a_0 + a_1^2 \right)\omega^8 + \left( 100w^4 - 10a_0 w^2 + 5w^2a_1^2 + a_0^2 \right)\omega^6 \\
+ \left( -200w^4a_0 + 100w^4a_1^2 + 5w^2a_0^2 - 100w^6 \right)\omega^4 + \left( 100w^4a_0^2 - 5w^8 \right)\omega^2 - w^{10} = 0
\]

(93)

When \( w \gg a_0 \), and \( w \gg a_1 \), the equation could be simplified to
\[ \omega^0 + 5w^2\omega^6 + 100w^4\omega^6 - 100w^6\omega^4 - 5w^8\omega^2 - w^0 = 0 \]  

indicating that \( \omega \) and \( w \) are of the same order of magnitude. We can then assume that \( \omega = k \cdot w \) and equation (80) becomes

\[ \left( k^0 + 5k^8 + 100k^6 - 100k^4 - 5k^2 - 1 \right)w^0 = 0 \]

since \( w \neq 0 \), we have

\[ k^0 + 5k^8 + 100k^6 - 100k^4 - 5k^2 - 1 = 0 \]

And, again, the only real positive solution of this equation is \( k = 1 \). This way we know that the phase and delay margin of the system can be calculated with \( \omega \approx w \). We can then calculate the value of \( G_c(j\omega)G_p(j\omega) \) as

\[
G_c(j\omega)G_p(j\omega) \approx \frac{-j\left(-9w^4 + 5jw^4\right)}{\left(9w^2 + 5jw^2\right)\left(-w^2 + a_1w + a_0\right)} = \frac{(90 + 56j)(a_0 - w^2 - a_1w)w^2}{106\left((a_0 - w^2)^2 + (a_1w)^2\right)}
\]

We have \( \angle(a_0 - w^2 - a_1w) > -\pi \) and \( \angle(90 + 56j) = 0.5566 \), we can then calculate the phase margin as

\[ PM_2 = \pi + \angle(G_c(jw)G_p(jw)) > \pi - \pi + 0.5566 = 0.5566 \text{ rad} = 31.89^\circ \]

and the delay margin as

\[ DM \approx \frac{PM_2}{w} > \frac{0.5566}{w} \]

Therefore, it is concluded that for the second-order system with third-order ESO, the minimum phase margin is \( PM_2 > 31.89^\circ \) and, similarly, for the sake of stability the closed-loop bandwidth must satisfy \( w < \frac{PM_2}{t_d} \) in the presence of time delay \( t_d \), or

\[ w < \frac{0.5566}{t_d} \]
4.1.3 Higher Order System

In this section, we will discuss the phase and delay margins for higher order systems with conventional ADRC controller. For system

\[
G_p(s) = \frac{Y(s)}{U(s)} = \frac{b}{s^n + \sum_{i=0}^{n-1} a_i s^i}
\]  

(101)

The conventional ESO and control law design can be given as

\[
\dot{z} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}_{(n+1\times n+1)} z + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ b \end{bmatrix}_{(n+1\times 1)} u + \begin{bmatrix} \beta_0 \omega_c \\ \beta_1 \omega_c^2 \\ \beta_2 \omega_c^3 \\ \vdots \\ \beta_n \omega_c^n \\ \beta_{n+1} \omega_c^{n+1} \end{bmatrix}_{(n\times 1)} (y - z) 
\]  

(102)

where

\[
\beta_i = \frac{(n+1)!}{i!(n+1-i)!}, 1 \leq i \leq n+1
\]  

(103)

and

\[
u = \frac{k_r - \sum_{i=1}^{n+1} k_i z_i}{b}
\]  

(104)

respectively, where the controller gains \( k_i \) are designed as

\[
k_i = \frac{n!}{(i-1)!(n+1-i)!} \omega_c^{n+1-i}, 1 \leq i \leq n+1
\]  

(105)

\( z_1 \) estimates the system output \( y \), \( z_2 \) through \( z_n \) the first to \( (n-1)^{th} \) order derivative of output, \( z_{n+1} \) estimates the equivalent total input disturbance, \( \omega_c \) and \( \omega_o \) are the controller and observer bandwidth. In order to discuss the stability margin of the system, the closed loop system can be reformulated into the form in Figure 14. Here the \( G_c(s) \) and \( H(s) \) for third through sixth order system is given.
For third order ADRC:

\[
G_c(s) = \frac{1}{b} \frac{\omega_o \left( 12s^3 \omega_o^2 \omega_c + 18s^3 \omega_o \omega_c^2 + \omega_o^3 \omega_c^3 + 6\omega_c^3 \omega_o s^2 + 4\omega_c^3 \omega_o^3 \right)}{s^3 + 4s^2 \omega_o + 6s\omega_o^2 + 3\omega_o^3 s + 12\omega_o^2 \omega_c s + 18\omega_o^2 \omega_c^2 + 12\omega_o \omega_c^2 + 4\omega_c^3} \quad (106)
\]

\[
H(s) = \frac{\omega_c^3 \left( s^4 + 4s^3 \omega_o + 6s^2 \omega_o^2 + 4s\omega_o^3 + \omega_o^4 \right)}{\omega_o \left( 12s^2 \omega_o^2 \omega_c + 18s^2 \omega_o \omega_c^2 + \omega_o^3 \omega_c^3 + 6\omega_c^3 \omega_o s + 4\omega_c^3 \omega_o^3 \right)} \quad (107)
\]

For fourth order ADRC:

\[
G_c(s) = \frac{1}{b} \frac{\omega_o \left( \omega_c^4 \omega_o^4 + s^4 \omega_o^4 + 5s^4 \omega_c^4 + 5s^2 \omega_o^3 \omega_c^2 + 40s^4 \omega_c^3 \omega_o \right)}{s^4 + 5s^3 \omega_o + 10s^2 \omega_o^2 + 10s\omega_o^3 + 4\omega_c s^2 + 20\omega_c \omega_o^2} \quad (108)
\]

\[
H(s) = \frac{\omega_c^4 \left( 10s^2 \omega_o^3 + 5\omega_o^4 s + \omega_c^5 + s^5 + 5s^4 \omega_o + 10s^2 \omega_o^2 \right)}{\omega_o \left( \omega_c^4 + s^4 \omega_o^4 + 5s^4 \omega_c^4 + 5s^2 \omega_o^3 \omega_c^2 + 40s^4 \omega_c^3 \omega_o + 4s^3 \omega_o^3 \omega_c \right)} \quad (109)
\]

For fifth order ADRC:
For sixth order ADRC:

\[ G_c(s) = \frac{1}{b} \left( \frac{60s^4 \omega_o^5 + 10s^2 \omega_o^5 \omega_o^2 + 30s \omega_o^5 \omega_o^4 + 150s^5 \omega_o^2 \omega_o^3}{s^5 + 6s^4 \omega_o^2 + 15s^3 \omega_o^3 + 20s^2 \omega_o^4 + 15s \omega_o^4 \omega_o^5 + 5 \omega_o^5} + 7s \omega_o^5 \omega_o^3} + 200 \omega_c^2 s^2 \omega_o^2 + 75 \omega_c^4 s^2 \omega_o^4 + 15 \omega_c^5 s^2 \omega_o^5 + 6 \omega_c^6 \omega_o^4 \right) \]  

(110)

\[ H(s) = \frac{20s^3 \omega_o^5 + 15s^2 \omega_o^5 + 6s^3 \omega_o^5 + 6s^5 \omega_o^5 + 15s^4 \omega_o^5}{6s^4 \omega_o^5 + 10s^2 \omega_o^5 + 30s^3 \omega_o^5 + 150s^5 \omega_o^2 \omega_o^3} \]  

(111)

For sixth order ADRC:

\[ G_c(s) = \frac{1}{b} \left( \frac{15s^5 \omega_o^3 + 105s^4 \omega_o^3 \omega_o^2 + 6s^5 \omega_o^5 + 42s^4 \omega_o^2 \omega_o^3 + 7s^5 \omega_o^5}{126s^5 \omega_o^4 + 24 \omega_c^2 s^4 \omega_o^3 + 525s^5 \omega_o^3 + 420s^4 \omega_o^2 + 420s^5 \omega_o^3 + 315s^4 \omega_o^2 + 210s^5 \omega_o^3} + 210s^4 \omega_o^2 + 35s^5 \omega_o^3 + 35s^6 \omega_o^4 + 35s^5 \omega_o^3 + 35s^6 \omega_o^4 + 35s^5 \omega_o^3 + 35s^6 \omega_o^4 + 35s^5 \omega_o^3 + 35s^6 \omega_o^4 \right) \]  

(112)
Then the positive real solution of the equation $\left| \left( G_c(j\omega)G_p(j\omega) \right)^2 \right| = 1$ for each system order is solved. As long as the controller and observer bandwidths are sufficiently high, we have

$$\omega \approx w$$  \hspace{1cm} (114)

for all systems. Then the phase margins of the systems can be calculated.

For third order system:

$$G_c(j\omega)G_p(j\omega) \approx \frac{-j \left( -28j\omega^6 - 20\omega^6 \right)}{(20j\omega^3 + 28\omega^3)(-j\omega^3 - a_2\omega^2 + ja_1\omega + a_0)} \hspace{1cm} (115)$$

$$PM \approx 180 - 270 + 108.9246 = 18.9246$$  \hspace{1cm} (116)

For fourth order system:

$$G_c(j\omega)G_p(j\omega) \approx \frac{-j \left( 91\omega^8 - 75j\omega^8 \right)}{(91\omega^4 + 75j\omega^4)(\omega^4 - ja_2\omega^3 - a_2\omega^2 + ja_1\omega + a_0)} \hspace{1cm} (117)$$

$$PM \approx 180 + 0 - 168.99 = 11.01^\circ$$  \hspace{1cm} (118)

For fifth order system:

$$G_c(j\omega)G_p(j\omega) \approx \frac{-j \left( 276\omega^{10} + 308j\omega^{10} \right)}{(276j\omega^5 + 308\omega^5)(j\omega^5 + a_4\omega^4 - ja_3\omega^3 - a_2\omega^2 + ja_1\omega + a_0)} \hspace{1cm} (119)$$

$$PM \approx 180 - 90 - 83.7273 = 6.2727^\circ$$  \hspace{1cm} (120)
For sixth order system:

\[
G_c(j\omega)G_p(j\omega) \approx \frac{-j(-1078w^{12} + 1014jw^{12})}{(1014jw^6 + 1078w^6)(-w^6 + j\alpha_2w^3 + \alpha_4w^4 - j\alpha_5w^3)}
\]  

(121)

\[
PM \approx 180 - 180 + 3.5046 = 3.5046^\circ
\]

(122)

### 4.2 Phase and Delay Margin of the Custom Designed ADRC

The ADRC controller is known for its ability to deal with internal and external disturbances without the need for an accurate system model. However, with the help of some offline information on partial system dynamics, the ESO performance can be improved greatly with limited bandwidth. In this section, an example of using offline information in the ESO design is presented. The stability margins of ESO designed with accurate system dynamics information is discussed.

For an n-th order system

\[
\frac{Y(s)}{U(s)} = \frac{b}{s^n + a_{n-1}s^{n-1} + \cdots + a_1s + a_0}
\]

(123)

the differential equation representation of the system can be given as

\[
y^{(n)} = bu - a_1y - a_2\dot{y} - \cdots - a_{n-1}y^{(n-1)}
\]

(124)

Define states as

\[
X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n^{(n-1)} \end{bmatrix} = \begin{bmatrix} y \\ \dot{y} \\ \vdots \\ y^{(n-1)} \end{bmatrix}
\]

(125)

and the state-space representation of the system can be given as
Define the total disturbance an extended state as

\[ f = x_{n+1} = -a_0 x_1 - a_1 x_2 - \cdots - a_{n-1} x_n \]  \hfill (127)

we have

\[ \dot{x}_n = bu - a_0 x_1 - a_1 x_2 - \cdots - a_{n-1} x_n = x_{n+1} + bu \]  \hfill (128)

and the derivative of extended state can be given as

\[ \dot{x}_{n+1} = -a_0 \dot{x}_1 - a_1 \dot{x}_2 - \cdots - a_{n-1} \dot{x}_n = -a_0 x_2 - a_1 x_3 - \cdots - a_{n-2} x_n - a_{n-1} (x_{n+1} + bu) \]  \hfill (129)

the new extended state system can then be written as

\[ \dot{X} = A_{ma} X + B_{ma} u \]
\[ y = C_{ma} X \]  \hfill (130)

where

\[ A_{ma} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & -a_{n-2} & -a_{n-1} & \cdots & 0 \\ -a_0 & -a_1 & \cdots & -a_{n-2} & -a_{n-1} & 0 \end{bmatrix}, \quad B_{ma} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \vdots \\ -a_{n-1} b_{(n+1)} \end{bmatrix} \]

and

\[ C_{ma} = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}_{1 \times (n+1)}. \]  Then the ESO can be designed as

\[ \dot{Z} = A_{ma} Z + B_{ma} u + L_{ma} (y - z_1) \]  \hfill (131)

where the observer gains \( L_{ma} \) are designed so that the eigenvalues of the matrix \( (A_{ma} - L_{ma} C_{ma}) \) are all placed at \(-\omega_o\). Here \( Z \) estimates \( X \) and, more importantly, \( z_{n+1} \) estimates the total disturbance \( f \).
The dynamics of the system can then be reformatted to a perfect cascade integrator as

\[ \frac{Y(s)}{U_0(s)} = \frac{1}{s^n} \]  

(132)

with the control law

\[ u = \frac{u_0 - z_{n+1}}{b} \]  

(133)

The feedback control law can be designed as

\[ u_0 = k_r r - \sum_{i=1}^{n} k_i z_i \]  

(134)

where the controller gains \( k_i \) are designed as

\[ k_i = \frac{n!}{(i-1)(n+1-i)!} \omega_c^{n+1-i}, 1 \leq i \leq n \]  

(135)

and \( \omega_c \) is the controller bandwidth.

The closed-loop transfer function of the system with the given ESO and control law design will then be

\[ \frac{Y(s)}{R(s)} = \frac{\omega_c^n}{(s + \omega_c)^n} \]  

(136)

For the first-order system in (17), ESO design in (20) and control law design in (21), system can be reduced to (22), which can be controlled with

\[ u_0 = \omega_c (r - z_1) \]  

(137)

The observer gain in (20) can be designed as

\[ l_i = 2\omega_o - a \]  

\[ l_2 = (\omega_o - a)^2 \]  

(138)
making observer bandwidth $\omega_o$. Substituting the plant, ESO and control law design together, the system can be reconstructed into transfer function form as shown in Figure 14, where

$$G_c(s) = \frac{1}{b} \left( \frac{-\omega_c a + 2\omega_c \omega_o + a^2 - 2a\omega_o + \omega_o^2}{s + 2\omega_o + \omega_c - a} \right) s + \omega_o^2 \omega_c$$

(139)

and

$$H(s) = \frac{s^2 + 2s\omega_c s + \omega_c^2}{(-\omega_c a + 2\omega_c \omega_o + a^2 - 2a\omega_o + \omega_o^2) s + \omega_o^2 \omega_c}$$

(140)

Let $\omega_c = \omega_o = w$ and $s = j\omega$, then we have

$$\left| (G_c(j\omega)G_p(j\omega)) \right|^2 = \frac{w^6 + 15w^2\alpha^2 - 18w^2a^2 - 6w^2\alpha^3 + 9w^4\omega^2 + a^4\omega^2}{(\alpha^2 - 6aw + 9w^2 + \omega^2)(\alpha^2 + \omega^2)} \omega^2$$

(141)

To calculate the phase margin and delay margin, let $\left| (G_c(j\omega)G_p(j\omega)) \right|^2 = 1$ and solve it for the real positive solution. In the case of $w \gg a$, or the closed-loop bandwidth is much higher than the open loop one, we can get the only solution as $\omega = w$. The phase and delay margin of the system can then be calculated at the frequency of $\omega = w$, where

$$G_c(j\omega)G_p(j\omega) = \frac{\left( -3jw^2\alpha + 3jw^3 + ja^2w + w^3 \right)}{(-jw + a - 3w)(jw + a)}$$

$$\quad \quad = \frac{-6w^4 + 6a^3w - 13a^2w^2 + 12aw^3 - a^4 - 2jw^2(a^2 - 3aw + 4w^2)}{(\alpha^2 - 6aw + 10w^2)(\alpha^2 + w^2)}$$

(142)

In the case of $w \gg a$, we can calculate the phase margin as

$$PM = \pi + \angle (G_c(j\omega)G_p(j\omega)) \approx 0.9273 \text{ rad} = 53.13^\circ$$

(143)

which is the same as the phase margin of conventional ADRC design when $w \gg a$. This shows that incorporating the offline information in the ADRC design does not affect the phase and delay margin when the controller and observer bandwidth are sufficiently large.
4.3 Simulation Verification of the Custom Designed ADRC

As discussed in the previous chapters, the offline information on system dynamics could be used during the ESO design. The information processing performance could be increased with the same set of controller and observer bandwidth. However, it is an important question whether the actual system bandwidth increased or not since higher system bandwidth will lead to higher noise sensitivity and requires higher sampling rate in discrete implemented. In this section, we will try to illustrate the performance increase and the actual bandwidth of the closed-loop system in a case study.

For a second-order system

\[
\frac{Y(s)}{U(s)} = \frac{b}{s^2 + a_1 s + a_0}
\]  

(144)

the conventional ESO can be written as

\[
\dot{Z} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} Z + \begin{bmatrix} 0 \\ b \\ 0 \end{bmatrix} u + L(y - z_1)
\]

(145)

and \( L \) can be designed as

\[
L = \begin{bmatrix} 3\omega_o \\ 3\omega_o^2 \\ \omega_o^3 \end{bmatrix}
\]

(146)

to place all eigenvalues at \(-\omega_o\).

To use the offline information in the controller design, part of the model information can be incorporated to enhance the estimation performance. The differential equation representation of the system (144) can be written as

\[
\dot{y} = -a_0 y - a_1 \dot{y} + bu
\]

(147)

Define

\[
f = -a_0 y - a_1 \dot{y}
\]

(148)
the system can be written as $\dot{y} = f + bu$. Assume the first term $f_1 = -a_0y$ is unknown and the second term $f_2 = -a_1\dot{y}$ is known, we have

$$\dot{y}_2 = -a_1\dot{y} = -a_1(f + bu)$$

(149)

Define the states as

$$X = \begin{bmatrix} y \\ \dot{y} \\ f \end{bmatrix}$$

(150)

the system can be rewritten as

$$\dot{X} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -a_1 \end{bmatrix} X + \begin{bmatrix} 0 \\ b \\ -a_1b \end{bmatrix} u$$

(151)

and the custom designed ESO can be designed as

$$\dot{Z} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -a_1 \end{bmatrix} Z + \begin{bmatrix} 0 \\ b \\ -a_1b \end{bmatrix} u + Lc(y - z_1)$$

(152)

Then the observer gain $Lc$ can be designed as

$$Lc = \begin{bmatrix} -a_1 + 3\omega_o \\ a_1^2 - 3a_1\omega_o + 3\omega_o^2 \\ 3a_1^2\omega_o - 3a_1\omega_o^2 + \omega_o^3 - a_1^3 \end{bmatrix}$$

(153)

to make observer bandwidth $\omega_o$.

To discuss the system bandwidth, we can derive the closed loop transfer function of the two solutions and calculate their bandwidths. The closed-loop transfer function for conventional ADRC can be derived as
The closed-loop transfer function for the custom designed ADRC can be derived as

\[
\frac{Y}{R} = \frac{\omega_c^2 \left( 3s\omega_o^2 + s^3 + 3s^2\omega_o + \omega_o^3 \right)}{3\omega_c^2 s^3 + 3\omega_c^2 \omega_o a_0 + s^2 \omega_c^2 \omega_o a_0 + s \omega_c^2 \omega_o a_0 + s^3 + s^4 a_1 + 2s^4 \omega_c a_1 + 2s^2 \omega_o a_0 + 3s^4 \omega_o + 3s^3 \omega_o a_1 + 3s^2 \omega_o a_0 + 6s^3 \omega_c \omega_o a_1 + 6s \omega_c \omega_c a_0 + 3\omega_c^3 \omega_o a_1 + 3s^2 \omega_c a_0 + 3s^2 \omega_c a_0 + 3s^2 \omega_c a_0 + 6s^2 \omega_c \omega_o a_1 + 2s^3 \omega_c a_0 + 2s^3 \omega_c a_0 + 2s^3 \omega_c a_0}
\]

We can then discuss the bandwidth of both systems.

To discuss the performance and system bandwidth of the two solutions, we choose a sample system as

\[
SYS = \frac{10}{s^2 + 3s + 1}
\]

and the controller and observer bandwidths are chosen as \(\omega_c = 4, \omega_o = 20\). The system response is given in Figure 15, where the red line is the response of the conventional ADRC and the blue line is the response of the ADRC designed with offline information. We can see that the green line does not have overshoot as the blue line and gets into steady state faster.
On the other hand, the closed-loop system bandwidth of conventional ADRC and custom designed ADRC are 2.5278 and 2.4807 respectively. This shows that the system bandwidth is not increased when offline information is incorporated in the ESO design.
Chapter V
C当前和速度调节设计

Based on the PMSM control problems described in Section 2.2 and the existing solutions summarized in Section 2.3.3, we will discuss the current and velocity control of PMSM as a case study in terms of the selection, extraction and processing of information. In particular, we will reformulate the PMSM current and velocity control problems in the framework of ADR and offer a new perspective in addressing the difficulties in decoupling, uncertain and time-varying dynamics, and disturbance rejection.

First, the mathematical model is given to describe both the electrical and mechanical dynamics of PMSM. Then, a set of Extended State Observer (ESO) designs with different methods of using known dynamics and coupling information of the PMSM are discussed, followed by the introduction of an ADR based solution for velocity control using only position feedback. In addition, the use of motion profile in feedforward control is discussed. Finally, the proposed controllers are verified in simulation and hardware experiment.
5.1 PMSM Characteristics

Although the ADR based controller design does not require a detailed model of the target system, some basic offline information, such as the order and the high-frequency gain, is still needed. The ESO could take care of the rest of the unknown dynamics and the external disturbances by estimating them as an equivalent total input disturbance and negating their effects using the control signal. In addition, the information processing quality of the ESO could be enhanced by taking advantage of the available offline information such as the coil resistance and inductance.

As we had already discussed earlier that the three phase currents of the motor winding could be defined as space vector and be projected into a rotating orthogonal reference frame, where direct axis and quadrature axis separated the magnetization and torque generation effect of the winding current and made it possible to analyze and control them independently. Based on this idea, the phase voltage applied to the stator winding could also be projected as

\[
\begin{align*}
\frac{v_a}{v_b} &= v_a \\
\frac{v_b}{v_b} &= \frac{1}{\sqrt{3}} v_a + \frac{2}{\sqrt{3}} v_b
\end{align*}
\]

(157)

and

\[
\begin{align*}
v_d &= v_a \cos \theta + v_b \sin \theta \\
v_q &= -v_a \sin \theta + v_b \cos \theta
\end{align*}
\]

(158)

with \(v_a\) and \(v_b\) as \(\alpha\) axis and \(\beta\) axis voltages, \(v_d\) and \(v_q\) as direct axis and quadrature axis voltages, respectively and \(\theta\) as the electrical position of the rotor.

Assuming that the motor under investigation has a round shape rotor and sinusoidal bank-EMF waveform, the law of physics governing the dynamics of the stator can be described as
where $i_d$ and $i_q$ are direct axis and quadrature axis currents, $\omega_e = \frac{P}{2} \omega_m$ is the electrical velocity of the motor, $\omega_m$ is the mechanical velocity of the motor, $P$ is the number of poles in the motor, $R$ is the stator phase resistance, $L_d$ and $L_q$ are d axis and q axis inductances, $d_d$ and $d_q$ are the effects of unknown dynamics and external disturbances in d axis and q axis, and $\psi_f$ is the rotor flux.

The mechanical dynamics of the motor could be described as

$$\dot{\omega}_m = \frac{1}{J} \left( T_e - B \omega_m - T_m \right)$$

(160)

where $T_e = \frac{3P}{4} \psi_f i_q$ is the electromagnetic torque, $B$ is the friction coefficient and $T_m$ is the load torque applied to the motor. When the motor is saturated, the electromagnetic torque $T_e$ could be calculated by magnetic co-energy as

$$T_e = -\frac{\partial W(i, \theta)}{\partial \theta}$$

(161)

where $W$ is the magnetic co-energy, $i$ is armature current, and $\theta$ is the angle of the rotor. The $W$ can be expressed as

$$W = W_i + W_{mi} + W_m$$

(162)

where $W_i$ is the magnetic co-energy from the armature current, $W_{mi}$ is the magnetic co-energy from the interaction of armature and the magnet, $W_m$ is the magnetic co-energy from the magnet. These three variables are governed by the law of physics in terms of
\[
\frac{dW_l}{d\theta} = \frac{1}{2} \frac{P}{2} \frac{dL}{d\theta} i^2 
\]  
(163)

\[
\frac{dW_{mi}}{d\theta} = \frac{P}{2} \frac{d\psi_f}{d\theta} i 
\]  
(164)

\[
\frac{dW_m}{d\theta} = T_{cogging} 
\]  
(165)

where \( T_{cogging} \) is the cogging torque. Then, based on equation (161), the electromagnetic torque can be calculated as

\[
T_e = \frac{1}{2} \frac{P}{2} \frac{dL}{d\theta} i^2 + \frac{P}{2} \frac{d\psi_f}{d\theta} i + T_{cogging} 
\]  
(166)

Note that the second term in (166) is the control signal; the first and third terms are part of the internal disturbance which is generally unknown.

### 5.2 Current Loop Design

Apart from the online information required to close the loop, including the phase current feedback and electrical angle of the rotor, there are many online and offline information that we could use in the controller design. First, since the dynamics of the PMSM is well modeled under the normal operating condition and the parameters can be extracted offline from the datasheet or measurement, we could certainly use the mathematical model of PMSM in the controller design. In addition, since the PMSM model has the electrical speed of the motor in its current equation, the velocity information could be extracted online from the motion loop.

To be sure, however, since the parameters of PMSM will vary under the high current condition due to heat and saturation, the offline information cannot describe all the dynamics of the motor in its operational range. Furthermore, there will be unknown external disturbances in the system during the motor operation. An ESO could be used to
extract the model uncertainties and external disturbance online as a total disturbance. This total disturbance is the key to ADR based controller design.

The phase lag is a big issue in the controller design, especially for systems with fast dynamics, such as the PMSM current loop. As a result, minimizing the phase lag is the key. In this section, three ADR based controller designs are presented, where the phase lag in the system is reduced progressively.

5.2.1 Keep all Dynamics in Total Disturbance

In the first solution, all the known and unknown dynamics are put in the total disturbance term. The offline information on system dynamics will be used in the ESO design to help information processing. Based on the mathematical model obtained in (159), let

\[
\begin{align*}
    f_d &= -\frac{R}{L_d} i_d + \frac{L_q}{L_d} \omega d + d_d \\
    f_q &= -\frac{R}{L_q} i_q - \frac{L_d}{L_q} \omega d - \frac{\psi_f}{L_q} \omega + d_q
\end{align*}
\]  

where \( f_d \) and \( f_q \) are the total disturbances in the dynamics of the \( d \) axis and \( q \) axis. Then, equation (159) can be rewritten as
\[
\begin{align*}
\dot{i}_d &= \frac{1}{L_d} v_d + f_d \\
\dot{i}_q &= \frac{1}{L_q} v_q + f_q \\
\dot{j}_d &= -\frac{R}{L_d} i_d + \frac{L_q}{L_d} \dot{\omega} j_q + \frac{L_q}{L_d} \omega j_d + h_d \\
&= -\frac{R}{L_d} \left( \frac{1}{L_d} v_d + f_d \right) + \frac{L_q}{L_d} \dot{\omega} j_q + \frac{L_q}{L_d} \omega j_d + \frac{1}{L_q} v_q + f_q + h_d \\
\dot{j}_q &= -\frac{R}{L_q} i_q - \frac{L_d}{L_q} \dot{\omega} j_d - \frac{L_d}{L_q} \omega j_d - \frac{\psi_f}{L_q} \dot{\omega} z + h_q \\
&= -\frac{R}{L_q} \left( \frac{1}{L_q} v_q + f_q \right) - \frac{L_d}{L_q} \dot{\omega} j_d - \frac{L_d}{L_q} \omega j_d - \frac{1}{L_d} v_d + f_d \frac{\psi_f}{L_q} \dot{\omega} z + h_q
\end{align*}
\] (168)

where \( h_d = \dot{d}_d \) and \( h_q = \dot{d}_q \). Then define the states as \( X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} i_d \\ i_q \\ f_d \end{bmatrix} \), input as \( U = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} v_d \\ v_q \end{bmatrix} \), output as \( Y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} i_d \\ i_q \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \), \( \Omega = \dot{\omega} \) and \( H = \begin{bmatrix} h_1 \\ h_2 \\ h_q \end{bmatrix} \), we could rewrite the system as

\[
\begin{align*}
\dot{x}_1 &= \frac{1}{L_d} u_1 + x_3 \\
\dot{x}_2 &= \frac{1}{L_q} u_2 + x_4 \\
\dot{x}_3 &= -\frac{R}{L_d} u_1 + \frac{1}{L_d} \omega u_2 - \frac{R}{L_d} x_3 + \frac{L_q}{L_d} \omega x_4 + \frac{L_q}{L_d} \dot{\omega} x_2 + h_1 \\
\dot{x}_4 &= -\frac{1}{L_q} \omega u_1 - \frac{R}{L_q} u_2 - \frac{L_d}{L_q} \omega x_3 - \frac{R}{L_q} x_4 - \frac{L_d}{L_q} \Omega x_1 - \frac{\psi_f}{L_q} \Omega + h_2
\end{align*}
\] (169)

Reconstruct equation (169) into state space form we have

\[
\dot{X} = AX + BU + EH + G\Omega
\]
\[Y = CX\] (170)
where \( A = \begin{bmatrix}
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
\frac{L_d}{L_q} \dot{\omega}_e & \frac{R}{L_d} \frac{L_q}{L_d} \dot{\omega}_e & 0 \\
-\frac{L_d}{L_q} \dot{\omega}_e & 0 & -\frac{L_d}{L_q} \dot{\omega}_e & -\frac{R}{L_q} \\
\end{bmatrix} \), \( B = \begin{bmatrix}
\frac{1}{L_d} & 0 \\
0 & 1 \\
-\frac{R}{L_d^2} & \frac{1}{L_d} \omega_e \\
-\frac{1}{L_q} \omega_e & -\frac{R}{L_q} \\
\end{bmatrix} \).

\[ C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad E = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \] and \( G = \begin{bmatrix} 0 \\ 0 \\ \omega_f \\ \frac{\omega_f}{L_q} \end{bmatrix} \). Then extended state observer can be constructed as

\begin{align*}
\dot{\hat{Z}} &= AZ + BU + G\Omega + L(Y - \hat{Y}) \\
\hat{Y} &= CZ
\end{align*}

(171)

With the observer gain \( L = \begin{bmatrix} l_{11} & l_{12} \\
l_{21} & l_{22} \\
l_{31} & l_{32} \\
l_{41} & l_{42} \end{bmatrix} \) selected appropriately to provide the estimated states as

\[ \begin{bmatrix} \hat{z}_1 \\ \hat{z}_2 \\ \hat{z}_3 \\ \hat{z}_4 \end{bmatrix} = \begin{bmatrix} \dot{\hat{x}}_1 \\ \dot{\hat{x}}_2 \\ \dot{\hat{x}}_3 \\ \dot{\hat{x}}_4 \end{bmatrix} \]. Most importantly, the third state of the observer \( z_3 \) approximates \( f_d \) and the fourth state \( z_4 \) approximates \( f_q \).

To make sure the observer is stable and to simplify the tuning process, we place all eigenvalues of the observer at \( -\omega_o \) and the observer gain \( L \) by solving the equation

\[ \left| sI - (A - LC) \right| = (s + \omega_o)^n \]

(172) and get
Then control law

\[ u_d = L_d (u_{0d} - z_3) \]  
\[ u_q = L_q (u_{0q} - z_4) \]  

(174)

reduces (159) approximately to

\[ i_d = u_{0d} \]  
\[ i_q = u_{0q} \]  

(175)

which could easily be controlled by a proportional controller.

One big issue with this design is the \( \dot{\omega}_e \) term in \( A \) and \( L \) is not normally directly measurable. Extracting \( \dot{\omega}_e \) from the position feedback information will normally lead to noisy signal and significant phase lag. To overcome this difficulty, we explore an alternative design next to eliminate the need for the \( \dot{\omega}_e \) measurement by taking part of the known dynamics out of the total disturbance term and putting it into ESO.
5.2.2 Remove Some Known Dynamics from Total Disturbance

In this solution, a part of the known dynamics obtained offline is used to alter the
system model used in the ESO design. By removing this part of the known dynamics
from the total disturbance, the need of $\dot{\omega}_e$ is eliminated. Based on the mathematical
model obtained in (159), we can redefine the disturbance as

$$f_d = -\frac{R}{L_d} i_d + d_d$$
$$f_q = -\frac{R}{L_q} i_q + d_q$$

Then, equation (159) can be rewritten as

$$\begin{cases}
  i_d = \frac{1}{L_d} v_d + \frac{L_q}{L_d} \omega j_q + f_d \\
  i_q = \frac{1}{L_q} v_q - \frac{L_d}{L_q} \omega j_d - \frac{\psi_f}{L_q} \omega_e + f_q \\
  \dot{j}_d = -\frac{R}{L_d} i_d + h_d \\
  = -\frac{R}{L_d} \left( \frac{1}{L_d} v_d + \frac{L_q}{L_d} \omega j_q + f_d \right) + h_d \\
  \dot{j}_q = -\frac{R}{L_q} i_q + h_q \\
  = -\frac{R}{L_q} \left( \frac{1}{L_q} v_q - \frac{L_d}{L_q} \omega j_d - \frac{\psi_f}{L_q} \omega_e + f_q \right) + h_q \\
\end{cases}$$

(177)

where $h_d = \dot{d}_d$ and $h_q = \dot{d}_q$. Define the states as $X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ \end{bmatrix}$, input as $U = \begin{bmatrix} u_1 \\ u_2 \\ \end{bmatrix}$, output as $Y = \begin{bmatrix} y_1 \\ y_2 \\ \end{bmatrix}$, $\Omega = \omega_e$ and $H = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ \end{bmatrix}$, we could rewrite the system as
Reconstruct equation (178) into state space form we have

\[
\begin{align*}
\dot{X} &= AX + BU + EH + G\Omega \\
Y &= CX
\end{align*}
\]

(179)

where

\[ A = \begin{bmatrix}
0 & \frac{L_q}{L_d} \omega_e & 1 & 0 \\
-\frac{L_d}{L_q} \omega_e & 0 & 0 & 1 \\
0 & -\frac{L_q R}{L_d} \omega_e & -\frac{R}{L_d} & 0 \\
\frac{L_d R}{L_q^2} \omega_e & 0 & 0 & -\frac{R}{L_q}
\end{bmatrix}, \\
B = \begin{bmatrix}
\frac{1}{L_d} & 0 \\
0 & \frac{1}{L_q} \\
-\frac{R}{L_d^2} & 0 \\
0 & -\frac{R}{L_q^2}
\end{bmatrix}, \quad C = \begin{bmatrix}
1 & 0 & 0 & 0
\end{bmatrix}
\]

\[ E = \begin{bmatrix}
0 & 0 \\
0 & 0 \\
1 & 0 \\
0 & 1
\end{bmatrix} \quad \text{and} \quad G = \begin{bmatrix}
0 & \frac{\psi_f}{L_q} \\
0 & 0 \\
0 & \frac{\psi_f R}{L_q^2}
\end{bmatrix}
\]

Then the extended state observer is constructed as

\[
\dot{Z} = AZ + BU + G\Omega + L(Y - \hat{Y})
\]

(180)

\[ \hat{Y} = CZ \]
with the observer gain \( L = \begin{bmatrix} l_{11} & l_{12} \\ l_{21} & l_{22} \\ l_{31} & l_{32} \\ l_{41} & l_{42} \end{bmatrix} \) selected appropriately to provide the estimated states as
\[
\begin{bmatrix}
    z_1 \\
    z_2 \\
    z_3 \\
    z_4
\end{bmatrix} = \begin{bmatrix}
    \dot{x}_1 \\
    \dot{x}_2 \\
    \dot{x}_3 \\
    \dot{x}_4
\end{bmatrix}.
\]
Most importantly, the third state of the observer \( z_3 \) approximates \( f_d \) and the fourth state \( z_4 \) approximates \( f_q \).

To make sure the observer is stable and to simplify the tuning process, we place all eigenvalues of the observer at \(-\omega_o\) and the observer gain \( L \) by solving equation
\[
\left| SL - (A - LC) \right| = (s + \omega_o)^4
\]
and this yields
\[
\begin{align*}
    l_{11} &= \frac{2L_d \omega_o - R}{L_d} \\
    l_{12} &= \frac{L_q \omega_e}{L_d} \\
    l_{21} &= 0 \\
    l_{22} &= \frac{2L_q \omega_o - R}{L_q} \\
    l_{31} &= \frac{-2R \omega_o L_d + R^2 + \omega_o^2 L_d^2}{L_d^2} \\
    l_{32} &= \frac{L_q R \omega_e}{L_d^2} \\
    l_{41} &= 0 \\
    l_{42} &= \frac{-2R \omega_o L_q + L_q^2 \omega_e^2 + R^2}{L_q^2}
\end{align*}
\]

Then the control law is given as
or

\[ u_d = I_d \left( u_{0d} - z_3 - \frac{L_e}{L_d} \omega_e z_2 \right) \]

\[ u_q = I_q \left( u_{0q} - z_4 + \frac{L_d}{L_q} \omega_e z_1 + \frac{\psi_f}{L_q} \omega_e \right) \]  

(183)

which reduces (159) to

\[ \dot{i}_d = u_{0d} \]

\[ \dot{i}_q = u_{0q} \]  

(185)

to be controlled simply by a proportional controller.

This solution removes the need for \( \dot{\omega}_e \) measurement. Note that the difference between (183) and (184) is the use of ESO output \( z_1 \) and \( z_2 \) versus the direct feedback signal \( i_d \) and \( i_q \). They are both current information, but the \( z_1 \) and \( z_2 \) extracted by ESO has been filtered, which means that using ESO output can reduce the noise level in the control signal, whereas using the direct feedback can reduce the phase lag in the feedback loop. Since the actuator in PMSM current loop is power switch, the noise level in the control signal is not a critical issue and the design in (184) design is adopted.

The above new design and comparison suggest that perhaps there is merit in PMSM control to use the direct feedback variable whenever possible, except the total disturbance estimation, so that the phase lag in the ESO output can be avoided. Based on this design principle, another solution is explored next, which takes all available offline information of system dynamics and moves it from the total disturbance to the plant.
model used in ESO. In doing so, ESO is tasked only to estimate the effect of the uncertain dynamics and external disturbances.

5.2.3 Remove all Known Dynamics from Total Disturbance

In this section, all the offline information of system dynamics are removed from the total disturbance, to further reduce the phase lag in ESO and the control law. Based on the mathematical model obtained in (159), we now redefine the total disturbance as

\[
\begin{align*}
   f_d &= d_d \\
   f_q &= d_q
\end{align*}
\] (186)

Note that, as stated above, \(d_d\) and \(d_q\) are the effects of unknown dynamics and external disturbances in d axis and q axis. Then, equation (159) can be rewritten as

\[
\begin{align*}
   \dot{i}_d &= \frac{1}{L_d} v_d - \frac{R}{L_d} i_d + \frac{L_q}{L_d} \omega_q \dot{i}_q + f_d \\
   \dot{i}_q &= \frac{1}{L_q} v_q - \frac{R}{L_q} i_q - \frac{L_d}{L_q} \omega_d - \frac{\psi_f}{L_q} \omega_e + f_q \\
   \dot{f}_d &= h_d \\
   \dot{f}_q &= h_q
\end{align*}
\] (187)

where \(h_d = \dot{d}_d\) and \(h_q = \dot{d}_q\).

Define the states as \(X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}\), input as \(U = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}\), and output as \(Y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}\), with \(\Omega = \omega_e\) and \(H = \begin{bmatrix} h_1 \\ h_2 \end{bmatrix}\), we could rewrite the system as
Reconstruct equation (188) into state space form we have

\[
\dot{X} = AX + BU + EH + G\Omega
\]

\[
Y = CX
\]

where

\[
A = \begin{bmatrix}
-\frac{R}{L_d} & \frac{L_q}{L_d} \omega_e & 1 & 0 \\
\frac{L_d}{L_q} \omega_e & -\frac{R}{L_q} & 0 & 1 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}, \quad B = \begin{bmatrix}
\frac{1}{L_d} & 0 \\
0 & \frac{1}{L_q} \\
0 & 0 \\
0 & 0
\end{bmatrix}, \quad C = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0
\end{bmatrix}, \quad E = \begin{bmatrix}
0 & 0 \\
0 & 0 \\
1 & 0 \\
0 & 1
\end{bmatrix}
\]

and \( G = \begin{bmatrix}
0 \\
\frac{\psi_f}{L_q} \\
0 \\
0
\end{bmatrix} \). Then extended state observer is constructed as

\[
\dot{Z} = AZ + BU + G\Omega + L \left( Y - \hat{Y} \right)
\]

\[
\hat{Y} = CZ
\]

with the observer gain \( L = \begin{bmatrix}
l_{11} & l_{12} \\
l_{21} & l_{22} \\
l_{31} & l_{32} \\
l_{41} & l_{42}
\end{bmatrix} \) selected appropriately to provide the estimated states of equation (189) as

\[
\begin{bmatrix}
z_1 \\
z_2 \\
z_3 \\
z_4
\end{bmatrix} = \begin{bmatrix}
\hat{x}_1 \\
\hat{x}_2 \\
\hat{x}_3 \\
\hat{x}_4
\end{bmatrix}
\]

Most importantly, the third state of the observer \( z_3 \) approximates \( f_d \) and the fourth state \( z_4 \) approximates \( f_q \).
To make sure the observer is stable and to simplify the tuning process, we place all eigenvalues of the observer at \(-\omega_o\) and the observer gain \(L\) by solving equation
\[
\begin{align*}
\left| sI - (A - LC) \right| &= (s + \omega_o)^4
\end{align*}
\]
and it yields
\[
\begin{align*}
&I_{11} = \frac{2L_d \omega_o - R}{L_d} \\
&I_{12} = 0 \\
&I_{21} = -\frac{L_q \omega_o}{L_q} \\
&I_{22} = \frac{2L_q \omega_o - R}{L_q} \\
&I_{31} = \omega_o^2 \\
&I_{32} = 0 \\
&I_{41} = 0 \\
&I_{42} = \omega_o^2
\end{align*}
\]
Then the control law can be constructed as
\[
\begin{align*}
u_d &= L_d \left( u_{\theta d} - z_3 + \frac{R}{L_d} i_d - \frac{L_q}{L_d} \omega_e i_q \right) \\
u_q &= L_q \left( u_{\theta q} - z_4 + \frac{R}{L_q} i_q + \frac{L_d}{L_q} \omega_e i_d + \frac{\psi_f}{L_q} \omega_e \right)
\end{align*}
\]
which reduces (159) to
\[
\begin{align*}
i_d &= u_{\theta d} \\
i_q &= u_{\theta q}
\end{align*}
\]
to be controlled by a proportional controller. It can be seen that this solution minimized the phase lag by using all direct feedback in the control law (193).
5.3 Velocity Loop Design

Since most modern motion system only uses position feedback, the velocity information needs to be extracted from position information online for speed control. The traditional method uses an approximate differentiator for the information extraction and closes the loop with a PI controller. This operation introduced an extra order into the loop system and more phase lag.

In this section, two ADR based design solutions will be proposed with velocity and total disturbance information both extracted by the ESO to eliminate the need for a separate differentiator. The first solution only uses the offline information such as inertia and torque constant, whereas the second solution uses additional offline information of the friction coefficient in its ESO design. As discussed in Chapter 3.1, all the offline information used here can be estimated using data-driven methods.

The motion of PMSM could be described based on Newton’s law as

\[ \dot{\omega}_m = \frac{1}{J} (T_e - B\omega_m - T_m) \]  

where \( T_e = \frac{3P}{4} \psi_f i_q \) is the electromagnetic torque, \( B \) is the friction coefficient and \( T_m \) is the load torque applied to the motor. Based on the law of physics, the relationship between the speed \( \omega_m \) and the position of the motor shaft \( \theta_m \) can be described as \( \dot{\theta}_m = \omega_m \). The mechanical system model (195) could be rewritten as

\[ \ddot{\theta}_m = \frac{3P}{4} \frac{\psi_f}{J} i_q - \frac{B}{J} \dot{\theta}_m - \frac{1}{J} T_m \]  

and the total disturbance \( f \) can be defined as

\[ f = -\frac{B}{J} \dot{\theta}_m - \frac{1}{J} T_m \]  

with which equation (196) could be rewritten as
where \( b = \frac{3P \psi_f}{4J} \) represents the input gain. Then the control law \( i_q = -\dot{f} + \dot{u}_0 \) reduces (198) to the cascade integral from \( \dot{\theta}_m = u_0 \) which can be controlled easily.

Define the states in the motion system as \( X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} \theta_m \\ \omega_m \\ f \end{bmatrix} \), input as \( u = i_q \), output \( y = \theta_m \) and \( h = \dot{f} \), we could rewrite the system as

\[
\dot{X} = AX + Bu + Eh \\
y = CX
\] (199)

where \( A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \), \( B = \begin{bmatrix} b \\ 0 \\ 0 \end{bmatrix} \), \( C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \) and \( E = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \). Then state observer

\[
\dot{Z} = AZ + Bu + L(y - \dot{y}) \\
\hat{y} = CZ
\] (200)

with the observer gain \( L \) selected appropriately to provide an estimate of the states of equation (200) as \( Z = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \end{bmatrix} \). Most importantly, the second state of the observer \( z_2 \) approximates \( \omega_m \) and the third state \( z_3 \) approximates \( f \).

To make sure the observer is stable and to simplify the tuning process, we place all eigenvalues of the observer at \(-\omega_o\) by selecting the observer gain \( L \) as

\[
L = \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix} = \begin{bmatrix} 3\omega_o \\ 3\omega_o^2 \\ \omega_o^3 \end{bmatrix}
\] (201)
As we discussed earlier, the $\frac{B}{J}$ value can easily be estimated in the form of $a \approx \frac{B}{J}$.

Incorporating this information in the controller design leads to better performance and lower bandwidth requirement. Rewrite (197) as

$$f = -a \dot{\theta}_m - \left(\frac{B}{J} - a\right) \dot{\theta}_m - \frac{1}{J} T_m$$

(202)

and (199) becomes

$$\dot{X} = A_i X + B_i u + E h_i$$
$$y = C X$$

(203)

where $A_i = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -a \end{bmatrix}$, $B_i = \begin{bmatrix} 0 \\ b \\ -ab \end{bmatrix}$ and $h_i = -\left(\frac{B}{J} - a\right) \dot{\theta}_m - \frac{1}{J} \dot{i}_m$. Then extended state observer is designed as

$$\dot{Z} = A_i Z + B_i u + L_1 (y - \hat{y})$$
$$\hat{y} = C Z$$

(204)

with the observer gain $L_1$ chosen as

$$L_1 = \begin{bmatrix} -a + 3 \omega_o \\ a^2 - 3a \omega_o + 3 \omega_o^2 \\ 3a^2 \omega_o - 3a \omega_o^2 + \omega_o^3 - a^3 \end{bmatrix}$$

(205)

For both ESO designs, the control law is the same

$$u = \frac{\omega_c (\omega_{ref} - z_2) - z_2}{b}$$

(206)

where $\omega_{ref}$ is the reference to the motor speed. This completes the velocity loop design, to be tested in simulation and experimentation as shown below.
5.4 Simulation Verification

In this section, the proposed current loop designs are put to test in simulation and compared against the widely used PI plus decoupling method, followed by the velocity loop test. All current loop designs use the same amount of offline information and were set at the same bandwidth. The parameters of the motor used in the simulations are shown in Table 1.

Table 1 Parameters of Motor in Simulation

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>d-Axes Inductance</td>
<td>$L_d$</td>
<td>0.0002 H</td>
</tr>
<tr>
<td>q-Axes Inductance</td>
<td>$L_q$</td>
<td>0.0002 H</td>
</tr>
<tr>
<td>Stator Phase Resistance</td>
<td>$R$</td>
<td>0.36 Ω</td>
</tr>
<tr>
<td>Rotor Flux</td>
<td>$\psi_f$</td>
<td>0.0191 V·s</td>
</tr>
<tr>
<td>Inertia</td>
<td>$J$</td>
<td>0.000004802 kg·m²</td>
</tr>
<tr>
<td>Number of Poles</td>
<td>$P$</td>
<td>8</td>
</tr>
</tbody>
</table>

5.4.1 Current loop Verification

In this section, the two ADR based controller design was put to the test against the commonly used industrial solution, the PI controller with model-based decoupling, as shown in (207)

\[
\begin{align*}
K_p_d &= L_d \omega_c \\
K_p_q &= L_q \omega_c \\
K_i &= R \omega_c \\
u_d &= K_p_d (r_d - i_d) + K_d \int (r_d - i_d) - i_q L_q \omega_c \\
u_q &= K_p_q (r_q - i_q) + K_d \int (r_q - i_q) + i_d L_d \omega_c + \psi_f \omega_c
\end{align*}
\]  

(207)

where $\omega_c$ is the controller bandwidth. With this solution, the closed-loop response becomes
when the parameters used in the controller design match that of the system model perfectly.

We can see that the PI control solution uses the same amount of offline information of system dynamics. When the motor is running under the normal condition, i.e. when the motor parameters are accurate, this solution will have excellent performance, as the zero in the PI controller cancels the effect of the pole in the system model and all the coupling between the two loops are canceled by the feedforward control. However, since there is no online extraction of the model information, the performance will deteriorate when the motor enters the high current condition, where the motor parameters vary due to temperature and saturation.

This design was used as a benchmark to test the two ADR based solutions in Chapter 5.2.2 (ADRC_H) and Chapter 5.2.3 (ADRC_P). All controller has their bandwidth set to \(2\pi \cdot 4000\) rad/s, and a periodic step signal was given as the q-axis current reference.
Figure 16  ADRC vs. PI plus Decoupling

Figure 17  ADRC vs. PI plus Decoupling Zoom-In
The simulation result is shown in Figure 16, and a zoom-in version is given in Figure 17. We can see that all three controllers have outstanding tracking and disturbance rejection performance when the motor is running in normal condition. The PI plus decoupling method works best since it compensated for all dynamics based on the offline information. The ADRC\textsubscript{P} has a very similar response, and the ADRC\textsubscript{H} response is slightly slower.

However, the motor parameters vary greatly throughout the operating range of the motor due to reasons such as temperature variations. To verify their performance under different working conditions, the three controllers are put to simulation with 30% motor parameter inaccuracy, which is not big at all in real motor operation.

![Figure 18 ADRC vs. PI plus Decoupling with Parameter Uncertainty](image)
The simulation result is shown in Figure 18, and a zoom-in version is given in Figure 19. We can see that the two ADR based solutions have very similar tracking and disturbance rejection performance with the simulation without parameter variation, whereas the PI plus decoupling performance becomes slower. The ADRC_P still has a slightly faster response than the ADRC_H. This confirms that the ADR based solutions have much better tolerance to parameter uncertainties because the model uncertainty information can be extracted online with ESO and used in the controller.

5.4.2 Velocity Loop Verification

In the industrial environment, one big challenge to motor control is to compensate for the load variation under normal operation. As discussed earlier, most motion control systems do not have direct velocity feedback. The velocity signal has to be extracted from the encoder reading or position estimator instead. The ESO in ADR based velocity
loop controller solved both problems by extracting the load variation information as a part of the total disturbance and the velocity information together from the position feedback and control signal in real time. In this section, the two ESO based velocity loop control designs in Chapter 5.3 are put to test in comparison with the industrial standard PI solution in the form of

\[
K_p = \frac{\omega_c}{b}
\]

\[
K_i = \frac{\omega_c}{b} a
\]

\[
i_{\text{ref}} = K_p (\omega_{\text{ref}} - \omega_m) + K_i \int (\omega_{\text{ref}} - \omega_m)
\]  \hspace{1cm} (209)

where the same amount of offline information as the second ADR based design is used.

The current loop was controlled with the ADRC_H solution.

The reference signal is initially set to 80, followed by a setpoint change to 160 at t=0.5 seconds. Then two step disturbances are introduced at t=0.8 and t=1.2 seconds respectively to test the disturbance rejection performance. The response of the three controllers are given in Figure 20, and a zoomed in version is given in Figure 21.
We can see that the PI controller has the slowest response in both tracking and disturbance rejection tasks. Both ADR based controllers perform well in the test, with the second design using more offline information and performing slightly better as expected.

5.5 Hardware Validation

To verify the proposed current and velocity loop solutions, a hardware testbed is designed and built. The hardware system has power and control subsystems and is equipped PLC-based protection system, which monitors the voltages and currents in real time and shuts the system down in emergency. The power subsystem includes the power entry and protection circuits and a power inverter generating the 300 V DC bus and an IGBT power converter to generate the three-phase power to the motor.

The heart of the control subsystem is a Texas Instruments TMS320F28377D dual-core MCU, equipped with Floating-Point Unit (FPU) and Trigonometric Math Unit
(TMU), and two programmable Control Law Accelerators, four 16-bit differential ADCs, eight PWM outputs with built-in synchronizing and dead-zone implementation. The phase voltage and current are measured with LEM current transformer and isolated sigma-delta modulator. The motor shaft position is measured with high-resolution sin-cos encoder and the absolute position can be read via an RS-485 channel in the encoder module.

A Rockwell Automation MPL-A330P-M motor with a rated torque of 4.18 Nm and a rated output power of 1.8 kW is chosen in the test, driving a MAGTOR dynamometer system HD-715-6N-0100 as the simulated load. The load comes with DSP7001 controller, which could run the load at both constant speed and constant torque mode.

A detailed description of the construction of the hardware testbed is given in Appendix A and the schematics of the designed motor drive PCBs are given in Appendix B. The parameters of the motor used in the simulations are shown in Table 2.

**Table 2** Parameters of Motor in Hardware Experiment

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>d-Axes Inductance</td>
<td>$L_d$</td>
<td>0.0002068772 H</td>
</tr>
<tr>
<td>q-Axes Inductance</td>
<td>$L_q$</td>
<td>0.0002068772 H</td>
</tr>
<tr>
<td>Stator Phase Resistance</td>
<td>$R$</td>
<td>0.3654691 Ω</td>
</tr>
<tr>
<td>Rotor Flux</td>
<td>$\psi_f$</td>
<td>0.04052209 V·s</td>
</tr>
<tr>
<td>Number of Poles</td>
<td>$P$</td>
<td>8</td>
</tr>
<tr>
<td>Sampling frequency</td>
<td>$f_s$</td>
<td>40 kHz</td>
</tr>
</tbody>
</table>

The current loop control solution in Chapter 5.2.3 and the velocity loop control solution in Chapter 5.3 are discretized with Tustin approximation, implemented in C
program and deployed in the main MCU. The current loop controller and observer bandwidths are set to $2\pi \cdot 4000$ rad/s and the velocity loop controller and observer bandwidths are set to $2\pi \cdot 200$ rad/s. The load dynamometer is set to constant torque mode, with load setting increased at $t = 2.1, 6.1$ and $9.1$ seconds. The result in Figure 22 shows that the proposed ADRC solutions work well as expected in hardware implementation.

![Graphs showing current, control signal, position, and eso value over time.](image)

**Figure 22** Hardware Experiment Result
As was discussed in the previous chapters, high-quality torque, speed, or position control of a PMSM requires knowledge of rotor shaft position. Traditional PMSM servo systems use resolvers or optical encoders with an index or absolution position to measure the position information online [21]. This is also true in high precision servo systems. However, more and more manufacturers are leaning toward sensorless control solutions in their new design. The so-called sensorless control reduces cost and system complexity and increases reliability by eliminating the position transducer by extracting the position information from online data such as the voltage and current data [22].

In this chapter, the fundamental idea of the back-EMF observer based sensorless control is discussed, followed by an ESO based solution that we proposed. Then, the effective bandwidth and limitation of a popular industrial solution, sliding mode observer, are discussed.

6.1 The Fundamental Idea of Back-EMF Observer Based Sensorless Control

As was discussed earlier in Chapter 2.2.3, Back-EMF observer is the most common way to extract the rotor position information from the voltage and current
signals online. The mathematical model of the dynamics from voltage to current in a PMSM can be written in the $\alpha - \beta$ reference frame as

$$i_\alpha = -\frac{R_s}{L_s} i_\alpha + \frac{1}{L_s} v_\alpha - \frac{1}{L_s} e_\alpha$$

$$i_\beta = -\frac{R_s}{L_s} i_\beta + \frac{1}{L_s} v_\beta - \frac{1}{L_s} e_\beta$$

(210)

with

$$\begin{bmatrix} e_\alpha \\ e_\beta \end{bmatrix} = \omega_r \lambda_m \begin{bmatrix} -\sin(\theta_r) \\ \cos(\theta_r) \end{bmatrix}$$

(211)

Since the currents and voltages are measurable information, we could acquire the value of $e_\alpha$ and $e_\beta$ from the equation (210) with various methods. Then from the equation (211), we can obtain the electrical angle of rotor $\theta_r$ independent of the value of $\omega_r$ and $\lambda_m$. The most straightforward method is arctangent calculation by

$$\theta_r = \arctan \left( \frac{-e_\alpha}{e_\beta} \right) + \frac{\omega_r \lambda_m}{|e_\beta|} \frac{\pi}{2}$$

(212)

An alternative way to calculate the $\theta_r$ is with phase lock loop (PLL), where the electrical velocity $\omega_r$ can be obtained with $\theta_r$ at the same time through a closed loop regulation where the tracking error of $\theta_r$ can then be calculated as

$$\begin{align*}
-e_\alpha \cos(\dot{\theta}_r) - e_\beta \sin(\dot{\theta}_r) \\
= \omega_r \lambda_m \sin(\theta_r) \cos(\dot{\theta}_r) - \omega_r \lambda_m \cos(\theta_r) \sin(\dot{\theta}_r) \\
= \omega_r \lambda_m \sin(\theta_r - \dot{\theta}_r) \approx \omega_r \lambda_m (\theta_r - \dot{\theta}_r)
\end{align*}$$

(213)

6.2 The Proposed ESO Based Sensorless Estimator

In this section, the back-EMF information $e_\alpha$ and $e_\beta$ are extracted by formulating them as states and estimating them online with state observers. Since the two
axes are independent and symmetrical, the following ESO are designed and applied to both axes. The differential equation of each axis could be reformulated as

\[
\begin{align*}
\dot{x}_1 &= -\frac{R_x}{L_x} x_1 - \frac{1}{L_x} x_2 + \frac{1}{L_x} v_y \\
\dot{x}_2 &= \dot{e}_s
\end{align*}
\]  

(214)

where \( x_1 = i_{sa}(i_{sβ}), x_2 = e_{sa}(e_{sβ}) \). Then the corresponding ESO is

\[
\begin{align*}
\dot{Z} &= \begin{bmatrix} \frac{R_x}{L_x} & -\frac{1}{L_x} \\ 0 & 0 \end{bmatrix} Z + \begin{bmatrix} \frac{1}{L_x} \\ \frac{1}{L_x} \end{bmatrix} v_y + \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} (i_y - \hat{i}_y) \\
\hat{i}_s &= [1 \ 0] Z
\end{align*}
\]  

(215)

where \( Z = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \) is the estimate of \( X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \). In particular, \( z_2 \) estimates \( e_s \) to be used in the angle calculation in (212). The observer gain

\[
L = \begin{bmatrix} l_1 \\ l_2 \end{bmatrix} = \begin{bmatrix} -\frac{R_x + 2\omega_o L_x}{L_x} \\ \frac{\omega_o^2 L_x}{-\omega_o^2 L_x} \end{bmatrix}
\]  

(216)

assign both observer eigenvalues at \(-\omega_o\).

To get better estimation with a limited sampling rate, nonlinear gains can be used in the ESO in the form of

\[
\begin{align*}
\dot{Z} &= \begin{bmatrix} \frac{R_x}{L_x} & -\frac{1}{L_x} \\ 0 & 0 \end{bmatrix} Z + \begin{bmatrix} \frac{1}{L_x} \\ \frac{1}{L_x} \end{bmatrix} v_y + \begin{bmatrix} fαl((i_y - \hat{i}_y),\alpha_1,\delta)l_1 \\ fαl((i_y - \hat{i}_y),\alpha_2,\delta)l_2 \end{bmatrix} \\
\hat{i}_s &= [1 \ 0] Z
\end{align*}
\]  

(217)

where the nonlinear function
with $0 < \alpha < 1$, can make the observer gain higher when error is small. This allows the error to reach zero more quickly. In addition, when $\alpha = 0$, the observer is similar to the sliding mode observer; when $\alpha = 1$, it corresponds to the linear ESO.

6.3 Simulation Verification

The ESO based position estimator is put to test in simulation in this section, using the same simulation platform in Chapter 5.4. The position estimation result is used to replace the position feedback signal.

With an initial position error of $55^\circ$, the position estimation result is shown in Figure 23. We can see that the estimated position converges to the real motor position quickly and closely follows it afterward.

![Figure 23 Simulation Result of ESO Based Position Estimation](image-url)
The velocity loop response is given in Figure 24 where the position estimation result is used as the position feedback and fed to the ESO. The TD based profile generator, as discussed in Chapter 3.5, is used to produce the reference signal. Two load disturbances are introduced at \( t=1 \) second and \( t=1.5 \) seconds to test the disturbance rejection performance. We can see that the estimated velocity closely follows the real motor velocity and both signals follow the reference signal.

![Figure 24 Simulation Result of ESO Based Position Estimation](image)

6.4 The Analysis of Sliding Mode Observer with Small Error

Sliding mode observer, with its performance and elegant mathematical deduction, is one of the most popular back-EMF based solutions nowadays in PMSM sensorless control. However, mathematical proof for the convergence of sliding mode observer is based on the bang-bang control mechanism, which often leads to chattering at the level not suitable for industrial applications. As a result, the switching function was replaced with a saturation function in some industrial solutions [20, 23]. In actual practice, the sliding mode observer works in the linear range most of the time, i.e., with a small
tracking error, which could be analyzed using the existing tools in control theory, as shown below.

The block diagram of the enhanced sliding mode observer (eSMO) is given in Figure 25 [20, 23].

![Block Diagram of the Enhanced Sliding Mode Observer](image)

**Figure 25** Block Diagram of the Enhanced Sliding Mode Observer

The dynamics of the linear range of the eSMO is given as

\[
\dot{X} = \begin{bmatrix} \frac{R_s + K}{L_s} & -\frac{1}{L_s} \\ \omega_s K & -\omega_s \end{bmatrix} X + \begin{bmatrix} \frac{1}{L_s} \\ 0 \end{bmatrix} v_s
\]

\[
\dot{i}_s = \begin{bmatrix} 1 & 0 \end{bmatrix} X
\]

\[
\dot{\omega}_s = \begin{bmatrix} 0 & 1 \end{bmatrix} X
\]

where \( K = \frac{k}{E_{\omega_0}} \) is the slope of the linear approximation of the switching mechanism and \( \omega_s \) is the bandwidth of the low pass filter. The eigenvalues of the \( A \) matrix of the eSMO system can be chosen as

\[
E = \begin{bmatrix} \frac{1}{2} \omega_s I_s + R_s + K - \sqrt{(\omega_s I_s - R_s - K)^2 - 4\omega_s I_s K} \\ \frac{L_s}{2} \\ \frac{1}{2} \omega_s I_s + R_s + K + \sqrt{(\omega_s I_s - R_s - K)^2 - 4\omega_s I_s K} \end{bmatrix}
\]
Since $K$ is usually much larger than $\omega_s$, $L_s$, $R_s$ are very small, a reasonable approximation of the eigenvalues could be given as

$$E \approx \begin{bmatrix} -\frac{1}{2} \frac{\omega_s L_s + R_s + K - (R_s + K - \omega_s L_s)}{L_s} \\ -\frac{1}{2} \frac{\omega_s L_s + R_s + K + (R_s + K - \omega_s L_s)}{L_s} \end{bmatrix} \approx \begin{bmatrix} -\omega_s \\ -\frac{K}{L_s} \end{bmatrix}$$ (221)

Since the two poles in the eSMO system are $\omega_s$ and $\frac{K}{L_s}$. With $\frac{K}{L_s} \gg \omega_s$, the effective bandwidth of the observer is dominated by $\omega_s$. This means that the eSMO has similar effective bandwidth with the conventional Luenberger observer of the similar bandwidth, and the much-advertised sliding mode observers, as they are implemented in the industry to avoid excessive chattering, is very similar to the conventional state observers. What makes the eSMO different is the extra feedback signal from the output of the low pass filter. Although this proves effective, the reason is not quite clear. The ESO based solution discussed above helps to explain the success of eSMO in that the back-EMF, $e_s$, is treated as an additional state to be estimated. The ESO formulation is more general and flexible. In particular, the bandwidth concept in the ESO design and tuning allow additional compensation to be easily designed to mitigate the unavoidable phase lag in the ESO, which is critically important in FOC based motor control.
Chapter VII

CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

In this dissertation, we discussed an information-driven design strategy based on the active disturbance rejection framework. The key problem in control system design is the selection, extraction and processing of online and offline information. Some ESO based information extraction methods were discussed including the ESO based parameter estimation and the estimation of motor position from voltage and current information. The two most common ways of using offline information in the industry, profile and feedforward, were then discussed. Some ADR based online information processing solutions were also proposed including the error-based ESO implementation; also discussed were the pros and cons of different methods of using offline system dynamics information in ADR based controller design. The ADRC deployment issues were also addressed, including the scaling of ESO for better data processing accuracy fixed-point implementation and the restructuration of error-based ESO.

A case study was carried out to address some core problems in the PMSM control. The proposed ESO design with offline information was discussed for the decoupling and control problem in the PMSM current loop. An active disturbance rejection based integrated current loop control solution was presented. Three designs were presented progressively, with the idea of minimizing phase lag. Then two ADR based velocity loop
control designs were proposed with only position information from either sensor feedback or position estimator. The velocity information was extracted with the proposed ESO, together with the total disturbance information, eliminating the need for a separate differentiator. Both current and voltage controller designs were put to the test in simulation and hardware experiment against the industrial leading PI plus decoupling design with the same amount of offline information. It is shown all controllers work great when the offline system parameter used in the controller design was accurate. Once the system parameter changes, the ADR based solutions shows its strong ability in dealing with uncertainties.

7.2 Future Work

Although many topics were discussed in this dissertation, there are still a lot of unsolved problems or other possible topics to work on. In Chapter 6.2, we discussed nonlinear ESO based position estimator, while focused on linear ESO based designs throughout the rest of the dissertation. Trying to introduce nonlinearity into the ESO designs in other topics can potentially achieve higher performance.

In the frequency characteristics, we focused on the limitation due to delay and phase lag. Another big part of the limitation of bandwidth in industrial applications was the noises in the feedback signal, including white noise, electromagnetic interference and quantization noise. The relationship of those noises and the system bandwidth can be explored in the future.

Although we discussed some key problems in the PMSM control, there are still some very important topics that can be put together in the information-based design framework. For example, vector-based control for the d-q voltage can be done for better performance on the field weakening and the maximum torque per ampere (MTPA) control.
BIBLIOGRAPHY


International Middle East Power Systems Conference (MEPCON), 2016, pp. 468-473.


APPENDICES

A. Design and Build of Hardware Test Bed

To verify the proposed results in this dissertation, a hardware testbed is designed and built. The hardware system contains power entry and protection, power inverter, motor drive and a PLC-based protection system. Figure 26 shows a photo of the motor drive rack and Figure 27 shows a picture of the embedded servo drive tray.
Figure 26  Picture of Motor Drive Rack
Figure 27  Picture of the Power Inverter and Controller Tray
A.1. Power Subsystem

A.1.1. Power Entry

A three-phase 208 volts input line with 50 amps capacity was fed into the system with a main switch on the front door. A three-phase AC line filter was used to reduce the switching noise injected into the power grid. A switch-mode power supply behind an AC line filter was used to generate the 24 volts control power rail in the rack.

Multiple levels of protection were implemented to ensure the safety of the operator and the device. An isolation transformer, with delta-to-delta connection and 3kVA phase capacity to make the high voltage circuit in the system, floats with respect to earth ground and ensure the safety of the operators even when accidentally touched one point in the system. A three-phase ground fault circuit interrupter (GFCI) is also used for another layer of protection. Then an over power and phase loss interrupter was added onto the main circuit breaker to protect the drive and motor. Also, three current sensors were used to measure the currents on motor stator coils and interrupt the system on over current situations.

A block diagram of this module is given in Figure 28, and a picture of the hardware panel is shown in Figure 29.
Figure 28 Block Diagram of Power Supply and Protection Module

Figure 29 Photo of Power Supply and Protection Panel
A.1.2. Power Inverter

The three-phase power from the power supply module was sent through a three-phase rectifier to generate the 300 volts DC bus to be fed to the power inverter module. A capacitor array with an equipment capacitance of 1500 microfarads was then used to reduce the ripple on the DC bus. Two power resistors were used in parallel as shunt resistor to accelerate the drain of bus capacitors after turning off the main power.

A voltage divider was used as the voltage sensor for bus voltage followed by a voltage isolator chip AMC1200 with built-in eight-time gain. The differential analog output of AMC1200 was then fed into the differential ADC on the MCU as bus voltage feedforward in the PWM signal generation process. A schematic of the voltage sensing system is shown in Figure 30.

![Figure 30 Schematic of the Voltage-Sensing System](image)

A.1.3. Power Switch and Driver

In the first version of the design, a SLLIMM Intelligent Power Module (IPM) STGIPS20C60 from STMicroelectronics was selected and evaluated with evaluation board STEVAL-IHM028V2. This solution was then abandoned for its lack of isolation and poor over current and short circuit protection capability.

An IGBT Intelligent Power Module PM50RL1A060 from Mitsubishi Electric was then selected as power switching device. The unit packed six gates for a full three-phase H-bridge and an extra gate for brake application. It has an isolated heat spreading base and is designed for power switching applications operating at frequencies up to 20 kHz. The built-in control circuit includes gate drive and protections, include short circuit
protection, over temperature protection, and under voltage protection. The target application of this unit were power inverters, ups power supply (UPS), motion and servo drives and power supplies. A schematic of the PM50RL1A060 module is shown in Figure 31.

Figure 31 Schematic of PM50RL1A060 IGBT IPM

An interface circuit including optically coupled isolation for control signals and isolated power supplies for the IPM’s built-in gate drive and protection circuits is constructed as in Figure 32. This circuit provides three phase isolated interface circuit with brake control and fault feedback, features 2500 V RMS isolation for control power and signals and can operate from a single 24 V DC supply.
Figure 32  Schematic of IPM Interface Circuit
A.2. Control Subsystem

A.2.1. The MCU Controller

A Texas Instruments (TI) TMS320F28377D MCU is picked as the main controller in the system. This MCU has two 200 MHz CPU cores, each with IEEE 754 single-precision Floating-Point Unit (FPU) and Trigonometric Math Unit (TMU), and two programmable Control Law Accelerators (CLAs), also running at 200 MHz, with IEEE 754 Single-precision floating point instructions and executes code independently of the main CPU cores.

On the peripheral side, it contains four 16-bit differential ADC channels, eight PWM outputs with built-in synchronizing and dead-zone implementation, three quadrature encoder pulse (QEP) module, SPI and SCI module.

A.2.2. Current Feedback

In a three-phase AC drive system, at least two phase currents are needed to implement field oriented control. The quality of current feedback, such as accuracy, noise level and bandwidth, directly affects the quality of current control.

In this system, a LEM closed-loop current sensor LAH 25-NP with 1000:1 current reduction ratio was selected for its good accuracy and linearity, low temperature drift, fast response time and wide frequency bandwidth. This sensor could be used in one, two and three turn mode for the different current ranges. For our application, we are running the sensor in one turn mode to allow the maximum measuring range. A schematic of the current sensor is shown in Figure 33.
Figure 33 Schematic of the Current Sensor

The output was then converted to a voltage signal with a precision shunt resistor of 249 Ohms. Then the signal was conditioned with a fully differential operational amplifier THS4521 to be read by the on-chip ADC of the MCU. A schematic of the current signal conditioning system is shown in Figure 34.

Figure 34 Schematic of the current signal conditioning system

A.2.3. Position Feedback

Absolute shaft position information is critical for accurate position, speed and torque control of PMSM. The motor under test consists a HIPERFACE sin/cos absolute encoder providing super high-resolution position feedback. Since both analog and digital information are embedded within the differential sin/cos signal, the circuit in Figure 35
was designed to split them and feed them separately into the ADC and quadrature encoder peripherals of the MCU.

A common mode choke inductor was used to filter out the common mode noise coming from the differential signal. A termination resistor of 121 ohms is used for impedance matching. An operational amplifier without feedback was used as a comparator to extract the digital portion of the encoder signal. The fully differential operational amplifier THS4521 was used to condition the 1 V pick-to-pick signal for ADC module of the MCU.

Figure 35 Schematic of the Sin/Cos Encoder Feedback

There is a separate RS-485 channel in the HIPERFACE encoder feedback for acquiring the absolute shaft position and other motor status information such as temperature. An RS-485 transceiver MAX 3485 is used. A schematic of the RS-485 feedback is shown in Figure 36.
A.2.4. Command and Data Acquisition System

An RS-232 serial channel is used as a command receiver. A 5-byte binary command is used, where the first byte contains the command code and the remaining 4 bytes were used to transmit a 32-bit integer or floating point number depending on the command type.

An SPI interface is used for data acquisition running at 25 MHz clock frequency. Sixteen 32-bit floating-point data could be sent out at 20k samples per second, which is the same as the sampling rate of the current loop controller. One of which is used as a sequence number and another is used as the checksum. Therefore, 14 numbers could be collected in each cycle. This way the data is flowing at an average of 10 Mbps, which is the maximum data rate of the DLN-4S USB-SPI interface used on PC side to collect the data.

On the PC side, a command sending program and a data collection program was separately implemented and an automated testing system was constructed with them using a script file. Then the binary data was decoded and imported into MATLAB for plotting and analysis using a MATLAB m-file script.
A.2.5. Power Supply

In the drive system, a main +5 V power rail for generating the 3.3V digital and analog voltage rails and 1.8 V core voltage for MCU, a ±15 V power rail for current sensor operation and an isolated +5V power rail is needed for the DC bus voltage sensing.

In the beginning, a solution using DC-DC converter module followed by high power supply rejection ratio (PSRR) linear regulator was implemented as shown in Figure 37. This implementation leaves a voltage ripple higher than 0.3 V peak-to-peak in the ground plane, which yields to noise in the ADC result of more than 10 % total dynamic range.

![Figure 37 Schematic of ± 15 V in Power Supply Version 1](image)

To reduce the ripple in the power rails, the power supply was redesigned with power transformers and linear power regulator. The noise level was then reduced to less than 0.01 volts peak-to-peak. The schematic of the power entry for power supply is
shown in Figure 38 and the schematics of the 5 V and ±15 V power regulator designs are shown in Figure 39 and Figure 40 respectively.

**Figure 38  Schematic of Power Supply Power Entry**

![Schematic of Power Supply Power Entry](image)

**Figure 39  Schematic of 5 V Power in Supply Version 2**

![Schematic of 5 V Power in Supply Version 2](image)
A.2.6. Motor and Load

The motor used in this study is a Rockwell Automation MPL-A330P-M motor with a rated torque of 4.18 Nm and a rated output power of 1.8 kW. This motor comes with a multi-turn, 1024 sin/cos absolute encoder with HIPERFACE protocol interface.
A MAGTOR dynamometer system HD-715-6N-0100 with controller DSP7001 is used as the test load. It could run at both constant speed and constant torque mode. A LabVIEW-based software is provided by the manufacturer to tune the speed and torque controller, operate and monitor the system and collect its running data via the GPIB port on the dynamometer controller. A picture of the testbed is shown in Figure 41.

![Figure 41 Picture of Hardware Test Bed](image)

### A.3. MCU Software Architecture

The CPU 1 of MCU is running TI-RTOS and boot from FLASH memory. The flow of the program is shown in the following chapter. The entire firmware code is documented with Doxygen.

In FOC control scheme, two phase currents are measured with sensors and sent through the Clark and Park transformations to get the $d$ and $q$ axis currents. Two current regulators are used to drive the two currents to their desired values by manipulating the $d$ and $q$ axis voltages. The two voltages then go through the inverse version of Park and
Clark transformations and the three-phase voltage commands are generated. Then space vector PWM signals are generated accordingly.

**A.3.1. CPU 1 Boot Sequence**

The boot sequence is shown in Figure 42. The CPU 1 runs from start point “codestart”, then a custom function is called to set clock source to the external crystal. Then boot module is called to initialize PLL and FLASH memory. After that, the C and C++ runtime is initialized. The main function is then called after initialization. At the end of the main function, BIOS_Start function is called and BIOS scheduler is started.
A.3.2. Software Flow of CPU 1 Main Function

The software flow of CPU 1 main function is shown in Figure 43.
Figure 43  Software Flow of CPU 1 Main Function
A.3.3. Software Flow of Main Hwi

The software flow of main Hwi is shown in Figure 44.

![Software Flow of Main Hwi](image)

Figure 44  Software Flow of Main Hwi

A.3.4. Software Flow of Torque Swi

The software flow of torque Swi is shown in Figure 45.
Figure 45  Software Flow of Torque Swi
B. Schematics of Motor Control Test Bed

B.1. MCU and Sensor Board Schematics
Use 0.1% resistors or better.
Use 0.1% resistors or better.

Put near Op Amp Pin 3.

Put near Op-Amp Pin 8.
Use 0.1% resistors or better.
Use 0.1% resistors or better.

Put near Op-Amp Pin 8.
DE and nRE should be tied together to a GPIO pin.

Put near Pin 8.

Put near Op-Amp Pin 8.
B.2. High Voltage Board Schematics
B.3. Power Supply Board Schematics