UTS UNIVERSITY OF TECHNOLOGY SYDNEY

ROBUST PREDICTIVE CONTROL OF PERMANENT MAGNET SYNCHRONOUS MACHINE DRIVES

By NABIL SALEM YAHYA FARAH

Thesis submitted in fulfilment of the requirements for the degree of

Doctor of Philosophy

Under the supervision of

Dr. Gang Lei
Prof. Jianguo Zhu
Prof. Youguang Guo

University of Technology Sydney

Faculty of Engineering and Information Technology

June 2023

CERTIFICATE OF ORIGINAL AUTHORSHIP

I, Nabil Farah, declare that this thesis is submitted in fulfillment of the requirements for the award of Doctor of Philosophy in the Faculty of Engineering and Information Technology at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis. I also certify that the work in this thesis has not previously been submitted for a degree, nor has it been submitted as part of the requirements for a degree except as fully acknowledged within the text.

This research is supported by the Australian Government Research Training Program.

Signature: Production Note: Signature removed prior to publication.

Date : 27/10/2023

ACKNOWLEDGEMENTS

I would like to express my heartfelt gratitude and appreciation to the following individuals who have played a significant role in the completion of my Ph.D. thesis:

First and foremost, I extend my deepest gratitude to my supervisor, Dr. Gang Lei. Your guidance, expertise, and unwavering support throughout this journey have been invaluable. Your insightful feedback and constructive criticism have constantly pushed me to excel and strive for excellence.

I am also indebted to my co-supervisors, Professor Jianguo Zhu and Professor Youguang Guo. Your extensive knowledge and invaluable contributions have enriched my research and broadened my understanding of the subject matter. I am grateful for the time you have dedicated to mentoring and advising me, which has profoundly impacted the quality of my work.

I would like to extend a special thanks to my friend Abdulrahman for his exceptional support and assistance throughout my studies. Your willingness to lend a helping hand and your invaluable insights have been instrumental in shaping the direction of my research. Your presence and encouragement during this journey's ups and downs have been invaluable.

I am deeply grateful to my family for their unwavering support and understanding. To my father, Salem Farah, and my mother, Harsah Yahya, thank you for instilling in me a love for knowledge and always encouraging me to pursue my dreams. To my loving wife, Lina Mohammed, and my wonderful children, Maria, and Ryan, thank you for your patience, understanding, and sacrifice during this demanding period. Your unwavering belief in me has been my constant source of motivation.

To all those mentioned above and those who may not be explicitly named, please accept my heartfelt gratitude. Without your support, this accomplishment would not have been possible. Thank you for being an integral part of my journey and for your unwavering belief in my abilities.

ABSTRACT

Permanent Magnet Synchronous Machines (PMSMs) are widely used in industry due to their high power density, high torque/current ratio, low power losses, and high efficiency. Model predictive control (MPC) is a popular control method for PMSMs, but conventional MPC methods have limitations in terms of unsatisfactory steady-state performance, variable switching frequency, and reliance on weighting factors. To overcome these drawbacks, two enhanced MPC methods based on current and torque control have been proposed. These approaches can eliminate weighting factors, generate two switching vectors per control cycle, and exhibit superior performance compared to the conventional MPC.

However, model uncertainties and parameter mismatching are unavoidable in PMSM drives, significantly affecting the control performance. To evaluate the robustness of a control system and determine the robustness level, a novel and systemic robustness evaluation method based on the concept of Six-Sigma methodology has been proposed. This method is validated based on the conventional MPC and five other robust predictive control methods.

Data-driven controls have emerged as a promising alternative to robust MPC, such as model-free predictive current control (MFPCC) for PMSM drives. However, inaccuracies in prediction and performance degradation can occur when the switching vectors remain unchanged for consecutive control cycles, causing unapplied switching to stagnate. To overcome this limitation, an adaptive MFPCC (A-MFPCC) has been proposed, which incorporates a modified current difference updating mechanism. By generating a reference vector based on current tracking error, the A-MFPCC method enforces the update of current differences, preventing stagnation and optimizing the current tracking performance.

Reinforcement learning (RL) based control is another data-driven method, but standard RL-based control usually is trained over a single training task with specific operating conditions and a fixed parameter set. To address this challenge, multi-set robust RL (MSR-RL) based current control of PMSM drives has been proposed. MSR-RL aims to learn a single optimal policy that remains robust across multiple parameter sets or contexts. The proposed A-MFPCC and MSR-RL methods have been validated through numerical simulations, experimental tests, and robustness evaluations, demonstrating superior performance across various operating conditions compared to their conventional counterparts.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

AC	-	Alternating Current
A-MFPCC	-	Adaptive Model Free Current Control
CCS-MPC	-	Continuous Control Set Model Predictive Control
CMDP	-	Context Markov Decision Process
DC	-	Direct Current
DDPG	-	Deep Deterministic Policy Gradient
DNN	-	Deep Neural Network
DQN	-	Deep Q-learning Network
DSP	-	Digital Signal Processor
DTC	-	Direct Torque Control
DSPACE	-	Digital Signal Processing And Control Engineering
FCS-MPC	-	Finite Control Set Model Predictive Control
FDM	-	Fuzzy Decision Making
FOC	-	Field-Oriented Control
IGBT	-	Insulated Gate Bipolar Transistor
MDP	-	Markov Decision Process
MF	-	Membership Function
MPC	-	Model Predictive Control
MPTC	-	Model Predictive Torque Control
MPC	-	Model Predictive Current Control
MFPC	-	Model Free Predictive Control

MSR-RL	-	Multi-Set Robust Reinforcement Learning
NN	-	Neural Network
PI	-	Proportional Integral
POF	-	Probability of Failure
POMDP	-	Partially Observable Markov Decision Process
PMSM	-	Permanent Magnet Synchronous Motor
PWM	-	Pulse Width Modulation
RL	-	Reinforcement Learning
RMSE	-	Root Mean Square Error
RPC	-	Robust Predictive Control
RTI	-	Real-Time Interface
SPWM	-	Sinusoidal Pulse Width Modulation
SVPWM	-	Space Vector Pulse Width Modulation
THD	-	Total Harmonics Distortion
VSI	-	Voltage Source Inverter
USL	-	Upper Specification Limit

LIST OF SYMBOLS

$\boldsymbol{B}$	-	Friction
$\boldsymbol{\alpha\beta}$-frame	-	Stator reference frame
$\boldsymbol{dq}$-frame	-	Rotor reference frame
$\boldsymbol{d}$-axis	-	Direct axis
$\boldsymbol{q}$-axis	-	Quadrature axis
$\boldsymbol{i_a, i_b, i_c}$	-	Stator phase (a, b, and c) current
$\boldsymbol{i_d}$	-	Direct axis stator current
$\boldsymbol{i_q}$	-	Quadrature axis stator current
$\boldsymbol{K_p}$	-	Proportional gain for PI
$\boldsymbol{K_i}$	-	Integral gain for PI
$\boldsymbol{L_d}$	-	Direct axis inductance
$\boldsymbol{L_q}$	-	Quadrature axis inductance
$\boldsymbol{P}$	-	Number of poles pairs
$\boldsymbol{J}$	-	Motor inertia
OS	-	Overshoot
$\boldsymbol{R_s}$	-	Stator Resistance
$\boldsymbol{T_e}$	-	Electromagnetic torque
$\boldsymbol{T_m}$	-	Mechanical torque
$\boldsymbol{T_L}$	-	External load
$\boldsymbol{T_s}$	-	Settling time
$\boldsymbol{t_s}$	-	Sampling time
$\boldsymbol{v_a, v_b, v_c}$	-	Stator phase (a, b, and c) voltage
$\boldsymbol{v_s}$	-	voltage vector
$\boldsymbol{v_d}$	-	Direct axis voltage
$\boldsymbol{v_q}$	-	Quadrature axis voltage
Ψ_{PM}	-	Permanent magnet flux linkage
Ψ_d	-	Direct axis flux
Ψ_q	-	Quadrature axis flux

$\boldsymbol{\sigma}$	-	Standard deviation
$\boldsymbol{\mu}$	-	Mean
$\boldsymbol{\varepsilon}$	-	Noise
$\boldsymbol{\theta_r}$	-	Rotor position angle
$\boldsymbol{\omega_r}$	-	Rotor speed or actual speed
$\boldsymbol{\omega_r}$ *	-	Reference speed
$\boldsymbol{S}$	-	RL state
$\boldsymbol{a}$	-	RL action
$\boldsymbol{P(s,a)}$	-	RL transition probability
$\boldsymbol{Q}$	-	Q-learning value function
$\boldsymbol{o_k}$	-	RL observation
$\boldsymbol{r_k}$	-	RL reward
$\boldsymbol{R_h}$	-	RL regret
$\Delta_{\boldsymbol{O}}$	-	Operational uncertainty
$\Delta_{\boldsymbol{M}}$	-	Manufacturing uncertainty

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Background and Significance

Electrical machines are increasingly used in various industrial, domestic, and transportation applications. A tremendous number of electrical machines are being used worldwide, utilizing about 46.2% of the total global electricity consumption [1.1]. With the global concerns of energy conservation, environmental preservation, and sustainability development, high-efficiency electrical machines have attracted much attention in the academic and industrial fields [1.2-1.3]. Permanent magnet synchronous machines (PMSMs) with features of high-power density, high efficiency, higher torque, and less electrical losses became the ideal machines for several applications, including traction systems (e.g., electric vehicle (EV)) [1.4], robotics [1.5-1.6], machine tools [1.7], actuators [1.8], servo drives [1.9], air conditioning [1.10-1.11], washing machine [1.12-1.13], and vacuum cleaner [1.14]. To optimize the performance of PMSMs in these applications, machine controls are introduced [1.15].

For decades, conventional vector control methods have been implemented for PMSM drives. Vector control offers superior performance and overcomes all the issues of scalar control [1.16]. Two vector control methods are commonly used in PMSM drives, field-oriented control (FOC) and direct torque control (DTC) [1.17]. FOC has the merits of fast dynamic response and good state-steady performance, but it incorporates a complex structure comprising two current controllers and a PWM modulator [1.18]. On the other hand, DTC has a simple structure and a rapid dynamic response. However, the

conventional DTC produces high torque ripples, current harmonics, variable switching frequency, and degraded performance at lower speed operations [1.19-1.20].

In the last decade, MPC methods for PMSM drives have garnered industrially and academically growing attention. This is due to the merits of a basic concept, fast dynamic response, multi-variable control, nonlinearity control, and constraint inclusion [1.21-1.26]. MPC reduces the drive system complexity by eliminating the modulation scheme and/or current controllers used in FOC [1.27]. Also, it selects a switching vector based on minimizing a cost function instead of a heuristic switching table used in DTC. Thus, a more accurate and optimum switching vector is obtained [1.28]. However, the conventional MPC still faces some challenges, such as high torque and current ripples, variable switching frequency, proper selection of weighting factor, and high computational requirements. Therefore, numerous studies have focused on improving the conventional MPC to eliminate these challenges. For instance, additional vectors are applied along with the optimum vector during a control cycle [1.29-1.31]. In addition, weighting factor elimination and switching frequency regulation are implemented to improve the conventional MPC [1.32-1.33].

Despite these improvements, a key disadvantage of MPC is its dependency on uncertain machine models. Various manufacturing and operational uncertainties exist in the practical operation of PMSMs [1.34], which may not all have been captured in the machine model used for prediction. Thus, prediction accuracy and control performance are significantly affected when uncertainties occur. To reduce the effect of uncertainties on the PMSM drive performance, RPC methods are introduced to enhance the system's robustness and compensate for the impact caused by these uncertainties. Various RPC methods have been introduced by implementing serval techniques and mechanisms, such

as using observers [1.35], optimizing cost function [1.36], combining control techniques [1.37], using prediction error correction [1.38], and using the model-free method [1.39].

Model-free predictive control (MFPC) has recently emerged as a promising alternative to robust MPC methods. MFPC can eliminate the prediction dependency on a simplified parametric machine model by developing a prediction model independent of the machine model and parameters [1.40]. MFPC can be achieved using an ultra-local model [1.41], where an ultra-local model replaces a complex system model with one or two unknowns that can be estimated based on measured input and output data [1.42-1.43]. On the other hand, MFPCC can be achieved by solely using the system's measured input and/or output data and their variations [1.44-1.46]. However, inaccurate estimation and stagnation can occur depending on the technique used to achieve MFPC (e.g., current difference, ultra-local model), and higher computational effort may be required [1.39, 1.44].

Recently, reinforcement learning (RL) has emerged as a promising approach for achieving data-driven control in PMSM drives. A computationally efficient controller optimized offline is obtained by training an RL agent with appropriate rewards based on PMSM data [1.47-1.48], unlike MFPC, which requires continuous optimization during online control and can be computationally intensive. The effectiveness of RL-based controllers depends on the amount and quality of the data used for training. In the standard RL-based controller, an agent learns an optimal policy that maximizes its expected cumulative reward over a single training task with specific operating conditions and a single parameter set. Thus, new operating conditions and different parameter sets (due to parameter mismatching) can lead to poor performance, robustness, or instability in the controlled system [1.49].

Although there are various robust control methods, there is a lack of knowledge on the definition and criteria of robustness. This raises questions about the basis for determining whether a control system is robust or non-robust and how to quantify the level of robustness of a control system. In other words, it is unclear what criteria should be used to define a robust control system and how to compare the robustness of different control systems. This highlights the need for further research to understand better the definition and criteria of robustness in control systems and to develop methods for measuring the robustness index of a control system.

This research proposes a clear and adequate robustness definition, demonstrated based on a simple second-order system and DC motor drives. A control system's robustness criteria and boundaries are established by considering different performance indicators. In addition, practical parameter variation (mismatching) ranges are generated considering operating conditions and operational factors such as temperature changes and manufacturing tolerances. Then, the robustness indices are computed using the Six-Sigma concept by considering various performance indicators and setting their boundaries (acceptance level for different application requirements). This method is validated based on six different predictive control methods of PMSM drives.

To improve the performance of data-driven control and achieve higher sigma levels, this research proposes two novel data-driven control methods for PMSM drives. The first method is an adaptive model-free predictive current control (A-MFPCC) with a modified current difference updating technique. A reference vector is generated considering the tracking error and the position of the reference current vector, ensuring regular updates of the current difference and preventing stagnation. The second method is a multi-set robust reinforcement learning (MSR-RL) based current control of PMSM drives. The MSR-RL aims to learn a single optimal policy robust to several different

parameter sets. Instead of learning a policy over a single training task with a single parameter set, the proposed MSR-RL learns a single policy over multiple training tasks with various parameter sets. The resultant policy can be robust to all these parameter sets and generalized to the new ones. The effectiveness of the proposed A-MFPCC and MSR-RL is validated through comparison with conventional methods based on numerical simulation, experimental tests, and robustness evaluation.

1.2 Research Contributions

The main contributions of this research are summarized as follows:

- Two improved MPC methods are proposed to eliminate the issue of conventional MPCs. These methods apply two vectors in each control cycle to improve steady-state performance and regulate the switching frequency and utilize the fuzzy decision-making criteria to eliminate the use of weighting factors.
- A novel robustness evaluation method based on the Six-Sigma concept is proposed. A control system's robustness index or sigma level can be determined by defining specific indicators and evaluating the control system against them for a number of samples within bounded uncertainty ranges. The proposed method is applied to MPC and some existing RPC methods for PMSMs. Their robustness indexes (sigma levels) are obtained by evaluating them with different bounded uncertainty ranges and based on three different application requirements.
- An adaptive MFPCC (A-MFPCC) with a modified current difference updating technique is introduced. A reference vector is generated considering the tracking error and the position of the reference current vector, ensuring regular updates of the current difference and preventing stagnation.

- A novel MSR-RL-based current control of PMSM drives is introduced. MSR-RL learns a single policy over multiple training tasks with various parameter sets, resulting in a robust policy that can be generalized to the new ones.

1.3 Thesis Outline

This thesis consists of seven chapters, including this chapter that provides an overview of the research background, significance, contributions, and the overall structure of the thesis. The subsequent chapters are outlined as follows:

Chapter 2 offers a comprehensive literature survey on PMSM drives and commonly employed control methods. Additionally, this chapter critically reviews various RPC methods, including model-free control and data-driven RL-based control.

Chapter 3 delves into the proposed two-vector dimensionless MPCs based on fuzzy decision-making criteria. This chapter presents mathematical modelling, simulation, and experimental tests of these methods applied to PMSM drives.

Chapter 4 discusses the novel robustness evaluation method based on Six Sigma methodology. The definition and criteria of system robustness are presented and illustrated through a simple second-order system. Furthermore, the validation of the proposed method is showcased in this chapter based on six predictive control methods for PMSM drives.

Chapter 5 focuses on the proposed A-MFPCC scheme for PMSM drives. It encompasses mathematical modelling and comparisons with two other MFPCC schemes through simulation, experiments, and robustness evaluation.

Chapter 6 introduces the proposed MSR-RL-based current control for PMSM drives. The concept of RL-based control, MSR-RL modelling, and multi-task training are

discussed extensively. Additionally, this chapter presents a comparison between MSR-RL and standard RL through numerical simulation, experimental tests, and robustness evaluation.

Chapter 7 summarizes the main findings and contributions achieved in this thesis. Furthermore, this chapter outlines the future research perspective for this field.

REFERENCES

[1.1] B. C. Waide P, "Energy-efficiency policy opportunities for electric motor-driven systems," in "Energy Efficiency Series," OECD iLibrary, International Energy Agency: Paris, France, 2011.

[1.2] D. G. Dorrell, "A Review of the Methods for Improving the Efficiency of Drive Motors to Meet IE4 Efficiency Standards," *Journal of Power Electronics,* vol. 14, no. 5, pp. 842-851, 2014, doi: 10.6113/jpe.2014.14.5.842.

[1.3] G. Lei, J. Zhu, and Y. Guo, *Multidisciplinary design optimization methods for electrical machines and drive systems* (Power Systems). Berlin Heidelberg: Springer, 2016.

[1.4] M. Zeraoulia, M. E. H. Benbouzid, and D. Diallo, "Electric Motor Drive Selection Issues for HEV Propulsion Systems: A Comparative Study," *IEEE Transactions on Vehicular Technology,* vol. 55, no. 6, pp. 1756-1764, 2006, doi: 10.1109/TVT.2006.878719.

[1.5] T. M. R. Antony, J. Jose, S. Sakthivel, D. Jagadishan, S. J. Winston, and S. Venugopal, "Design and Development of Two Axis Control and Drive for PMSM Motor of In-Service Inspection Module for PFBR Steam Generator," *Procedia Engineering,* vol. 86, pp. 520-528, 2014, doi: 10.1016/j.proeng.2014.11.076.

[1.6] M. Sadeghijaleh, "Adaptive Voltage-based Control of Direct-drive Robots Driven by Permanent Magnet Synchronous Motors," *International Journal of Engineering,* vol. 30, no. 4, 2017, doi: 10.5829/idosi.ije.2017.30.04a.08.

[1.7] F. Mohd Zaihidee, S. Mekhilef, and M. Mubin, "Robust Speed Control of PMSM Using Sliding Mode Control (SMC)—A Review," *Energies,* vol. 12, no. 9, 2019, doi: 10.3390/en12091669.

[1.8] Y. A. R. I. Mohamed, "A Newly Designed Instantaneous-Torque Control of Direct-Drive PMSM Servo Actuator With Improved Torque Estimation and Control Characteristics," *IEEE Transactions on Industrial Electronics,* vol. 54, no. 5, pp. 2864-2873, 2007, doi: 10.1109/tie.2007.901356.

[1.9] Y. Meng and X. Meng, "Design and implementation of a PMSM servo drive system applied to intelligent patrol robots," in *IOP Conference Series: Materials Science and Engineering*, vol. 397, no. 1: IOP Publishing, p. 012064, 2018.

[1.10] S. I. Kim, G. H. Lee, J. J. Lee, and J. P. Hong, "Simple design approach for improving characteristics of interior permanent magnet synchronous motors for electric air-conditioner systems in HEV," *International Journal of Automotive Technology,* vol. 11, no. 2, pp. 277-282, 2010, doi: 10.1007/s12239-010-0035-z.

[1.11] K. Sung-Il, L. Geun-Ho, H. Jung-Pyo, and J. Tae-Uk, "Design Process of Interior PM Synchronous Motor for 42-V Electric Air-Conditioner System in Hybrid Electric Vehicle," *IEEE Transactions on Magnetics,* vol. 44, no. 6, pp. 1590-1593, 2008, doi: 10.1109/tmag.2007.916136.

[1.12] S. Chi, Z. Zhang, and L. Xu, "Sliding-Mode Sensorless Control of Direct-Drive PM Synchronous Motors for Washing Machine Applications," *IEEE Transactions on Industry Applications,* vol. 45, no. 2, pp. 582-590, 2009, doi: 10.1109/tia.2009.2013545.

[1.13] O. Payza, Y. Demir, and M. Aydin, "Investigation of Losses for a Concentrated Winding High-Speed Permanent Magnet-Assisted Synchronous Reluctance Motor for Washing Machine Application," *IEEE Transactions on Magnetics,* vol. 54, no. 11, pp. 1-5, 2018, doi: 10.1109/tmag.2018.2848881.

[1.14] V. Dmitrievskii, V. Prakht, V. Kazakbaev, and S. Sarapulov, "Optimal Design of a High-Speed Single-Phase Flux Reversal Motor for Vacuum Cleaners," *Energies,* vol. 11, no. 12, 2018, doi: 10.3390/en11123334.

[1.15] B. K. Bose, "Adjustable speed AC drives—A technology status review," *Proceedings of the IEEE,* vol. 70, no. 2, pp. 116-135, 1982, doi: 10.1109/PROC.1982.12259.

[1.16] M. Štulrajter, V. Hrabovcova, and M. Franko, "Permanent magnets synchronous motor control theory," *Journal of electrical engineering,* vol. 58, no. 2, pp. 79-84, 2007.

[1.17] M. Merzoug and F. Naceri, "Comparison of field-oriented control and direct torque control for permanent magnet synchronous motor (PMSM)," *World Academy of Science, Engineering, and Technology,* vol. 45, pp. 299-304, 2008.

[1.18] W. Kim, C. Yang, and C. C. Chung, "Design and Implementation of Simple Field-Oriented Control for Permanent Magnet Stepper Motors Without DQ Transformation," *IEEE Transactions on Magnetics,* vol. 47, no. 10, pp. 4231-4234, 2011, doi: 10.1109/tmag.2011.2157956.

[1.19] Z. Wang, J. Chen, M. Cheng, and K. T. Chau, "Field-Oriented Control and Direct Torque Control for Paralleled VSIs Fed PMSM Drives With Variable Switching Frequencies," *IEEE Transactions on Power Electronics,* vol. 31, no. 3, pp. 2417-2428, 2016, doi: 10.1109/tpel.2015.2437893.

[1.20] G. Abad, M. A. Rodriguez, and J. Poza, "Two-Level VSC Based Predictive Direct Torque Control of the Doubly Fed Induction Machine With Reduced Torque and Flux Ripples at Low Constant Switching Frequency," *IEEE Transactions on Power Electronics,* vol. 23, no. 3, pp. 1050-1061, 2008, doi: 10.1109/tpel.2008.921160.

[1.21] M. Preindl and E. Schaltz, "Sensorless Model Predictive Direct Current Control Using Novel Second-Order PLL Observer for PMSM Drive Systems,"

IEEE Transactions on Industrial Electronics, vol. 58, no. 9, pp. 4087-4095, 2011, doi: 10.1109/TIE.2010.2100331.

[1.22] M. Preindl and S. Bolognani, "Model Predictive Direct Speed Control with Finite Control Set of PMSM Drive Systems," *IEEE Transactions on Power Electronics,* vol. 28, no. 2, pp. 1007-1015, 2013, doi: 10.1109/tpel.2012.2204277.

[1.23] W. Xie *et al.*, "Finite-Control-Set Model Predictive Torque Control With a Deadbeat Solution for PMSM Drives," *IEEE Transactions on Industrial Electronics,* vol. 62, no. 9, pp. 5402-5410, 2015, doi: 10.1109/TIE.2015.2410767.

[1.24] X. Zhang, B. Hou, and Y. Mei, "Deadbeat Predictive Current Control of Permanent-Magnet Synchronous Motors with Stator Current and Disturbance Observer," *IEEE Transactions on Power Electronics,* vol. 32, no. 5, pp. 3818-3834, 2017, doi: 10.1109/tpel.2016.2592534.

[1.25] X. Zhang and Y. He, "Direct Voltage-Selection Based Model Predictive Direct Speed Control for PMSM Drives Without Weighting Factor," *IEEE Transactions on Power Electronics,* vol. 34, no. 8, pp. 7838-7851, 2019, doi: 10.1109/tpel.2018.2880906.

[1.26] S. G. Petkar, K. Eshwar, and V. K. Thippiripati, "A Modified Model Predictive Current Control of Permanent Magnet Synchronous Motor Drive," *IEEE Transactions on Industrial Electronics,* pp. 1-1, 2020, doi: 10.1109/tie.2020.2970671.

[1.27] A. D. Alexandrou, N. Adamopoulos, and A. Kladas, "Development of a Constant Switching Frequency Deadbeat Predictive Control Technique for Field-Oriented Synchronous Permanent-Magnet Motor Drive," *IEEE Transactions on Industrial Electronics,* vol. 63, no. 8, pp. 5167-5175, 2016, doi: 10.1109/tie.2016.2559419.

[1.28] M. J. Navardi, J. Milimonfared, and H. A. Talebi, "Torque and Flux Ripples Minimization of Permanent Magnet Synchronous Motor by a Predictive-Based Hybrid Direct Torque Control," *IEEE Journal of Emerging and Selected Topics in Power Electronics,* vol. 6, no. 4, pp. 1662-1670, 2018, doi: 10.1109/jestpe.2018.2834559.

[1.29] Y. Liu, S. Cheng, Y. Zhao, J. Liu, and Y. Li, "Optimal two-vector combination-based model predictive current control with compensation for PMSM drives," *International Journal of Electronics,* vol. 106, no. 6, pp. 880-894, 2019/06/03 2019, doi: 10.1080/00207217.2019.1570565.

[1.30] H. Lin and W. Song, "Three-Vector Model Predictive Current Control of Permanent Magnet Synchronous Motor Based on SVM," in *IEEE International Symposium on Predictive Control of Electrical Drives and Power Electronics (PRECEDE)*, vol. 1, 2019.

[1.31] Y. Zhang, L. Huang, D. Xu, J. Liu, and J. Jin, "Performance evaluation of two-vector-based model predictive current control of PMSM drives," *Chinese Journal of Electrical Engineering,* vol. 4, no. 2, pp. 65-81, 2018.

[1.32] X. Zhang and B. Hou, "Double Vectors Model Predictive Torque Control Without Weighting Factor Based on Voltage Tracking Error," *IEEE Transactions on Power Electronics,* vol. 33, no. 3, pp. 2368-2380, 2018.

[1.33] X. Zhang, B. Hou, Y. He, and D. Gao, "Model predictive torque control of surface mounted permanent magnet synchronous motor drives with voltage cost functions," *Journal of Power Electronics,* vol. 18, no. 5, pp. 1369-1379, 2018.

[1.34] J. Yang, W.-H. Chen, S. Li, L. Guo, and Y. Yan, "Disturbance/Uncertainty Estimation and Attenuation Techniques in PMSM Drives—A Survey," *IEEE Transactions on Industrial Electronics,* vol. 64, no. 4, pp. 3273-3285, 2017, doi: 10.1109/tie.2016.2583412.

[1.35] R. Errouissi, M. Ouhrouche, W.-H. Chen, and A. M. Trzynadlowski, "Robust cascaded nonlinear predictive control of a permanent magnet synchronous motor with anti-windup compensator," *IEEE Transactions on Industrial Electronics,* vol. 59, no. 8, pp. 3078-3088, 2011.

[1.36] A. Sivaprakasam and L. N. Ramya, "A new approach to minimize torque ripple and noise in model predictive control of permanent magnet synchronous motor drives," *Journal of Vibration and Control,* p. 1077546320933743, 2020, doi: 10.1177/1077546320933743.

[1.37] Z. Zhou, C. Xia, Y. Yan, Z. Wang, and T. Shi, "Disturbances attenuation of permanent magnet synchronous motor drives using cascaded predictive-integral-resonant controllers," *IEEE Transactions on Power Electronics,* vol. 33, no. 2, pp. 1514-1527, 2017.

[1.38] J. Stumper, S. Kuehl, and R. Kennel, "Predictive torque control for AC drives: Improvement of parametric robustness using two-degree-of-freedom control," in *2013 IEEE Energy Conversion Congress and Exposition*, pp. 1170-1175, 15-19 Sept. 2013.

[1.39] Y. Zhou, H. Li, and H. Zhang, "Model-free deadbeat predictive current control of a surface-mounted permanent magnet synchronous motor drive system," *Journal of Power Electronics,* vol. 18, no. 1, pp. 103-115, 2018.

[1.40] M. Khalilzadeh, S. Vaez-Zadeh, J. Rodriguez, and R. Heydari, "Model-Free Predictive Control of Motor Drives and Power Converters: A Review," *IEEE Access,* vol. 9, pp. 105733-105747, 2021, doi: 10.1109/access.2021.3098946.

[1.41] M. Fliess and C. Join, "Model-free control," *International Journal of Control,* vol. 86, no. 12, pp. 2228-2252, 2013, doi: 10.1080/00207179.2013.810345.

[1.42] Z. Sun, Y. Deng, J. Wang, T. Yang, Z. Wei, and H. Cao, "Finite Control Set Model-Free Predictive Current Control of PMSM With Two Voltage Vectors Based on Ultralocal Model," *IEEE Transactions on Power Electronics,* vol. 38, no. 1, pp. 776-788, 2023, doi: 10.1109/TPEL.2022.3198990.

[1.43] X. Li, Y. Wang, X. Guo, X. Cui, S. Zhang, and Y. Li, "An Improved Model-Free Current Predictive Control Method for SPMSM Drives," *IEEE Access,* vol. 9, pp. 134672-134681, 2021, doi: 10.1109/ACCESS.2021.3115782.

[1.44] C. Lin, T. Liu, J. Yu, L. Fu, and C. Hsiao, "Model-Free Predictive Current Control for Interior Permanent-Magnet Synchronous Motor Drives Based on Current Difference Detection Technique," *IEEE Transactions on Industrial Electronics,* vol. 61, no. 2, pp. 667-681, 2014.

[1.45] P. G. Carlet, F. Tinazzi, S. Bolognani, and M. Zigliotto, "An Effective Model-Free Predictive Current Control for Synchronous Reluctance Motor Drives," *IEEE Transactions on Industry Applications,* vol. 55, no. 4, pp. 3781-3790, 2019, doi: 10.1109/TIA.2019.2910494.

[1.46] C. Ma, H. Li, X. Yao, Z. Zhang, and F. D. Belie, "An Improved Model-Free Predictive Current Control With Advanced Current Gradient Updating Mechanism," *IEEE Transactions on Industrial Electronics,* vol. 68, no. 12, pp. 11968-11979, 2021, doi: 10.1109/TIE.2020.3044809.

[1.47] A. Traue, G. Book, W. Kirchgässner, and O. Wallscheid, "Toward a Reinforcement Learning Environment Toolbox for Intelligent Electric Motor Control," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 33, no. 3, pp. 919-928, 2022, doi: 10.1109/TNNLS.2020.3029573.

[1.48] G. Book *et al.*, "Transferring Online Reinforcement Learning for Electric Motor Control From Simulation to Real-World Experiments," *IEEE Open Journal of Power Electronics,* vol. 2, pp. 187-201, 2021, doi: 10.1109/OJPEL.2021.3065877.

[1.49] A. Ez-zizi, S. Farrell, D. Leslie, G. Malhotra, and C. J. H. Ludwig, "Reinforcement Learning Under Uncertainty: Expected Versus Unexpected Uncertainty and State Versus Reward Uncertainty," *Computational Brain & Behavior* 2023, doi: 10.1007/s42113-022-00165-y.

CHAPTER 2

A LITERATURE SURVEY ON PERMANENT MAGNET SYNCHRONOUS MACHINE DRIVES

2.1 Introduction

The role of AC machines in control systems is to convert electric energy into mechanical energy to move/drive a load. Several types of AC machines are currently being used in various applications. Permanent magnet synchronous machines (PMSMs) are among the most commonly used and preferred AC machines due to their high torque to current ratio, high power density, low power losses, and high efficiency. However, the higher cost, low robustness (in comparison with induction machines (IMs), and high complexity control are drawbacks of PMSMs. Various control strategies have been developed to cope with these issues and utilize the high efficiency PMSMs in various applications, including vector control techniques or field-oriented control (FOC) and direct torque control (DTC). Over the past ten years, predictive control, or MPC methods have been introduced as exemplary control methods for PMSM drives with several advantages over vector control techniques. Intensive efforts have recently been made to apply, develop, and improve MPC methods for PMSM drives, including improved and robust MPC methods.

This chapter presents a literature survey on PMSM drives, including conventional MPC and robust MPC methods. First, the state of the art of PMSM fundamentals is introduced in Section 2.2. Then, a brief discussion on the PMSMs drives, including scalar

control and vector control (FOC and DTC), is presented in Section 2.3. Next, a detailed discussion on predictive control methods for PMSM drives, including the concept of predictive control, conventional MPC, and improved MPC methods, are presented in Sections 2.4 and 2.5. After that, a discussion and investigation of various PMSM drive uncertainties and their effects on the drive performance are presented in Section 2.6. A discussion of various RPC methods is presented in Section 2.7. The last section (Section 2.8) discusses data-driven control methods, including model-free and RL-based control.

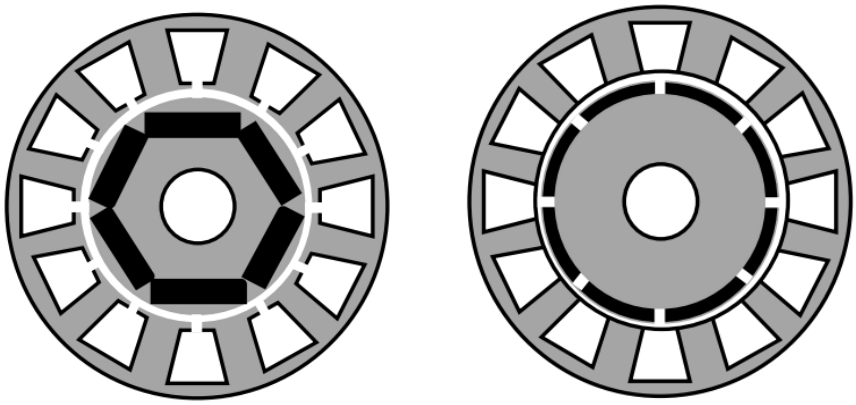
2.2 PMSM Fundamentals

PMSMs are AC machines that utilize permanent magnets (PMs) for excitation. Depending on the location of PMs on the machine, two types of PMSMs are classified: PMSMs with PMs on the rotor or PMSMs with PMs on the stator.

2.2.1 PMSMs with PMs on Rotor

This is a common type of PMSMs widely used in various applications. Depending on the location of PMs on the rotor configuration, they can be divided into two main types: exterior or surface-mounted PMSMs (SPMSMs) and interior PMSMs (IPMSMs). In SPMSMs, the PMs are located on the surface of the rotor to directly face the air gap and stator winding, while in IPMSMs, the PMs are buried inside the rotor cores [2.1-2.3]. The structures of SPMSMs and IPMSMs are presented in Fig. 2.1. Regarding dynamic models, the main difference between an SPMSM and an IPMSM is that the IPMSM has a variable reluctance that varies with the rotor angle.

In contrast, the SPMSM has a fixed reluctance for any rotor angle. That leads to a uniform air gap and, thus, an equal magnetizing inductance for the direct and quadrature axis [2.4-2.5]. Besides, other novel PMSMs with PMs on rotor designs have been proposed to achieve additional features for specific applications, including PM hysteresis hybrid rotor machines that can produce high starting torque [2.6], 4-layer hybrid windings synchronous machines that can achieve high air-gap and power density [2.7] and double rotor permanent magnet machine to be used as traction machine and can achieve high torque and fewer torque ripples [2.8].



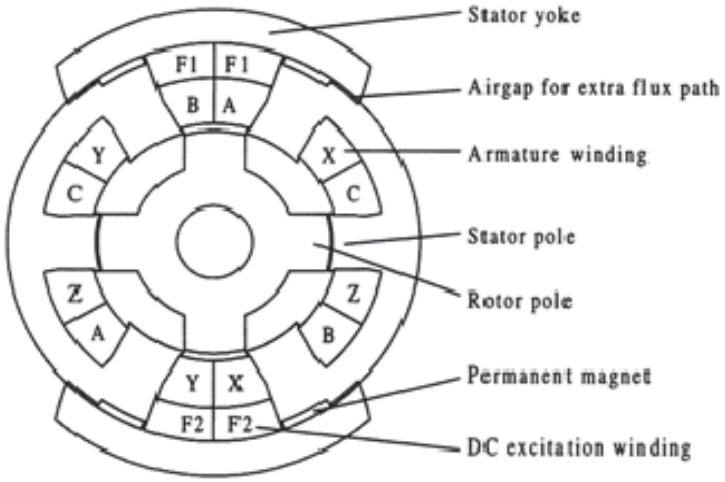
(a) (b)

Fig. 2.1. Structures of PMSMs with PMs on the rotor, (a) IPMSM, (b) SPMSM [2.9].

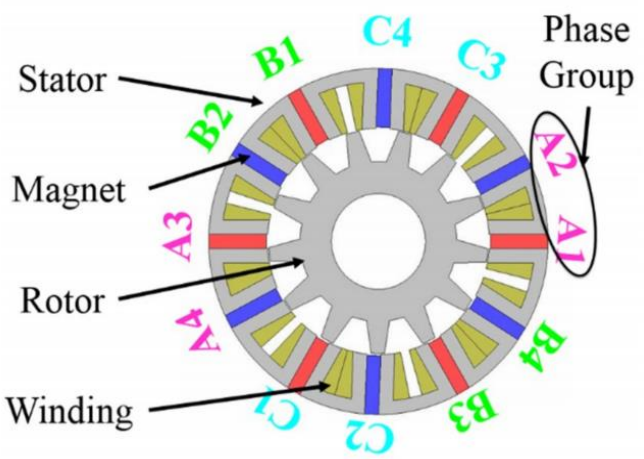
2.2.2 PMSMs with PMs on Stator

With the PMs located on the stator, the rotor must have a silent pole configuration, thus being similar to the switched reluctance machines (SRMs). Such PMSMs types have the merits of simplicity, rugged structure, and high-speed operations. If the PMs are located on the stator back-iron, it is called doubly salient permanent magnet machines (DSPMMs) [2.10-2.11]. If the PMs are fitted to the stator teeth, it is called flux-switching permanent magnet

machines (FSPMMs) [2.12-2.13]. The configurations of DSPMMs and FSPMMs are presented in Fig. 2.2.



(a)



(b)

Fig. 2.2. Structures of PMSMs with PMs on the stator, (a) stator doubly fed DSPMM [2.11], (b) FSPMM with phase-group concentrated-coil (PGCC) windings [2.14].

2.3 Conventional PMSM Drives

PMSM control techniques are evolving like other AC machines. In the past, PMSM drives have been controlled utilizing scalar control (V/F) by controlling the magnitude of voltage and frequency of the supply to maintain a constant (V/Hz) over the entire speed operation. Scalar control is an open-loop control suitable for applications that do not require a good dynamic performance [2.15]. However, for high-performance PMSM drives, scalar control is inadequate, and advanced control methods need to be applied [2.16]. Vector control offers superior performance and overcomes all the issues of scalar control. It utilizes the space vector concept and phase-transformation techniques to obtain an independent flux and torque control, therefore, incorporating a separately excited DC motor control [2.17]. Two popular vector control methods are commonly used in PMSM drives, namely FOC and DTC [2.18].

FOC was first introduced in the early 1970s as an attempt to control AC machines, similar to separately excited DC machines with independent torque and flux control [2.19]. The principle of FOC incorporates flux and torque decoupling control by transforming the stator currents into *d*-*q* rotating frames where the flux is controlled with a *d*-component. In contrast, the torque is controlled with a *q*-component [2.20-2.21]. In contrast, DTC was first introduced in the mid of 1980s [2.22-2.23] to control the torque and flux of AC machines and generate the inverter switching pulses based on a pre-defined switching table. Compared to FOC, DTC directly generates the inverter pulses based on the switching table without using a modulator [2.24]. The configuration of vector control consists of PMSM, phase-transformation, speed controller, current/torque and flux controllers, and inverter, as shown in Fig. 2.3.

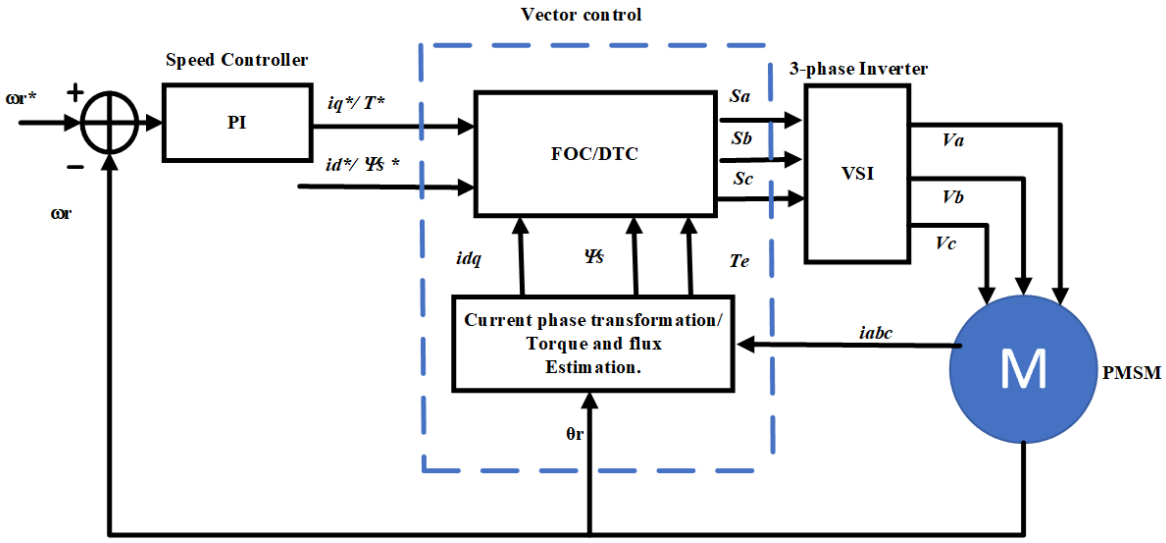


Fig. 2.3. Block diagram of vector control technique for PMSM drives.

2.4 Model Predictive Control of PMSM Drives.

MPC is an advanced control method that has recently emerged in machine drives. This is due to nonlinearity control, constraint inclusion, simplicity, elimination of current controllers compared to FOC, and selection of more accurate switching vectors compared to DTC. The principle of MPC is predicting the future states of machine variables based on their present states. The operation of MPC includes measurement of the machine variables, estimation of the variables that cannot be measured, and finally, prediction of future values based on the measured and estimated values [2.25-2.26]. Based on the controlled variables, MPC can be classified into predictive torque control (PTC), which uses torque and flux as the control variables [2.27]. Predictive current control (PCC) uses the stator currents as the control variables [2.28], predictive speed control (PSC) uses the speed of the machine as the control variable [2.29], and cascade predictive control which regulates the torque/ current

and motor speed with two separate MPCs [2.30].MPC can be classified into a continuous control set (CCS-MPC) and a finite control set (FCS-MPC) based on the switching signal state. The implementation of CCS-MPC is based on voltage vector calculation by minimizing the cost function for reference tracking and requires a PWM modulator to generate the switching pulses.

In contrast, FCS-MPC is based on determining an optimal voltage vector that minimizes a pre-defined cost function and directly generates the switching signals without a modulator [2.31]. A comparison of FCS-MPC and CCS-MPC for machine drives was discussed in [2.32] and [2.33] compared the performance of FCS-MPC and CCS-MPC and showed both obtained similar performance. Considering the discrete nature of the power converter, FCS-MPC is commonly used in power converters and machine drives [2.31]. The block diagram of the general MPC is shown in Fig. 2.4.

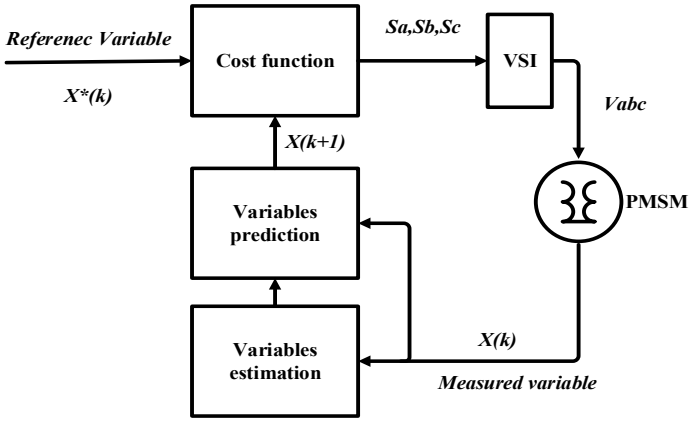


Fig. 2.4. General block diagram of MPC-based PMSM drives.

2.4.1 Continuous Control Set MPC (CCS-MPC)

In CCS-MPC, the discrete nature of the power converter is not considered; instead, a PWM modulator is used to abstract the integer nature of the converter. The principle of CCS-

MPC is similar to vector control (FOC), where a modulator is used, and a regulated switching frequency is attained, except that CCS-MPC achieves a faster transient response with a slight overshoot [2.34]. A continuous reference voltage is computed via the cost function minimization and then synthesized by a PWM modulator to obtain the switching pulses. The design methodology of CCS-MPC is based on Taylor series expansion to solve an optimization problem. Unlike FCS-MPC, the constraints on the switch changes cannot be handled in CCS-MPC because a PWM modulator is always required to generate the switching pulses for the converter. The advantages of CCS-MPC include regulated switching frequency, less computation, and smaller current and torque ripples, however; it is very sensitive to parameters mismatching and external disturbance.

CCS-MPC has been applied for PMSM drives in [2.35], where a disturbance observer was combined with CCS-MPC to estimate the lumped disturbance and compensate for their effects. Also, [2.36] has proposed a CCS-MPC for PMSM drives using a multi-step tracking error technique to reduce the overshoot of the current response and incorporate external disturbance and parameter mismatching. Besides, [2.37] utilized CCS-MPC to control PMSM drives for the PV water pumping system, and [2.38] implemented CCS-MPC for PMSM drives with the consideration of input voltage constraints.

2.4.2 Finite control Set MPC (FCS-MPC)

Finite or direct control set MPC considers the discrete nature of the power converter and minimizes a cost function to directly generate finite switching pulses for the converter. The FCS-MPC method incorporates the concept of DTC, where both attain high sampling frequency and variable switching frequency, except that FCS-MPC method can achieve a better current response and a lower sampling frequency [2.39-2.40]. The design methodology

of FCS-MPC is based on the Euler discretization method to solve the optimization problem. The advantages of this method include less complexity where the PWM modulator is eliminated, good steady-state response, and flexibility to define the control objectives [2.41]. However, high torque, current ripples, and high computation requirements are the drawbacks of FCS-MPC [2.42-2.43]. A detailed discussion of the advancement in FCS-MPC for power converters and machine drives was presented in [2.44], which shows that FCS-MPC has a comparable performance with the classical control methods and is generally superior in transient response and flexibility.

FCS-MPC is the most popular MPC method usually applied for PMSM drives, such as speed control [2.29, 2.45], torque control [2.46-2.52], and current control [2.53-2.58]. Besides, several studies have focused on eliminating the drawbacks of FCS-MPC-based PMSM drives, such as variable switching frequency [2.59-2.60], higher torque ripples [2.61-2.62], higher computational burden [2.63-2.64], and the effect of weighting factor [2.65-2.66].

2.4.3 Cost Function Selection

The cost function can include several types of terms related to different control requirements for the system [2.31]. One such type is the reference following cost function, which includes terms representing a variable following a reference, such as current control, torque control, and flux control. These terms can be expressed generally as the error between the predicted variable and its reference [2.48].

Actuation constraint cost function, where any measure of control effort is considered an additional term in the cost function. In power converters and drives, the control effort is related to the voltage or current variations, the switching frequency, or the switching losses.

For example, in a three-phase inverter, the control effort can be represented by the change in the voltage vector, where the difference between the previous and the current-voltage vector is added as an additional term to the cost function [2.67].

2.5 Improved MPC Methods

Despite the merits of MPC control methods, their conventional structure still faces some challenges, such as high torque and current ripples, variable switching frequency, proper selection of weighting factor, and high computational requirements. Therefore, numerous studies have focused on improving conventional MPC types to eliminate these challenges.

The first essential improvement made to the MPC is the time-delay compensation. In the conventional MPC, the machine variables are measured at time instant (k) and then predicted at time instant ($k+1$). However, the actuating signals (voltage vector) are only available at time instant ($k+2$); this creates a step time delay [2.63]. To compensate for this delay, the variables are predicted at time instant ($k+2$). Various literature has included the time-delay compensation for MPC; for example, [2.68] proposed an MPC for PMSM drive based on real-time optimization considering one-step delay compensation. Similarly, [2.69-2.71] have used two-step prediction to compensate for the time delay between state measurement and actuation state and avoid drive performance deterioration. Due to the simplicity and influence of time-delay compensation in MPC based PMSM drive, it has been embedded improvement to MPC schemes including other modified MPC.

2.5.1 Multi-Vector MPC

In the conventional MPC methods, only one optimum voltage vector is applied for the entire control cycle, which tends to produce a variable switching frequency and increase

the torque ripple. To eliminate this issue, additional vectors are applied along with the optimum vector during a control cycle with an appropriate duty ratio control. In [2.61], two-vector-based PCC for PMSM drives with vector duration control was proposed. The first vector was selected similarly to conventional MPC, while the second vector was among the adjacent vectors to the first optimum vector to ensure only one change of switching state at each control cycle, thus avoiding high switching frequency. In addition, a duty cycle optimization was used to ensure adequate time duration of the first vector.

Furthermore, [2.62] has proposed a three-vector PCC for PMSM drives based on the space vector modulation (SVM) technique. In this method, two active voltage vectors and a zero vector are selected to be applied for one control cycle, and their durations are calculated based on deadbeat control. The cost function is evaluated for three voltage vectors, then applied in one control cycle with the help of space vector modulation (SVM) to predict the next switching state. Unlike [2.61], which firstly evaluates the cost function to obtain the optimum first vector and calculates its duration, then selects the second vector from a combination of two adjacent active vectors to the first vector and zero vector and applies it for the rest of the duration. Similarly, [2.72] proposed a two-vector-based PTC with duty ratio control for PMSM drives, where the selection of the first and second vectors is similar to [2.61], except the vector duration is evaluated in the cost function. In addition, a two-vector-based PCC of PMSM drives was proposed in [2.73] to improve the steady-state performance. Firstly, a reference vector was calculated based on deadbeat current control; the first optimum vector was selected as the nearest active vector to the reference vector. While the second vector was selected among three candidates to be two adjacent active vectors to the first vector and zero vector, and thus a candidate with minimal distance from

the reference vector was selected. Unlike other two-vector MPC, this method does not require the calculation of the current slope to obtain the vector duration. In the same context, [2.74] proposed a generalized multi-vector MPC for PMSM drives, which combines two MPC methods to obtain the reference vector and calculates the duties cycle based on the space vector or sinusoidal pulse width modulation (PWM) concept.

In summary, multi-vector-based MPC methods are an essential improvement to the conventional MPC, where several studies consider it for PMSMs [2.75-2.76] and other machine drives [2.77-2.78]. It has been reported to improve the steady-state performance, reduce the torque and current ripples, regulate the switching frequency, and accurately select the voltage vector.

2.5.2 Weighting Factor Selection

The cost function of PTC method is typically based on torque and flux, which have different magnitudes and units. Therefore, a weighting factor is required to balance the performance of torque and flux. The selection of the weighting factor is based on the rated value of torque and flux. However, the selection of the weighting factor significantly impacts the drive performance; thus, this fixed value weighting factor might be inadequate for enhanced drive performance. Several research papers have considered different approaches to obtain an appropriate weighting factor. In [2.65], PTC without weighting factor for PMSM drives was proposed where a cost function-based voltage vector tracking error was used instead of a torque-flux error-based function. The deadbeat-direct torque flux control (DB-DTFC) principle was utilized to obtain the reference voltage vector that ensures the torque and flux error converge to zero. Thus, a cost function-based voltage vector was proposed, which does not require a weighting factor. The advantages of such a method are eliminating

non-trivial tuning of the weighting factor and reducing the computational burden associated with the cost function-based flux-torque error.

A similar principle has been adopted in [2.66], where PTC of PMSM drive-based voltage vector cost function was proposed. In addition, [2.79] has eliminated the weighting factor from PTC based PMSM drive by using a new lookup table of DTC, and only three voltage vectors are predicted and evaluated in the cost function. Torque-flux error-based cost function is commonly used for PTC; alternatively, the proposed method predicts the angle between the stator and rotor flux vectors and evaluates a cost function based on the error between the predicted angle and a reference angle. This method eliminates the weighting factor issue and reduces the system computation cost since it uses a new lookup table of DTC with only three voltage vectors. Furthermore, an improved weighting factor selection of PTC for PMSM drives has been proposed in [2.80]. An optimal weighting factor was selected using a radial basis function neural network (RBFNN).

In general scope, weighting factor selection techniques and guidelines in MPC methods are discussed in [2.81-2.82]. In addition, a weighting factor selection method for MPC in power converter using a neural network method has been proposed in [2.83]. With the neural network, a powerful and fast optimization is obtained, the responses from the network very well match the responses derived from the model, and the acquired weighting factor is robust to load variations.

2.5.3 Switching Frequency Regulation

Common-mode voltage (CMV) suppression to regulate the switching frequency is another improvement of conventional MPC methods for PMSM drives. CMV is the voltage between the midpoint DC-link capacitor and the neutral point of the load. If the frequency of

CMV is very high, the leakage current and electromagnetic interface can increase, and the motor shaft can be damaged [2.84-2.86]. Various studies have proposed different methods to address the effects caused by CMV in PMSM drives, including improvement to the conventional MPC techniques. In [2.59], an MPC with constant switching frequency was proposed to suppress the CMV in PMSM drives. Compared to the conventional MPC, where only one active vector is applied, this method utilized four active vectors in the next control cycle. The switching sequence model is developed to keep the switching frequency fixed and equal to the control frequency. This method applies the four active voltage vectors [2.87], where a zero vector is avoided to restrict the amplitude of CMV, and two non-adjacent vectors are used to create the equivalent zero vector.

The CMV and switching frequency can be improved with a total prohibition of zero vector, but the current signal quality is significantly affected. Therefore, [2.60] has proposed an MPC for PMSM drives with the realization of both CMV suppression and current distortion. A variable MPC was developed to suppress the CMV while maintaining good current quality and low switching frequency. The proposed scheme did not completely forbid the use of zero vector; alternatively, it sets an evaluation mechanism to determine whether to use zero vector or not. Three active and zero vectors are obtained with four cost functions during each control cycle. As the cost function of MPC represents the difference between the reference current and predicted current, a current error limit was introduced based on the current distortion requirement [2.88].

In addition, common mode current (CMC) suppression was proposed for MPC of PMSM drives in [2.89]. Because the variation in CMV primarily causes CMC, the study suggested decreasing the CMV variation by optimizing the output sequence of the voltage

vectors. Considering three output vectors (two active and zero vectors) for PTC-based PMSM drives, if these vectors are optimized, there is no variation in the next control cycle. This can be achieved by incorporating the CMV variation in the cost function. The issues of this method are that the calculation and prediction of CMV will increase the computational requirement of the system, the selection of appropriate weighting factors for torque-flux and CMV variation is an ambiguous process, and their values significantly impact the system performance.

Furthermore, the CMV suppression method based on two-vector MPC for PMSM drives was proposed in [2.90]. Like [2.59] and [2.87], the proposed method eliminates the use of zero vector and utilizes only six active vectors considered in the cost function optimization. It also utilizes the concept of two-vector-based MPC similar to [2.91] with different calculation methods of the optimal durations of the selected vectors. Under the premise of better current quality, reduced CMV, and less computational burden, an efficient method was designed to calculate the optimal duration of the two selected vectors. The first vector is selected based on the conventional MPC, while the second vector is selected in accordance with the error between reference and predicted currents. For six active vectors, the current error is classified into four categories. Based on these categories, the second vector should be selected such that a minimal cost function is obtained.

Moreover, a new CMV reduction method was proposed in [2.92], where an H7 inverter (VSI with additional switch *S*7) using Zener diode-based PCC for PMSM drive was implemented to block the DC-bus voltage when a zero vector is applied. Compared to other CMV reduction methods, the proposed method reduces drive complexity and considers the

dead-time effects in the PMSM drive system. However, this method requires an additional compensation algorithm for the gate signal of the additional switch (*S7*).

Finally, CMV suppression in PMSM drives with MPC is an outstanding improvement for the conventional MPC since it can reduce the switching frequency and/or improve the current quality. An effective MPC with CMV reduction should realize the trade-off between switching frequency, CMV suppression, and current distortion.

2.6 PMSM Drive Uncertainties

In PMSMs drives, uncertainties are unavoidable and generated from diverse sources such as load changes, environments, and the mechanical or electrical parts of the motor systems [2.93, 2.94]. PMSM drives' uncertainty can be classified into manufacturing or structural and operational or environmental uncertainties. The structural uncertainties are related to the machine structure and dimension, material diversity, assembly imperfection, frictions, and mechanical factors [2.95-2.102]. In contrast, operational ones are related to the uncertainties generated during the system operation, including machine parameter mismatching, inverter dead-time, measurement error, and external load disturbance [2.103-2.107]. Depending on the type and level of uncertainty, PMSM drives may fail or be significantly affected, resulting in poor or unsatisfactory performance, such as high torque ripples, high current harmonics, slow dynamic response, and low efficiency [2.108-2.109].

The dynamics of PMSMs are essentially nonlinear and subjected to a wide range of disturbances or uncertainties in many high-performance applications. The disturbances or

uncertainties in PMSM systems can be classified as unmodelled dynamics, parametric uncertainties, and external disturbances.

2.6.1 Unmodeled Dynamics

The actual PMSM drive system has complex dynamics in which some properties are not modelled, constituting the unmodelled dynamics. The unmodelled dynamics of PMSM drives are as follows.

A. Machine Body Structure Induced Torque

Due to the utilization of different rotor materials in PMSMs, the body structure may induce various pulsating torques. For example, the cogging torque is generated by the interaction of the rotor magnetic flux and angular variations in the stator magnetic reluctance [2.95]. The cogging torque even exists when the system is disconnected from the power source, as the cogging torque is generated from the structure of the motor. From the energy perspective, the energy storage in the air gap of the stator slot is not constant, and the fluctuation in this energy causes torque fluctuation. Cogging torque compensation using a mathematical method is very difficult because of the lack of a precise cogging torque model. However, it can be simplified as a periodic function of the rotor position [2.110] and represented by a Fourier series [2.111].

Cogging torque is a primary source of machine torque ripples, a noise factor, and it is difficult to start the motor with high cogging torque. Also, it causes several adverse effects on the PMSM operation, such as mechanical vibrations that may reduce the machine bearings' lifetime, acoustic noise, and positioning errors in the case of precision systems [2.112-2.113]. To mitigate the effect of cogging torque, the machine body structure can be improved by considering different methods, such as optimizing pole shape and stator teeth

[2.114], adequate selection of the number of poles and slots [2.115], skewing the stator core or PMs [2.116], asymmetrical positioning of PMs or stator teeth [2.117-2.118], segmentation of PMs [2.119], and filling the stator slots with magnetic wedges.

In addition, magnetic flux harmonics are a significant cause of torque pulsation. The most widely used material in the magnet of PMSMs is Neodymium Iron Boron (NdFeB), whose flux density is easily affected by temperature variation. The resultant demagnetization phenomenon of PMs due to temperature rise significantly impacts the maximum torque capability and the efficiency of PMSMs [2.96]. Owing to the motor structure and processing defect, the motor's air-gap magnetic density is non-sinusoidal; for example, the motor slot structure's existence can destroy the flux density's sinusoidal nature. This results in an imperfect sinusoidal flux-density distribution that produces periodic torque ripples interacting with standard stator currents. An improvement in structure design was introduced for ideal excitation magnetic field to suppress harmonic components using finite element analysis [2.120-2.122].

B. Dead-Time Effects

In a power electronics device, a dead-time is a short blanking time between the device's ON- and OFF-state to prevent the phase shortage of inverter arms. The dead-time causes a loss of a portion of the duty cycle, thus distorting the voltage applied to the drives [2.123-2.125]. Such effects become extremely severe near the zero crossing of the current. The resultant current deterioration leads to ripples in the electromagnetic torque. A voltage-fed inverter is never ideal; in practice, the switching dead-time, the device's ON-state voltage drop, and the dc-bus voltage variations can adversely affect the control performance, particularly during the steady-state operation. The most obvious effect is the distortion of the

output current caused mainly by the low-order harmonics [2.105]. In the current control loop, the dead-time effect introduces periodic disturbances, which could lead to distortion of stator currents, especially in extremely low speed and heavy load conditions [2.126-2.128]. In the MPC of PMSM, the dead-time induced harmonics lead to prediction error. Even though these harmonics are quite small, the prediction error will be amplified when the prediction step increases. Their effects on the control system cannot be neglected to obtain high-performance control [2.129].

C. Measurement Error Effects

In AC machine drives, position or current measurement errors inevitably cause torque ripples. For example, the offset error in current measurements superimposing directly on the phase currents via the Clarke and Park transformations causes ripples on stator currents in the *dq* frame [2.130]. During the current measurement, inaccurate current acquisition introduces measurement noise, causing a DC offset. The output of the current sensor must be scaled to match the input of the A/D converter. In the digital form, the controller rescales the value of the A/D output to obtain the actual value of the current, thus introducing a scaling error [2.106].

In real applications, stator currents are measured through the Hall sensor or high-precision resistance, which can lead to periodic measurement errors. The current measurement errors include current dc offsets and scaling errors. The DC offsets can cause the measured current error in the *dq* coordinates to oscillate at the electrical angle frequency, and the scaling errors can cause the measured current error in the *dq* coordinates to oscillate at twice the electrical angle frequency [2.131].

2.6.2 Parametric Uncertainties

PMSMs consist of different parameters subjected to variations during operation due to temperature rise or other environmental effects. These include electrical parameters of stator resistance (R_s), stator direct inductance (L_d), stator quadrature inductance (L_q), the permanent magnet flux linkage (Ψ_{pm}), and mechanical parameters of rotor moment of inertia (J), and viscous friction coefficient (B). These parameters' typical or nominal values are normally obtained with offline measurements at nominal operating conditions or provided by the motor manufacturer [2.132-2.134]. However, considering the manufacturing tolerances and changes in the operating conditions, the actual values of these parameters may differ from the nominal ones. Thus, PMSM parameter variation will occur due to manufacturing tolerances and/or operational factors. Manufacturing tolerance is a certain inaccuracy range in a given typical value of a machine variable due to geometric dimensions and material properties tolerances. Thus, the manufacturing tolerance of each parameter depends on the machine manufacturers and what geometric dimensions and materials are used. Besides, operational factors are changed in operating conditions, such as temperature and load variations.

Parametric uncertainties can significantly affect PMSM drive performance, especially with parameters-dependent control methods like MPC. For instance, the stator resistance (R_s) varies with the variation of winding temperature, which significantly impacts the current-loop regulation performance. These effects become severe at low speeds or high-load torque conditions [2.135]. Besides, the PM flux linkage (Ψ_{pm}) depends on the dq-axis currents and the magnet temperature. A mismatch of the rotor flux has a significant influence on medium and high speeds because the back electromotive force (EMF) is proportional to

the rotor flux. If the rotor flux is inaccurate, a constant current error occurs in steady-state, and overcurrent or undercurrent occurs in transient state [2.104]. The variation of PM flux can also lead to a steady-state current error with nonzero velocity [2.136]. The machine inductances (L_d,L_q) mainly influenced by the flux density and dq-axis currents and slightly vary with temperature changes. Also, the inductances vary nonlinearly with respect to the load conditions due to magnetic saturation. The effect of inductance variation is mainly coupled with the current change, and the transient performance is primarily affected in the current dynamic period [2.137].

Furthermore, as the variations of electrical parameters significantly affect the performance of current and torque control loops, mechanical parameters also vary during real-time operations and significantly influence the speed control loop. For example, the machine inertia (J) varies when a load is applied to the machine. The shape and the dimensions of mechanical loads mainly affect the variation of mechanical parameters [2.138-2.140]. The inertia of the whole drive system for some applications is time-varying [2.103]. If the system's inertia increases to some values more than the original, the speed response will have a bigger overshoot and a longer settling time [2.141].

The performance of MPC of PMSM drives highly depends on machine parameters, and variations in these parameters distort the measured and predictive current. Consequently, errors between the predictive, measured, and reference currents will be introduced. Thus, the final voltage vector cannot be precisely predicted by MPC.

2.7 Robust Predictive Control Methods of PMSM Drives

PMSM drives are generally formed with feedback controllers, including feedback speed, currents, torque, and/or flux controls. In control theory, the primary aim of feedback control is to force a system output to track a reference (desired) input of that system, as well as reject disturbances and suppress measurement noises [2.142-2.144]. Various control techniques can be used to design a feedback controller for a process [2.145-2.148]; however, the controller is always designed based on an approximate model representing the dynamic behaviour of that process [2.148]. The model's accuracy varies but never perfectly describes the actual process; also, the behaviour of the process change with time, and these changes are mostly not captured by the model [2.149]. This is referred to as model uncertainty which can degrade the controller performance in the real process [2.150]. Uncertainties are inherent in real-world processes, and robust control methods are introduced to cope with these uncertainties [2.151].

Robust control implies the ability of a control system to maintain desired performance (robustness) in the presence of uncertainties. Control system robustness is the property of tolerating uncertainties in the system without exceeding predefined tolerance bounds in the vicinity of some nominal dynamic performance [2.152-2.154]. Sometimes, robustness is evaluated as the system stability is far from being affected by uncertainties /disturbances. However, although stability is necessary for robustness, it is not the only desired control objective, and system performance has to be considered [2.149, 2.155, 2.156]. Therefore, robust analysis of a control system with uncertainties not only evaluates the stability property but also assesses whether system performance remains within predefined bounds in the vicinity of nominal performance for a complete set of system uncertainties [2.157-2.158].

With various uncertainties and their diverse effects on the performance of PMSM drives, robust PMSM control methods have been introduced to enhance the system's robustness. Due to the merits of predictive controls and their high dependency on the machine model and parameters, RPCs have been massively investigated for PMSM drives. Based on the robust mechanisms, RPCs can be divided into different types: RPC-based, prediction error correction, observers, model-free, optimized cost function, and hybrid techniques.

2.7.1 RPC-Based Prediction Error Correction

One of the simplest robust methods is the RPC-based prediction error correction, where the prediction error is included in the prediction stage to compensate for any control effort. Such a method was proposed in [2.159]**,** where a current error correction controller was implemented to generate a more accurate reference current. This can compensate for the output current incapability of tracking the reference current due to parameter variation and the non-linear operation of the inverter. Although this method has obtained specific good results, however; it increases the complexity of the current regulation scheme and neglects the effects of winding resistance and other drive uncertainties.

Another RPC-based prediction error was proposed in [2.160] using the principle feedforward linearization method to solve the problem of parameter sensitivity of PMSMs. The weighted errors between the predicted and measured values in the last sampling instant are added to prediction equations in the next sampling instant to compensate for parameter mismatching. This method has reported a reduction in the parametric sensitivity of PTC compared to the conventional PTC while maintaining excellent dynamic performance. However, only the effect of machine inductance was considered, while other parameters were neglected. Also, the prediction errors of the switching states applied during each sampling

instant are different. Therefore, using the prediction error of the last sampling might not be accurate to compensate for parameter variation in the next sampling.

RPC-based prediction error was proposed in [2.161], where the same principle in [2.160] was applied to the PCC of PMSM drives with the difference that the prediction error of each switching state is added to the prediction equations of the same switching state. Thus, by adding the prediction errors to the current's prediction stage with weighting factors, the predicted currents are close enough to the motor currents' real behaviour, enabling the PCC algorithm to select a better switching state for the next control cycle. The same method was applied to PTC-based PMSM drives in [2.162]. With a variation of machine inductance, this method showed a better response than the conventional MPC and RPC in [2.160]. However, this method only considers the effects of machine inductance, while other uncertainties were neglected. Also, a stagnant prediction update may occur if the present switching state is the same as the last one.

Moreover, another RPC-based prediction error was proposed in [2.163], where a current variation mechanism was implemented for MPC-based PMSM drives to compensate for parameter mismatching and improve stagnant current updates. Also, this method utilizes a modified current prediction equation to predict future currents. The performance of this method has been investigated under different PMSM parameters variation. Still, it comprises intensive computation steps, which may increase the computational burden.

RPC-based prediction error was discussed in [2.164], where the current prediction error caused by parameter variation is used to design a self-regulation technique and compensate for the parameter mismatch. Then, the performance is enhanced by correcting the current mathematical model in the control algorithm. Despite improving parameter

robustness in the proposed method, the estimation strategy's complexity requires massive computation, and not all system uncertainties were considered. Another RPC-based current error and parallel compensation terms were introduced in [2.195]. A robust MPC based on a current error with parallel compensation terms onto predictive deadbeat control was designed to compensate for the effects of parameter mismatch in real-time implementation. The proposed method utilizes deadbeat control, which requires a PWM modulator to generate the switching pulses and enhance the system's complexity. Besides, only the effects of inductance and resistance variations were considered, while other uncertainties were neglected.

Furthermore, RPC-based prediction error was discussed in [2.165], where a current prediction error reduction method based on online inductance correction was proposed. It can directly correct the inductance value of the prediction model only using one proportional regulator, which is simple and easy to implement. Compared with other methods, the proposed method can reduce the calculation burden. However, only the effect of inductance mismatch was considered, while other uncertainties were neglected.

2.7.2 RPC-Based Observer

Another RPC type for PMSM drives is based on observers. In this type, single or multiple observers are employed to mitigate the effects of PMSM drive uncertainties. Observers can be used to estimate the disturbances and/or uncertainties in the drive system and compensate for their impact. An example of this RPC was proposed in [2.166], where an incremental prediction model was implemented to eliminate the permanent magnet flux linkage parameter. An inductance disturbance controller that includes a simple disturbance observer and inductance extraction algorithm was implemented to reduce the effects of

machine inductance mismatch. Though specific performances have been improved with this method, it enhances the system complexity, and only the impact of machine inductance was considered, while other parameters were neglected.

In addition, a non-linear RPC scheme with a disturbance observer in a cascaded structure for a PMSM drive in the presence of input constraints was proposed in [2.167]. It consists of two MPC: the inner MPC is used to regulate the armature current by acting on the armature voltage, whereas the outer MPC is employed to track the speed reference by considering the q-axis component of the armature current (i_q) as the input control. Besides, to eliminate the undesired side effect known as integrator windup, an anti-windup compensator is derived from the design of the disturbance observer. The design complexity involving two MPC methods and an observer is one of the drawbacks of this method. Also, the performance of this method was only investigated for low-speed operation.

The RPC-based observer was introduced in [2.168], where a state and disturbance observer was used to estimate the variable parameters, compensating for parameter mismatching and unmodeled uncertainties. Though this method has reported good stability and constraint satisfaction, the two observers used require a high-capability processor to implement in real time. Another RPC-based observer was discussed in [2.104], where an incremental model was adopted to eliminate rotor flux, thus improving flux robustness. Also, the incremental model was combined with an extended state observer (ESO) to estimate the error caused by machine inductance variation and compensate in the prediction model. This method improved the robustness against flux and inductance uncertainties. However, other uncertainties were neglected.

Another RPC-based observer was proposed in [2.169], where a new PMSM model consisting of ideal and disturbance parameters was considered. A speed controller-based sliding mode and a current controller-based deadbeat MPC were designed based on the ideal part. Also, a high-order sliding mode observer was developed based on the disturbance part to estimate the parameters and disturbance uncertainties and compensate for them in the current and speed controller. This method has achieved quick transient response and good steady-state performance. However, the complexity of the control method requires massive computation capability for real-time implementation, and it is time-consuming.

RPC-based observer was proposed in [2.170], where a predictive stator flux control for PMSM drives was designed to ensure good drive performance despite disturbance. A composite discrete sliding mode observer was utilized to estimate the disturbances and compensate for the flux prediction. This method has reduced the torque ripple and current distortion regardless of the disturbances. However, the effects of machine inductance and resistance were not considered. Besides that, [2.171] proposed a robust MPC for PMSM drive, where the optimal voltage vector combination selection of the three-vector MPCC is simplified, reducing the computational complexity. The super twisting algorithm-based second-order sliding-mode observer was designed to observe the lump disturbance. The estimated lump disturbance was used to compensate for the original PMSM model so that the problem of steady-state current error under parameter mismatch can be solved. Finally, the robustness against the motor parameters variation was effectively improved. The shortcoming of this method is that not all parameters variation were taken into consideration.

RPC-based observer was discussed in [2.172], where an accurate PMSM model was analyzed considering the influence of parameter mismatches and measurement errors. A

modified MPC containing an accurate PMSM voltage and nonperiodic and periodic disturbance models were developed. Then, a novel current and disturbance observer (NCDO) in the modified MPC. The disturbances that the NCDO predicts are regarded as feedforward voltage compensation and are directly added into a modified PMSM voltage model. A good performance of the proposed DPCC with NCDO under parameters mismatch and current-measurement error conditions were achieved. However, only the machine inductance mismatch was considered, and the current regulation schemes and observer enhance the system complexity and computational burden.

Besides that, robust predictive current control for PMSM drives based on disturbance observer was introduced in [2.173]. A new predictive current control with a discrete-time DOB estimates the disturbance of the parameter variation online for an IPMSM drive was developed. The proposed observer aims to overcome the parameter sensitivity from the resistance and inductance uncertainties and make a prediction without the need for rotor flux information. The estimated disturbances are compensated with the predicted reference voltage model considering a digital delay. Compared to the conventional MPC, the proposed method can eliminate a steady-state current and transient-state error caused by system disturbances. The PMSM drive with this method under mismatched parameters showed a good performance. However, the method comprises two observers and utilizes a PWM modulator which enhances the system complexity and computational burden, thus increasing the cost of digital implementation.

Virtual vector-based robust MPC for PMSM drives was proposed in [2.174]; the fundamental concept of this method is to reduce the required parameter information in the predictive model. Then, the influence of remaining parameter mismatches was suppressed

through the discrete disturbance observer. This discrete disturbance observer has a unified form with a predictive model, which can simplify the drive system. The virtual and basic vectors tend to increase the computation iterations, leading to a high computational burden on the drive system. Also, [2.175] has proposed robust MPC for PMSM drive based on observer using a multi-step error tracking technique. An extended PMSM model was incorporated by considering the external disturbance and parameter variation in the disturbance part. Then a sliding mode observer is employed to compensate for the effects of these disturbances. The proposed method reduces the overshoot and keeps good steady-state performance. Still, the complexity of the current control and computation required by the observer are among the disadvantages of this method.

In addition, another RPC-based observer was discussed in [2.176], where a robust predictive torque controller is designed based on an unknown torque disturbance observer, which can enhance robustness against parameter mismatch and load torque disturbance. However, this method utilizes an SVM modulator which increases the computation requirement of the system. RPC-based observer was discussed in [2.177], where a novel disturbance method is used based on equivalent input disturbance, a signal applied to the input voltage, and producing the same effects as the actual disturbance. Then, this observer is combined with MPC for PMSM drives to eliminate the effects caused by machine uncertainties. The issue with this method is that the actual disturbance may differ from the observer's estimated disturbance.

Another RPC-based observer was discussed in [2.137], where a robust predictive current control-based adaptive gain disturbance observer for PMSM drive was developed. First, an online adaptation mechanism was designed to extend the robust inductance limit.

Then an adaptive disturbance with the adaptation mechanism is combined with MPC to eliminate the static current error and enhance the transient response and parameter robustness. The issue with this method is that adding an observer enhances the control algorithm complexity and increases the implementation difficulty and the hardware computational burden. In addition, an explicit MPC-based disturbance observer was discussed in [2.178], where an improved disturbance observer based on the augmented model, in conjunction with the concept of offset-free MPC, to estimate both the disturbance terms and the state variables from the predicted and measured outputs. The estimated total disturbance removes all the influences of plant/model mismatches and unmodeled nonlinear terms within the closed-loop framework of explicit MPC. Besides the observer and explicit MPC, the space vector pulse width modulator (SVPWM) requirement enhances the complexity and computational burden of the drive system.

2.7.3 Model-Free RPC

Model-free predictive control is another type of RPC for PMSM drives. In this RPC, the prediction process is performed without using the machine parameters, thus avoiding the effects of parametric uncertainties. An example of this RPC was proposed in [2.179], where a model free based on the current difference detection technique was implemented. This method does not require any knowledge of the parameters of the motor. The stator current during each switching interval is assumed to be a linear. Thus, the current difference within each interval can be precisely computed. The stator current is detected twice at each switching instant to improve the accuracy of the current difference, and a simple subtraction operation is used to compute the current difference. Despite this method's simplicity and model independency, the accuracy of the current prediction at low speed is not good enough.

Another RPC-based model-free was discussed in [2.180], where the PMSM model was designed considering the effects of inverter nonlinearity and parametric uncertainties. Then, model-free predictive control was designed based on the ultra-local model and combined with the PMSM model and deadbeat predictive control. The proposed method does not require the knowledge of parameters. Still, the complexity of the predictive control with two predictive models of the *d*- and *q*-axes and the SVM PWM modulator can enhance the drive system complexity, which may increase the computational burden and hardware cost.

In [2.181], an improved RPC-based model-free was proposed for PMSM by introducing an advanced current gradient updating mechanism. This method does not require a mathematical model and instead employs information about the current gradients to predict future currents. Another RPC-based model-free was presented in [2.182], where a model-free predictive current control based on the current difference was proposed for PMSM drives. The concept is based on the idea in [2.179], with a simplified synchronized update mechanism of current differences. Each of the seven basic voltage vectors of the two-level inverter was updated in real-time without extensive calculations. This method's drawbacks are system complexity with long calculations and the inaccuracy of current difference estimation may still lead to stagnation. In [2.183], RPC was proposed based on model-free PMSM drives based on an ultra-local model and using an extended state observer. It applies the same concept as [2.180], but the disturbances are estimated using an extended state observer. Besides that, [2.184] proposed an improved RPC-based model-free SPMSM drives. It applies the same concept as [2.180], but the disturbances are estimated using a sliding model observer.

2.7.4 RPC-Based Optimized Cost Function

Another RPC type for PMSM drives is based on an optimized cost function, where the cost function is modified to achieve drive robustness. An example of this RPCC type was proposed in [2.185]. A novel cost function was proposed based on the predicted integral action of the tracking error, and the controller was developed under the assumption that the system is free from any disturbance and mismatched parameters. It was shown that this leads to an integral action in the controller, which is exploited to improve the disturbance attenuation without using an offset observer. Based on simulation and experimental results, the proposed method has shown high performance concerning speed tracking and current control of the motor. Another RPC-based optimized cost function was proposed in [2.186], where a novel cost function that utilizes torque tracking, maximum torque per ampere condition, switching losses minimization, and system constraints to reduce the current, torque, flux ripples, and acoustic noise. Compared with DTC and conventional MPC, this method has shown improved performance, reduced torque ripple, lower current harmonics, reduced switching frequency, and less acoustic noise. However, compared to conventional MPTC, the new cost function utilizes four weighting factors. Their fixed values may degrade the performance, and the effects of parameters variation were not considered. Also, it requires the prediction of current, torque, and flux, which increases the system's calculation steps.

In [2.29], a direct speed MPC was developed based on a novel cost function with three terms: tracking term to track the speed reference, attraction region term to attract the system state in steady-state efficiently, and limitation term to limit the current. Besides, the proposed cost function utilized a new technique to reject the disturbance and noise. Although, this method reduces the system complexity by combining speed and current control in one

controller. However, the effects of parameter variations and other model uncertainties were not considered. Also, the proposed cost function contains three weighting factors in which the selection of their values influences system performance. The same method was applied for predictive direct torque control in [2.187] based field weakening operation and [2.188] based maximum torque per ampere operation.

Besides, robust MPC for PMSM with a newly designed cost function was proposed in [2.189]. A robust FCS-PCC strategy with a cost function in proportional-integral (PI) form for PMSM drives is very simple and practical to implement. The accumulated errors are weighted with the sampling time, and the integral action is activated in a predefined range. In this way, the design of the Integral coefficients is facilitated. This method demonstrates superior robustness compared to the conventional MPC under parameter mismatch. The drawback of this method is it requires intensive calculation steps and time to obtain the best performance.

2.7.5 RPC-Based Hybrid Technique

Another RPC type is based on hybrid techniques, combining two or more control techniques to produce a robust control strategy capable of dealing with machine uncertainties. An example of this RPC type was proposed in [2.190], where the complementary features of continuous input and discrete input MPC techniques were combined to ensure stability, robustness, optimal nonlinearity, and constraint inclusion for VSI-fed PMSM drive. A continuous control was designed based on the Lyapunov function to ensure stability with feedback control and robustness with adaptive control. Because the stability is guaranteed by at least one discrete switching state, the continuous control is converted into relevant constraints of the discrete control MPC. Thus, utilizing the discrete MPC optimization with

exhaustive search to find the switching pulses for VSI. The disadvantages of this method are the system complexity and possible computational burden increment. In addition, different gains and weighting factors are used, which require tuning and proper selections to avoid performance degradation in which their values may affect the performance.

Another RPC-based hybrid technique was introduced in [2.46], where a deadbeat (DB) solution was combined with MPTC to select the best voltage vector (VV). The cost function needs to evaluate only three VVs selected by the DB solution instead of evaluating eight feasible VVs for a 2-level Inverter. The calculation iteration is reduced, and the parameter variation is investigated. Although a good steady-state response was achieved, the parameter mismatch affects the system stability during the transient response. The effects of magnet saturation and other machine uncertainties were neglected. Another RPC-based hybrid technique was discussed in [2.128], where integral-resonant control composed of several paralleled quasi-resonant controllers is embedded in the standard MPC algorithm to restrain the periodic disturbances. The proposed method does not need to store a large number of past time variables so that the computational complexity is reduced. In addition, it has relatively strong frequency robustness because the resonant internal model can adjust its control bandwidth conveniently. This method has the merits of suppressing periodic disturbance. However, the effect of other uncertainties was not considered.

RPC-based hybrid technique was introduced in [2.129], where a less computational simplified repetitive controller with two resonant units and phase compensation was combined with MPC for PMSM drives to realize the system's robustness against disturbances. Based on simulation and experimental results, the current ripple significantly reduced with the proposed SRC method in the presence of disturbances. A similar approach

was proposed in [2.191], which attempted to combine the features of both MPC and RC for linear motion drives to reduce the tracking error from the periodic disturbances. However, these methods require a long adjustment time to produce the best performance and can only be applied to a repetitive task. Moreover, another RPC-based hybrid technique was proposed in [2.192]. A hybrid flux prediction that combines the voltage and current model was developed in the prediction stage. A closed-loop current prediction model was designed to improve the performance in the presence of parameter variation. Though the parameter mismatching was compensated, the complexity of the proposed current, voltage, and flux model enhanced the overall computation requirement of the system.

2.8 Data-Driven Control of PMSM Drives

Model uncertainties and parameter mismatching are inevitable in PMSM drives. The PMSM model fails to capture various dynamics and changing parameters with different operating conditions [2.109]. These uncertainties can significantly impact the performance of model-based control, leading to reduced system robustness [2.166]. Although various techniques have been developed to enhance the robustness of model-based control methods, they often increase system complexity and computational requirements and may involve effective tuning of multiple parameters.

A practical alternative to robust control methods, offering efficient computation, is data-driven control approaches. These approaches enable control based on system input/output data, eliminating the need for a system model and its associated parameters. One approach is model-free predictive control (MFPC), which performs online prediction solely based on measured input data without relying on a parametric system model. In contrast,

reinforcement learning (RL) achieves data-driven control by training the RL algorithm offline, resulting in a computationally efficient controller compared to MFPC.

2.8.1 Model-Free Predictive Control (MFPC)

MFPC-based current control (MFPCC) has received significant attention in the field of PMSM drives. Depending on the technique used to achieve MFPC (such as current difference and ultra-local model), stagnation effects can occur, and higher computational effort may be required [2.179-2.180]. MFPCC can be achieved using an ultra-local model [2.193], where an ultra-local model replaces a complex system model with one or two unknowns that can be estimated based on measured input and output data [2.184, 2.194]. The effectiveness of such MFPCCs highly depends on the estimation accuracy; also, some estimation methods may increase the computation requirements of the system.

On the other hand, MFPCC can be achieved by solely using the measured currents and their variations. This type of MFPCC mainly depends on the current differences. The current difference due to the recently applied voltage vector is used to estimate the current differences due to the remaining possible vectors of a two-level inverter [2.179]. However, inaccurate prediction can occur when the same voltage vector is applied for long control intervals, resulting in stagnation of the other vectors. This issue was attempted in [2.195] and [2.181] by using the current differences due to the past two/three successive voltage vectors to estimate the differences due to the remaining vectors.

However, the voltage vector applied over two/ three consecutive control cycles cannot be the same for these techniques. This could cause the applied voltage vector to change between two vectors for long control intervals; thus, current difference updating becomes ineffective, and stagnation may occur. Furthermore, most existing MFPCCs utilize

the measured current and applied voltage variations to compensate for the effect of parameter variations. However, parameter inaccuracies influence the reference current when a speed control loop is used. This results in suboptimal tracking performance and high current ripple.

Two-vector modulation MFPCC based on the current tracking error slope was proposed in [2.196]. The current differences due to the possible active voltage vectors were determined using the method proposed in [2.181]. Two adjacent voltage vectors were identified, assuming the tracking error slope falls between two adjacent current differences. Then, these two vectors and a zero vector create three candidates of two-vector combinations that can be applied in each control cycle. This method considers tracking error and two-vector modulation to reduce current ripple. However, when the tracking error slope does not change for a few successive control cycles, only two vectors will be applied, and stagnation may occur. This issue can be more severe in nominal cases with constant steady-state tracking error slope causing a prolonged stagnation, leading to inaccurate prediction, and degrading the performance.

2.8.2 Reinforcement Learning-Based Control

RL-based control has emerged as a promising approach for achieving data-driven control in PMSM drives [2.197-2.198]. An optimal control action is obtained by training an RL agent with appropriate rewards based on measured PMSM data. This results in a computationally efficient controller optimized offline during the training process. RL can be employed to enhance the performance of standard PMSM control strategies (i.e., FOC, DTC, and MPC). For instance, [2.199] utilized RL to obtain the weight coefficients of an improved MPC for PMSM drives, and [2.200-2.201] implemented deep RL to optimize the parameters of active disturbance rejection control of PMSM.

Furthermore, RL can replace the standard control methods of PMSM drives entirely. RL-based current control [2.202] and DTC [2.203] of PMSM drives were implemented by training a deep Q-learning network to learn optimal controllers. These learning-based controllers were then deployed to a real-world drive system and demonstrated comparable performance to the standard controllers [2.198]. RL-based speed control was also trained to achieve optimal speed tracking and replace the standard speed control [2.204-205].

The effectiveness of RL-based controllers depends on the amount and quality of the data used for training. In the standard RL-based controller, an agent learns an optimal policy that maximizes its expected cumulative reward over a single training task with specific operating conditions and a single parameter set. Thus, new operating conditions and different parameter sets (due to parameter mismatching) can lead to poor performance and robustness or instability in the controlled system [2.206].

The optimal learned policy varies for different parameter sets, making it difficult to generalize and adapt a learned policy to new operating conditions with new parameter sets. To generate a policy that can adapt to new tasks (e.g., new motors), meta-RL [2.207] was used to learn a policy that can adapt to new tasks more efficiently and quickly by leveraging prior experience on similar tasks. With a data set of different motor parameters, the environment of each motor data is pictured as a partially observable Markov decision process (POMDP), where the environment state is not fully available to the agent. Then, additional contexts (variables) containing information about the momentary environment are included in the environment state.

However, these contexts can be static within each measurement, and incorporating them into the state creates a larger COMDP and reduces the learned policy's generalizing

power [2.208]. Furthermore, Meta-RL can be computationally intensive since it requires much data to learn the meta-policy and adapt to new tasks efficiently. Additionally, there is a risk of overfitting to the training tasks, where the agent memorizes the training tasks and cannot adequately generalize to test tasks. This can lead to poor performance on unseen tasks or tasks that are significantly different from the training tasks [2.209].

2.9 Summary

This chapter presents a comprehensive analysis of the literature on PMSM drives. Mainly, RPC methods are critically investigated and classified into five types based on the robust mechanism employed. A robust model-free type with no machine model or parameters is found to be a better alternative to robust MPC methods, which tend to increase system complexity. Various model-free control techniques for PSMSM drives are discussed intensively in this chapter. Data-driven control methods, including model-free and RL-based controls, are also investigated in this chapter.

Furthermore, the literature studies of RPC methods are quantitatively analyzed against a set of indicators, as presented in Appendix B. Various research gaps are identified in the literature, and corresponding solutions are proposed to fill these gaps, as will be presented in the following chapters. The gaps identified can be summarized as follows:

- The drawbacks of conventional MPCs have been realized by various improvements such as multi-vector, weighting factor elimination, and switching frequency regulation. However, it is essential to maintain performance trade-off while dealing with the various issues of conventional MPCs. Most existing improved MPCs deal with specific issues and neglect the others, resulting in overall performance degradation. For

instance, regulating the switching frequency by prohibiting zero vectors can result in high current ripples. Two improved MPCs are proposed in this research (see Chapter 3) that consider various issues of MPCs while maintaining an overall performance trade-off.

- Various robust controls were introduced in the literature. However, it is unclear what criteria should be used to define a robust control system and how to compare the robustness of different control systems. A systemic robustness evaluation method based on Six-Sigma Methodology is introduced (see Chapter 4) to numerically obtain any control systems' sigma levels (robustness index) subjected to a bounded range of uncertainties.
- Model-free control methods were found to be a better approach to eliminating the effects of uncertainties, particularly parametric uncertainty. The main issue of the various model-free predictive current control (MFPCC) is the switching vector stagnation when the applied switching vector is not updated for a long interval. This research addresses this issue by developing an adaptive MFPCC (see Chapter 5) that utilizes a reference vector to force updating the current differences and improve tracking performance.
- Reinforcement learning (RL) based controls were implemented to achieve a data-driven PMSM drive based on offline optimized policy, resulting in a computationally efficient controller compared to MFPCC. However, standard RL is usually trained in specific operating conditions with single parameter sets. Thus, new operating conditions with different parameter sets (due to parameter mismatching) can lead to poor performance, robustness, or instability in the controlled system. This issue is

addressed by proposing a multi-set robust reinforcement learning (see Chapter 6) to learn a single optimal policy robust to several different parameter sets.

REFERENCES

[2.1] L. Qinghua, "Analysis, design and control of permanent magnet synchronous motors for wide-speed operation," Doctor of Philosophy, Department of Electrical Engineering, National University of Singapore, ScholarBank@NUS Repository, 2005.

[2.2] S. Meier, "Theoretical design of surface-mounted permanent magnet motors with field-weakening capability," Master, Department of Electrical Engineering Electrical Machines and Power Electronics, Royal Institute of Technology Stockholm, Emetor, 2002.

[2.3] R. G. Shriwastava, M. Diagavane, and S. Vaishnav, "Literature Review of Permanent Magnet AC Motors and Drive for Automotive Application," *Bulletin of Electrical Engineering and Informatics,* vol. 1, no. 1, pp. 7-14, 2011.

[2.4] D. Vindel Muñoz, "Design, Simulation and implementation of a PMSM drive system," Master of Science Thesis in Electric Power Engineering, Chalmers University of Technology, Göteborg, 2011.

[2.5] S. Chi, "Position-sensorless control of permanent magnet synchronous machines over wide speed range," The Ohio State University, 2007.

[2.6] M. A. Rahman and R. Qin, "A permanent magnet hysteresis hybrid synchronous motor for electric vehicles," *IEEE Transactions on Industrial Electronics,* vol. 44, no. 1, pp. 46-53, 1997.

[2.7] W.-H. Kim, J.-N. Bae, I.-S. Jang, and J. Lee, "A study on 4-layer hybrid winding layout of the IPMSM and location of the permanent magnets," in *Digests of the 2010 14th Biennial IEEE Conference on Electromagnetic Field Computation*, IEEE, pp. 1-1, 2010.

[2.8] P. Pišek, B. Štumberger, T. Marič, and P. Virtič, "Design and FE analysis of a double rotor synchronous PM machine," in *Digests of the 2010 14th Biennial IEEE Conference on Electromagnetic Field Computation*, IEEE, pp. 1-1,2010.

[2.9] S.-M. Liu, C.-H. Tu, C.-L. Lin, and V.-T. Liu, "Field-Oriented Driving/Braking Control for Electric Vehicles," *Electronics,* vol. 9, no. 9, 2020, doi: 10.3390/electronics9091484.

[2.10] Y. Liao, F. Liang, and T. A. Lipo, "A novel permanent magnet motor with doubly salient structure," *IEEE transactions on industry applications,* vol. 31, no. 5, pp. 1069-1078, 1995.

[2.11] K. Chau, J. Jiang, and Y. Wang, "A novel stator doubly fed doubly salient permanent magnet brushless machine," *IEEE Transactions on Magnetics,* vol. 39, no. 5, pp. 3001-3003, 2003.

[2.12] S. Rauch and L. Johnson, "Design principles of flux-switch alternators [includes discussion]," *Transactions of the American Institute of Electrical Engineers. Part III: Power Apparatus and Systems,* vol. 74, no. 3, pp. 1261-1268, 1955.

[2.13] E. Hoang, M. Lecrivain, and M. Gabsi, "A new structure of a switching flux synchronous polyphased machine with hybrid excitation," in *2007 European Conference on Power Electronics and Applications*, IEEE, pp. 1-8,2007.

[2.14] J.-W. Kwon, J.-h. Lee, W. Zhao, and B.-I. Kwon, "Flux-Switching Permanent Magnet Machine with Phase-Group Concentrated-Coil Windings and Cogging Torque Reduction Technique," *Energies,* vol. 11, no. 10, 2018, doi: 10.3390/en11102758.

[2.15] M. Štulrajter, V. Hrabovcova, and M. Franko, "Permanent magnets synchronous motor control theory," *Journal of electrical engineering,* vol. 58, no. 2, pp. 79-84, 2007.

[2.16] G.-D. Andreescu, C.-E. Coman, A. Moldovan, and I. Boldea, "Stable V/f control system with unity power factor for PMSM drives," in *2012 13th International Conference on Optimization of Electrical and Electronic Equipment (OPTIM)*, IEEE, pp. 432-438,2012.

[2.17] R. K. Sharma, V. Sanadhya, L. Behera, and S. Bhattacharya, "Vector control of a permanent magnet synchronous motor," in *2008 Annual IEEE India Conference*, vol. 1: IEEE, pp. 81-86 2008.

[2.18] M. Merzoug and F. Naceri, "Comparison of field-oriented control and direct torque control for permanent magnet synchronous motor (PMSM)," *World Academy of Science, Engineering and Technology,* vol. 45, pp. 299-304, 2008.

[2.19] F. Blaschke, "Apparatus for field-oriented control or regulation of asynchronous machines," ed: Google Patents, 1974.

[2.20] K. Gulez, A. A. Adam, and H. Pastaci, "Torque ripple and EMI noise minimization in PMSM using active filter topology and field-oriented control," *IEEE Transactions on Industrial Electronics,* vol. 55, no. 1, pp. 251-257, 2008.

[2.21] X. Wang, N. Liu, and R. Na, "Simulation of PMSM field-oriented control based on SVPWM," in *2009 IEEE Vehicle Power and Propulsion Conference*, IEEE, pp. 1465-1469,2009.

[2.22] M. Depenbrock, "Direct self-control (DSC) of inverter fed induktion machine," in *1987 IEEE Power Electronics Specialists Conference*, IEEE, pp. 632-641,1987.

[2.23] I. Takahashi and T. Noguchi, "A new quick-response and high-efficiency control strategy of an induction motor," *IEEE Transactions on Industry applications,* no. 5, pp. 820-827, 1986.

[2.24] Z. Wang, J. Chen, M. Cheng, and K. Chau, "Field-oriented control and direct torque control for paralleled VSIs fed PMSM drives with variable switching frequencies," *IEEE Transactions on Power Electronics,* vol. 31, no. 3, pp. 2417-2428, 2015.

[2.25] Z. Ma, S. Saeidi, and R. Kennel, "FPGA implementation of model predictive control with constant switching frequency for PMSM drives," *IEEE Transactions on Industrial Informatics,* vol. 10, no. 4, pp. 2055-2063, 2014.

[2.26] Z. Mynar, L. Vesely, and P. Vaclavek, "PMSM model predictive control with field-weakening implementation," *IEEE Transactions on Industrial Electronics,* vol. 63, no. 8, pp. 5156-5166, 2016.

[2.27] T. Englert, S. Grüner, and K. Graichen, "Model predictive torque control of permanent magnet synchronous machines," *IFAC-PapersOnLine,* vol. 50, no. 1, pp. 758-763, 2017.

[2.28] W. Wang, Z. Lu, W. Hua, Z. Wang, and M. Cheng, "Simplified model predictive current control of primary permanent-magnet linear motor traction systems for subway applications," *Energies,* vol. 12, no. 21, p. 4144, 2019.

[2.29] M. Preindl and S. Bolognani, "Model Predictive Direct Speed Control with Finite Control Set of PMSM Drive Systems," *IEEE Transactions on Power Electronics,* vol. 28, no. 2, pp. 1007-1015, 2013.

[2.30] S. Chai, L. Wang, and E. Rogers, "A Cascade MPC Control Structure for a PMSM With Speed Ripple Minimization," *IEEE Transactions on Industrial Electronics,* vol. 60, no. 8, pp. 2978-2987, 2013.

[2.31] J. Rodriguez and P. Cortes, *Predictive control of power converters and electrical drives*. John Wiley & Sons, 2012.

[2.32] A. A. Ahmed, B. K. Koh, and Y. I. Lee, "A Comparison of Finite Control Set and Continuous Control Set Model Predictive Control Schemes for Speed Control of Induction Motors," *IEEE Transactions on Industrial Informatics,* vol. 14, no. 4, pp. 1334-1346, 2018, doi: 10.1109/TII.2017.2758393.

[2.33] M. Preindl and S. Bolognani, "Comparison of direct and PWM model predictive control for power electronic and drive systems," in *2013 Twenty-Eighth Annual IEEE Applied Power Electronics Conference and Exposition (APEC)*, IEEE, pp. 2526-2533,2013.

[2.34] S. Bolognani, S. Bolognani, L. Peretti, and M. Zigliotto, "Design and Implementation of Model Predictive Control for Electrical Motor Drives," *IEEE Transactions on Industrial Electronics,* vol. 56, no. 6, pp. 1925-1936, 2009.

[2.35] R. Errouissi, A. Al-Durra, S. M. Muyeen, and S. Leng, "Continuous-time model predictive control of a permanent magnet synchronous motor drive with disturbance decoupling," *IET Electric Power Applications,* vol. 11, no. 5, pp. 697-706, 2017.

[2.36] F. Wang, L. He, and J. Rodriguez, "FPGA based Continuous Control Set Model Predictive Current Control for PMSM System Using Multi-step Error Tracking Technique," *IEEE Transactions on Power Electronics,* pp. 1-1, 2020.

[2.37] M. Kashif, S. Murshid, and B. Singh, "Continuous Control Set Model Predictive Controller for PMSM Driven Solar PV Water Pumping System," in *2019 IEEE International Conference on Environment and Electrical Engineering and 2019 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe)*, IEEE, pp. 1-6, 2019.

[2.38] F. Toso, P. G. Carlet, A. Favato, and S. Bolognani, "On-line Continuous Control Set MPC for PMSM drives current loops at high sampling rate using qpOASES," in *2019 IEEE Energy Conversion Congress and Exposition (ECCE)*: IEEE, pp. 6615-6620,2019.

[2.39] T. Geyer, G. Papafotiou, and M. Morari, "Model Predictive Direct Torque Control—Part I: Concept, Algorithm, and Analysis," *IEEE Transactions on Industrial Electronics,* vol. 56, no. 6, pp. 1894-1905, 2009.

[2.40] G. Papafotiou, J. Kley, K. G. Papadopoulos, P. Bohren, and M. Morari, "Model Predictive Direct Torque Control—Part II: Implementation and Experimental Evaluation," *IEEE Transactions on Industrial Electronics,* vol. 56, no. 6, pp. 1906-1915, 2009.

[2.41] S. Hanke, S. Peitz, O. Wallscheid, J. Böcker, and M. Dellnitz, "Finite-Control-Set Model Predictive Control for a Permanent Magnet Synchronous Motor Application with Online Least Squares System Identification," in *2019 IEEE International Symposium on Predictive Control of Electrical Drives and Power Electronics (PRECEDE)*, pp. 1-6, 31 May-2 June 2019.

[2.42] J. Rodriguez *et al.*, "State of the art of finite control set model predictive control in power electronics," *IEEE Transactions on Industrial Informatics,* vol. 9, no. 2, pp. 1003-1016, 2012.

[2.43] Y. Zhang and H. Yang, "Model Predictive Torque Control of Induction Motor Drives With Optimal Duty Cycle Control," *IEEE Transactions on Power Electronics,* vol. 29, no. 12, pp. 6593-6603, 2014.

[2.44] J. Rodriguez *et al.*, "State of the Art of Finite Control Set Model Predictive Control in Power Electronics," *IEEE Transactions on Industrial Informatics,* vol. 9, no. 2, pp. 1003-1016, 2013.

[2.45] A. Formentini, A. Trentin, M. Marchesoni, P. Zanchetta, and P. Wheeler, "Speed Finite Control Set Model Predictive Control of a PMSM Fed by Matrix Converter," *IEEE Transactions on Industrial Electronics,* vol. 62, no. 11, pp. 6786-6796, 2015.

[2.46] W. Xie *et al.*, "Finite-control-set model predictive torque control with a deadbeat solution for PMSM drives," *IEEE Transactions on Industrial Electronics,* vol. 62, no. 9, pp. 5402-5410, 2015.

[2.47] A. Mora, O. Á, J. Juliet, and R. Cárdenas, "Model Predictive Torque Control for Torque Ripple Compensation in Variable-Speed PMSMs," *IEEE Transactions on Industrial Electronics,* vol. 63, no. 7, pp. 4584-4592, 2016.

[2.48] Z. Zhou, C. Xia, Y. Yan, Z. Wang, and T. Shi, "Torque Ripple Minimization of Predictive Torque Control for PMSM With Extended Control Set," *IEEE Transactions on Industrial Electronics,* vol. 64, no. 9, pp. 6930-6939, 2017.

[2.49] W. Xie, X. Wang, F. Wang, W. Xu, R. Kennel, and D. Gerling, "Dynamic Loss Minimization of Finite Control Set-Model Predictive Torque Control for Electric Drive System," *IEEE Transactions on Power Electronics,* vol. 31, no. 1, pp. 849-860, 2016.

[2.50] X. Wu, W. Song, and C. Xue, "Low-Complexity Model Predictive Torque Control Method Without Weighting Factor for Five-Phase PMSM Based on Hysteresis Comparators," *IEEE Journal of Emerging and Selected Topics in Power Electronics,* vol. 6, no. 4, pp. 1650-1661, 2018.

[2.51] T. Wang, C. Liu, G. Lei, Y. Guo, and J. Zhu, "Model predictive direct torque control of permanent magnet synchronous motors with extended set of voltage space vectors," *IET Electric Power Applications,* vol. 11, no. 8, pp. 1376-1382, 2017.

[2.52] F. Ban, G. Lian, J. Zhang, B. Chen, and G. Gu, "Study on a Novel Predictive Torque Control Strategy Based on the Finite Control Set for PMSM," *IEEE Transactions on Applied Superconductivity,* vol. 29, no. 2, pp. 1-6, 2019.

[2.53] Y. Zhou, H. Li, R. Liu, and J. Mao, "Continuous Voltage Vector Model-Free Predictive Current Control of Surface Mounted Permanent Magnet Synchronous Motor," *IEEE Transactions on Energy Conversion,* vol. 34, no. 2, pp. 899-908, 2019.

[2.54] W. Zhao, T. Tao, J. Zhu, H. Tan, and Y. Du, "A Novel Finite-Control-Set Model Predictive Current Control for Five-Phase PM Motor With Continued Modulation," *IEEE Transactions on Power Electronics,* vol. 35, no. 7, pp. 7261-7270, 2020.

[2.55] C. Xiong, H. Xu, T. Guan, and P. Zhou, "A Constant Switching Frequency Multiple-Vector-Based Model Predictive Current Control of Five-Phase PMSM With Nonsinusoidal Back EMF," *IEEE Transactions on Industrial Electronics,* vol. 67, no. 3, pp. 1695-1707, 2020.

[2.56] W. Wang, Y. Fan, S. Chen, and Q. Zhang, "Finite control set model predictive current control of a five-phase PMSM with virtual voltage vectors and adaptive control set," *CES Transactions on Electrical Machines and Systems,* vol. 2, no. 1, pp. 136-141, 2018.

[2.57] A. Brosch, S. Hanke, O. Wallscheid, and J. Boecker, "Data-Driven Recursive Least Squares Estimation for Model Predictive Current Control of Permanent Magnet Synchronous Motors," *IEEE Transactions on Power Electronics,* pp. 1-1, 2020.

[2.58] A. Abbaszadeh, D. A. Khaburi, R. Kennel, and J. Rodríguez, "Hybrid exploration state for the simplified finite control set-model predictive control with a deadbeat solution for reducing the current ripple in permanent magnet synchronous motor," *IET Electric Power Applications,* vol. 11, no. 5, pp. 823-835, 2017.

[2.59] H. Li, S. Chen, X. Wu, and G. Tan, "Model predictive control method with constant switching frequency to reduce common-mode voltage for pmsm drives," *Journal of Electrical and Computer Engineering,* vol. 2018, 2018.

[2.60] X. Wang, X. Fang, S. Lin, F. Lin, and Z. Yang, "Predictive Common-Mode Voltage Suppression Method Based on Current Ripple for Permanent Magnet Synchronous Motors," *IEEE Journal of Emerging and Selected Topics in Power Electronics,* vol. 7, no. 2, pp. 946-955, 2019.

[2.61] L. Sheng, D. Li, and Y. Ji, "Two-Vector FCS-MPC for Permanent-Magnet Synchronous Motors Based on Duty Ratio Optimization," *Mathematical Problems in Engineering,* vol. 2018, p. 9061979, 2018/05/31 2018, doi: 10.1155/2018/9061979.

[2.62] H. Lin and W. Song, "Three-Vector Model Predictive Current Control of Permanent Magnet Synchronous Motor Based on SVM," in *IEEE International Symposium on Predictive Control of Electrical Drives and Power Electronics (PRECEDE)*, 2019.

[2.63] P. Cortes, J. Rodriguez, C. Silva, and A. Flores, "Delay compensation in model predictive current control of a three-phase inverter," *IEEE Transactions on Industrial Electronics,* vol. 59, no. 2, pp. 1323-1325, 2011.

[2.64] M. Siami, D. A. Khaburi, and J. Rodriguez, "Simplified Finite Control Set-Model Predictive Control for Matrix Converter-Fed PMSM Drives," *IEEE Transactions on Power Electronics,* vol. 33, no. 3, pp. 2438-2446, 2018.

[2.65] X. Zhang and B. Hou, "Double Vectors Model Predictive Torque Control Without Weighting Factor Based on Voltage Tracking Error," *IEEE Transactions on Power Electronics,* vol. 33, no. 3, pp. 2368-2380, 2018.

[2.66] X. Zhang, B. Hou, Y. He, and D. Gao, "Model predictive torque control of surface mounted permanent magnet synchronous motor drives with voltage cost functions," *Journal of Power Electronics,* vol. 18, no. 5, pp. 1369-1379, 2018.

[2.67] J. Rodriguez *et al.*, "Predictive Current Control of a Voltage Source Inverter," *IEEE Transactions on Industrial Electronics,* vol. 54, no. 1, pp. 495-503, 2007.

[2.68] J.-F. Stumper, A. Dötlinger, J. Jung, and R. Kennel, "Predictive control of a permanent magnet synchronous machine based on real-time dynamic optimization," in *Proceedings of the 2011 14th European Conference on Power Electronics and Applications*, IEEE, pp. 1-8, 2011.

[2.69] P. Kakosimos, M. Beniakar, Y. Liu, and H. Abu-Rub, "Model predictive control for permanent magnet synchronous motor drives considering cross-saturation effects," in *2017 IEEE Applied Power Electronics Conference and Exposition (APEC)*, IEEE, pp. 1880-1885,2017.

[2.70] Y. Han, C. Gong, L. Yan, H. Wen, Y. Wang, and K. Shen, "Multiobjective Finite Control Set Model Predictive Control Using Novel Delay Compensation Technique for PMSM," *IEEE Transactions on Power Electronics,* vol. 35, no. 10, pp. 11193-11204, 2020.

[2.71] J. Gao, C. Gong, W. Li, and J. Liu, "Novel Compensation Strategy for Calculation Delay of Finite Control Set Model Predictive Current Control in PMSM," *IEEE Transactions on Industrial Electronics,* vol. 67, no. 7, pp. 5816-5819, 2020.

[2.72] C. Ma, X. Yao, H. Li, and F. De Belie, "An Improved Two-Vector Model Predictive Torque Control based on RMS Duty Ratio Optimization for PMSM," in *2019 IEEE International Electric Machines & Drives Conference (IEMDC)*, IEEE, pp. 1674-1679, 2019.

[2.73] Y. Zhang, L. Huang, D. Xu, J. Liu, and J. Jin, "Performance evaluation of two-vector-based model predictive current control of PMSM drives," *Chinese Journal of Electrical Engineering,* vol. 4, no. 2, pp. 65-81, 2018.

[2.74] Y. Zhang, D. Xu, and L. Huang, "Generalized multiple-vector-based model predictive control for PMSM drives," *IEEE Transactions on Industrial Electronics,* vol. 65, no. 12, pp. 9356-9366, 2018.

[2.75] J. Lin, G. Xie, and X. Shi, "Improved Double-Vector Model Predictive Current Control of Permanent Magnet Synchronous Linear Motor," in *Proceedings of the 2019 3rd International Conference on Innovation in Artificial Intelligence*, pp. 150-156, 2019.

[2.76] Y. Xu, T. Shi, Y. Yan, and X. Gu, "Dual-Vector predictive torque control of permanent magnet synchronous motors based on a candidate vector table," *Energies,* vol. 12, no. 1, p. 163, 2019.

[2.77] Y. Zhang and H. Yang, "Two-vector-based model predictive torque control without weighting factors for induction motor drives," *IEEE Transactions on Power Electronics,* vol. 31, no. 2, pp. 1381-1390, 2015.

[2.78] Y. Zhang and H. Yang, "Generalized two-vector-based model-predictive torque control of induction motor drives," *IEEE Transactions on Power Electronics,* vol. 30, no. 7, pp. 3818-3829, 2014.

[2.79] I. Jlassi and A. J. M. Cardoso, "Lookup-Table-Based Model Predictive Torque Control Without Weighting Factors for PMSM Drives," in *IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society*, vol. 1, pp. 1165-1170, 14-17 Oct. 2019.

[2.80] T. Liu and Y. Cai, "Improved Weighting Factor Selection Method of Predictive Torque Control for High Speed Surface-mounted PMSM based on RBFNN," in *2019 22nd International Conference on Electrical Machines and Systems (ICEMS)*, 11-14 Aug. 2019.

[2.81] P. Cortes *et al.*, "Guidelines for weighting factors design in Model Predictive Control of power converters and drives," in *2009 IEEE International Conference on Industrial Technology*, 10-13 Feb. 2009.

[2.82] M. Mamdouh, M. A. Abido, and Z. Hamouz, "Weighting Factor Selection Techniques for Predictive Torque Control of Induction Motor Drives: A Comparison Study," *Arabian Journal for Science and Engineering,* vol. 43, no. 2, pp. 433-445, 2018/02/01 2018, doi: 10.1007/s13369-017-2842-2.

[2.83] T. Dragičević and M. Novak, "Weighting Factor Design in Model Predictive Control of Power Electronic Converters: An Artificial Neural Network Approach," *IEEE Transactions on Industrial Electronics,* vol. 66, no. 11, pp. 8870-8880, 2019.

[2.84] H. Chen and H. Zhao, "Review on pulse-width modulation strategies for common-mode voltage reduction in three-phase voltage-source inverters," *IET Power Electronics,* vol. 9, no. 14, pp. 2611-2620, 2016.

[2.85] M. Asefi and J. Nazarzadeh, "Survey on high-frequency models of PWM electric drives for shaft voltage and bearing current analysis," *IET Electrical Systems in Transportation,* vol. 7, no. 3, pp. 179-189, 2017.

[2.86] N. Mousavi, T. Rahimi, and H. M. Kelk, "Reduction EMI of BLDC motor drive based on software analysis," *Advances in Materials Science and Engineering,* vol. 2016, 2016.

[2.87] M. Preindl, E. Schaltz, and P. Thogersen, "Switching frequency reduction using model predictive direct current control for high-power voltage source inverters," *IEEE Transactions on Industrial Electronics,* vol. 58, no. 7, pp. 2826-2835, 2010.

[2.88] X. WANG, X. FANG, F. LIN, and Z. YANG, "Common Mode Voltage Suppression Method Based on Model Predictive Control For A Permanent Magnet Synchronous Motor Considering Current Error Limit," *Computers in Railways XVI: Railway Engineering Design and Operation,* vol. 181, p. 111, 2018.

[2.89] F. Niu *et al.*, "Common mode current suppression for permanent magnet synchronous motor based on model predictive control," in *2018 Thirteenth International Conference on Ecological Vehicles and Renewable Energies (EVER)*, IEEE, pp. 1-6, 2018.

[2.90] C. Xiong, H. Xu, C. Fang, and H. Zhang, "Model predictive control method to reduce common-mode voltage for permanent-magnet synchronous machine drives," in *2017 IEEE Transportation Electrification Conference and Expo, Asia-Pacific (ITEC Asia-Pacific)*, IEEE, pp. 1-6.,2017.

[2.91] S. Kwak and S.-k. Mun, "Model predictive control methods to reduce common-mode voltage for three-phase voltage source inverters," *IEEE Transactions on Power Electronics,* vol. 30, no. 9, pp. 5019-5035, 2014.

[2.92] W.-s. Jung, K.-m. Choo, J.-c. Kim, W.-j. Kim, and C.-y. Won, "H7 Inverter Using Zener Diode With Model Predictive Current Control for Common-Mode Voltage Reduction in PMSM Drive System," in *2018 IEEE International Power Electronics and Application Conference and Exposition (PEAC)*, IEEE, pp. 1-6,2018.

[2.93] W.-H. Chen, J. Yang, L. Guo, and S. Li, "Disturbance-Observer-Based Control and Related Methods—An Overview," *IEEE Transactions on Industrial Electronics,* vol. 63, no. 2, pp. 1083-1095, 2016, doi: 10.1109/tie.2015.2478397.

[2.94] J. Ren, Y. Ye, G. Xu, Q. Zhao, and M. Zhu, "Uncertainty-and-disturbance-estimator-based current control scheme for PMSM drives with a simple parameter tuning algorithm," *IEEE Transactions on Power Electronics,* vol. 32, no. 7, pp. 5712-5722, 2016.

[2.95] T. M. Jahns and W. L. Soong, "Pulsating torque minimization techniques for permanent magnet AC motor drives-a review," *IEEE Transactions on industrial electronics,* vol. 43, no. 2, pp. 321-330, 1996.

[2.96] T. Sebastian, "Temperature effects on torque production and efficiency of PM motors using NdFeB magnets," *IEEE Transactions on Industry Applications,* vol. 31, no. 2, pp. 353-357, 1995.

[2.97] E. Richter and T. Neumann, "Line start permanent magnet motors with different materials," *IEEE Transactions on Magnetics,* vol. 20, no. 5, pp. 1762-1764, 1984.

[2.98] T. Z. Htet, Z. Zhao, and K. Li, "PM material analysis of permanent magnet synchronous generator in wind turbines," in *2017 2nd International Conference on System Reliability and Safety (ICSRS)*, IEEE, pp. 332-336, 2017.

[2.99] A. D. Gerlando, G. M. Foglia, M. F. Iacchetti, and R. Perini, "Effects of Manufacturing Imperfections in Concentrated Coil Axial Flux PM Machines: Evaluation and Tests," *IEEE Transactions on Industrial Electronics,* vol. 61, no. 9, pp. 5012-5024, 2014, doi: 10.1109/TIE.2013.2274427.

[2.100] A. Meyer, J. v. Lindenfels, A. Mayr, and J. Franke, "Manufacturing Imperfections in Electric Motor Production with Focus on Halbach Array Permanent Magnet Rotor Assembly," in *2018 8th International Electric Drives Production Conference (EDPC)*, pp. 1-7, 4-5 Dec. 2018, doi: 10.1109/EDPC.2018.8658282.

[2.101] H. Olsson, K. J. Åström, C. Canudas de Wit, M. Gäfvert, and P. Lischinsky, "Friction Models and Friction Compensation," *European Journal of Control*, vol. 4, no. 3, pp. 176-195, 1998/01/01/ 1998, doi: https://doi.org/10.1016/S0947-3580(98)70113-X

[2.102] S. Bognatz, "Alignment of critical and non critical machines," *Orbit,* vol. 4, pp. 23-25, 1995.

[2.103] S. Li and Z. Liu, "Adaptive speed control for permanent-magnet synchronous motor system with variations of load inertia," *IEEE Transactions on industrial electronics,* vol. 56, no. 8, pp. 3050-3059, 2009.

[2.104] M. Yang, X. Lang, J. Long, and D. Xu, "Flux Immunity Robust Predictive Current Control With Incremental Model and Extended State Observer for PMSM Drive," *IEEE Transactions on Power Electronics,* vol. 32, no. 12, pp. 9267-9279, 2017.

[2.105] D. Leggate and R. J. Kerkman, "Pulse based dead time compensator for PWM voltage inverters," in *Proceedings of IECON'95-21st Annual Conference on IEEE Industrial Electronics*, vol. 1: IEEE, pp. 474-481, 1995.

[2.106] Q. Fei, Y. Deng, H. Li, J. Liu, and M. Shao, "Speed ripple minimization of permanent magnet synchronous motor based on model predictive and iterative learning controls," *IEEE Access,* vol. 7, pp. 31791-31800, 2019.

[2.107] S. N. Vukosavic and M. R. Stojic, "Suppression of torsional oscillations in a high-performance speed servo drive," *IEEE Transactions on Industrial Electronics,* vol. 45, no. 1, pp. 108-117, 1998.

[2.108] J. Ren, Y. Ye, G. Xu, Q. Zhao, and M. Zhu, "Uncertainty-and-Disturbance-Estimator-Based Current Control Scheme for PMSM Drives With a Simple Parameter Tuning Algorithm," *IEEE Transactions on Power Electronics,* vol. 32, no. 7, pp. 5712-5722, 2017, doi: 10.1109/tpel.2016.2607228.

[2.109] J. Yang, W.-H. Chen, S. Li, L. Guo, and Y. Yan, "Disturbance/Uncertainty Estimation and Attenuation Techniques in PMSM Drives—A Survey," *IEEE Transactions on Industrial Electronics,* vol. 64, no. 4, pp. 3273-3285, 2017, doi: 10.1109/tie.2016.2583412.

[2.110] L. Zhu, S. Jiang, Z. Zhu, and C. Chan, "Analytical methods for minimizing cogging torque in permanent-magnet machines," *IEEE Transactions on magnetics,* vol. 45, no. 4, pp. 2023-2031, 2009.

[2.111] L. Dosiek and P. Pillay, "Cogging torque reduction in permanent magnet machines," *IEEE Transactions on industry applications,* vol. 43, no. 6, pp. 1565-1571, 2007.

[2.112] R. Islam and I. Husain, "Analytical model for predicting noise and vibration in permanent-magnet synchronous motors," *IEEE Transactions on industry applications,* vol. 46, no. 6, pp. 2346-2354, 2010.

[2.113] N. Bianchi and S. Bolognani, "Design techniques for reducing the cogging torque in surface-mounted PM motors," *IEEE Transactions on industry applications,* vol. 38, no. 5, pp. 1259-1265, 2002.

[2.114] A. Jabbari, M. Shakeri, and A. N. Niaki, "Iron pole shape optimization of IPM Motors using an Integrated Method," *Advances in Electrical and Computer Engineering Journal,* vol. 10, no. 1, 2010.

[2.115] T. Tudorache and M. Popescu, "Optimal design solutions for permanent magnet synchronous machines," *Advances in Electrical and Computer Engineering,* vol. 11, no. 4, pp. 77-82, 2011.

[2.116] T. Tudorache, L. Melcescu, and M. Popescu, "Methods for cogging torque reduction of directly driven PM wind generators," in *2010 12th International Conference on Optimization of Electrical and Electronic Equipment*, IEEE, pp. 1161-1166, 2010.

[2.117] D. Wang, X. Wang, Y. Yang, and R. Zhang, "Optimization of magnetic pole shifting to reduce cogging torque in solid-rotor permanent-magnet synchronous motors," *IEEE Transactions on magnetics,* vol. 46, no. 5, pp. 1228-1234, 2010.

[2.118] D. Wang, X. Wang, D. Qiao, Y. Pei, and S.-Y. Jung, "Reducing cogging torque in surface-mounted permanent-magnet motors by nonuniformly distributed teeth method," *IEEE Transactions on Magnetics,* vol. 47, no. 9, pp. 2231-2239, 2011.

[2.119] M. Ashabani and Y. A.-R. I. Mohamed, "Multiobjective shape optimization of segmented pole permanent-magnet synchronous machines with improved torque characteristics," *IEEE Transactions on Magnetics,* vol. 47, no. 4, pp. 795-804, 2011.

[2.120] K.-J. Tseng and S.-B. Wee, "Analysis of flux distribution and core losses in interior permanent magnet motor," *IEEE Transactions on Energy Conversion,* vol. 14, no. 4, pp. 969-975, 1999.

[2.121] X. Xiao and C. Chen, "Reduction of torque ripple due to demagnetization in PMSM using current compensation," *IEEE Transactions on applied superconductivity,* vol. 20, no. 3, pp. 1068-1071, 2010.

[2.122] G.-H. Lee, S.-I. Kim, J.-P. Hong, and J.-H. Bahn, "Torque ripple reduction of interior permanent magnet synchronous motor using harmonic injected current," *IEEE Transactions on Magnetics,* vol. 44, no. 6, pp. 1582-1585, 2008.

[2.123] S.-H. Hwang and J.-M. Kim, "Dead time compensation method for voltage-fed PWM inverter," *IEEE Transactions on energy conversion,* vol. 25, no. 1, pp. 1-10, 2009.

[2.124] S.-Y. Kim, W. Lee, M.-S. Rho, and S.-Y. Park, "Effective dead-time compensation using a simple vectorial disturbance estimator in PMSM drives," *IEEE Transactions on Industrial Electronics,* vol. 57, no. 5, pp. 1609-1614, 2009.

[2.125] J. Shi and S. Li, "Analysis and compensation control of dead-time effect on space vector PWM," *Journal of Power Electronics,* vol. 15, no. 2, pp. 431-442, 2015.

[2.126] D.-M. Park and K.-H. Kim, "Parameter-independent online compensation scheme for dead time and inverter nonlinearity in IPMSM drive through waveform analysis," *IEEE Transactions on Industrial Electronics,* vol. 61, no. 2, pp. 701-707, 2013.

[2.127] Y. Zhao, W. Qiao, and L. Wu, "Dead-time effect analysis and compensation for a sliding-mode position observer-based sensorless IPMSM control system," *IEEE Transactions on Industry Applications,* vol. 51, no. 3, pp. 2528-2535, 2014.

[2.128] Z. Zhou, C. Xia, Y. Yan, Z. Wang, and T. Shi, "Disturbances attenuation of permanent magnet synchronous motor drives using cascaded predictive-integral-resonant controllers," *IEEE Transactions on Power Electronics,* vol. 33, no. 2, pp. 1514-1527, 2017.

[2.129] Y. Liu, S. Cheng, B. Ning, and Y. Li, "Robust model predictive control with simplified repetitive control for electrical machine drives," *IEEE Transactions on Power Electronics,* vol. 34, no. 5, pp. 4524-4535, 2018.

[2.130] D.-W. Chung and S.-K. Sul, "Analysis and compensation of current measurement error in vector-controlled AC motor drives," *IEEE Transactions on industry applications,* vol. 34, no. 2, pp. 340-345, 1998.

[2.131] W. Qian, S. Panda, and J.-X. Xu, "Speed ripple minimization in PM synchronous motor using iterative learning control," *IEEE Transactions on Energy Conversion,* vol. 20, no. 1, pp. 53-61, 2005.

[2.132] A. Zentai and T. Daboczi, "Offline parameter estimation of permanent magnet synchronous machines by means of LS optimization," in *2008 IEEE/SICE International Symposium on System Integration*, IEEE, pp. 36-41,2008.

[2.133] I. Vesely, L. Vesely, and Z. Bradac, "MRAS identification of permanent magnet synchronous motor parameters," *IFAC-PapersOnLine,* vol. 51, no. 6, pp. 250-255, 2018.

[2.134] K. Liu, Q. Zhang, Z.-Q. Zhu, J. Zhang, A.-W. Shen, and P. Stewart, "Comparison of two novel MRAS based strategies for identifying parameters in permanent magnet synchronous motors," *International Journal of automation and computing,* vol. 7, no. 4, pp. 516-524, 2010.

[2.135] Y. A.-R. I. Mohamed and E. F. El-Saadany, "A current control scheme with an adaptive internal model for torque ripple minimization and robust current regulation in PMSM drive systems," *IEEE Transactions on Energy Conversion,* vol. 23, no. 1, pp. 92-100, 2008.

[2.136] X. Zhang, B. Hou, and Y. Mei, "Deadbeat predictive current control of permanent-magnet synchronous motors with stator current and disturbance observer," *IEEE Transactions on Power Electronics,* vol. 32, no. 5, pp. 3818-3834, 2016.

[2.137] R. Yang, M.-Y. Wang, L.-Y. Li, C.-M. Zhang, and J.-L. Jiang, "Robust predictive current control with variable-gain adaptive disturbance observer for PMLSM," *IEEE Access,* vol. 6, pp. 13158-13169, 2018.

[2.138] Z.-H. Liu, H.-L. Wei, X.-H. Li, K. Liu, and Q.-C. Zhong, "Global identification of electrical and mechanical parameters in PMSM drive based on dynamic self-learning PSO," *IEEE Transactions on Power Electronics,* vol. 33, no. 12, pp. 10858-10871, 2018.

[2.139] K. Liu and Z. Zhu, "Fast determination of moment of inertia of permanent magnet synchronous machine drives for design of speed loop regulator," *IEEE Transactions on Control Systems Technology,* vol. 25, no. 5, pp. 1816-1824, 2016.

[2.140] R. Garrido and A. Concha, "Inertia and friction estimation of a velocity-controlled servo using position measurements," *IEEE Transactions on industrial electronics,* vol. 61, no. 9, pp. 4759-4770, 2013.

[2.141] S. Li and H. Gu, "Fuzzy adaptive internal model control schemes for PMSM speed-regulation system," *IEEE Transactions on Industrial Informatics,* vol. 8, no. 4, pp. 767-779, 2012.

[2.142] H. Ozbay, *Introduction to feedback control theory*. CrC Press, Boca Raton, Florida ,USA, 1999.

[2.143] M. Fu and L. Xie, "The sector bound approach to quantized feedback control," *IEEE Transactions on Automatic control,* vol. 50, no. 11, pp. 1698-1711, 2005.

[2.144] S. Engell, "Feedback control for optimal process operation," *Journal of process control,* vol. 17, no. 3, pp. 203-219, 2007.

[2.145] G. E. Stewart, D. M. Gorinevsky, and G. A. Dumont, "Feedback controller design for a spatially distributed system: The paper machine problem," *IEEE Transactions on Control Systems Technology,* vol. 11, no. 5, pp. 612-628, 2003.

[2.146] M. S. Mahmoud and Y. Chen, "Design of feedback controllers by two-stage methods," *Applied Mathematical Modelling,* vol. 7, no. 3, pp. 163-168, 1983.

[2.147] G. C. Newton, "Analytical Design of Linear Feedback Controls," pp. 336-381, 1957 1957. [Online]. Available: https://ci.nii.ac.jp/naid/10003233380/en/.

[2.148] N. E. Leonard and J. G. Graver, "Model-based feedback control of autonomous underwater gliders," *IEEE Journal of oceanic engineering,* vol. 26, no. 4, pp. 633-645, 2001.

[2.149] A. Weinmann, *Uncertain models and robust control*. Springer Science & Business Media, 2012.

[2.150] R. S. Patwardhan, S. L. Shah, and B. Huang, "The Effect of Uncertainty on Controller Performance," *IFAC Proceedings Volumes,* vol. 32, no. 2, pp. 6962-6967, 1999/07/01/ 1999, doi: https://doi.org/10.1016/S1474-6670(17)57188-8.

[2.151] M. Morari and E. Zafiriou, *Robust process control*. Englewood Cliffs, N.J: Prentice Hall, p. 488, 1989.

[2.152] Z. Qu, *Robust control of nonlinear uncertain systems*. John Wiley & Sons, Inc., 1998.

[2.153] C. Scherer, "Theory of robust control," *Delft University of Technology,* pp. 1-160, 2001.

[2.154] B. Wie and D. S. Bernstein, "Benchmark problems for robust control design," *Journal of Guidance, Control, and Dynamics,* vol. 15, no. 5, pp. 1057-1059, 1992.

[2.155] T. T. Georgiou, "Differential stability and robust control of nonlinear systems," *Mathematics of Control, Signals and Systems,* vol. 6, no. 4, pp. 289-306, 1993/12/01 1993, doi: 10.1007/BF01211498.

[2.156] Y. Li and E. B. Lee, "Stability robustness characterization and related issues for control systems design," *Automatica,* vol. 29, no. 2, pp. 479-484, 1993/03/01/ 1993, doi: https://doi.org/10.1016/0005-1098(93)90142-G.

[2.157] Y. Cheng and B. L. R. D. Moor, "Robustness analysis and control system design for a hydraulic servo system," *IEEE Transactions on Control Systems Technology,* vol. 2, no. 3, pp. 183-197, 1994, doi: 10.1109/87.317976.

[2.158] S. Skogestad and I. Postlethwaite, *Multivariable feedback control: analysis and design*. Citeseer, 2007.

[2.159] P. Wipasuramonton, Z. Q. Zhu, and D. Howe, "Predictive current control with current-error correction for PM brushless AC drives," *IEEE Transactions on Industry Applications,* vol. 42, no. 4, pp. 1071-1079, 2006.

[2.160] J. Stumper, S. Kuehl, and R. Kennel, "Predictive torque control for AC drives: Improvement of parametric robustness using two-degree-of-freedom control," in *2013 IEEE Energy Conversion Congress and Exposition*, pp. 1170-1175,15-19 Sept. 2013.

[2.161] M. Siami, D. A. Khaburi, A. Abbaszadeh, and J. Rodríguez, "Robustness Improvement of Predictive Current Control Using Prediction Error Correction for Permanent-Magnet Synchronous Machines," *IEEE Transactions on Industrial Electronics,* vol. 63, no. 6, pp. 3458-3466, 2016.

[2.162] M. Siami, D. A. Khaburi, and J. Rodríguez, "Torque Ripple Reduction of Predictive Torque Control for PMSM Drives With Parameter Mismatch," *IEEE Transactions on Power Electronics,* vol. 32, no. 9, pp. 7160-7168, 2017

[2.163] X. Yuan, S. Zhang, and C. Zhang, "Improved Model Predictive Current Control for SPMSM Drives With Parameter Mismatch," *IEEE Transactions on Industrial Electronics,* vol. 67, no. 2, pp. 852-862, 2020.

[2.164] F. Wang, K. Zuo, P. Tao, and J. Rodriguez, "High Performance Model Predictive Control for PMSM by using Stator Current Mathematical Model Self-regulation Technique," *IEEE Transactions on Power Electronics,* 2020.

[2.165] F. Niu *et al.*, "Current Prediction Error Reduction Method of Predictive Current Control for Permanent Magnet Synchronous Motors," *IEEE Access,* vol. 8, pp. 124288-124296, 2020.

[2.166] X. Zhang, L. Zhang, and Y. Zhang, "Model Predictive Current Control for PMSM Drives With Parameter Robustness Improvement," *IEEE Transactions on Power Electronics,* vol. 34, no. 2, pp. 1645-1657, 2019, doi: 10.1109/TPEL.2018.2835835.

[2.167] R. Errouissi, M. Ouhrouche, W.-H. Chen, and A. M. Trzynadlowski, "Robust cascaded nonlinear predictive control of a permanent magnet synchronous motor with anti-windup compensator," *IEEE Transactions on Industrial Electronics,* vol. 59, no. 8, pp. 3078-3088, 2011.

[2.168] S.-C. Carpiuc and C. Lazar, "Fast real-time constrained predictive current control in permanent magnet synchronous machine-based automotive traction drives," *IEEE Transactions on Transportation Electrification,* vol. 1, no. 4, pp. 326-335, 2015.

[2.169] Y. Jiang, W. Xu, C. Mu, and Y. Liu, "Improved deadbeat predictive current control combined sliding mode strategy for PMSM drive system," *IEEE Transactions on Vehicular Technology,* vol. 67, no. 1, pp. 251-263, 2017.

[2.170] S. Huang, G. Wu, F. Rong, C. Zhang, S. Huang, and Q. Wu, "Novel predictive stator flux control techniques for PMSM drives," *IEEE Transactions on Power Electronics,* vol. 34, no. 9, pp. 8916-8929, 2018.

[2.171] Y. Xu, X. Ding, J. Wang, and C. Wang, "Robust three-vector-based low-complexity model predictive current control with supertwisting-algorithm-based second-order sliding-mode observer for permanent magnet synchronous motor," *IET Power Electronics,* vol. 12, no. 11, pp. 2895-2903, 2019.

[2.172] X. Yuan, S. Zhang, and C. Zhang, "Enhanced robust deadbeat predictive current control for PMSM drives," *IEEE Access,* vol. 7, pp. 148218-148230, 2019.

[2.173] S.-W. Kang, J.-H. Soh, R.-Y. Kim, K.-J. Lee, and S.-I. Kim, "Robust predictive current control for IPMSM without rotor flux information based on a discrete-time disturbance observer," *IET Electric Power Applications,* vol. 13, no. 12, pp. 2079-2089, 2019.

[2.174] S. Liu and C. Liu, "Virtual-Vector based Robust Predictive Current Control for Dual Three-Phase PMSM," *IEEE Transactions on Industrial Electronics,* 2020.

[2.175] F. Wang, L. He, and J. Rodriguez, "FPGA based Continuous Control Set Model Predictive Current Control for PMSM System Using Multi-step Error Tracking Technique," *IEEE Transactions on Power Electronics,* 2020.

[2.176] G. Wu, S. Huang, Q. Wu, C. Zhang, F. Rong, and Y. Hu, "Predictive Torque and Stator Flux Control for N* 3-phase PMSM Drives with Parameter Robustness Improvement," *IEEE Transactions on Power Electronics,* 2020.

[2.177] X. Liu and Q. Zhang, "Robust Current Predictive Control-Based Equivalent Input Disturbance Approach for PMSM Drive," *Electronics,* vol. 8, no. 9, p. 1034, 2019.

[2.178] C. Jia, X. Wang, Y. Liang, and K. Zhou, "Robust current controller for IPMSM drives Based on explicit model predictive control with online disturbance observer," *IEEE Access,* vol. 7, pp. 45898-45910, 2019.

[2.179] C. Lin, T. Liu, J. Yu, L. Fu, and C. Hsiao, "Model-Free Predictive Current Control for Interior Permanent-Magnet Synchronous Motor Drives Based on Current Difference Detection Technique," *IEEE Transactions on Industrial Electronics,* vol. 61, no. 2, pp. 667-681, 2014.

[2.180] Y. Zhou, H. Li, and H. Zhang, "Model-free deadbeat predictive current control of a surface-mounted permanent magnet synchronous motor drive system," *Journal of Power Electronics,* vol. 18, no. 1, pp. 103-115, 2018.

[2.181] C. Ma, H. Li, X. Yao, Z. Zhang, and F. D. Belie, "An Improved Model-Free Predictive Current Control With Advanced Current Gradient Updating Mechanism," *IEEE Transactions on Industrial Electronics,* vol. 68, no. 12, pp. 11968-11979, 2021, doi: 10.1109/TIE.2020.3044809.

[2.182] C. A. Agustin, J. T. Yu, Y. S. Cheng, C. K. Lin, and Y. W. Yi, "A Synchronized Current Difference Updating Technique for Model-Free Predictive Current Control of PMSM Drives," *IEEE Access,* vol. 9, pp. 63306-63318, 2021, doi: 10.1109/ACCESS.2021.3075499.

[2.183] Y. Zhang, J. Jin, and L. Huang, "Model-Free Predictive Current Control of PMSM Drives Based on Extended State Observer Using Ultralocal Model," *IEEE Transactions on Industrial Electronics,* vol. 68, no. 2, pp. 993-1003, 2021, doi: 10.1109/TIE.2020.2970660.

[2.184] X. Li, Y. Wang, X. Guo, X. Cui, S. Zhang, and Y. Li, "An Improved Model-Free Current Predictive Control Method for SPMSM Drives," *IEEE Access,* vol. 9, pp. 134672-134681, 2021, doi: 10.1109/ACCESS.2021.3115782.

[2.185] R. Errouissi, M. Ouhrouche, W.-H. Chen, and A. M. Trzynadlowski, "Robust nonlinear predictive controller for permanent-magnet synchronous motors with an optimized cost function," *IEEE Transactions on Industrial Electronics,* vol. 59, no. 7, pp. 2849-2858, 2011.

[2.186] A. Sivaprakasam and L. N. Ramya, "A new approach to minimize torque ripple and noise in model predictive control of permanent magnet synchronous motor drives," *Journal of Vibration and Control,* p. 1077546320933743, 2020, doi: 10.1177/1077546320933743.

[2.187] M. Preindl and S. Bolognani, "Model predictive direct torque control with finite control set for PMSM drive systems, part 2: Field weakening operation," *IEEE Transactions on Industrial Informatics,* vol. 9, no. 2, pp. 648-657, 2012.

[2.188] M. Preindl and S. Bolognani, "Model predictive direct torque control with finite control set for PMSM drive systems, Part 1: Maximum torque per ampere operation," *IEEE Transactions on Industrial Informatics,* vol. 9, no. 4, pp. 1912-1921, 2013.

[2.189] X. Liu, L. Zhou, J. Wang, X. Gao, Z. Li, and Z. Zhang, "Robust Predictive Current Control of Permanent-Magnet Synchronous Motors with Newly Designed Cost Function," *IEEE Transactions on Power Electronics,* 2020.

[2.190] H. T. Nguyen and J. Jung, "Finite Control Set Model Predictive Control to Guarantee Stability and Robustness for Surface-Mounted PM Synchronous Motors," *IEEE Transactions on Industrial Electronics,* vol. 65, no. 11, pp. 8510-8519, 2018.

[2.191] R. Cao and K.-S. Low, "A repetitive model predictive control approach for precision tracking of a linear motion system," *IEEE Transactions on Industrial Electronics,* vol. 56, no. 6, pp. 1955-1962, 2008.

[2.192] L. Yan, F. Wang, P. Tao, and K. Zuo, "Robust Predictive Torque Control of Permanent Magnet Synchronous Machine Using Discrete Hybrid Prediction Model," *IEEE Transactions on Energy Conversion,* 2020.

[2.193] M. Fliess and C. Join, "Model-free control," *International Journal of Control,* vol. 86, no. 12, pp. 2228-2252, 2013, doi: 10.1080/00207179.2013.810345.

[2.194] Z. Sun, Y. Deng, J. Wang, T. Yang, Z. Wei, and H. Cao, "Finite Control Set Model-Free Predictive Current Control of PMSM With Two Voltage Vectors Based on Ultralocal Model," *IEEE Transactions on Power Electronics,* vol. 38, no. 1, pp. 776-788, 2023, doi: 10.1109/TPEL.2022.3198990.

[2.195] P. G. Carlet, F. Tinazzi, S. Bolognani, and M. Zigliotto, "An Effective Model-Free Predictive Current Control for Synchronous Reluctance Motor Drives," *IEEE Transactions on Industry Applications,* vol. 55, no. 4, pp. 3781-3790, 2019, doi: 10.1109/TIA.2019.2910494.

[2.196] C. Ma, J. Rodriguez, C. Garcia, and F. D. Belie, "Integration of Reference Current Slope Based Model-Free Predictive Control in Modulated PMSM Drives," *IEEE Journal of Emerging and Selected Topics in Power Electronics,* vol. 11, no. 2, pp. 1407-1421, 2023, doi: 10.1109/JESTPE.2022.3159586.

[2.197] A. Traue, G. Book, W. Kirchgässner, and O. Wallscheid, "Toward a Reinforcement Learning Environment Toolbox for Intelligent Electric Motor Control," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 33, no. 3, pp. 919-928, 2022, doi: 10.1109/TNNLS.2020.3029573.

[2.198] G. Book *et al.*, "Transferring Online Reinforcement Learning for Electric Motor Control From Simulation to Real-World Experiments," *IEEE Open Journal of Power Electronics,* vol. 2, pp. 187-201, 2021, doi: 10.1109/OJPEL.2021.3065877.

[2.199] J. Gao, J. Zhang, M. Fan, Z. Peng, Q. Chen, and H. Zhang, "Model Predictive Control of Permanent Magnet Synchronous Motor Based on State Transition Constraint Method," *Mathematical Problems in Engineering,* vol. 2021, p. 3171417, 2021/11/25 2021, doi: 10.1155/2021/3171417.

[2.200] Y. Wang, S. Fang, J. Hu, and D. Huang, "Multiscenarios Parameter Optimization Method for Active Disturbance Rejection Control of PMSM Based on Deep Reinforcement Learning," *IEEE Transactions on Industrial Electronics,* vol. 70, no. 11, pp. 10957-10968, 2023, doi: 10.1109/tie.2022.3225829.

[2.201] Y. Wang, S. Fang, and J. Hu, "Active Disturbance Rejection Control Based on Deep Reinforcement Learning of PMSM for More Electric Aircraft," *IEEE Transactions on Power Electronics,* vol. 38, no. 1, pp. 406-416, 2023, doi: 10.1109/tpel.2022.3206089.

[2.202] M. Schenke, W. Kirchgässner, and O. Wallscheid, "Controller Design for Electrical Drives by Deep Reinforcement Learning: A Proof of Concept," *IEEE Transactions on Industrial Informatics,* vol. 16, no. 7, pp. 4650-4658, 2020, doi: 10.1109/TII.2019.2948387.

[2.203] M. Schenke and O. Wallscheid, "A Deep Q-Learning Direct Torque Controller for Permanent Magnet Synchronous Motors," *IEEE Open Journal of the Industrial Electronics Society,* vol. 2, pp. 388-400, 2021, doi: 10.1109/OJIES.2021.3075521.

[2.204] J. Zhao, C. Yang, W. Gao, and L. Zhou, "Reinforcement Learning and Optimal Control of PMSM Speed Servo System," *IEEE Transactions on Industrial Electronics,* vol. 70, no. 8, pp. 8305-8313, 2023, doi: 10.1109/tie.2022.3220886.

[2.205] Z. Song, J. Yang, X. Mei, T. Tao, and M. Xu, "Deep reinforcement learning for permanent magnet synchronous motor speed control systems," *Neural Computing and Applications,* vol. 33, no. 10, pp. 5409-5418, 2020, doi: 10.1007/s00521-020-05352-1.

[2.206] A. Ez-zizi, S. Farrell, D. Leslie, G. Malhotra, and C. J. H. Ludwig, "Reinforcement Learning Under Uncertainty: Expected Versus Unexpected Uncertainty and State Versus Reward Uncertainty," *Computational Brain & Behavior,* 2023/03/20 2023, doi: 10.1007/s42113-022-00165-y.

[2.207] D. Jakobeit, M. Schenke, and O. Wallscheid, "Meta-Reinforcement-Learning-Based Current Control of Permanent Magnet Synchronous Motor Drives for a Wide Range of Power Classes," *IEEE Transactions on Power Electronics,* vol. 38, no. 7, pp. 8062-8074, 2023, doi: 10.1109/tpel.2023.3256424.

[2.208] A. Modi, N. Jiang, S. Singh, and A. Tewari, "Markov decision processes with continuous side information," in *Algorithmic Learning Theory*, PMLR, pp. 597-618, 2018.

[2.209] T. Yu *et al.*, "Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning," in *Conference on robot learning*, 2020: PMLR, pp. 1094-1100.

CHAPTER 3

TWO-VECTOR DIMENSIONLESS MODEL PREDICTIVE CONTROL OF PMSM DRIVES BASED ON FUZZY DECISION MAKING

3.1 Introduction

Model predictive controls (MPCs) with the merits of non-linear multi-variable control can perform better than other commonly used control methods for PMSM drives. However, the conventional MPCs have various issues, including unsatisfactory steady-state performance, variable switching frequency, and challenging selection of appropriate weighting factors. All these issues significantly impact the overall drive performance of conventional MPCs of PMSMs. Thus, a practical MPC should realize the trade-off between these different issues because focusing on one issue and neglecting the others can degrade the performance over various operating conditions.

This chapter proposes two improved MPC methods to deal with different issues of conventional MPCs. The first method is based on model predictive torque control (MPTC) and implements two-vector and two cost functions (torque and flux). Fuzzy decision-making eliminates the weighting factor and selects the first optimum vector. The torque cost function selects the second vector whose duty cycle is determined based on torque error to decrease torque ripples further. The second method is based on model predictive current control (MPCC) with two voltage vectors. The first vector is selected in the same way as in the

conventional MPCC. Two separate current cost functions and fuzzy decision-making are used to select the second vector, whose duty cycle is determined based on the current error. Both proposed methods utilize the space vector PWM modulator to regulate the switching frequency.

The rest of this chapter is organized as follows. Section 3.2 discusses the mathematical modelling of PMSM drive systems with MPC. Section 3.3 discusses the proposed improved MPC methods. Sections 3.4 and 3.5 present the numerical simulation and experimental tests, respectively. Section 3.6 presents the quantitative performance comparison of the proposed methods with conventional MPC. Section 3.7 presents the quantitative comparison of the proposed method and MPC with a regulated switching frequency. Section 3.8 summarizes the findings and outcomes of the chapter.

3.2 PMSM Drive Modelling

PMSM drive system involves various models and subsystems that need to be designed and mathematically modelled before developing a simulation or experimental design of the system. Mathematical models, inverters, and control methods are essential parts of the drive system.

3.2.1 PMSM Dynamic Model

In order to design an efficient PMSM drive system, it is crucial to develop an accurate PMSM mathematical model. Considering a three-phase PMSM in Fig. 3.1 represented in *abc* and *dq* reference frames. The three-phase voltages can be obtained as follows:

$$v_{a,b,c} = R_s i_{a,b,c} + \frac{d}{dt} \Psi_{a,b,c} \tag{3.1}$$

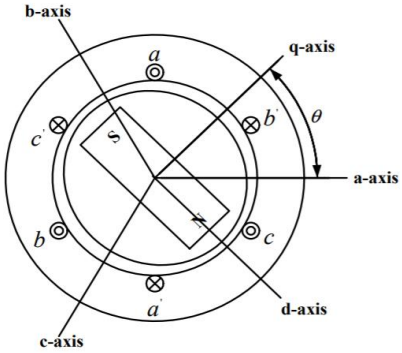


Fig. 3.1. PMSM in *abc* and *dq* frames.

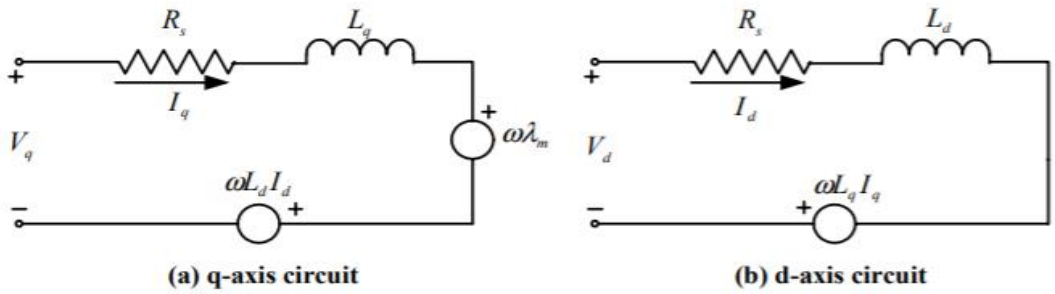


Fig. 3.2. Equivalent dq-circuit of PMSM.

Using phase transformation, the equivalent *dq*-circuits of PMSM can be obtained (Fig. 3.2), and the three-phase voltages can be transformed from an *abc*-frame into a *dq*-frame as:

$$v_d = R_s i_d + \frac{d\psi_d}{dt} - \omega_s \Psi_q \tag{3.2}$$

$$v_q = R_s i_q + \frac{d\psi_q}{dt} + \omega_s \Psi_d \tag{3.3}$$

and the flux equations are:

$$\Psi_d = L_d i_d + \Psi_{PM} \tag{3.4}$$

$$\Psi_q = L_q i_q \tag{3.5}$$

The mechanical and electrical torque can be expressed as follow:

$$T_e = \frac{3}{2}\frac{P}{2}\left[\left(L_d - L_q\right)i_d i_q + \Psi_{PM} i_q\right] \tag{3.6}$$

$$T_m = T_L + J\frac{d}{dt}\omega_r + B\omega_r \tag{3.7}$$

where v_d and v_q are the d- and q-axis voltages, i_d and i_q the d- and q-axis currents, L_d and L_q ($L_q = L_d$ in the case of surface-mounted PMSM) the d- and q-axis inductances, respectively; R_s is the stator resistance, ω is the machine speed, and Ψ_{pm} the permanent magnet flux in the rotor; T_m, T_e, and T_L are the mechanical torque, electrical torque, and load torque, respectively; J is the momentum of inertia, and B is the vicious friction coefficient.

3.2.2 Three-phase Inverter

A voltage Source Inverter (VSI) is a device that converts a DC voltage to an AC voltage of variable frequency and magnitude fed to the AC motors. Pulse Width Modulation (PWM) method is used to generate switching pulses to control the switches states of the inverter. Fig. 3.3 shows a three-phase VSI comprising six power switches, diodes, and DC link capacitors. The output voltages V_a, V_b, and V_c are applied to the stator windings of a motor. The basic operation can be understood by assuming a constant value for the DC link voltage in the three-phase VSI circuit, which implements the concept of switching pulses. Each inverter phase leg is independently operated. The switching pulses of the three inverter legs are denoted as S_a, S_b, and S_c. The line voltages can be derived using the switching pulses and DC voltage as:

$$V_{ab} = V_{dc}\left(S_a - S_b\right) \tag{3.8}$$

$$V_{bc} = V_{dc}\,(S_b - S_c) \tag{3.9}$$

$$V_{ca} = V_{dc}\,(S_c - S_a) \tag{3.10}$$

where V_{ab}, V_{bc} and V_{ca} are the line voltages and V_{dc} is the DC supply voltage.

From the line voltages equations, the phase voltages can be derived by assuming the system is balanced in the way of the summations of three-phase currents, and voltages are equal to zero. The phase voltages in terms of the line voltages are given as follows:

$$V_{an} = \frac{V_{ab} - V_{ca}}{3} \tag{3.11}$$

$$V_{bn} = \frac{V_{bc} - V_{ab}}{3} \tag{3.12}$$

$$V_{cn} = \frac{V_{ca} - V_{bc}}{3} \tag{3.12}$$

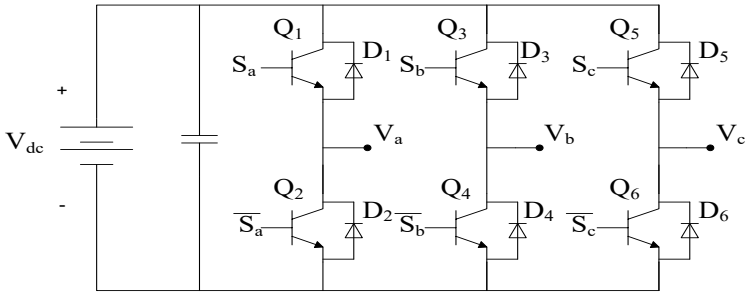


Fig. 3.3. Three-phase Voltage Source Inverter.

Substituting (3.11), (3.12), and (3.13) into (3.8), (3.9), and (3.10), one obtains the line-to-neutral voltages of the load as the following:

$$V_{an} = \frac{V_{dc}}{3}\,(2\mathrm{S_a} - \mathrm{S_b} - \mathrm{S_c}) \tag{3.14}$$

$$V_{bn} = \frac{V_{dc}}{3}\,(2\mathrm{S_b} - \mathrm{S_c} - \mathrm{S_a}) \tag{3.15}$$

$$V_{cn} = \frac{V_{dc}}{3}\,(2S_c - S_a - S_b) \tag{3.16}$$

3.2.3 Model Predictive Control

The purpose of the control method in a high-performance PMSM drive is to generate the switching pulses for the inverter. MPC generates the pulses by minimizing a cost function to select the optimum switching states. The operation of MPC includes measurement of the machine variables, estimation of the variables that cannot be measured, and finally, prediction of future values based on the measured and estimated values [3.1-3.2]. Based on the predicted variables, MPC can be classified into MPTC [3.3] and MPCC [3.4].

MPTC predicts future torque and flux values based on the measured stator currents and rotor speed. To obtain the prediction equations of MPTC, the voltage equations of PMSMs, (3.2) and (3.3) can be rearranged to solve for $\frac{di_d}{dt}$ and $\frac{di_q}{dt}$ as follows:

$$\frac{di_d}{dt} = -\frac{R_s}{L_d} i_d + \frac{L_q}{L_d}\, \omega i_q + \frac{1}{L_d} v_d \tag{3.17}$$

$$\frac{di_q}{dt} = -\frac{R_s}{L_q} i_q - \frac{L_d}{L_q}\, \omega i_d - \omega\, \Psi_m + \frac{1}{L_q} v_q \tag{3.18}$$

At sampling time T_s, using Euler derivative approximation for $\frac{di_{dq}}{dt}$, one can have:

$$\frac{di_{dq}}{dt} = \frac{i_{dq}(k+1) - i_{dq}(k)}{T_s} \tag{3.19}$$

Hence, the i_{dq} currents at $(k+1)th$ time instant can be calculated by substituting (3.19) into (3.17) and (3.18) as follows:

$$i_d(k+1) = \left(1 - \frac{R_s T_s}{L_d}\right) i_d(k) + \frac{L_q}{L_d} T_s\, \omega\, i_q(k) + \frac{T_s}{L_d} v_d \tag{3.20}$$

$$i_q(k+1) = \left(1 - \frac{R_s T_s}{L_q}\right) i_q(k) - \frac{L_d}{L_q} T_s\, \omega\, i_d(k) - T_s \omega\, \Psi_{PM} + \frac{T_s}{L_q} v_d \tag{3.21}$$

The stator flux at $(k+1)th$ time instant can be predicted using the predicted current in (3.20) and (3.21) as:

$$\Psi_d(k+1) = L_d i_d(k+1) + \Psi_{PM} \tag{3.22}$$

$$\Psi_q(k+1) = L_q i_q(k+1) \tag{3.23}$$

Also, the torque at $(k+1)th$ time instant can be predicted using predicted current and flux as:

$$T_e(k+1) = \frac{3}{2} p\big(\Psi_d(k+1) i_q(k+1) - \Psi_q(k+1) i_d(k+1)\big) \tag{3.24}$$

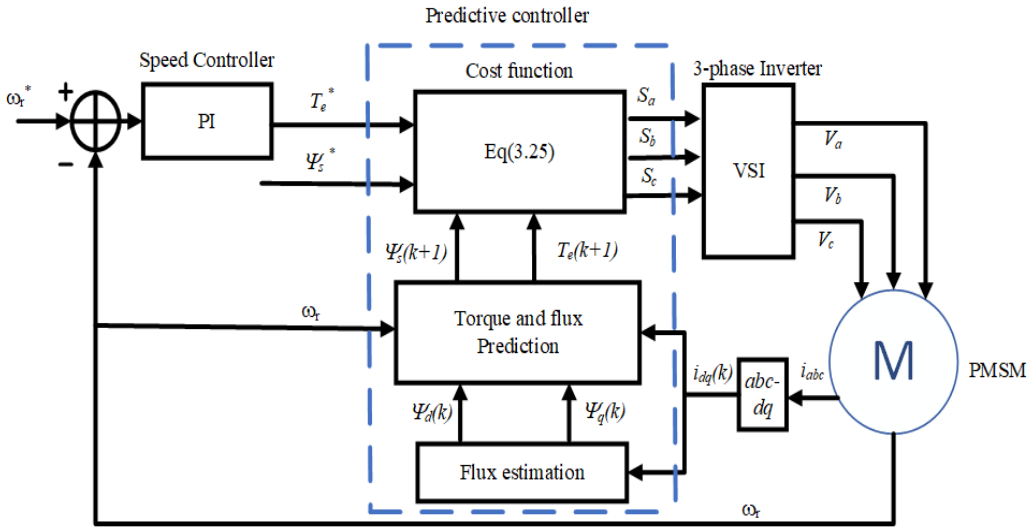
Finally, the cost function of MPTC can be expressed as:

$$g_{MPTC} = (T^* - T_e(k+1))^2 + \gamma(\Psi_s^* - \Psi_s(k+1))^2 \tag{3.25}$$

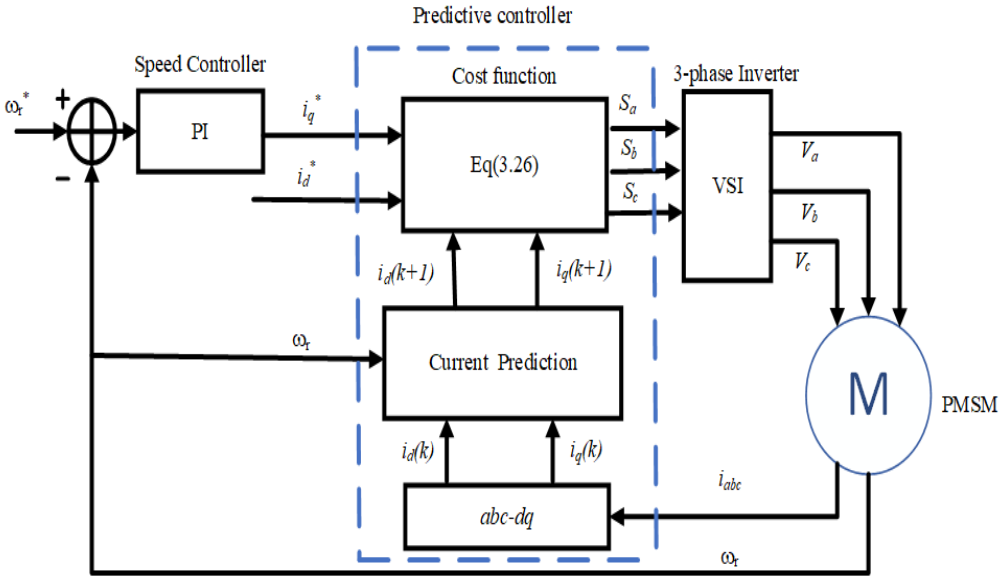
where Ψ_s is stator flux ($\Psi_s = \Psi_d + j\Psi_q$) and γ the weighting factor. Conversely, MPCC predicts the future values of the stator currents based on measured stator currents, rotor speed, and estimated voltages. MPCC works by obtaining the *dq* currents and machine position at sampling instant (*k*), and then the *dq* voltages are calculated and used to predict the *dq* currents for the *(k+1)th* sampling interval. The *dq* currents at *(k+1)th* time instant are predicted using (3.20) and (3.21) [3.5-3.6]. MPCC cost function to select the optimum voltage vector is:

$$g_{MPCC} = \left(i_d{}^* - i_d(k+1)\right)^2 + \left(i_q{}^* - i_q(k+1)\right)^2 \tag{3.26}$$

The block diagram of MPC for PMSM drives is shown in Fig. 3.4, where Fig. 3.4(a) shows MPTC and Fig. 3.4 (b) shows MPCC.



(a)



(b)

Fig. 3.4. Block diagrams of MPCs, (a) MPTC, (b) MPCC.

3.3 Proposed Two-Vector Dimensionless MPC

The proposed methods aim to improve the steady-state performance, regulate the switching frequency, and eliminate the weighting factor while maintaining an overall performance trade-off.

Prior to discussing the proposed method, it is essential to consider one-step delay compensation. In the conventional MPC, the machine variables are measured at time instant (*k*) and then predicted at time instant (*k+1*). However, the actuating signals (voltage vector) are only available at (*k+2*)*th* time instant. This creates a step time delay [3.7]. To compensate for this delay, the variables are predicted at (*k+2*)*th* time instant. Thus, the cost functions of the conventional MPCC and MPTC in (3.25) and (3.26) are rewritten as follows [3.8]:

$$g_{MPTC} = \left(T_e^* - T_e(k+2)\right)^2 + \gamma\left(\Psi_s^* - \Psi_s(k+2)\right)^2 \tag{3.27}$$

$$g_{MPCC} = \left(i_d{}^* - i_d(k+2)\right)^2 + \left(i_q{}^* - i_q(k+2)\right)^2 \tag{3.28}$$

3.3.1 Method-I: Two-Vector dimensionless FDM-MPTC

The most common issue in MPTC is the selection of the weighting factor. The cost function of MPTC in (3.27) contains two objective functions based on torque and flux. Each objective function has a different degree of importance, and torque and flux have different magnitudes and units. Thus, a weighting factor (γ) must be included to balance the performance. The selection of the weighting factor is an ambiguous process, and a significant performance effect can be experienced if an inappropriate value is selected. The weighting factor can be optimized manually (offline) based on empirical procedures [3.9] or based on other methods such as parameter sweep [3.10] and Genetic algorithm [3.11]. However, these methods are time-consuming and influenced by parameters and operating conditions [3.12]. In addition, different approaches are considered to eliminate the weighting factors in MPC

of PMSM drives, including using cost function-based voltage vector tracking error instead of torque-flux error [3.13] or using a new lookup table of DTC [3.14].

This research proposes a two-vector MPTC based on fuzzy decision-making (FDM) to eliminate the weighting factor and improve steady-state performance. Generally, choosing the weighting factor (γ) requires the absolute importance of both torque and flux objective functions. However, FDM depends on the relative importance of each objective function over another and can be chosen based on the decision maker's subjective and qualitative experience and judgment. The torque and flux objective functions in (3.27) can be rewritten individually as:

$$g_T = \left(T_e^* - T_e(k+2)\right)^2 \tag{3.29}$$

$$g_\Psi = \left(\Psi_s^* - \Psi_s(k+2)\right)^2 \tag{3.30}$$

The proposed method utilizes two switching vectors over one control cycle. The torque objective function in (3.29) is evaluated separately based on eight voltage vectors ($V_0 - V_7$) of a 2-level three-phase inverter and minimized to obtain an optimum first voltage vector V_1. The second optimum voltage vector V_2 is obtained based on FDM using the torque and flux objective function.

The torque and flux objective functions in (3.29) and (3.30) are evaluated based on eight voltage vectors ($V_0 - V_7$) of a 2-level three-phase inverter, the final optimum voltage vector is determined using FDM. FDM is used where insufficient and incomplete data exist for the solution [3.15]. It is a bit different from the conventional fuzzy approach and has been introduced for MPC in [3.16] but not for the aim of eliminating the weighting factor and was applied for MPC in power converters [3.17] and induction motors [3.18]. To apply FDM in MPC, the specification of membership and decision functions are required. The linear

membership function is the common form used in MPC. Therefore, a cost function is evaluated at eight voltage vectors $g_i(V_s)$, $(V_s = 0 - 7)$, and the linear membership function $\mathrm{m_i}(\mathrm{V_s})$ is:

$$\mathrm{m_i}(V_s) = \left(\frac{g_i^{max} - g_i(V_s)}{g_i^{max} - g_i^{min}}\right)^{R_i} \tag{3.31}$$

where R_i is a priority weight factor determined based on each objective function's relative importance.

In [3.19], the numbers for various relative importance cases are listed and assigned depending on the priority importance of an objective function over the other. For the proposed method, an intermediate value of two ($R_i = 2$) is selected as the priority weight for both torque and flux objective functions. Hence, the membership function of the torque objective function $\mathrm{m_T}(V_s)$ and flux objective function $\mathrm{m}_{\Psi}(V_s)$ are:

$$\mathrm{m_T}(V_s) = \left(\frac{g_T^{max} - g_T(V_s)}{g_T^{max} - g_T^{min}}\right)^2 \tag{3.32}$$

$$\mathrm{m}_{\Psi}(V_s) = \left(\frac{g_{\Psi}^{max} - g_{\Psi}(V_s)}{g_{\Psi}^{max} - g_{\Psi}^{min}}\right)^2 \tag{3.33}$$

The cost functions of torque $g_T(V_s)$ and flux $g_{\Psi}(V_s)$ are evaluated based on eight different voltage vectors ($V_s = 0 - 7$). Then, using the obtained values, their maximum ($g_T^{max}, g_{\Psi}^{max}$), and minimum values ($g_T^{min}, g_{\Psi}^{min}$) as in (3.32) and (3.33), the torque and flux membership functions ($\mathrm{m_T}, \mathrm{m}_{\Psi}$) in the range of [0 1] are obtained. Thus, an optimum voltage vector can be selected by minimizing and maximizing a decision function as:

$$\mathrm{m_D}(V_s) = \min\{m_T(V_s), m_{\Psi}(V_s)\} \tag{3.34}$$

The optimum voltage vector (V_{opt}) is selected as:

$$V_{opt} = \max_{s=0-7}\left(\mathrm{m_D}(V_s)\right) \tag{3.35}$$

With the previously obtained vector V_1 by minimizing the torque cost function and making the optimum vector obtained in (3.35) as $V_{opt} = V_2$, this will result in two different voltage vectors (V_1, V_2). By computing the dq voltages components of each of these two vectors as:

$$u_{dq_1} = u_{dq}(V_1) \tag{3.36}$$

$$u_{dq_2} = u_{dq}(V_2) \tag{3.37}$$

The final reference voltage u_{dq}^{ref} is obtained as the combinations of the dq voltages components of two vectors with different duty cycles as:

$$u_{dq}^{ref} = \frac{t_1 \times u_{dq_1}}{T_s} + \frac{(T_s - t_1) \times u_{dq_2}}{T_s} \tag{3.38}$$

where T_s is the sampling time, and t_1 $(0 < t_1 < T_s)$ the time assigned for the first vector (V_1). The duty cycle d_1 of V_1 is determined based on the torque error:

$$d_1 = \left| \frac{T_e^* - T_e(k+2)}{C_T} \right| \tag{3.39}$$

where C_T is a positive constant value to be chosen to minimize the torque ripple.

Hence, the proposed FDM-MPTC can eliminate the weighting factor by transforming the torque and flux terms into dimensionless quantities using FDM and obtaining the optimum voltage vector with the factorless MPTC. To improve the steady-state performance and reduce torque ripples, a second vector is obtained by evaluating the torque cost function. This vector and the optimum vector are applied in each control cycle. This will improve the torque performance and reduce torque ripple because the second vector is obtained by evaluating the torque cost function and its duty cycle based on the torque error. If the torque ripple is high, the vector obtained from the torque cost function (V_1) will be applied for a longer time to reduce torque ripple. Finally, PWM modulator-based space vector regulates

the switching frequency and accounts for a trade-off between the steady-state performance, weighting factor, and switching frequency. The reference u_{dq}^{ref} in (3.38) is fed to the SVM PWM modulator; thus, the inverter's switching pulses with a regulated switching frequency are generated. Fig. 3.5 shows a block diagram of the proposed FDM-MPTC.

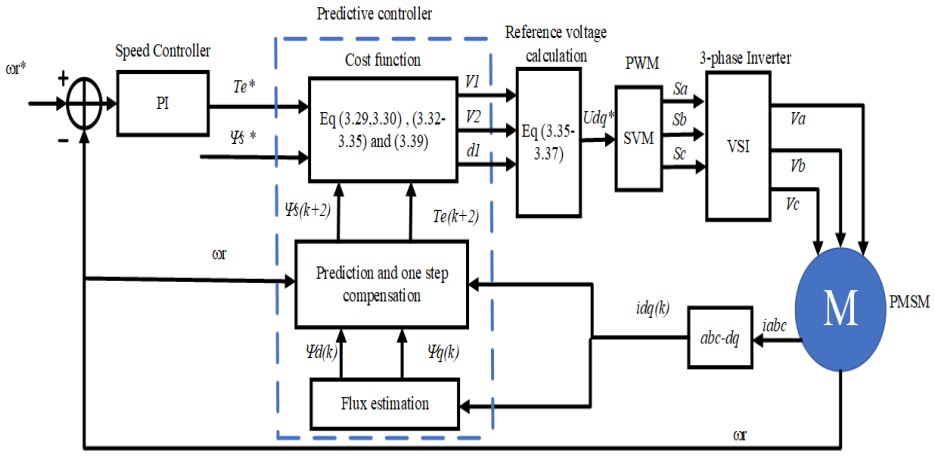


Fig. 3.5. The Block Diagram of the proposed FDM-MPTC for PMSM drives.

3.3.2 Method II: Two-vector FDM-MPCC

The conventional MPCC evaluates the cost function (3.31) based on the current difference to obtain the optimum switching vector for the next control cycle. Most improvement of MPCC applies one or more vectors along with the optimum vector over one control cycle. Usually, the weighting factor is not required in MPCC. However, weighting factors are used to render the cost functions comparable in magnitude and units. Clearly, the MPCC's cost function terms have the same unit but may have different magnitudes. Thus, the weighting factor can be used, like in [3.20], which presented an MPCC cost function with a weighting factor computed using a fuzzy logic controller based on current errors. This research applies two-vector MPCC with duty cycle control over one control cycle. The first vector is selected using FDM presented in Method-I by rewriting the cost function in (3.31)

into two separate objective functions. The i_{dq} current differences as the *d*- and *q*-objective functions are as follows:

$$g_d(V_s) = \left(i_d{}^* - i_d(k+2)\right)^2 \tag{3.40}$$

$$g_q(V_s) = \left(i_q{}^* - i_q(k+2)\right)^2 \tag{3.41}$$

The membership functions m_i for the current cost functions $g_d(V_s)$ and $g_q(V_s)$ can be expressed in the general form with different priority weights as:

$$\mathrm{m_i}(V_s) = \left(\frac{g_i^{max} - g_i(V_s)}{g_i^{max} - g_i^{min}}\right)^{R_i} \tag{3.42}$$

The priority weight R_i in (3.42) is determined based on the relative importance of each objective function. First, a pairwise comparison matrix Γ is constructed by comparing the objective function with each other. Since the current i_q directly influences the torque and current performance, the *q*-objective function is given moderate importance over the *d*-objective function to improve the torque performance. The comparison matrix Γ between the two objective functions (g_d, g_q) can be obtained as:

$$\Gamma = \begin{bmatrix} 1 & 3 \\ \frac{1}{3} & 1 \end{bmatrix} \tag{3.43}$$

By computing the eigenvector γ corresponding to the maximum value of the eigenvalues of Γ, the priority weight R_i can be expressed as:

$$R_i = [\, R_1 \,....\, R_i] = \frac{\gamma_{\mathrm{i}_{max}}}{\sum_{i=1}^{i} \gamma_{\mathrm{i}_{max}}} \tag{3.44}$$

In our case, the priority weight $R_i = [R_q \;\; R_d]$, which has been obtained as $R_q = 0.75$ and $R_d = 0.25$. More details on obtaining the comparison matrix and computing the priority

weight can be found in [3.19]. The membership function of the *d*- and *q*-objective functions($\mathrm{m_q}(V_s)$,$\mathrm{m}_d(V_s)$) are:

$$\mathrm{m_d}(V_s) = \left(\frac{g_d^{max} - g_d(V_s)}{g_d^{max} - g_d^{min}}\right)^{R_d} \tag{3.45}$$

$$\mathrm{m_q}(V_s) = \left(\frac{g_q^{max} - g_q(V_s)}{g_q^{max} - g_q^{min}}\right)^{R_q} \tag{3.46}$$

The *d*-objective function $g_d(V_s)$ and *q*-objective function $g_d(V_s)$ are evaluated based on eight different voltage vectors ($V_s = 0 - 7$). Then, using the obtained values, their maximum (g_d^{max}, g_d^{max}) and minimum values (g_d^{min}, g_d^{min}) as in (3.45) and (3.46), the membership functions ($\mathrm{m_d}, \mathrm{m}_d$) in the range of [0 1] are obtained. An optimum voltage vector is selected by minimizing a decision function as:

$$\mathrm{m_D}(V_s) = \min\{m_d(V_s), m_q(V_s)\} \tag{3.47}$$

Then, the first optimum vector is determined by maximizing the decision function in (3.47) as:

$$V_1 = \max_{s=0-7}\left(\mathrm{m_D}(V_s)\right) \tag{3.48}$$

The second vector is determined by minimizing the cost function in (3.26) as:

$$V_2 = \min_{s=0-7}(\mathrm{g_{MPCC}}) \tag{3.49}$$

By computing the *dq* voltage components of two vectors (V_1, V_2), the final reference voltage u_{dq} is obtained as the combinations of the *dq* voltages components of the two vectors with duty cycles as:

$$u_{dq}^{ref} = \frac{t_1 \times u_{dq_1}}{T_s} + \frac{(T_s - t_1) \times u_{dq_2}}{T_s} \tag{3.50}$$

The duty cycle d_1 of V_1 is determined based on i_q current error as:

$$d_1 = \left| \frac{i_q{}^* - i_q(k+2)}{C_q} \right| \tag{3.51}$$

where C_q is a positive constant value.

Similar to Method-I, a PWM modulator-based SVM is used to regulate the switching frequency, where the u_{dq}^{ref} in (3.50) is fed into the PWM modulator to generate the switching pulses. The block diagram of FDM-MPCC is shown in Fig. 3.6.

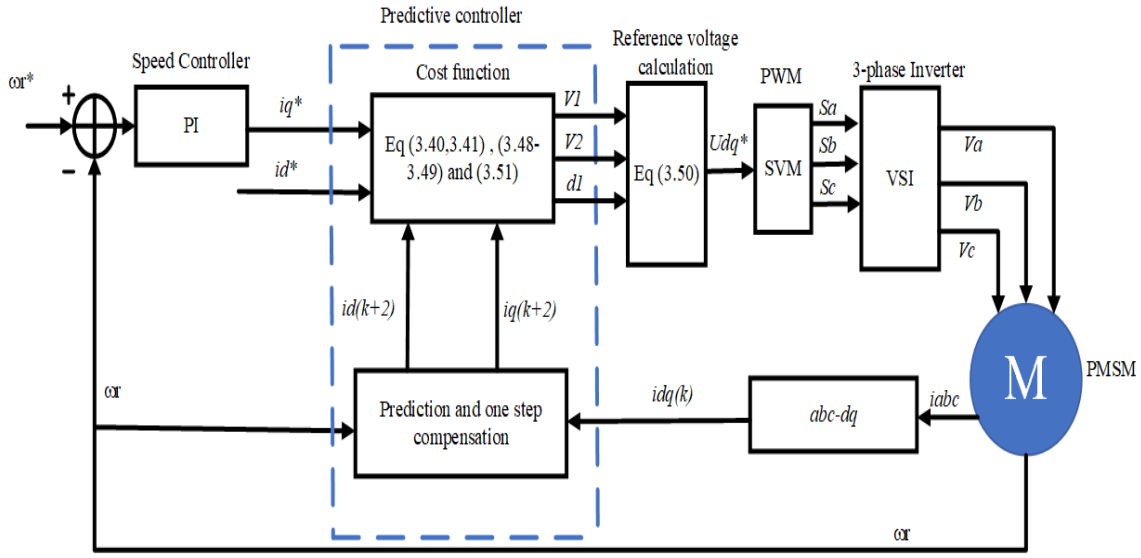


Fig. 3.6. The Block Diagram of the proposed FDM-MPCC for PMSM drives.

3.4 Numerical Simulation

In this section, the proposed two MPC methods (FDM-MPTC and FDM-MPCC) are designed and simulated using Matlab/Simulink based on a 1kW PMSM. To verify the effectiveness of these two methods, the conventional MPC (PCC) is applied to the PMSM and compared with the proposed methods considering different characteristics. The MPC applies one vector in each control cycle but has been optimized for better performance. The PMSM parameters are kept the same for all simulations, as in Table 3.1. The numerical simulation combines start-up, steady-state, and external load tests. The motor starts at 0s with a reference speed (500 rpm or 1000 rpm), and an external load (2 Nm) is applied at 0.1s.

Table 3.1 PMSM drives parameters.

Parameter	Symbol	Value and unit
Stator Resistance	R_s	0.47Ω
d-Axis Inductance	L_d	14.2mH
q-Axis Inductance	L_q	15.9mH
Permanent magnet Flux	Ψ_m	0.1057 Wb
Number of Pole Pairs	P	3
Rated Speed	ω_n	1000 rpm
Rated Torque	T_n	2 Nm
dc-Link voltage	V_{DC}	200V
Inertia	J	0.002 kg/m^2
viscous Friction	B	0.0006 Nm/rad/s
Sampling Time	t_s	100μs

Figs 3.7 and 3.8 show the dynamic responses at 500 and 1000 rpm with an external load of 2 Nm applied at t = 0.3 s for MPC and the proposed FDM-MPTC and FDM-MPCC. From top to bottom, the waveforms are stator current and torque on the left side of the figure and stator flux and rotor speed on the right side. As can be seen from the curves, the proposed FDM-MPTC and FDM-MPCC show an excellent dynamic response with lower overshoot, faster settling time, smaller flux and torque ripples, and smoother stator current responses than compared to MPC.

Fig. 3.9 presents the waveforms of switching frequency, the harmonic spectra of stator currents, and the selected voltage vectors at the steady state of 1000 rpm (the rated speed) with a load torque of 2 Nm. It is seen that the proposed FDM-MPTC and FDM-MPCC show almost fixed average switching frequency, which is better and more regulated than MPC. MPC has an unregulated average switching frequency because the switching pulses are directly generated without a PWM modulator. In addition, the total harmonic distortion (THD) of the stator current is calculated up to 6 kHz maximum frequency. The proposed FDM-MPTC and FDM-MPCC record THD values of 4.61% and 4.12%, respectively. In contrast, MPC has a broad harmonic spectrum, and the stator current THD is up to 6.31%, much higher than the proposed methods.

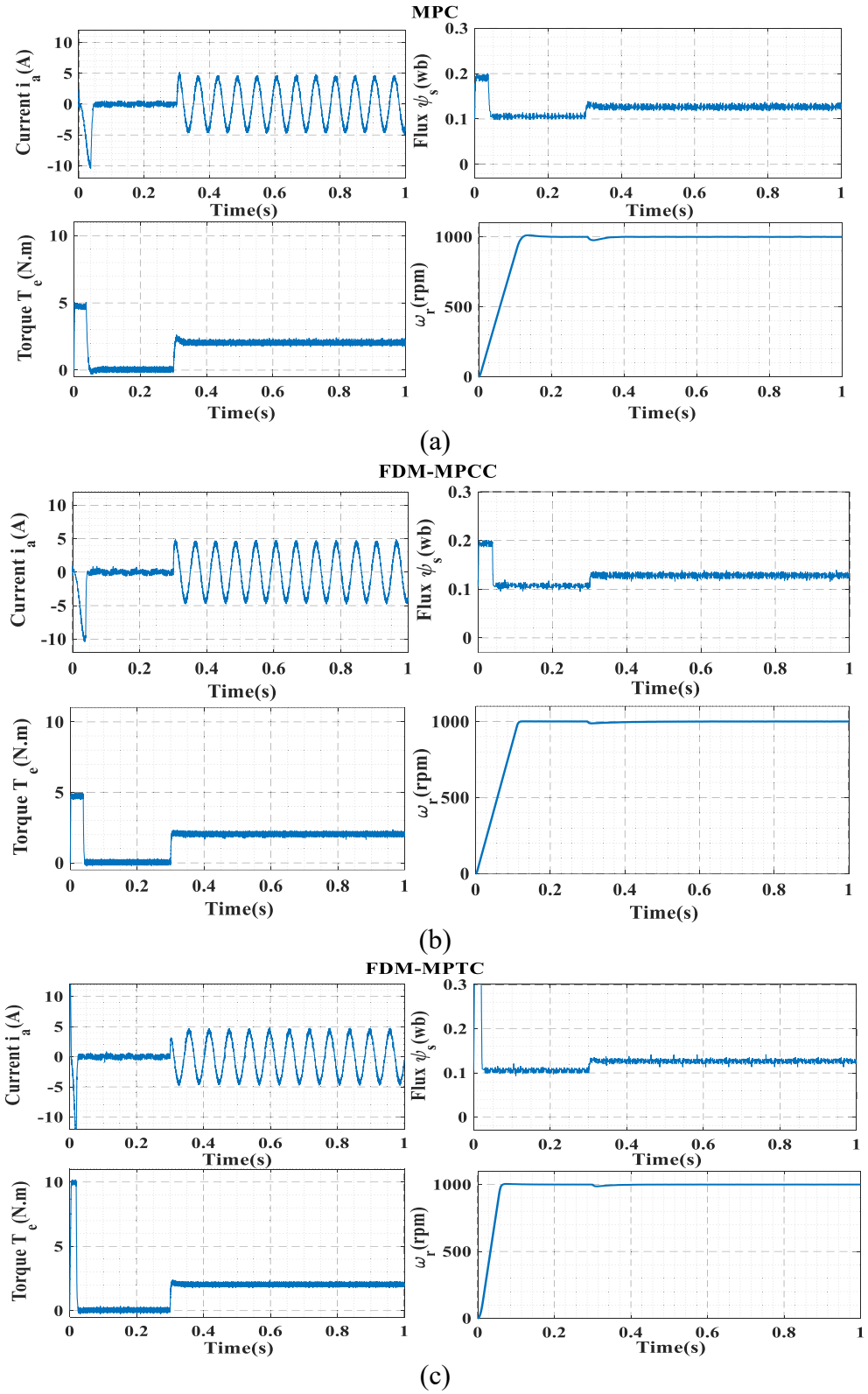


Fig. 3.7. Responses of rotor speed, stator current, torque, and stator flux at 1000 rpm with sudden load change for (a) MPC, (b) proposed FDM-MPCC, and (c) proposed FDM-MPTC.

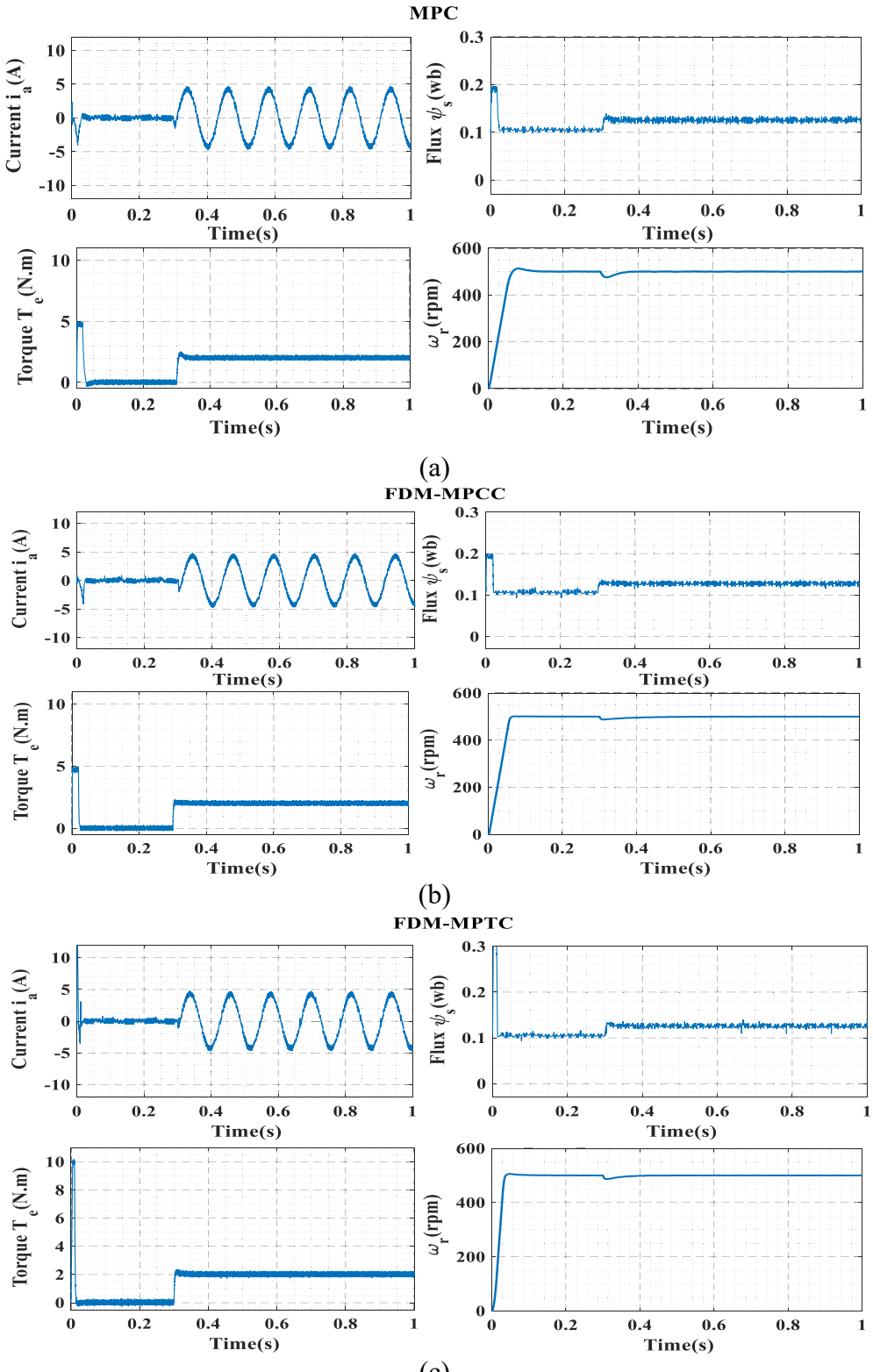
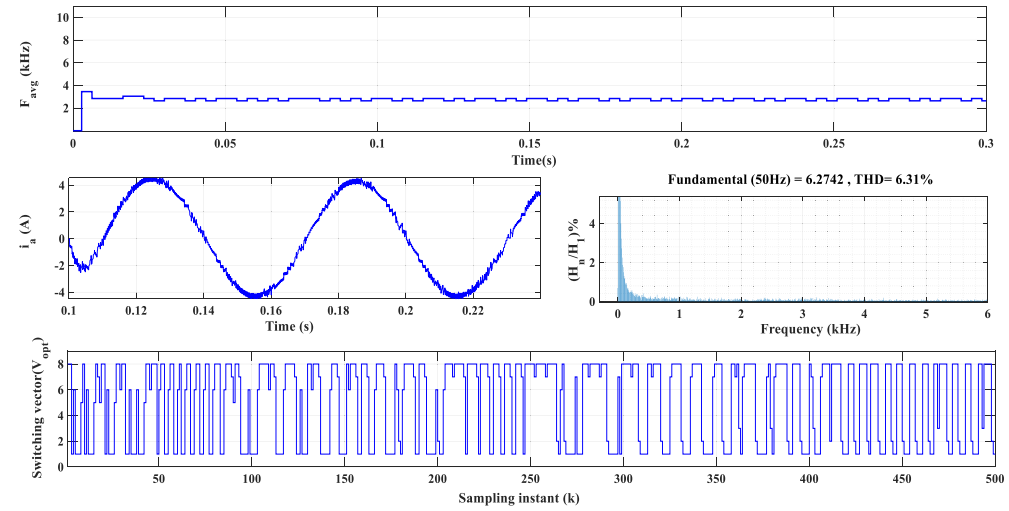
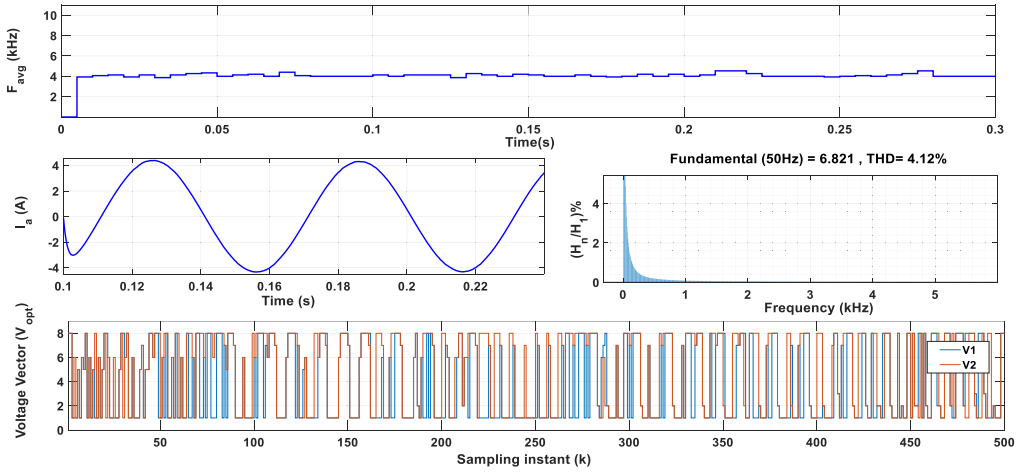


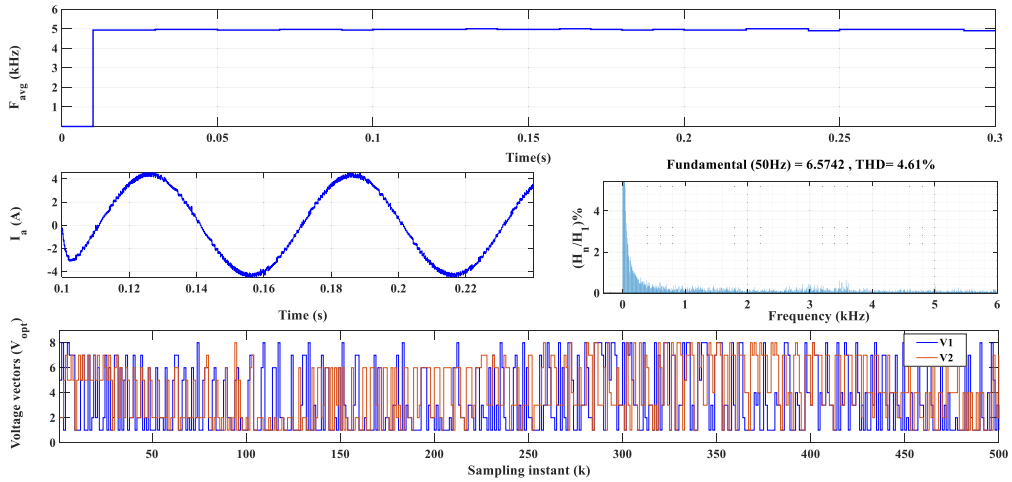
Fig. 3.8. Responses of rotor speed, stator current, torque, and stator flux at 500 rpm with sudden load change for (a) MPC, (b) proposed FDM-MPCC, and (c) proposed FDM-MPTC.



(a)



(b)



(c)

Fig. 3.9. Switching frequency, the harmonic spectrum of stator current, and selected switching vectors for (a) MPC, (b) proposed FDM-MPCC, and (c) proposed FDM-MPTC.

3.5 Experimental Test

3.5.1 Experimental Setup

To validate the effectiveness of the proposed designs and simulation results, an experimental setup is built based on a PMSM drive system. The experimental implementation is accomplished by making a real-time interface between MATLAB/ SIMULINK and dSPACE DS 1104 controller board. The experimental system is divided into two parts, which are software and hardware. The software includes modelling and controlling the drive system using MATLAB/SIMULINK and ControlDesk. In contrast, the hardware consists of a digital controller based on dSPACE DS 1104, gate drive modules, a three-phase power inverter, a DC source, current sensors, an encoder, a three-phase PMSM, and load. The complete experimental setup of the PMSM drive system is shown in Fig. 3.10. In addition, a detailed describtion of hardware components is presented in the next sections.

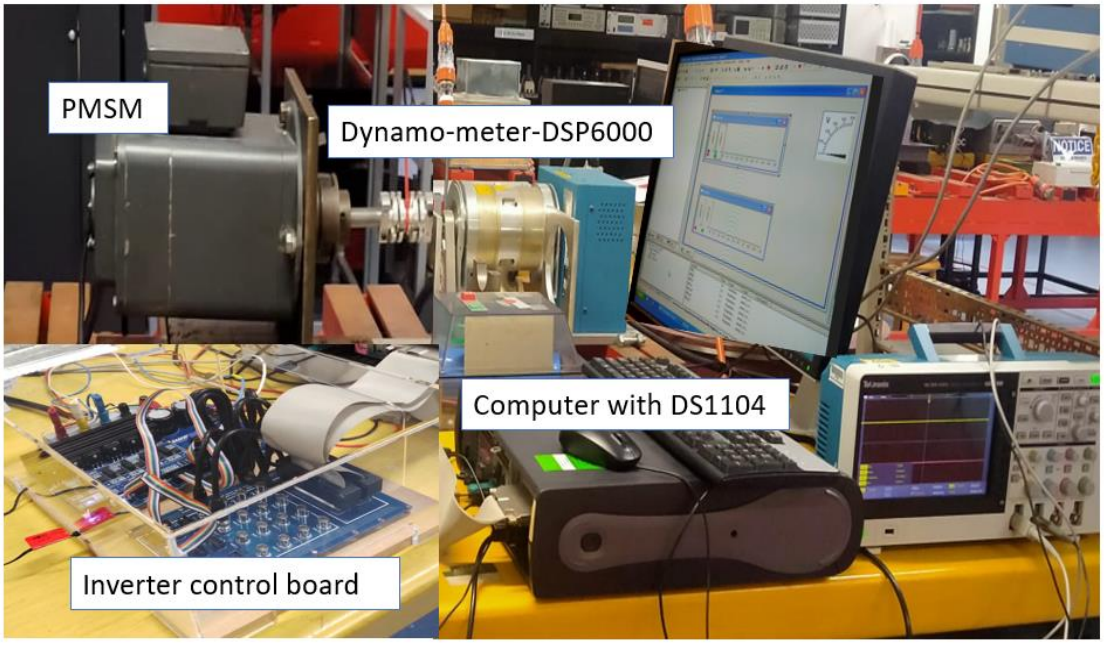


Fig. 3.10. Experimental setup of PMSM drives system.

3.5.1.1 ControlDesk

ControlDesk is a well-established experiment software from dSPACE, designed to provide users with comprehensive control, monitoring, and automation capabilities for experiments, while also facilitating the efficient development of controllers. The workflow typically begins by desgning a system model in MATLAB/ SIMULINK, then a real time code in C language is generataed and uploaded to dSPACE 1104 borad. Subsequently, the real-time code generated can be utilized within ControlDesk to construct control systems for real-world applications.Users can perform real time control of the hardware by fine-tuning system parameters online, such as specifying step input speed commands or adjusting scaling factors. In addition, ControlDesk enable real time monitoring of the system , in which the hardware measurement such as speed and currents are displayed and monitored in real time. Subsequently, the desired system measurement data are sampled at 5 kHz sampling frequency and saved as mat files for in-depth analysis in MATLAB. Fig. 3.11 illustrates a typical layout of ControlDesk.

Fig. 3.11. Layout of ControlDesk.

3.5.1.2 dSPACE

Most applications that require the high speed processing of a large amount of numerical data need to use the digital signal processors (DSP). Therefore, an evaluation module named digital signal processing and control engineering (dSPACE DS 1104) is used in this drive application. The dSPACE DS 1104 is a control board and a stand-alone card that serves as an excellent platform to develop and build the drive system as shown in Fig. 3.12. This type of DSP is equipped with an on-board peripherals such as Analog–to-Digital (ADC) and Digital-to-Analog (DAC) to be used in the digital control systems.

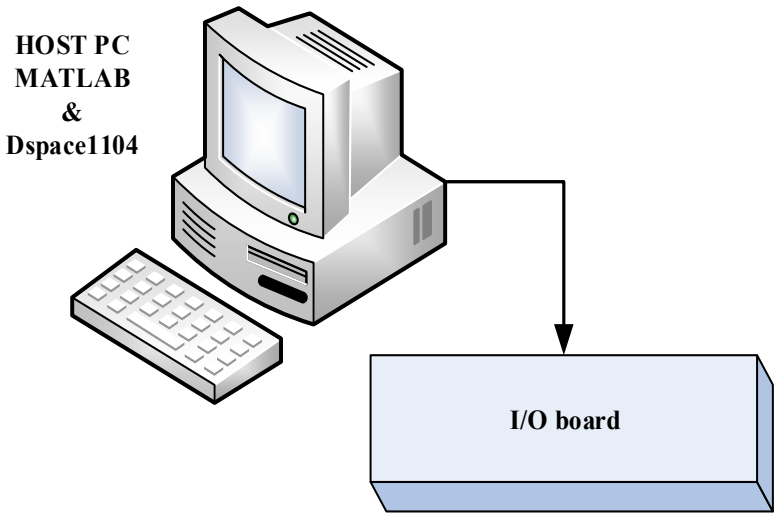


Fig. 3.12. dSPACE1104 illustration with I/O board.

3.5.1.3 Inverter Control Circuit

The control circuit shown in Fig.3.13 is a control interface board specifically designed to control the motor through three-phase inverter. This board is powered by 5 V_{DC} supply and can be divided into three parts,namedly inverter, gate drivers, and interface part. The inverter part comprise three-phase inverter (made of six INFINEON IKW40T120FKSA1 IGBTs and two DC-Link capacitors) and two current sensors. The gate drivers part consists of various

optocouplers to generate the required switching pulses for IGBTs. The interface part acts as interconnection panel between dspace and other hardware components. It takes the control signal from dSPACE and send them to the gate driver, also it receive the current sensors and speed encoder measuurments and feed it back to dSPACE for processing.

In addition, 200Vdc is supplied to the inverter through external DC power supply. The workflow of the control circuit begins by generating a control signals through dSPACE which are sent to gate driver circuit via interface panel. Then, the DC supply is powerd on and the switching pulses generated by the gate drivers are used to control the inverter to produce three phase voltage to operate the motor. The measurements of speed encoder and current sensors are fedbcak to the dSPACE via interface panel for processing and forming the closed loop system.

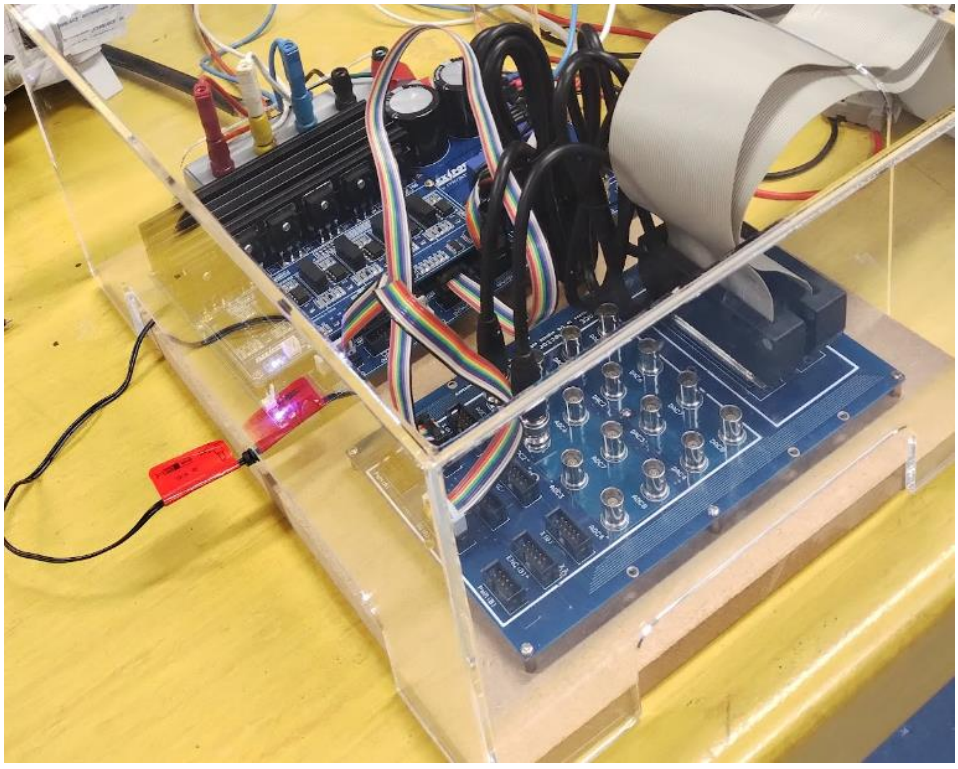


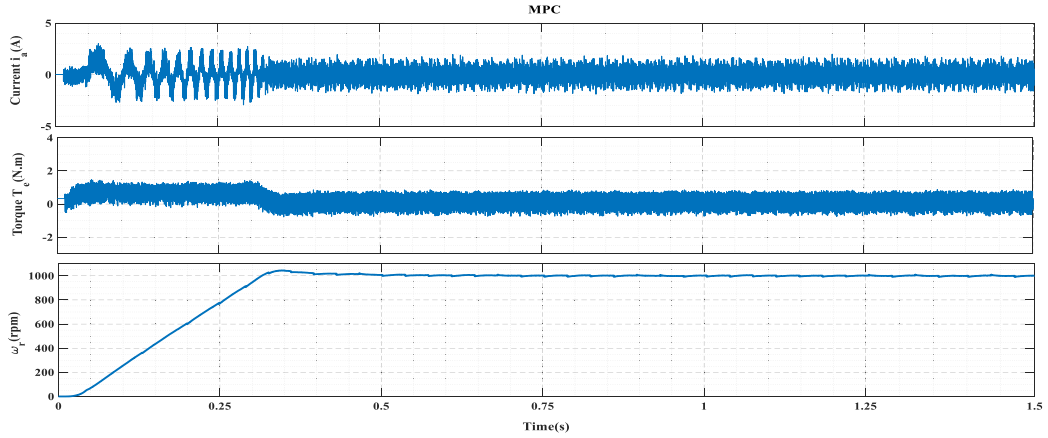
Fig. 3.13. Control circuit board.

3.5.2 Start-up and Load Tests

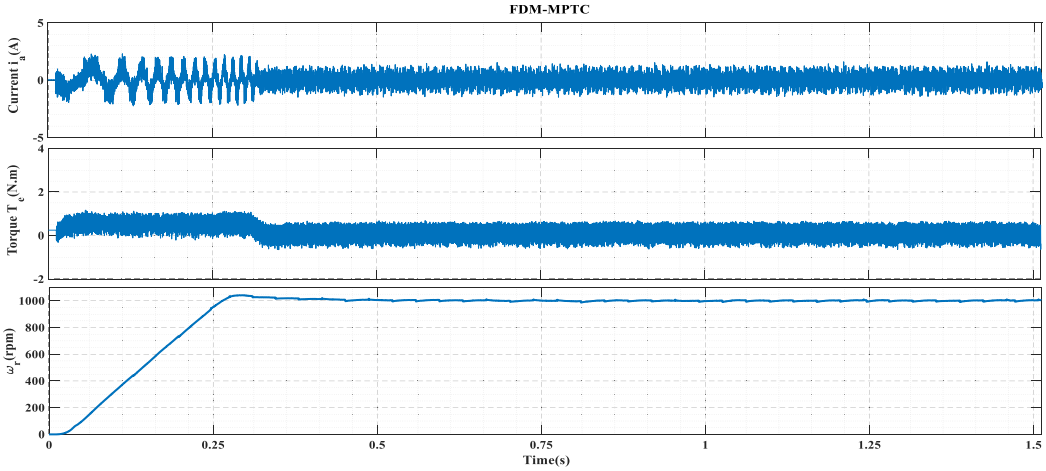
This section analyses the responses during start-up, deceleration, and in the presence of external load disturbances. The reference speed was set to 1000 rpm to examine the start-up behaviour, allowing the motor speed to accelerate steadily from a standstill to its rated speed of 1000 rpm. The response exhibits a slight overshoot, as depicted in Fig. 3.14. For the deceleration test, the motor operates at a speed of 500 rpm, then a reference speed of -1000 rpm is applied, and the obtained responses are depicted in Fig. 3.15. Furthermore, the load torque responses are investigated by operating the motor at a constant speed of 1000 rpm while subjecting it to a 2 Nm external load applied to the shaft. The corresponding responses to load disturbance are illustrated in Fig. 3.16. The curves from top to bottom in Figs 3.14 to 3.16 are stator current, torque, and rotor speed.

Upon analysis, it becomes evident that the proposed FDM-MPCC and FDM-MPTC exhibit superior dynamic performance compared to MPC. These methods showcase faster settling times, ensuring quicker stabilization of the motor speed. Moreover, they exhibit remarkable resilience against load disturbances. Compared to MPC, the motor speed quickly returns to its steady-state position following a disturbance, indicating excellent load disturbance rejection capabilities.

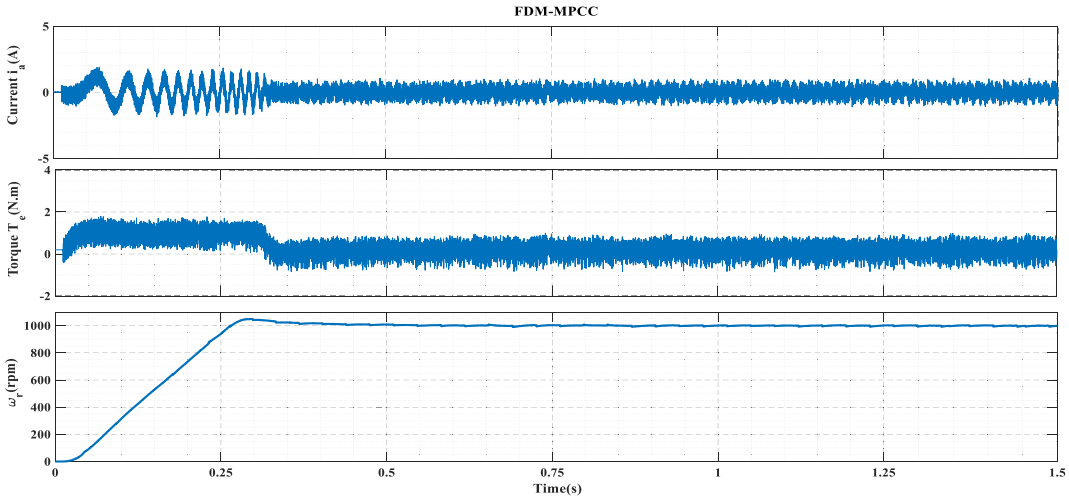
In summary, the FDM-MPCC and FDM-MPTC techniques demonstrate commendable dynamic performance and efficient load disturbance rejection, surpassing the performance of MPC in both start-up and external load scenarios.



(a)

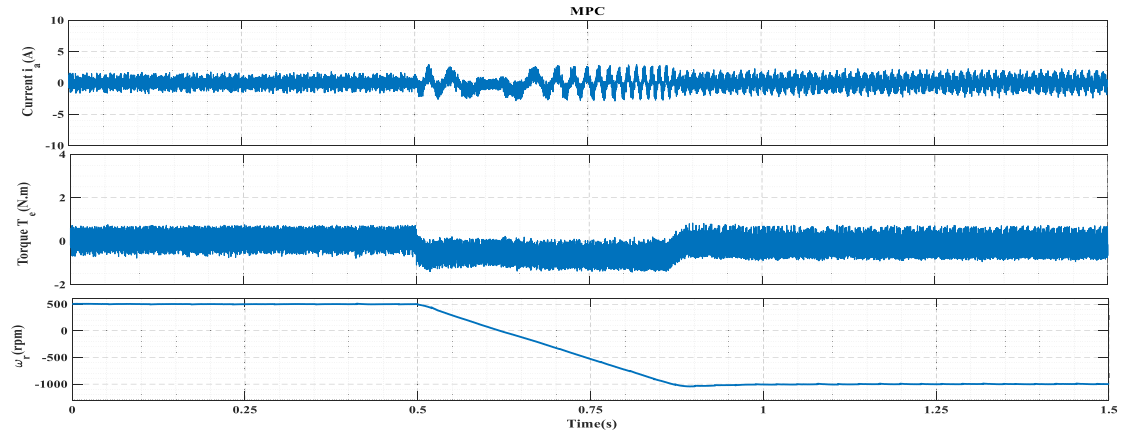


(b)

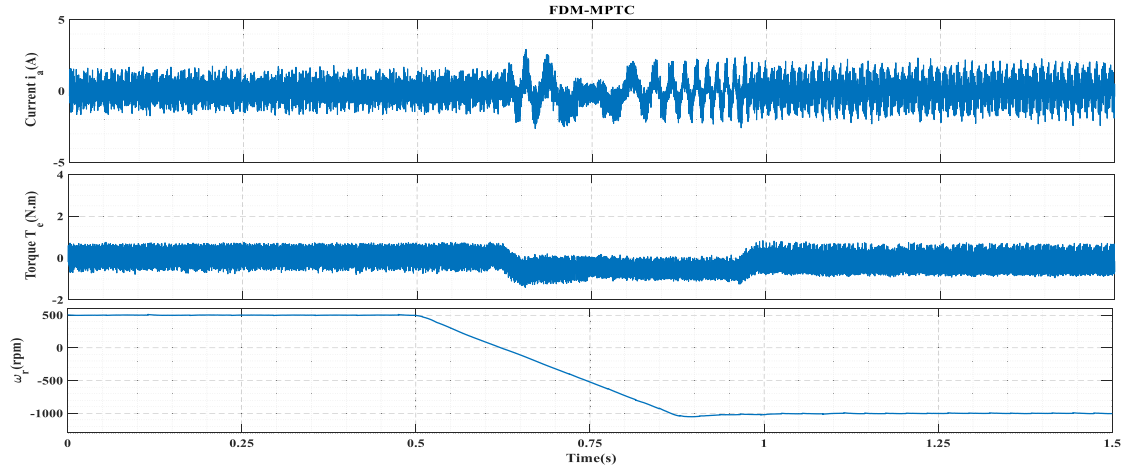


(c)

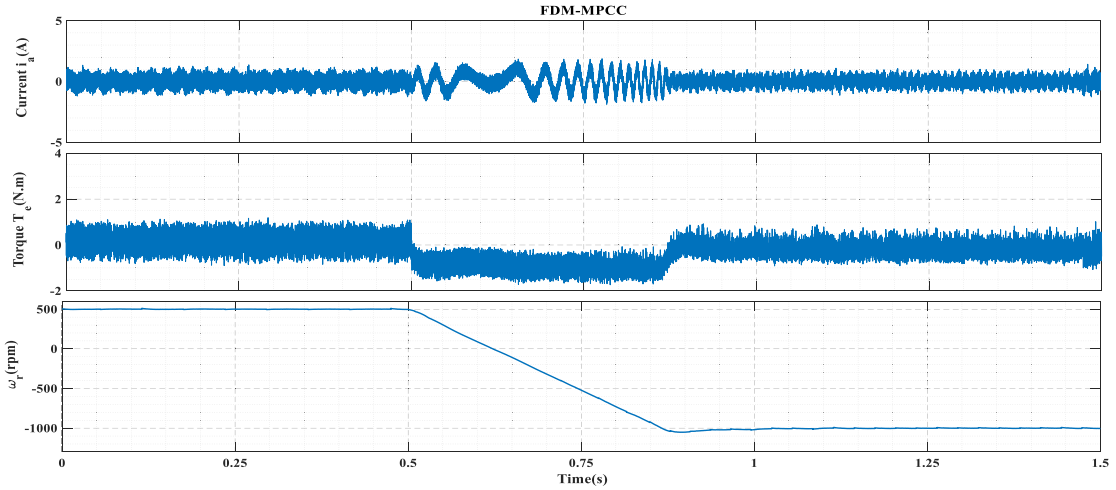
Fig. 3.14. Start-up response from standstill to 1000 rpm for (a) MPC, (b) FDM-MPTC, and (c) FDM-MPCC.



(a)

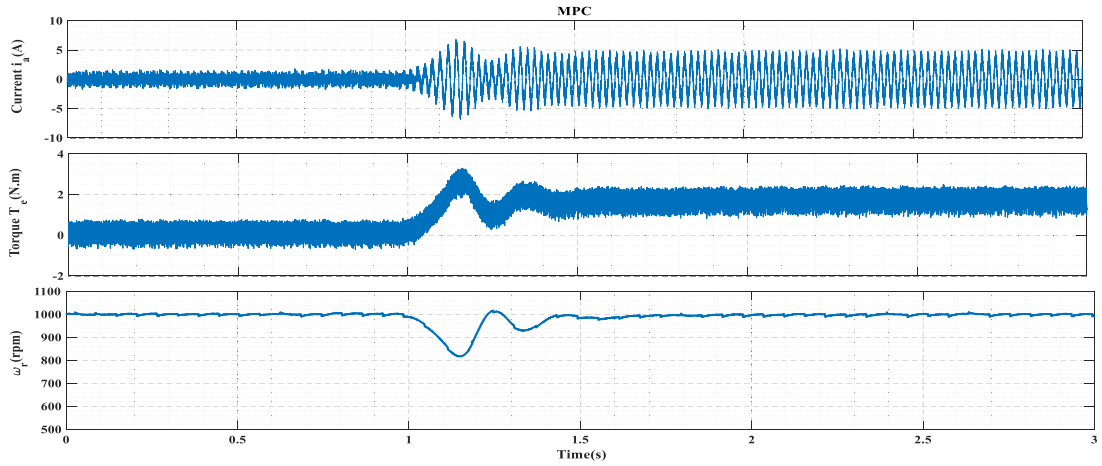


(b)

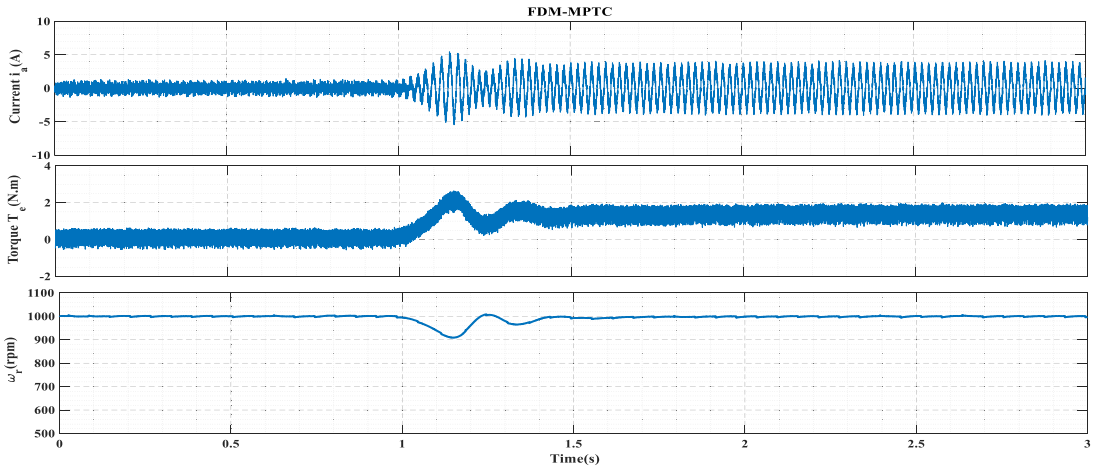


(c)

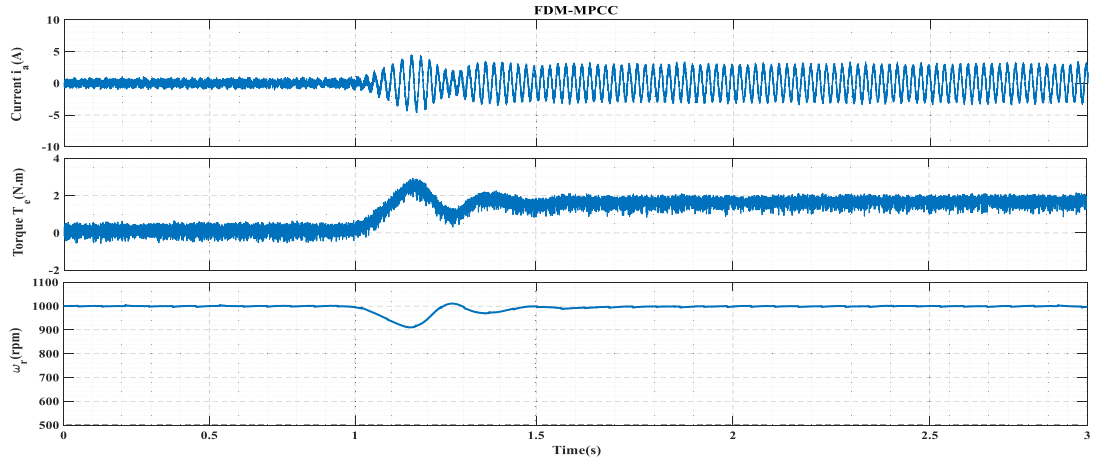
Fig. 3.15. Deceleration response from 500 to -1000 rpm for (a) MPC, (b) FDM-MPTC, and (c) FDM-MPCC.



(a)



(b)



(c)

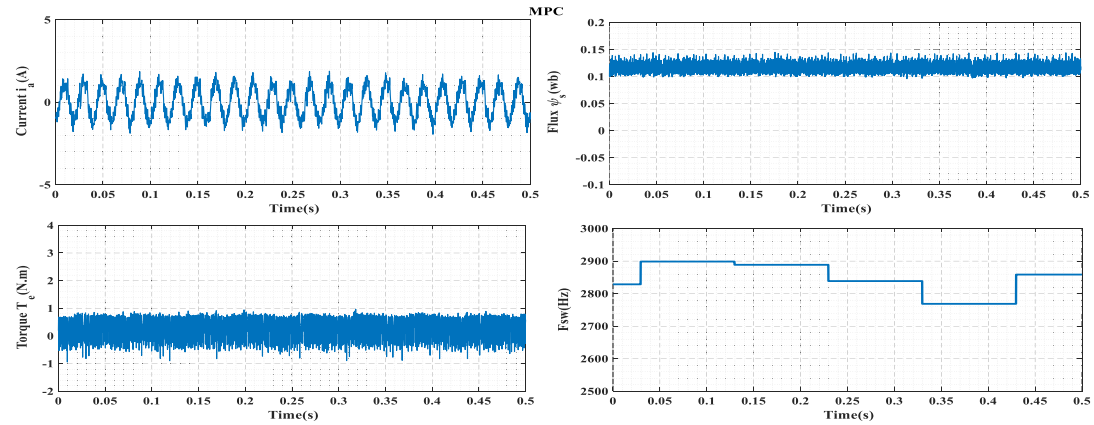
Fig. 3.16. Responses to external load disturbance (2 Nm) for (a) MPC, (b) FDM-MPTC, and (c) FDM-MPCC.

3.5.3 Steady-State Responses

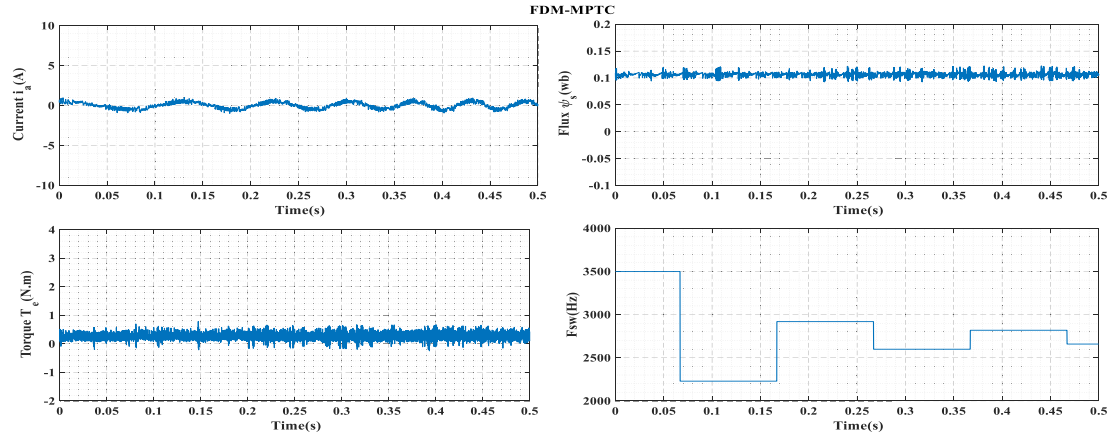
To delve deeper into the performance evaluation of the proposed FDM-MPTC and FDM-MPCC methods, an examination of their steady-state responses under various speed operations, both with and without load, is conducted. To highlight the effectiveness of these techniques, a performance comparison is conducted with the conventional MPC in different operating conditions. This evaluation uses two speeds (500 rpm and 1000 rpm) with and without load conditions.

Firstly, the responses at 500 rpm are analyzed to assess the system's behaviour. Fig. 3.17 illustrates the system's response without any load, while Fig. 3.18 depicts the response under a 2 Nm load torque. Furthermore, the responses at 1000 rpm are investigated to evaluate the performance of the FDM-MPTC and FDM-MPCC techniques at rated speed. Fig. 3.19 presents the system's response without any load, while Fig. 3.20 showcases the response under a load condition. The graphs in Figs 3.17 to 3.20 represent stator current, flux, torque, and applied switching vectors.

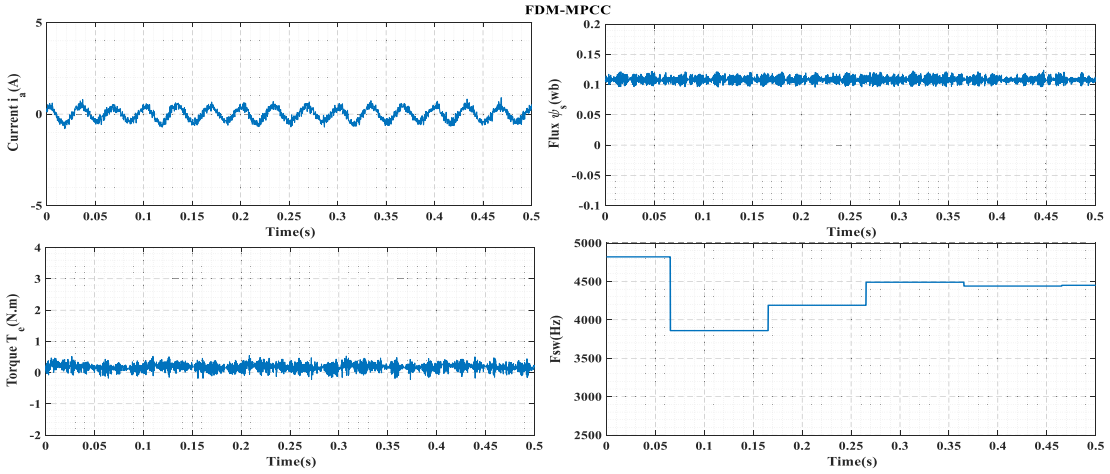
Analyzing these steady-state responses gives a more comprehensive understanding of the effectiveness of the proposed FDM-MPTC and FDM-MPCC methods compared to conventional MPC. The proposed methods exhibit excellent steady-steady torque, flux, and current responses with fewer ripples and distortions. This illustrates the significance of applying a two-vector and eliminating the weighting factor in enhancing the performance of the PMSM drive system.



(a)

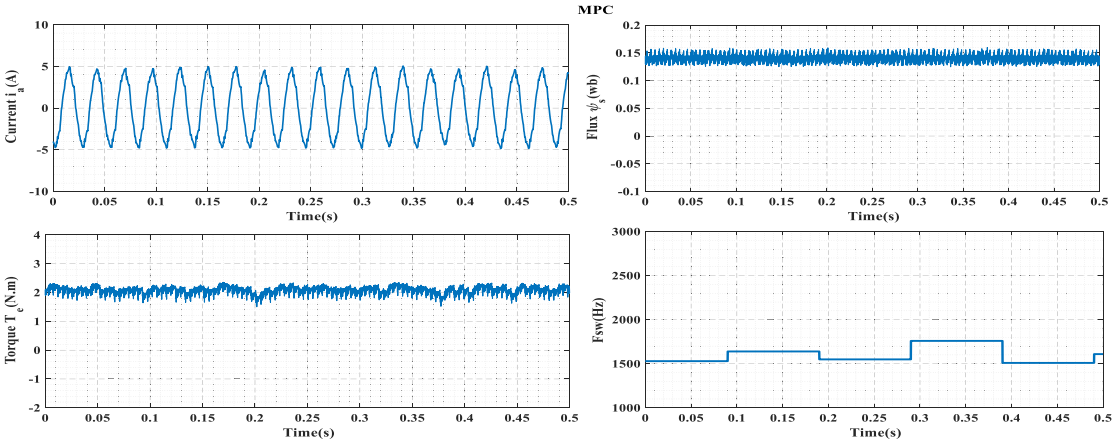


(b)

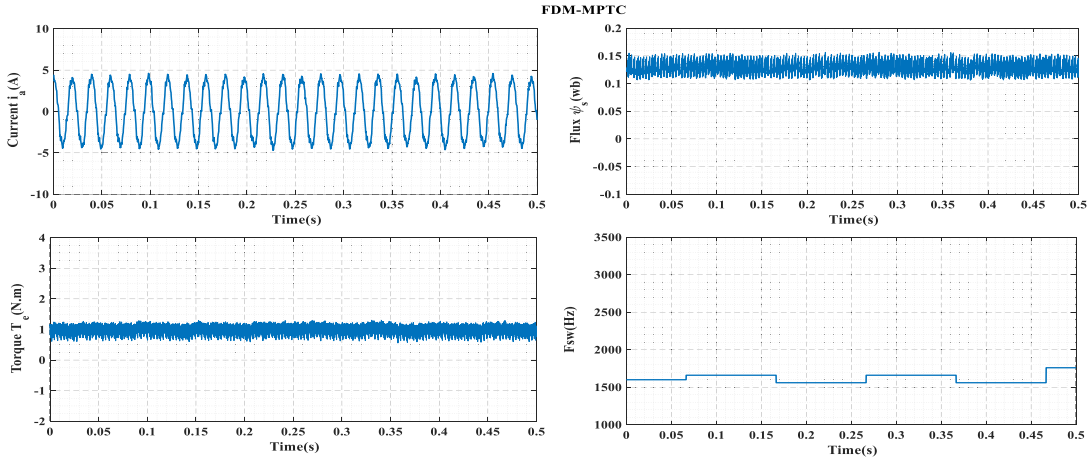


(c)

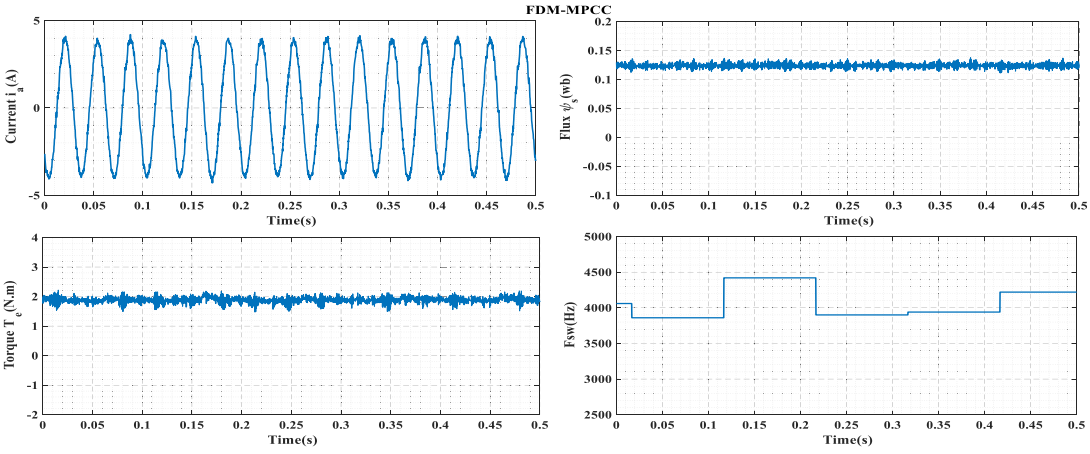
Fig. 3.17. Steady-state responses at 500 rpm (no load) for (a) MPC, (b) FDM-MPTC, and (c) FDM-MPCC.



(a)

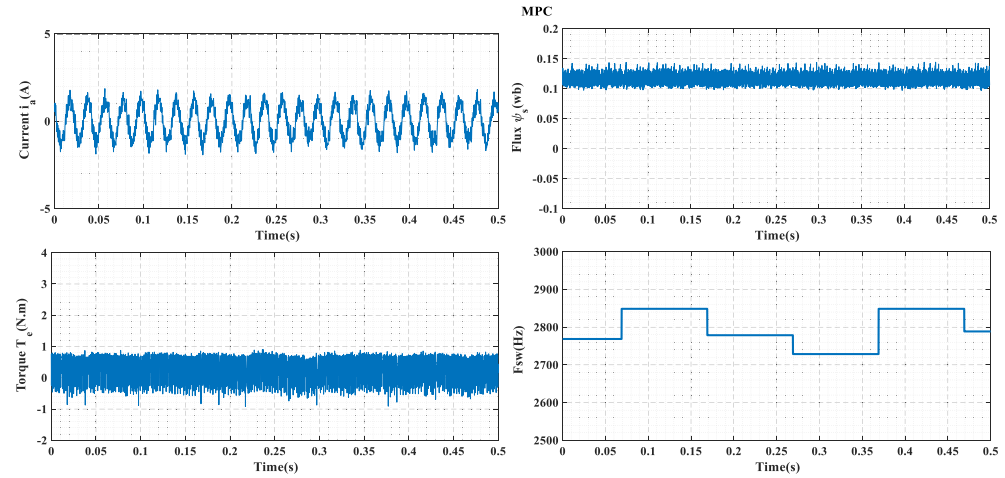


(b)

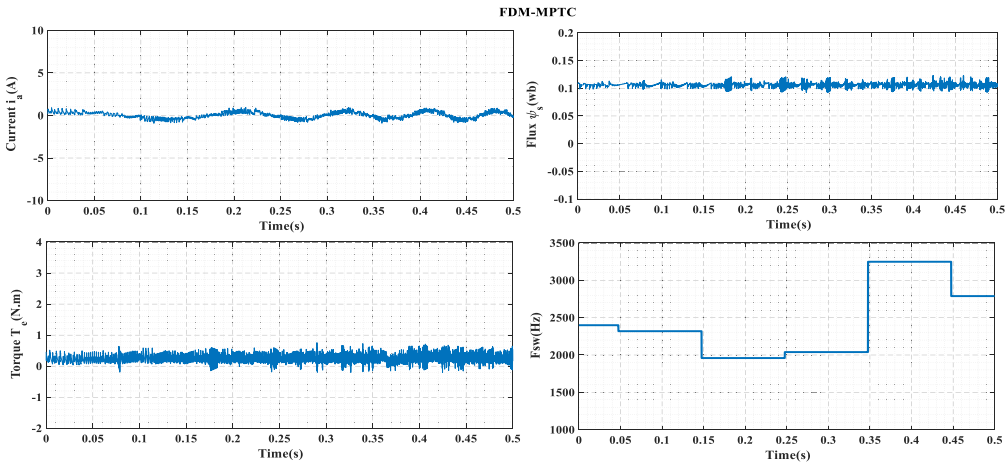


(c)

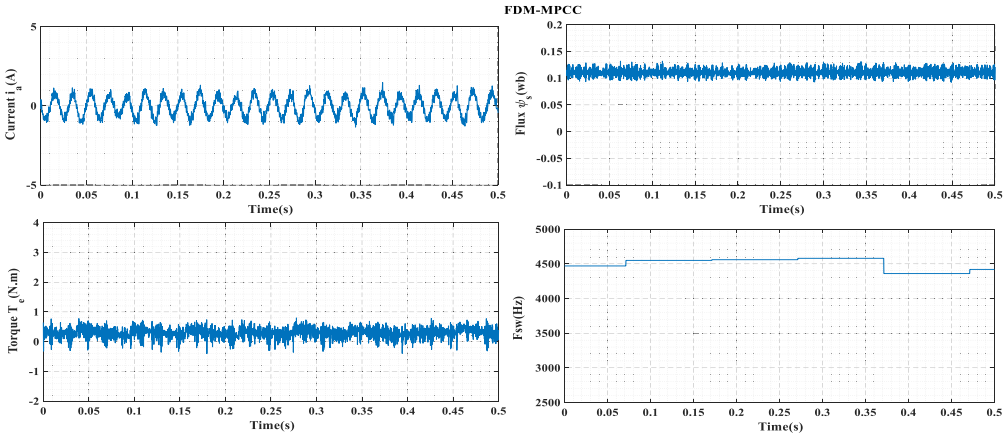
Fig. 3.18. Steady-state responses at 500 rpm (2 Nm load) for (a) MPC, (b) FDM-MPTC, and (c) FDM-MPCC.



(a)

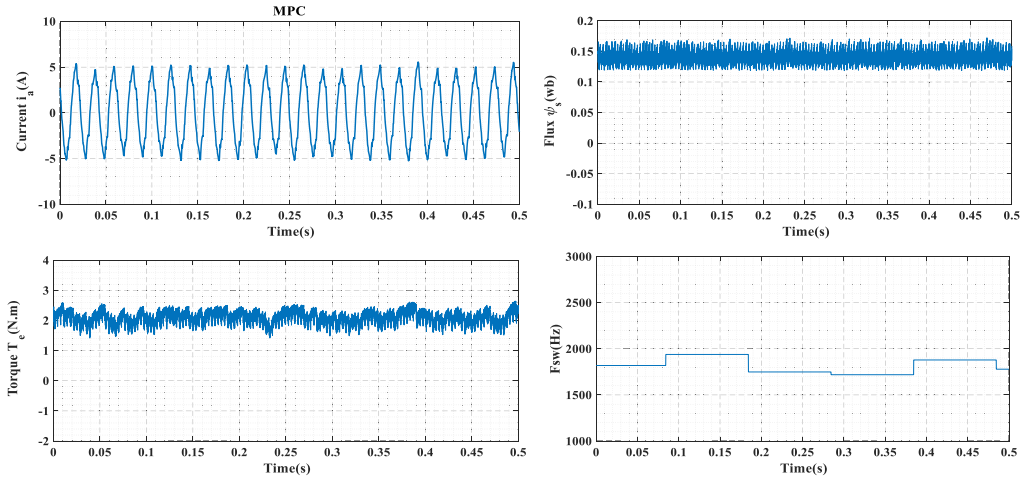


(b)

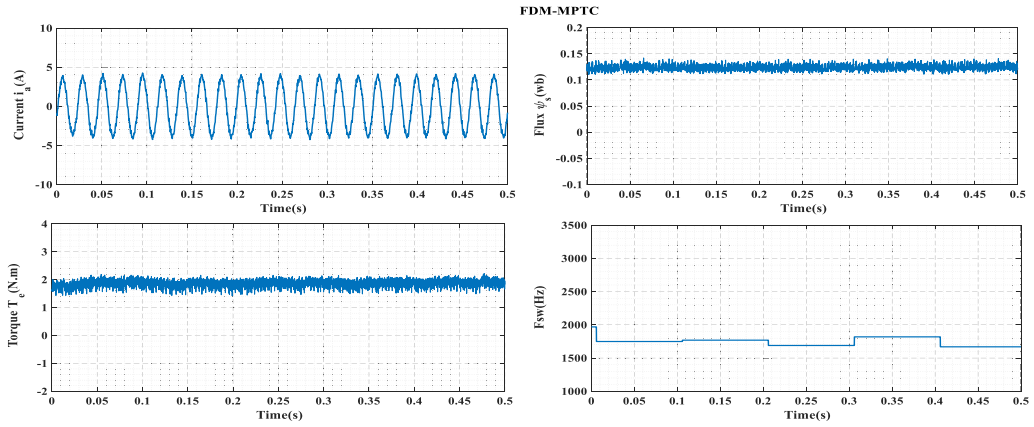


(c)

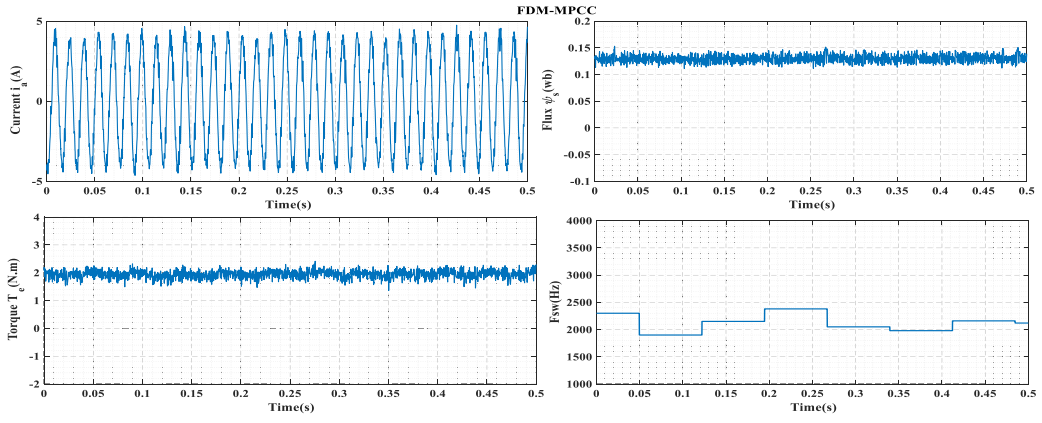
Fig. 3.19. Steady-state responses at 1000 rpm (no load) for (a) MPC, (b) FDM-MPTC, and (c) FDM-MPCC.



(a)



(b)



(c)

Fig. 3.20. Steady-state responses at 1000 rpm (2 Nm load) for (a) MPC, (b) FDM-MPTC, and (c) FDM-MPCC.

3.6 Quantitative Analysis and Comparison of Control Methods

The proposed FDM-MPTC and FDM-MPCC performances can be further evaluated under different operating conditions. The stator current and torque tracking performances and switching frequency at steady state with different speeds and torque are assessed. By varying the motor speed and applying different load torques, the current ripples, torque ripples, and average switching frequency are computed and recorded using the following formulas:

$$T_{rip} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(T_e(i) - T_{avg}\right)^2} \tag{3.52}$$

$$i_{rip} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(i_{a_{ref}}(i) - i_a\right)^2} \tag{3.53}$$

The average inverter switching frequency F_{avg} is obtained by counting the total switching jumps *N* of six legs of a two-level inverter over a fixed period of 0.05s.

$$F_{avg} = \frac{\frac{N}{6}}{\frac{00.05}{2}} \tag{3.54}$$

The current ripples i_{rip}, torque ripples T_{rip}, and average switching frequency F_{avg} of MPC, FDM-MPCC, and FDM-MPTC at three speeds with no load, 1 Nm load, and 2 Nm load are presented in Tables 3.2 to 3.10. The motor operates at a speed of (200, 500, or 1000) rpm, and the corresponding load is applied to the motor shaft. Then the load and current ripples are computed for a duration of 0.5 s. The average switching frequency is computed for the three controllers with a fixed duration of 0.05 s.

Table 3.2 Torque ripples of conventional MPC T_{rip} (Nm).

T_{rip} (Nm)		Speed (rpm)		
		200	**500**	**1000**
Torque (Nm)	**0**	0.16	0.29	0.43
	1	0.18	0.27	0.37
	2	0.15	0.25	0.36

Table 3.3 Torque ripples of FDM-MPCC T_{rip} (Nm).

T_{rip} (Nm)		Speed (rpm)		
		200	**500**	**1000**
Torque (Nm)	**0**	0.097	0.11	0.12
	1	0.075	0.091	0.11
	2	0.062	0.081	0.095

Table 3.4 Torque ripples of FDM-MPTC T_{rip} (Nm).

T_{rip} (Nm)		Speed (rpm)		
		200	**500**	**1000**
Torque (Nm)	**0**	0.061	0.082	0.092
	1	0.051	0.069	0.091
	2	0.048	0.064	0.084

Table 3.5 Current ripples of MPC i_{rip} (A).

i_{rip} (A)		Speed (rpm)		
		200	**500**	**1000**
Torque (Nm)	**0**	0.42	0.45	0.53
	1	0.48	0.53	0.56
	2	0.46	0.43	0.45

Table 3.6 Current ripples of FDM-MPCC i_{rip} (A).

i_{rip} (A)		Speed (rpm)		
		200	**500**	**1000**
Torque (Nm)	**0**	0.21	0.22	0.24
	1	0.23	0.27	0.28
	2	0.20	0.16	0.17

Table 3.7 Current ripples of FDM-MPTC i_{rip} (A).

i_{rip} (A)		Speed (rpm)		
		200	**500**	**1000**
Torque (Nm)	**0**	0.24	0.26	0.32
	1	0.27	0.31	0.35
	2	0.22	0.19	0.26

Table 3.8 Average switching frequency of MPC F_{avg} (Hz).

F_{avg} (Hz)		Speed (rpm)		
		200	**500**	**1000**
Torque (Nm)	**0**	2375.4	2228.1	1829.0
	1	2161.5	1985.5	1730.9
	2	2019.9	2029.7	1688.7

Table 3.9 Average switching frequency of FDM-MPCC F_{avg} (Hz).

F_{avg} (Hz)		Speed (rpm)		
		200	**500**	**1000**
Torque (Nm)	**0**	3924.0	3436.5	3199.1
	1	3813.2	3177.4	3024.0
	2	3369.9	3254.8	2976.4

Table 3.10 Average switching frequency FDM-MPTC F_{avg} (Hz).

F_{avg} (Hz)		Speed (rpm)		
		200	**500**	**1000**
Torque (Nm)	**0**	4217.4	3556.2	3321.0
	1	3934.8	3279.5	3161.7
	2	3449.8	3361.1	3189.3

The quantitative results in Tables 3.2 to 3.7 show that implementing the proposed FDM-MPCC and FDM-MPTC significantly reduced the torque and current ripples compared to the conventional MPC at different speed operation and load conditions. For instance, at 1000 rpm with a 2 Nm load, FDM-MPCC and FDM-MPTC reduced the torque ripples by 72.1% and 75.3% and the current ripples by 62.2% and 42.2%, respectively. Thus, it can be concluded that the performance of the proposed FDM-MPTC and FDM-MPCC is better than that of the conventional MPC in the high, medium, and low-speed and load regions.

In addition, based on the quantitative results, it appears that FDM-MPTC has the lowest torque ripples under all three operating conditions, with FDM-MPCC having the second lowest and MPC having the highest. Regarding current ripples, FDM-MPCC performs best under no load and 1 Nm and 2 Nm load conditions. The switching frequency is highest for FDM-MPTC under all three operating conditions and lowest for MPC. On the other hand, the trade-off between torque and current ripples is another essential aspect of comparing the controller's performance. For example, while FDM-MPTC has the lowest torque ripples, it has higher current ripples than FDM-MPCC.

3.7 Evaluation with Regulated Switching Frequency

The simulation, experimental tests, and quantitative analyses presented in previous sections were all conducted under the same control system sampling frequency. Due to the nature of the MPC algorithm, the switching frequencies of these control methods vary depending on the applied switching vectors. To achieve a fair comparison, the switching frequencies of these control methods should be maintained at a similar level. The quantitative analysis in Section 3.6 shows that the inverter's average switching frequency varies not only

with the control methods but also with the speed and load conditions. As a result, it will be challenging and complex to conduct all the tests under a similar inverter switching frequency. To thoroughly investigate the effectiveness of the proposed MPC methods and fairly compare it with conventional MPC, performance evaluation at rated conditions (1000 rpm and 2 Nm) is performed with the regulation of MPC switching frequency to be approximately at a similar level to the proposed method. To regulate the switching frequency of MPC, an additional cost function constraint is applied to reduce the number of switching commutations between two control cycles. The inverter switching states are represented as:

$$S = [S_a, S_b, S_c]^T \tag{3.54}$$

To regulate the switching frequency, the switching states change must be limited to no more than one change per control cycle. The change in switching states can be obtained by summing the switching states S , between two control cycles as follows:

$$\Delta S = \sum S(k+1) - S(k) \tag{3.55}$$

Then, a cost function constraint is included to limit the switching states change to no more than one per cycle as follows:

$$C_{sw}(k) = \text{Lim}(\Delta S) = \begin{cases} \infty & if\ \Delta S > 1 \\ 0 & if\ \Delta S \leq 1 \end{cases} \tag{3.56}$$

The constraint (C_{sw}) is included in the cost function evaluation in (3.28), which is rewritten as follows:

$$g_{MPCC} = \left(i_d{}^* - i_d(k+2)\right)^2 + \left(i_q{}^* - i_q(k+2)\right)^2 + w\, C_{sw}(k+2) \tag{3.57}$$

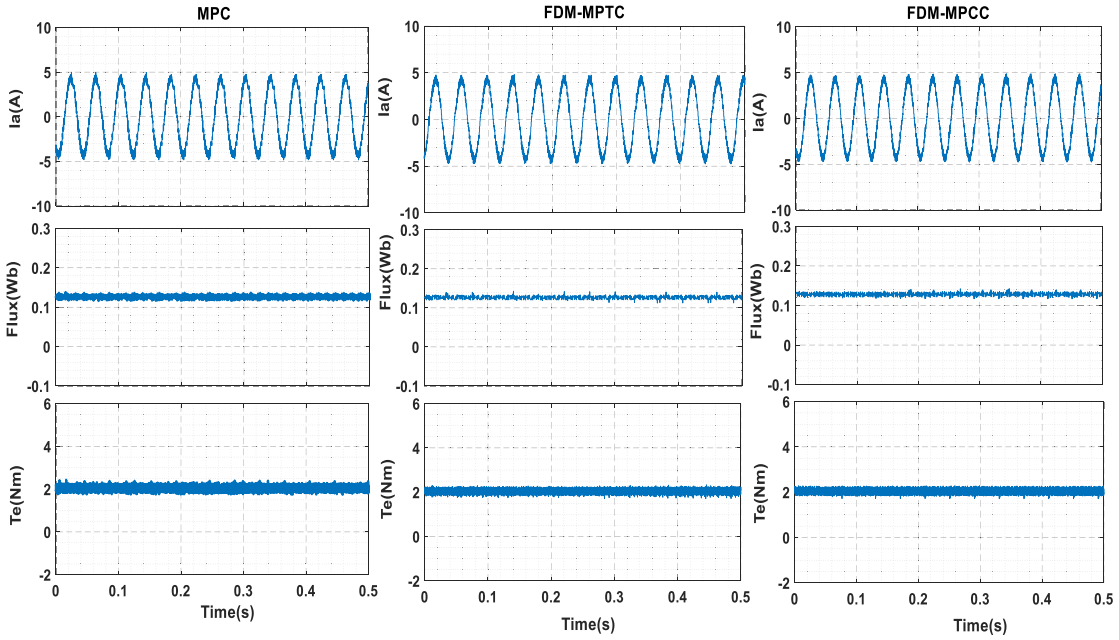
where w is a weighting factor.

The constraint C_{sw} will prevent the selection of a switching state with more than one switch change per cycle. Thus, a regulated and approximately fixed switching frequency can be obtained for MPC.

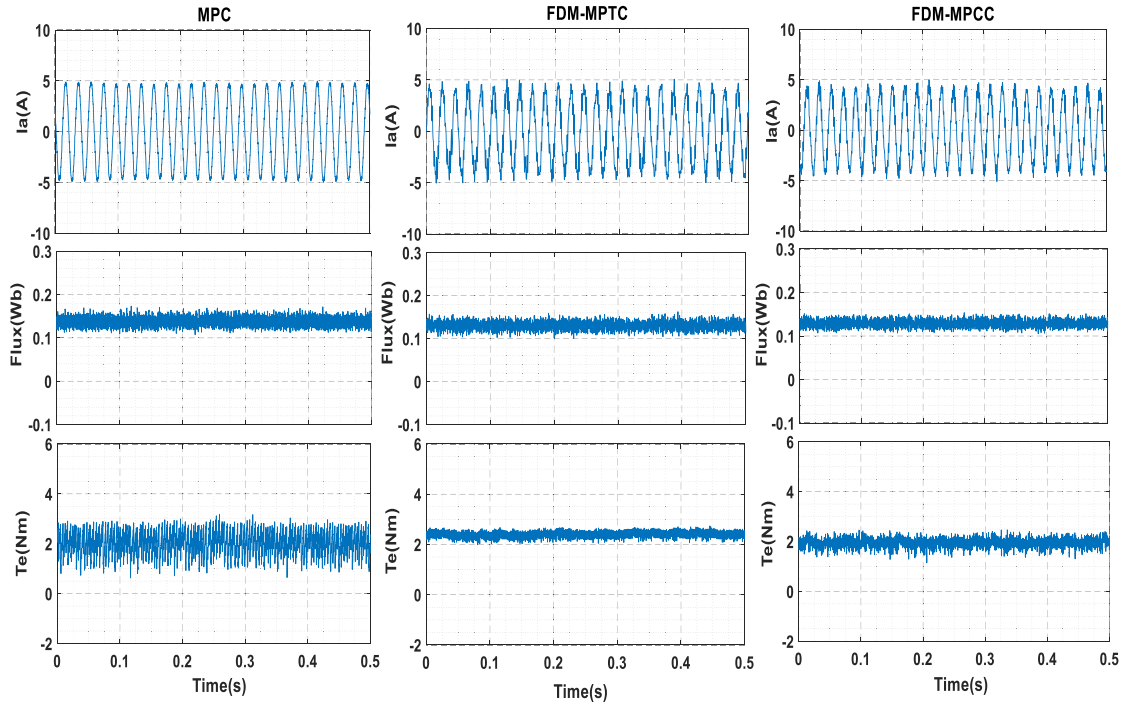
With a requlated switching frequency, the simulation results at the rated condition are obtained for the three controllers, as presented in Fig. 3.21(a). Similarly, the experimental tests with a requlated switching frequency are conducted at rated speed (1000 rpm) and torque (2 Nm), as shown in Fig. 3.21(b). It can be seen that the performance of the conventional MPC is improved with a fixed switching frequency. However, the performance of the proposed FDM-MPTC and FDM-MPCC is still superior, as confirmed by the quantitative data of torque ripples, current ripples, and current harmonics in Table 3.11.

Table 3.11 Quantitative comparison of MPC, FDM-MPTC, and FDM-MPCC with the same switching frequency.

Method	**Torque Ripple** T_{rip} **(Nm)**	**Current Ripple** i_{rip} **(A)**	**Current THD%**
MPC	0. 26	0.28	5.3%
FDM-MPCC	0. 094	0.17	4.1%
FDM-MPTC	0.086	0.24	4.6%



(a)



(b)

Fig. 3.21. Performance comparison of MPC, FDM-MPTC, and FDM-MPCC with regulated switching frequency, (a) simulation results, and (b) experimental results.

3.8 Summary

The unsatisfactory steady-state performance, unregulated switching frequency, and ambiguous process of weighting factor selection made conventional MPC methods less effective than conventional control methods for AC machine drives. Thus, various improvements, such as two or more vectors during one control cycle, weighting factor eliminations, and switching frequency regulation techniques, were implemented to maintain the effectiveness of MPCs. This chapter proposed two MPC methods based on predictive torque/flux and current controls to reduce the torque ripple, regulate switching frequency, and maintain good current quality. The proposed methods utilize two vectors for one control cycle and eliminate the issue of weighing factors using FDM. Compared with the conventional MPCs, the proposed methods have shown superiority in terms of different characteristics under transient and steady-state conditions. The proposed FDM-MPCC has shown better current response and maintained reduced current ripples compared to the proposed FDM-MPTC.

REFERENCES

[3.1] Z. Ma, S. Saeidi, and R. Kennel, "FPGA implementation of model predictive control with constant switching frequency for PMSM drives," *IEEE Transactions on Industrial Informatics,* vol. 10, no. 4, pp. 2055-2063, 2014.

[3.2] Z. Mynar, L. Vesely, and P. Vaclavek, "PMSM model predictive control with field-weakening implementation," *IEEE Transactions on Industrial Electronics,* vol. 63, no. 8, pp. 5156-5166, 2016.

[3.3] T. Englert, S. Grüner, and K. Graichen, "Model predictive torque control of permanent magnet synchronous machines," *IFAC-PapersOnLine,* vol. 50, no. 1, pp. 758-763, 2017.

[3.4] W. Wang, Z. Lu, W. Hua, Z. Wang, and M. Cheng, "Simplified model predictive current control of primary permanent-magnet linear motor traction systems for subway applications," *Energies,* vol. 12, no. 21, p. 4144, 2019.

[3.5] Y. Zhang, D. Xu, J. Liu, S. Gao, and W. Xu, "Performance improvement of model-predictive current control of permanent magnet synchronous motor drives," *IEEE Transactions on Industry Applications,* vol. 53, no. 4, pp. 3683-3695, 2017.

[3.6] H.-T. Moon, H.-S. Kim, and M.-J. Youn, "A discrete-time predictive current control for PMSM," *IEEE Transactions on Power Electronics,* vol. 18, no. 1, pp. 464-472, 2003.

[3.7] P. Cortes, J. Rodriguez, C. Silva, and A. Flores, "Delay compensation in model predictive current control of a three-phase inverter," *IEEE Transactions on Industrial Electronics,* vol. 59, no. 2, pp. 1323-1325, 2011.

[3.8] Z. Yongchang and W. Xianglong, "Torque ripple RMS minimization in model predictive torque control of PMSM drives," in *2013 International Conference on Electrical Machines and Systems (ICEMS)*, pp. 2183-2188, 26-29 Oct. 2013.

[3.9] P. Cortes *et al.*, "Guidelines for weighting factors design in Model Predictive Control of power converters and drives," in *2009 IEEE International Conference on Industrial Technology*, pp. 1-7,10-13 Feb. 2009, doi: 10.1109/ICIT.2009.4939742.

[3.10] C. A. Rojas, J. Rodriguez, F. Villarroel, J. Espinoza, and D. A. Khaburi, "Multiobjective Fuzzy Predictive Torque Control of an induction motor drive," in *The 6th Power Electronics, Drive Systems & Technologies Conference (PEDSTC2015)*, pp. 201-206, 3-4 Feb. 2015, doi: 10.1109/PEDSTC.2015.7093274.

[3.11] P. Zanchetta, "Heuristic multi-objective optimization for cost function weights selection in finite states model predictive control," in *2011 Workshop on Predictive Control of Electrical Drives and Power Electronics*, pp. 70-75, 14-15 Oct. 2011, doi: 10.1109/PRECEDE.2011.6078690.

[3.12] M. Mamdouh, M. A. Abido, and Z. Hamouz, "Weighting Factor Selection Techniques for Predictive Torque Control of Induction Motor Drives: A Comparison Study," *Arabian Journal for Science and Engineering,* vol. 43, no.2, pp. 433-445, 2018/02/01 2018, doi: 10.1007/s13369-017-2842-2.

[3.13] X. Zhang and B. Hou, "Double Vectors Model Predictive Torque Control Without Weighting Factor Based on Voltage Tracking Error," *IEEE Transactions on Power Electronics,* vol. 33, no. 3, pp. 2368-2380, 2018.

[3.14] I. Jlassi and A. J. M. Cardoso, "Lookup-Table-Based Model Predictive Torque Control Without Weighting Factors for PMSM Drives," in *IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society*, vol. 1, pp. 1165-1170, 14-17 Oct. 2019.

[3.15] K. Cengiz, O. Sezi Cevik, and O. Basar, "Fuzzy Multicriteria Decision-Making: A Literature Review," *International Journal of Computational Intelligence Systems,* vol. 8, no. 4, pp. 637-666, 2015, doi: https://doi.org/10.1080/18756891.2015.1046325.

[3.16] J. M. d. C. Sousa and U. Kaymak, "Model predictive control using fuzzy decision functions," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics),* vol. 31, no. 1, pp. 54-65, 2001, doi: 10.1109/3477.907564.

[3.17] F. Villarroel, J. R. Espinoza, C. A. Rojas, J. Rodriguez, M. Rivera, and D. Sbarbaro, "Multiobjective Switching State Selector for Finite-States Model Predictive Control Based on Fuzzy Decision Making in a Matrix Converter," *IEEE Transactions on Industrial Electronics,* vol. 60, no. 2, pp. 589-599, 2013, doi: 10.1109/TIE.2012.2206343.

[3.18] C. A. Rojas, J. R. Rodriguez, S. Kouro, and F. Villarroel, "Multiobjective Fuzzy-Decision-Making Predictive Torque Control for an Induction Motor Drive," *IEEE Transactions on Power Electronics,* vol. 32, no. 8, pp. 6245-6260, 2017, doi: 10.1109/TPEL.2016.2619378.

[3.19] V. Kumar, P. Gaur, and A. P. Mittal, "Finite-state model predictive control of NPC inverter using multi-criteria fuzzy decision-making," *International Transactions on Electrical Energy Systems,* vol. 25, no. 5, pp. 876-897, 2015, doi: 10.1002/etep.1880.

[3.20] H. Mahmoudi, M. Aleenejad, P. Moamaei, and R. Ahmadi, "Fuzzy adjustment of weighting factor in model predictive control of permanent magnet synchronous machines using current membership functions," in *2016 IEEE Power and Energy Conference at Illinois (PECI)*, pp. 1-5, 19-20 Feb. 2016, doi: 10.1109/PECI.2016.7459225.

CHAPTER 4

A NOVEL ROBUSTNESS EVALUATION METHOD BASED ON SIX-SIGMA METHODOLOGY FOR PREDICTIVE CONTROL OF PMSM DRIVES

4.1 Introduction

Several robust predictive controls (RPCs) for PMSMs have been investigated with different robustness techniques, like prediction error correction. The prediction error is included in the prediction stage to compensate for any control effort [4.1-4.2]. Besides, observers are employed to establish an RPC method, where a specific type of observer is used to deal with uncertainties, such as disturbance observer [4.3], extended state observer (ESO) [4.4], and sliding mode observer [4.5]. In addition, model-free control is another technique to achieve RPCs for PMSMs, where the prediction process is independent of the machine model and parameters. Using ultra-local models, estimating the system unknowns [4.6], and using the current differences at different samples [4.7] are some types of model-free RPC for PMSM drives. Moreover, RPCs can be achieved by modifying the cost function to include several constraints and objectives [4.8-4.9]. In addition, combining predictive control with other control techniques can form a variety of RPC methods, such as MPC with deadbeat (DB) solution [4.10], integral-resonant control [4.11], and repetitive control [4.12].

However, there is a lack of discussion on the control system's uncertainties and robustness fundamentals. Most existing RPCs do not adequately describe uncertainties and their effect on performance. The robustness to uncertainties differs from one control method to another,

and yet no systemic procedure to evaluate and quantify the robustness. Moreover, PMSM uncertainties (e.g., parametric uncertainty) can be estimated to be within a bounded range depending on the machine structure (e.g., manufacturing tolerances) and the expected operating conditions (e.g., rated and maximum temperature). Therefore, to effectively evaluate a robust control method, realistic and practical uncertainty ranges must be considered instead of random uncertainty values, such as a 200% error in specific parameters, which is unrealistic and unlikely to occur in practical situations.

This chapter presents a clear and systemic method for evaluating the control system's robustness. The concept of robust control and the effect of uncertainties on performance are illustrated. Realistic and practical uncertainty ranges based on manufacturing and operational sources are obtained. The six-sigma methodology is used to evaluate the robustness of a control system, including second-order, DC motor drive, and RPC of PMSM drives.

4.2 Uncertainties and Robustness Fundamentals

To design a control system for a specific real plant, an approximate mathematical model that represents the plant needs to be obtained. Thus, various controllers can be designed according to the system requirements. However, the model never accurately describes the dynamic behaviour of the real plant, where some plant dynamics are not captured in the model [4.13]. Moreover, the system parameters typically are fixed and estimated/ measured at specific operating conditions. In the real system, these parameters are subjected to change in response to the system's operational, environmental, and/or structural variations. Therefore, the plant missing (unmodelled) dynamics and system parameters variation normally are referred to as control system uncertainties [4.14]. Most control system

uncertainties are generated due to plant structure differences, materials diversity, assembly imperfection, friction and mechanical factors, and environmental and operating condition changes. The sources of uncertainties can be grouped into manufacturing and operational sources. For instance, uncertainties can be generated due to materials diversity caused by manufacturing tolerances or operating temperature variations [4.15]. These uncertainties can highly degrade the system and may lead to instability. Thus, the design of a control method that deals with uncertainties was considered, and this control method is referred to as robust control.

Since the late 1970s, robustness has become a primary objective of control research, and numerous control methods have been proposed as robust controls of processes with uncertainties. A control system is typically considered robust if a good control performance is achieved in the presence of uncertainties. However, a control system's robustness definition and robustness level have not been clearly described. Thus, clearly defining the term robustness is desirable and proposing a practical approach to evaluate the robustness level of a control system.

To evaluate the effects of uncertainties on the system performance, the uncertainties need to be determined and represented mathematically. For example, parametric uncertainty can be represented and quantified by assuming that each uncertain parameter is bounded within some range [*min max*]. By considering a control system shown in Fig. 4.1 that has an uncertain parameter X_i bounded in the range $X_{min} \leq X_i \leq C_{max}$. The uncertain parameter X_i can be modelled as:

$$X_i = \bar{X}(1 + r_x\Delta) \tag{4.1}$$

where $\bar{X} = \frac{X_{min}+X_{max}}{2}$ is the mean parameter (nominal) value; $r_x = \frac{X_{max}-X_{min}}{X_{max}+X_{min}}$ is the relative uncertainty in the parameter, and Δ any real scalar satisfying $|\Delta| \leq 1$. The parametric uncertainty (G_p) can be written in a multiplicative form as follows:

$$G_p = \frac{\bar{x}}{\tau s + 1}(1 + r_x \Delta), |\Delta| \leq 1 \tag{4.2}$$

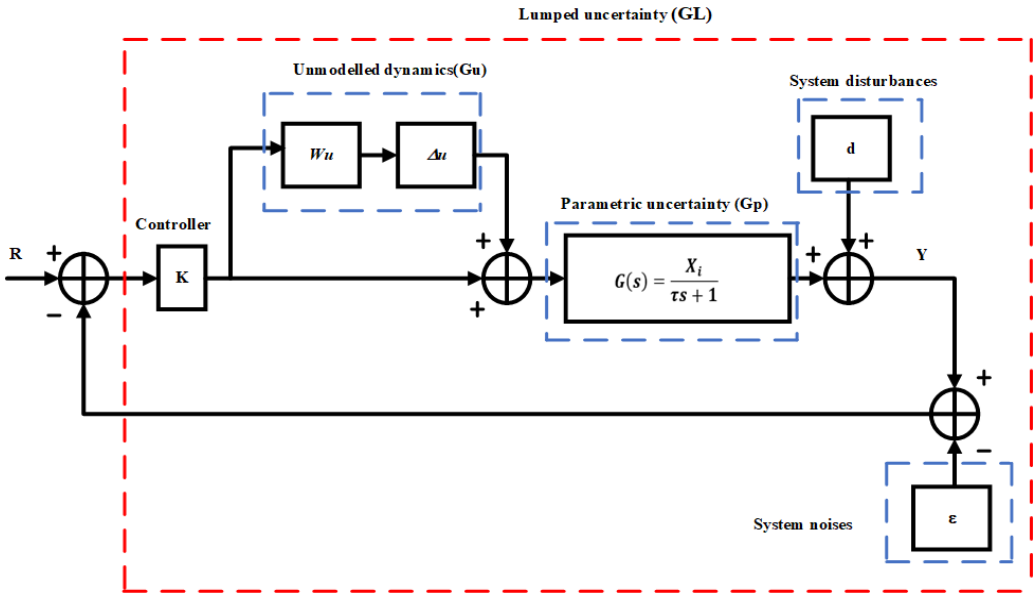


Fig. 4.1. Perturbed uncertain system.

Besides, to represent the unmodelled dynamics (Δu) mathematically with a weight (W_u) for the system shown in Fig. 4.2. The unmodelled dynamics uncertainty (G_u) can be expressed in multiplicative form as:

$$G_u = G(s)\big(1 + W_u(s)\Delta_u(s)\big), |\Delta_u(j\omega)| \leq 1 \; \forall \omega \tag{4.3}$$

Lumped uncertainty represents one or several sources of parametric and/or unmodelled dynamics uncertainty combined into single lumped perturbation of a chosen

structure. For instance, parametric and unmodelled dynamics uncertainties can be modelled as a single uncertatinty as follows:

$$G_p G_u = \frac{\bar{x}}{\tau s + 1}(1 + r_x \Delta)\big(1 + W_u(s)\Delta_u(s)\big), |\Delta| \leq 1, |\Delta_u(j\omega)| \leq 1 \; \forall \omega \tag{4.4}$$

Moreover, external disturbances and measurement noises are also primary sources of performance degradation. The lumed uncertainy (G_L) of the perturbed uncertain system, as shown in Fig. 4.1, considering the uncertain parameter X_i, controller (K), disturbances (d), measurement noise (ε), and unmodelled dynamics Δu can be represented as follows:

$$G_L = K\left(\frac{\bar{x}}{\tau s + 1}(1 + r_x \Delta)\big(1 + W_u(s)\Delta_u(s)\big)\right) + d - \varepsilon \tag{4.5}$$

The robustness of a control system is a measure of how well it can perform in the presence of uncertainties or how sensitive it is to system uncertainties. In other words, a control system (Fig. 4.1) with measured output (Y), uncertainties (U_i), the robustness of the control system depends on how sensitive the output Y to the uncertainties U_i. This can be expressed mathematically as the differential sensitivity ($S_{U_i}^{Y}$) of Y with respect to the uncertainties U_i , which is the percentage change in Y divided by the percentage change in U_i that has caused the change in Y to occur, as the following:

$$S_{U_i}^{Y} = \frac{\partial Y}{\partial U_i} = \frac{\Delta}{\Delta U_i} \tag{4.6}$$

Moreover, the effects of various uncertainties on a control system can be evaluated by considering the system in Fig. 4.1 with output Y having a number of uncertain elements ($U_1 \rightarrow U_i$). These uncertain elements include parametric and unmodelled dynamics uncertainties, disturbances, and noises. When these elements vary (ΔU_i) (uncertainties

occur), the system output Y is affected and consequently varied, which can be reflected as (ΔY). Thus, the effect on the system output Y due to the occurrence of uncertainties(ΔU_i) can be represented as incremental changes (derivative dy) and mathematically expressed as:

$$dy = \sum_{i=1}^{n} U_i Y_i dU_i \quad \text{for } n \text{ uncertainties} \tag{4.7}$$

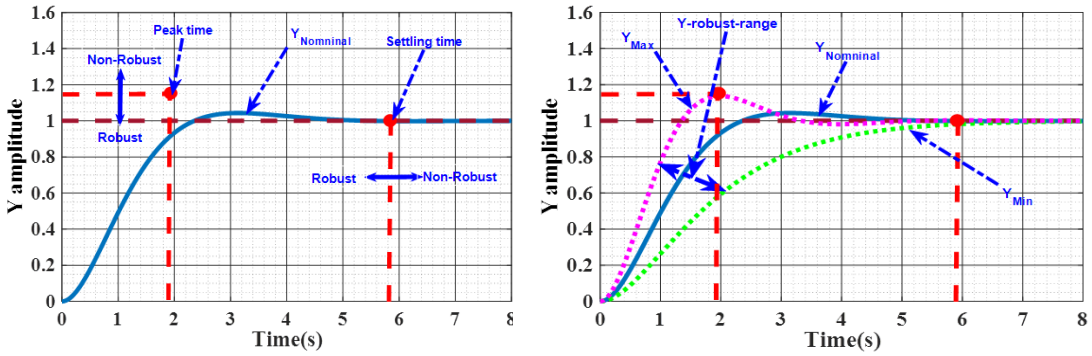
Thus, a small value of (dy) indicates high system robustness.

To illustrate the meaning of robustness graphically, the robustness of the control system in Fig. 4.1 can be described by its ability to have one or more properties Y within predetermined bounds in the presence of unknown parts or uncertain parameters. To obtain the robustness range of a control system performance, the nominal performance has to be obtained first. Then, the maximum and minimum bounds for robustness can be defined from the nominal point. Suppose the system Y has a nominal output Y_0 where there are no uncertainties. Thus, the range for robustness would be within the bounds$[Y_{min}, Y_{max}]$. For any system output Y_i to be robust, it should be within the bounds such that:

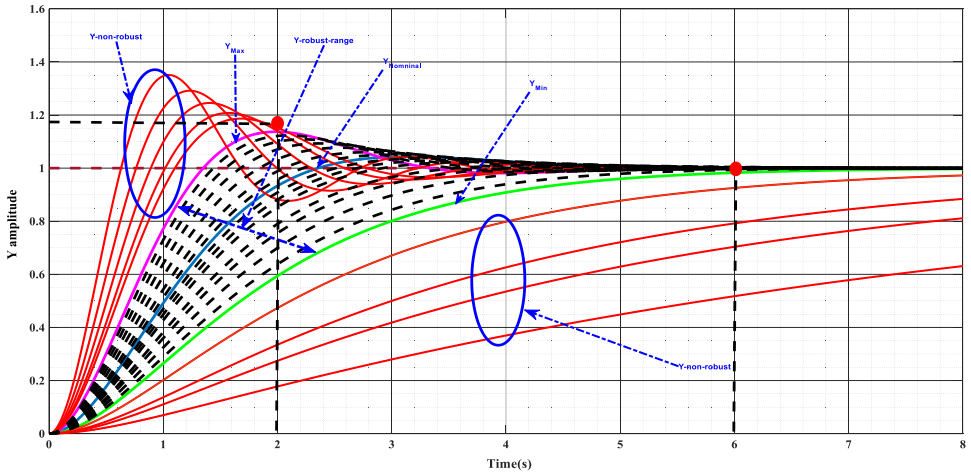
$$Y_i \xrightarrow{robust} if \;\; Y_{min} \leq Y_i \leq Y_{max} \tag{4.8}$$

Any output Y_i that is outside the pre-defined bounds $[Y_{min}, Y_{max}]$ is considered non-robust. To illustrate the concept, let's consider the control system (Fig. 4.1) with plant $G(s) = \frac{X_i}{0.5s+1}$. If the uncertain parameter x_i of plant $G(s)$ has a nominal value of 4 and is varied in the range [1, 7], and other uncertainties also occurred to the system. First, the nominal response $(Y_{nominal)}$ is generated $(X_{nominal} = 4)$, and no other uncertainties exist, as shown in Fig. 4.2(a). Then, the robust performance range is defined $[Y_{min}, Y_{max}]$, for instance, overshoot must be less than 15% (Y_{max}) and settling time must be less than 7s (Y_{max}) as

depicted in Fig. 4.2(b). Then, a set of step responses (Y_i) with different uncertainties generated. The responses that fall within the pre-defined bounds $[Y_{min}, Y_{max}]$ are considered robust responses (Y_{robust}), and this is called the robustness range ($Y_{robust-range}$). While the responses that fall outside the bounds range are considered non-robust ($Y_{non-robust}$) as shown in Fig. 4.2(c).



(a) (b)



(c)

Fig. 4.2. Feedback control system robustness illustration (a) nominal performance, (b) nominal and robust performance range, and (c) nominal, robustness ranges and non-robust performance.

4.3 PMSM Drive Uncertainties

In order to design an efficient control strategy for PMSMs, an accurate mathematical model of the machine needs to be developed. A three-phase PMSM with symmetrical winding and identical parameters can be transformed into a *dq* equivalent model. The voltage and flux linkage equations of a PMSM in *d-q* form are expressed as:

$$v_d = R_s i_d + L_d \frac{di_d}{dt} - \omega L_q i_q \tag{4.9}$$

$$v_q = R_s i_q + L_q \frac{di_q}{dt} + \omega L_d i_d + \omega \Psi_{pm} \tag{4.10}$$

$$\Psi_d = L_d i_d + \Psi_{pm} \tag{4.11}$$

$$\Psi_q = L_q i_q \tag{4.12}$$

$$T_m = T_L + J \frac{d\omega}{dt} + B\omega \tag{4.13}$$

$$T_e = \frac{3}{2} p\left[\Psi_{pm} i_q + \left(L_d - L_q\right) i_q i_d\right] \tag{4.14}$$

Typically, a PMSM control method is designed based on the developed model. However, PMSM nonlinear behavior and complex dynamics result in various unmodelled dynamics of the real PMSM. These dynamics are unavoidable in the real system, exist in different parts of the machine drive system, and are rarely captured in the model. Due to the material diversity of PMSM rotor, the machine structure can exhibit unmodelled dynamics in the form of induced pulsating torques such as cogging torque [4.16] and flux harmonics due to the PM material of PMSM. The machine's PM can be demagnetized with temperature rise, significantly affecting the maximum torque capability and PMSM efficiency [4.17].

In addition, the nonlinear operation of the inverter can generate unmodelled dynamics in the form of a deadtime effect. A voltage-fed inverter is never ideal; in practice, the switching dead time, the device's ON-state voltage drop, and the dc-bus voltage variations can adversely affect the control performance, particularly during the steady-state operation [4.18]. Besides, measurement errors due to sensors offset are another source of unmodelled dynamics, where the errors in measurements of either position or current inevitably cause torque ripples.

To design MPC for PMSM drives, the current at $(k+1)$ can be predicted based on the machine equations as follows:

$$i_d(k+1) = i_d(k) - \left(\frac{R_s T_s}{L_d}\right) i_d(k) + \frac{L_q}{L_d} T_s\, \omega\, i_q(k) + \frac{T_s}{L_d} v_d \tag{4.15}$$

$$i_q(k+1) = i_q(k) - \left(\frac{R_s T_s}{L_q}\right) i_q(k) - \frac{L_d}{L_q} T_s\, \omega\, i_d(k) - \frac{T_s \omega \Psi_{pm}}{L_q} + \frac{T_s}{L_q} v_d \tag{4.16}$$

The MPC of PMSM drives is subjected to uncertainties from manufacturing (e.g., assembly imperfection, PM material diversity) and operational (e.g., temperature variations, measurement offset), resulting in unmodelled dynamics and parametric uncertainties[4.16]. Parametric uncertainty due to machine parameter variations is the most common PMSMs drive uncertainty and severely affects performance [4.19]. Hence, this research considered only parametric uncertainty.

Generally, when designing a PMSM drive, the nominal values of both electrical parameters (R_s, L_d, L_q, Ψ_{pm}), and mechanical parameters (J, B) are used. However, these values can vary during operations due to temperature changes, load torque, and magnetic saturation. The mismatch or variation of any of the machine parameters will lead to an error

in the predicted variable. For instance, the predicted currents in (4.15) and (4.16) with parameters mismatching can be rewritten as follows:

$$i_d(k+1) = i_d(k) - \left(\frac{R_s + \Delta R_s}{L_d + \Delta L_d}\right) T_s i_d(k) + \left(\frac{L_q + \Delta L_q}{L_d + \Delta L_d}\right) T_s \, \omega \, i_q(k) + \left(\frac{1}{L_d + \Delta L_d}\right) T_s v_d \quad (4.17)$$

$$i_q(k+1) = i_d(k) - \left(\frac{R_s + \Delta R_s}{L_q + \Delta L_q}\right) T_s i_q(k) - \left(\frac{L_d + \Delta L_d}{L_q + \Delta L_q}\right) T_s \, \omega \, i_d(k)$$

$$- \left(\frac{Y_{pm} + \Delta Y_{pm}}{L_q + \Delta L_q}\right) T_s \omega + \left(\frac{1}{L_q + \Delta L_q}\right) T_s v_q \quad (4.18)$$

where ΔR_s , ΔL_q, ΔL_d, $\Delta \Psi_{pm}$ are parameter errors between the nominal parameter values and the mismatching values.

The stator resistance variation significantly impacts the current-loop regulation performance, and this effect becomes much more severe at low speeds or in high load torque conditions. The effect of inductance variation is mainly coupled with the current change, so the transient performance will be primarily affected in the current dynamic period [4.20]. A mismatch of the rotor flux significantly influences performance at medium and high speeds because the back electromotive force (EMF) is proportional to the rotor flux. If the rotor flux varies, a constant current error occurs in the steady state, and overcurrent or undercurrent occurs in the transient state [4.4]. Moreover, the inertia *J* of a PMSM system, including both rotor and load, is time-varying for some special applications, e.g., an electric winding machine, where the inertia of the whole system increases as time goes by [4.21]. If the system's inertia increases to a value higher than the original, the speed response will have a bigger overshoot and a longer settling time.

Parameter mismatching is one of the crucial PMSM drive uncertainties that can degrade the performance. The variations of PMSM parameters severely affect the prediction accuracy of predictive control methods, which is why several RPC methods primarily focus on compensating for the effects due to parameter mismatching. However, the expected variation range for each parameter based on actual practical conditions should be obtained to design an effective robust control. In other words, it is essential to obtain realistic and practical variations ranges for these parameters by considering the manufacturing tolerances and changes in the operating conditions. Manufacturing tolerance is a specific inaccuracy range in a typical value of a machine variable due to tolerances of geometric dimensions and material properties.

Operational factors are another cause of PMSM parameter variations. For example, the stator winding resistance (R_s) depends on the stator winding temperature. The nominal value of R_s or R_{s0} provided by the manufacturer is obtained at 25 °C (room temperature). The R_s at an operational temperature t (°C) of stator winding (if the temperature is known) can be calculated using the nominal resistance value R_{s0} measured at temperature $t_0 = 25$ °C by

$$R_s = R_{s0}[1 + \alpha(t - t_0)] \tag{4.19}$$

where α is a material constant (for copper, $\alpha = 0.004\ K^{-1}$).

The PM flux linkage (Ψ_{PM}) depends on the direct and quadrature axis currents (i_d, i_q), and the magnet temperature. The machine inductances (L_d, L_q) depend on the currents (i_d, i_q) and the flux density, slightly affected significantly by machine temperature. In addition, L_d and L_q vary non-linearly with respect to the load conditions due to magnetic saturation.

Mechanical parameters of PMSM also vary during real-time operations. For example, moment inertia (J) varies when a load is applied to the machine or connected to an external system. The shape and the dimensions of mechanical loads mainly affect the variation of mechanical parameters [4.22].

In general, the parametric uncertainties of both electrical and mechanical parameters of PMSM are caused by manufacturing tolerances and operational factors. Thus, if the manufacturing tolerance (Δ_M) and operational variation (Δ_O) of a parameter are expressed as a percentage of the nominal values, the total uncertainties (variation range) of a parameter, e.g., R_s, can be obtained as:

$$R_s = R_0\big(1 + \Delta_M(R_0)\big)(1 + \Delta_O(R_0)\,) \tag{4.20}$$

If $\Delta_M(R_0) = \pm 10\%$ and $\Delta_O(R_0) = [-5\%, +30\%]$, the uncertainties of R_s can be expressed as

$$R_s = R_0(1 + [-10\%, +10\%])(1 + [-5\%, +30\%\,])$$

$$R_s = R_0[-14.5\%, +43\%] \tag{4.21}$$

The manufacturing tolerances (Δ_M) can be obtained from the catalogue datasheets of different manufacturers, while the operational variation (Δ_O) can be obtained by considering two situations (rated and maximum operating conditions). For example, Δ_O of R_s due to temperature change can be obtained by using (4.19). For a machine with a rated temperature of $70\ °C$ and maximum temperature of $155\ °C$ (class F insulation material), R_s will increase by $+18\%$ and $+52\%$ of the nominal value at rated and maximum temperature. Also, R_s can be lower than the nominal values at a low temperature and before the machine warms up.

Thus minimum -5% and maximum -10% decrease of R_s from the nominal value can be experienced. The operational variation (Δ_O) of other parameters determined based on several PMSM parameters identification methods, such as recursive least-squares (RLS) algorithms, neural networks (NN), model reference adaptive system (MRAS) based algorithms, online clustering, and particle swarm optimization (PSO) [4.23-4.24]. Thus, with $\%\Delta_M$,$\%\Delta_O$ of each parameter and using (4.20), the total variation ranges of PMSM parameters at rated and maximum conditions are summarized in Table 4.1.

Table 4.1 PMSM parameters potential variations due to manufacturing tolerances and rated & maximum conditions.

Rated condition						
$\%\Delta$ \ parameter	R_s	L_d	L_q	Ψ_{pm}	J	B
$+\%$	30%	32%	32%	15.5%	26.5%	15.5%
$-\%$	14.5%	57.5%	57.5%	23.5%	14.5%	7.85%
Maximum condition						
$+\%$	78%	43.7%	43.7%	21%	32%	20.75%
$-\%$	19%	74.5%	74.5%	32.5%	19%	9.75%

4.4 The Proposed Six-Sigma Robustness Evaluation Method

Six-sigma is a quality measure that quantitatively describes a process or product's performance. The term "sigma" basically is standard deviation σ which measures how a set of data is dispersed around the mean value μ of this data. MOTOROLA and GE developed the six-sigma quality management system to design products that meet customer needs with very low defect levels [4.25]. Considering normal distribution data, the sigma levels ($\pm\sigma$) as

the number of defects per million are presented in Table 4.2. Fig. 4.3 illustrates the normal distribution curves of sigma levels 1 to 6 with $\mu = 0$, the upper specification limit (USL) $=$ 6, and the lower specification limit (LSL) $= -6$. The areas under the normal distribution in Fig. 4.3 associated with each σ-level relate directly to the probability of performance falling in that particular range (for example, $\pm 1\ \sigma$ is equivalent to a probability of 0.683).

Initially, $\pm 3\sigma$ approach was used, where 3σ is equivalent to the probability of 99.73% or the probability of failure (POF) is 0.27% (2,700 defects per million). This probability was deemed acceptable considering short-term quality control. However, in the long term, an approximate 1.5σ shift in the mean μ was experienced, according to MOTOROLA and GE [4.25]. Due to this 1.5σ shift, the 3σ quality control insufficient in long-term, thus the 6σ quality control was used to define the long-term sigma quality. The 6σ in short and long terms as percentage variation and number of defective per million (DPMO) are in Table 4.2 [4.26].

Table 4.2 Sigma level as percentage variation and defects per million.

Sigma level	Percentage variation	DPMO (S)	DPMO (L)
$\pm 1\sigma$	68.26%	317,400	697,700
$\pm 2\sigma$	95.46%	45,400	308,733
$\pm 3\sigma$	99.73%	2,700	66,803
$\pm 4\sigma$	99.9937%	63	6,200
$\pm 5\sigma$	99.999943%	0.57	233
$\pm 6\sigma$	99.9999998%	0.002	3.4

It is important to note that the "Six" in Six-sigma does not mean only 6σ-level is considered. Six-sigma is a quality measure or improvement technique that can be used to

obtain the corresponding or the desired ($n\sigma$). Sigma level is a key property of the Six Sigma method that measures the capability of a process to produce defect-free performance. Another essential property of the six-sigma method is the Z-value, which measures how many standard deviations, σ, a process specification, X, is away from the process's mean, μ. It calculates the process capability index, which indicates how well the process performs relative to its specifications. The concept of Z-value based on a defined USL is shown in Fig. 4.4.

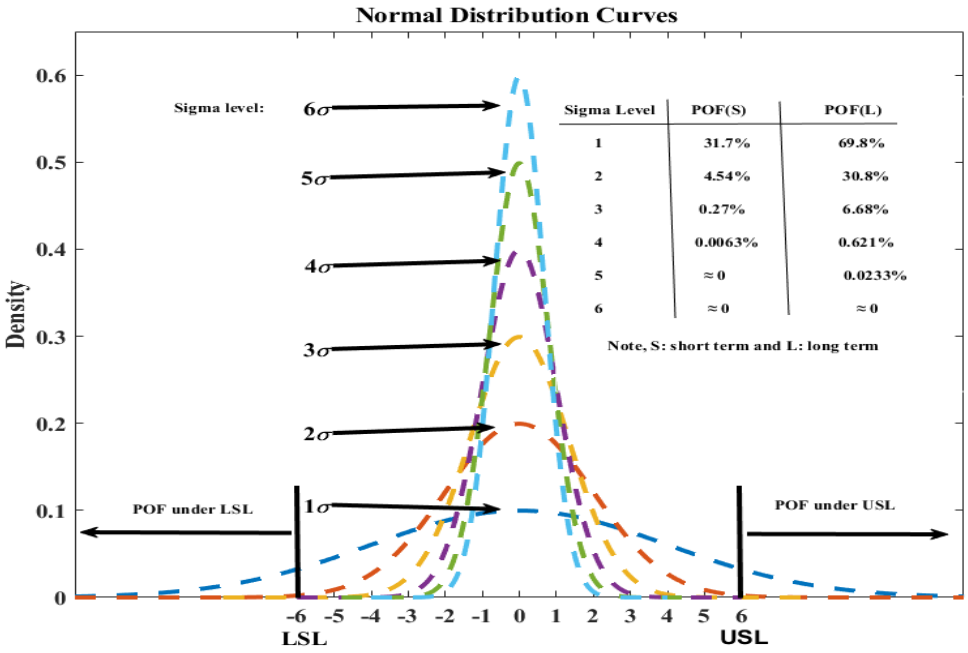


Fig. 4.3. Normal distribution curves with respect to sigma levels from 1 to 6 under the conditions of mean = 0, LSL = −6, and USL = 6 [4.27].

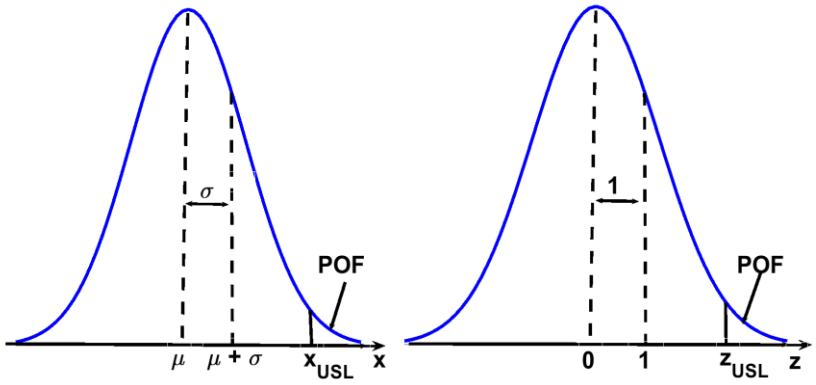


Fig. 4.4. Normal probability density function (NPDF) and Z_{USL} and its relationship to POF with one-sided hypothesis test conditions for cases with USL [4.27].

Six-sigma properties can be used to evaluate and quantify the performance robustness of a control system. First, a set of performance indicators, K_i, that adequately reflects the system performance, and their acceptance levels or upper specification limit, USL_i, must be defined. The robustness criteria of the control system with performance indicators (K_i) and specification limits (USL_i) are expressed as:

$$K_i \leq USL_i, i = 1,2, \dots, m \tag{4.22}$$

where K_i represents the i^{th} performance indicator of a control system, like torque ripple in a motor drive system should be less than 0.4 Nm.

The Z-value of the i^{th} performance indicator is defined as:

$$Z_i = \frac{USL_i - \mu_i}{\sigma_i}, i = 1,2, \dots, m \tag{4.23}$$

where μ_i and σ_i are the mean and standard deviation of the i^{th} performance indicator, respectively.

The Z-value, Z_i, accurately indicates the robustness level of an individual indicator, K_i, relative to the speciation limit, USL_i. However, it is difficult to indicate the overall system

robustness using the Z-value for a system with several indicators. Therefore, sigma levels, $n\sigma$, and POF are used to indicate the overall system robustness. Based on the number of defects, the POF of the system, POF_{sys}, can be obtained and used to compute the system sigma level, n_{sys}. For a control system with *N* samples (total number), if *ND* is the number of defects, the system's POF and equivalent sigma level can be obtained by [4.27]:

$$POF_{sys} = \frac{ND}{N} \tag{4.24}$$

$$n_{sys} = \Phi^{-1}(1 - \frac{POF_{sys}}{2}) \tag{4.25}$$

where $\Phi^{-1}(x)$ is the inverse transformation of a standard cumulative distribution function.

4.5 Robustness Evaluation of the Second-Order and DC Motor Drive Systems

To validate the proposed six-sigma robustness evaluation methods, a closed-loop second-order system with uncertain parameters ζ_i and ω_{ni} (Fig. 4.5) is considered. In order to evaluate the robustness of the stability and performance of the system, some indicators must be defined. The step response characteristics, such as overshoot (*OS*), settling time (*Ts*), and root-mean-square error (RMSE), can be used as performance indicators and the location of the real parts of closed-loop poles (P_{CLP}) can be used to indicate stability. Therefore, the performance and stability indicators (K) and their defined $USLs$ are set as follows:

$$\mathrm{K} = \begin{bmatrix} K_1 \\ K_2 \\ K_3 \\ K_4 \end{bmatrix} = \begin{bmatrix} RMSE(X_c) \\ Ts(X_c) \\ OS(X_c) \\ Re[P_{CLP}(X_c)] \end{bmatrix} \leq \begin{bmatrix} 0.03 \\ 0.02 \\ 0.04 \\ 0 \end{bmatrix} = \mathrm{USL} \tag{4.26}$$

where $X_c = [\zeta_i, \omega_{ni}]$ is the model's uncertain parameters.

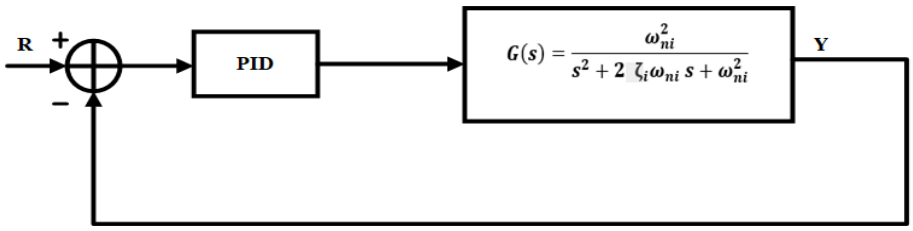


Fig. 4.5. Second-order closed-loop control system with uncertainties.

To evaluate the robustness numerically, the PID controller is designed for the system to achieve optimal performance at the nominal values of ζ_i *and* ω_{ni} as 0.6 and 5, respectively. Then, four different sets of bounded uncertainty (variation) ranges of ζ_i *and* ω_{ni} with 10,000 normally distributed samples for each set generated. The Z-value of settling time (Z_{Ts}), overshoot (Z_{os}) and RMSE (Z_{RMSE}) are obtained based on (4.23), and the sigma level of system performance (n_{sys}), system stability (n_{stab}) and *POF* of system performance are computed based on (4.24) and (4.25), as presented in Table 4.3.

Table 4.3 Robustness evaluation of second-order system with parameter uncertainties.

ω_{ni}	ζ_i	Z_{Ts}	Z_{OS}	Z_{RMSE}	n_{sys}	n_{stab}	POF_{sys}
[4,6]	[0.5,0.7]	57.4	45.3	119.1	6	6	0
[3, 7]	[0.4, 0.8]	25.5	20	29.0	6	6	0
[2, 8]	[0.3, 0.9]	4.9	4.2	6.6	3.7	6	0.0215%
[0,10]	[0.2,1]	1.5	1.8	2.4	1.4	5.4	16.15%

It can be observed from Table 4.3 that as the uncertainties increase, the sigma levels of the system performance and stability decrease. Besides, the robustness of stability is broader than performance, where in some cases, the performance robustness is low, but the

system is still robustly stable.In addition, considering a more practical system like a DC motor drive is simple to control and can be represented by a loop transfer function. Thus, with an open loop transfer function of the DC motor in (4.27) and a suitable controller, robustness analysis of DC motor drive can be performed in the same way as the system in Fig. 4.5.

$$G(s) = \frac{K_t}{JLs^3 + (JR + BL)s^2 + BRs + K_t^2} \tag{4.27}$$

where *J* is the moment of inertia, *B* the motor's viscous friction constant, K_t the motor torque constant, *R* the motor resistance, and *L* the motor inductance.

To numerically evaluate the DC motor drive robustness, the indicators (K) and $USLs$ are defined as follows:

$$\mathrm{K} = \begin{bmatrix} K_1 \\ K_2 \\ K_3 \\ K_4 \end{bmatrix} = \begin{bmatrix} RMSE(X_c) \\ Ts(X_c) \\ OS(X_c) \\ Re[P_{CLP}(X_c)] \end{bmatrix} \leq \begin{bmatrix} 0.03 \\ 0.04 \\ 0.03 \\ 0 \end{bmatrix} = \mathrm{USL} \tag{4.28}$$

where $X_c = [R_i, L_i, J_i, B_i]$.

A DC motor drive is optimized to achieve nominal performance at nominal parameters in Table 4.4. Then, the robustness evaluation results with three variation ranges of nominal parameters ($\pm 50\%$, $\pm 75\%$, $\pm 100\%$) are listed in Table 4.5.

Table 4.4 Nominal DC motor parameters.

Parameter	$R(\Omega)$	$L(H)$	$J(kg.m^2)$	$B(N.m.s)$	K_t
Value	4	$2.75e^{-6}$	$3.23e^{-6}$	$3.51e^{-6}$	0.0274

Table 4.5 Robustness evaluation of DC motor drive with parameters uncertainties.

Range	Z_{Ts}	Z_{OS}	Z_{rmse}	n_{sys}	n_{stab}	POF_{sys}
$\pm 50\%$	26.3	16.7	28.7	6	6	0.0
$\pm 75\%$	6.6	5.1	9.4	4.3	6	0.0017%
$\pm 100\%$	2.5	1.8	2.8	1.6	6	10.96%

4.6 Evaluation of Different RPC Methods of PMSM Drives

PMSM drives are subjected to various uncertainties, including parametric and unmodelled dynamics uncertainties. MPC is highly dependent on the machine model and parameters; thus, high uncertainties affect MPC's operations and lead to unacceptable performance for some applications. RPC methods have been introduced to maintain a good control performance in the presence of uncertainties. Various RPC methods for PMSM drives have been proposed in the literature, considering different techniques to deal with uncertainties. The robustness of these RPCs is commonly illustrated by considering a few cases with mismatching parameters applied (deterministic approach). However, how robust (robustness level) each of these RPCs with bounded uncertainties range has not been assessed against specific application requirements (stochastic approach). With the proposed six-sigma robustness evaluation method, the robustness levels of any RPCs or any control method for PMSMs can be determined numerically with a bounded uncertainties range considering

different applications requirement.

To validate and illustrate the proposed method, the conventional MPC and five existing RPC methods for PMSM drives are used to assess their robustness to uncertainties. From the literature survey (Chapter 2), RPC methods are classified into five types: RPC-based prediction error, observer, hybrid (combined) techniques, optimized cost function, and model-free. Thus, a controller from each category is selected to perform a robustness evaluation based on the proposed six sigma robustness evaluation method. Predictive current control-based prediction error correction proposed in [4.1] is used and will be referred to as RPC-I. To achieve robustness and compensate for any parameter mismatching, this method used the weighted errors between the predicted and measured values in the last sampling instant and added them to prediction equations in the next sampling instant. Robust MPC with simplified repetitive control introduced in [4.12] is also used and will be referred to as RPC-II. This method applies a less computationally simplified repetitive controller with two resonant units and phase compensation to MPC to realize the system's robustness against disturbances.

Model predictive current control based on an incremental model and disturbance observer proposed in [4.28] is another RPC to be used and will be referred to as RPC-III. In this method, an incremental prediction model was implemented to eliminate the permanent magnet flux link-age parameter, and an inductance disturbance controller that includes a simple disturbance observer and inductance extraction algorithm was implemented to reduce the effects of machine inductance mismatch. Robust model predictive direct torque control based on the optimized cost function proposed in [4.9] is also used and will be referred to as RPC-IV. This method's cost function is modified to include specific objectives and constraints to achieve drive robustness. A direct torque MPC based on maximum torque per

ampere (MTPA) criteria was developed with a novel cost function with three terms. They are: tracking term to ensure reference tracking, attraction region term to define where the steady-state control states should be, and limitation term to limit the control states to their admissible values.

The robust model-free predictive current control-based current detection technique proposed in [4.7] is also used and will be referred to as RPC-V. This method works by calculating the difference between the measured currents at different samples; then, these differences are used to predict the current in the next sampling. Thus, no machine parameters are required to perform the prediction.

4.6.1 Numerical Verification and Experimental Validation

Prior to conducting a robustness evaluation of conventional MPC and RPCs, these methods are to be validated by numerical simulation and experimental results. MPC and the five existing RPC methods were designed and implemented in Matlab/Simulink based on PMSM drive with parameters in Table 3.1. The sampling frequency of 10 kHz and the same machine parameters were applied to all controllers. The start-up responses from standstill to rated speed (1000 rpm) with load torque (2 Nm) applied at 0.2s are shown in Fig. 4.6 with nominal parameters and Fig. 4.7 with mismatching parameters ($1.5R_s, 0.5L_d$,$0.5L_q$,$0.7\psi_{Pm}$). From top to bottom, the graphs are phase A stator current (i_a), direct-axis and quadrature-axis currents (i_d, i_q), motor torque (T_e), and rotor speed (ω_r). As can be seen from the responses in Fig. 4.7, the performance is critically affected by mismatching parameters. Compared to the responses at nominal parameters, high current, and torque ripples are produced, and slow speed responses are experienced. Higher performance degradation is

recorded for the conventional MPC compared to the other RPC methods because no robustness mechanism was implemented for MPC.

In addition to the simulation studies, MPC and five RPC methods are experimentally validated on a two-level inverter-fed PMSM drive system, the same as shown in Fig. 3.10. A dSPACE DS1104 PPC/DSP control board is employed to implement the real-time algorithm, a 2500-pulse incremental encoder is used to obtain the motor speed and position, and dSPACE ControlDesk interfaced with DS1104 is used for real-time control, monitoring, and record all experimental results. Magtrol DSP6000 high-speed programmable dynamometer controller is used to apply external load.

First, the six control methods are tested during start-up, where motor speed starts from a standstill to rated speed (1000 rpm) are shown in Fig. 4.8, and the steady-state responses with load disturbances (2 Nm) applied at 0.5s are demonstrated in Fig. 4.9. The curves in Figs 4.8 and 4.9 from top to bottom are phase A stator current (i_a), direct-axis and quadrature-axis currents (i_d, i_q), motor torque (T_e), and rotor speed (ω_r).

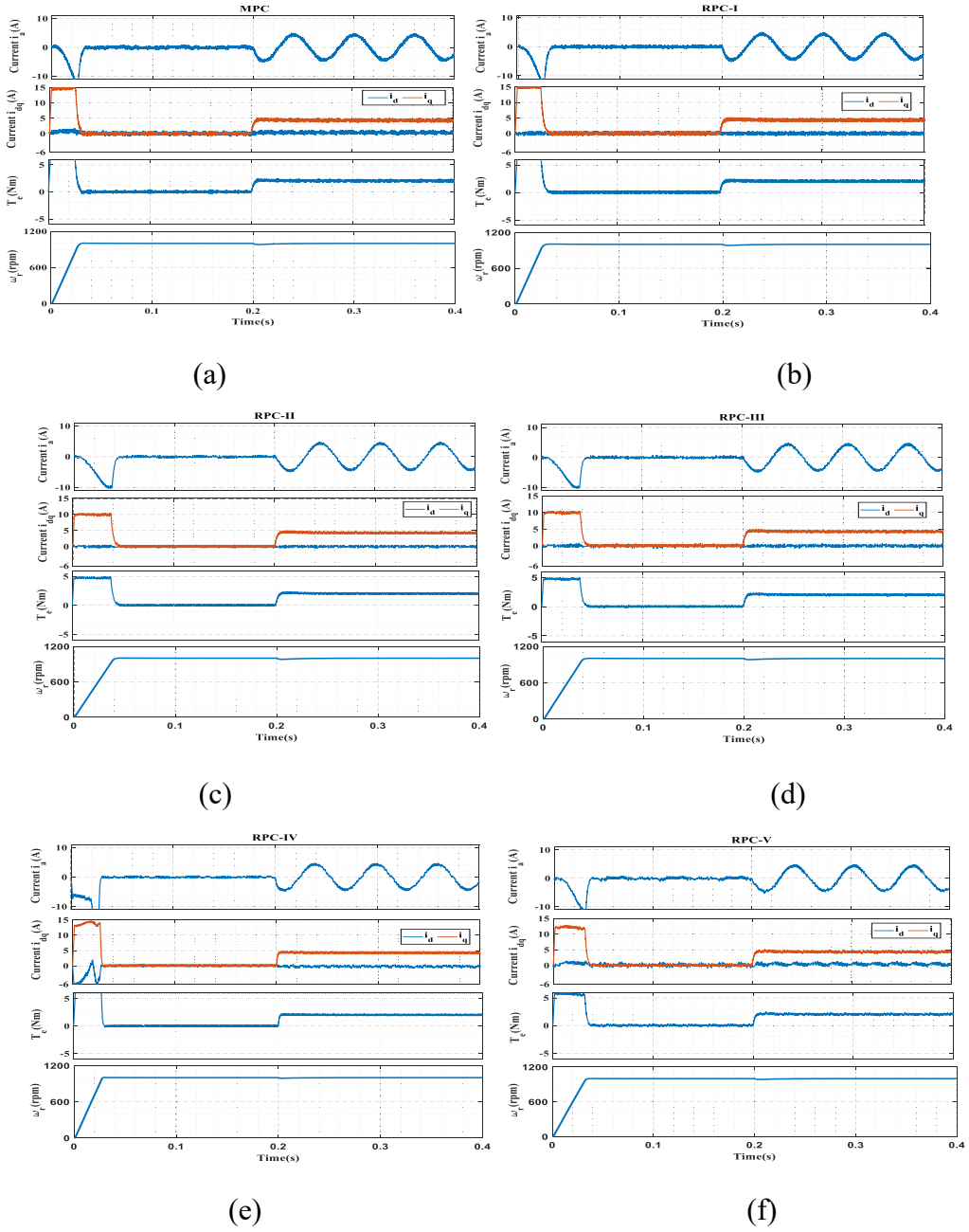


Fig. 4.6. Simulation start-up performances with nominal PMSM parameters for (a) MPC, (b)RPC-I, (c) RPC-II, (d) RPC-III, (e) RPC-IV, and (f) RPC-V.

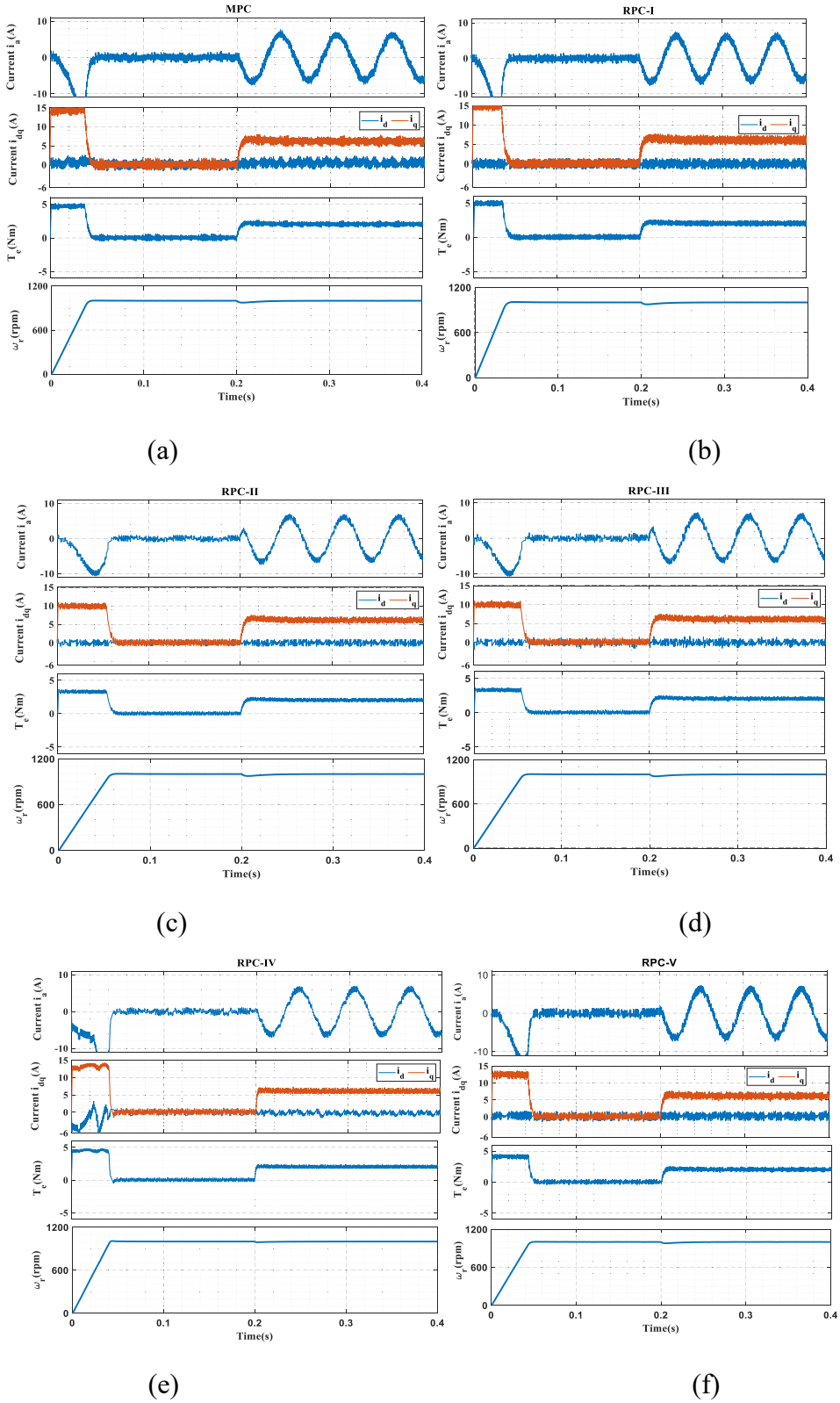


Fig. 4.7. Simulation start-up performances with mismatching PMSM parameters for (a) MPC, (b)RPC-I, (c) RPC-II, (d) RPC-III, (e) RPC-IV, and (f) RPC-V.

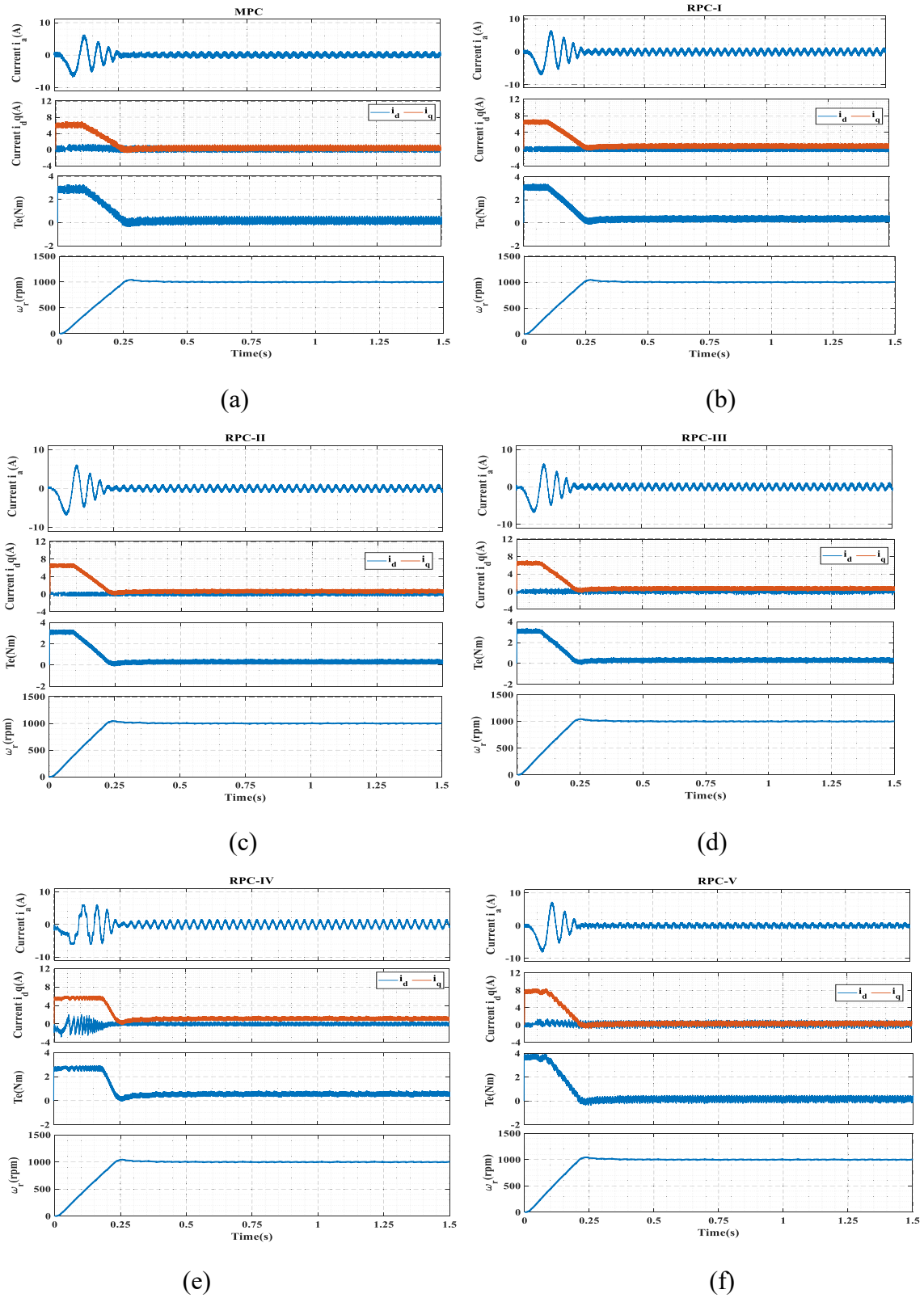


Fig. 4.8. Experimental start-up response from 0 to 1000 rpm for (a) MPC, (b)RPC-I, (c) RPC-II, (d) RPC-III, (e) RPC-IV, and (f) RPC-V.

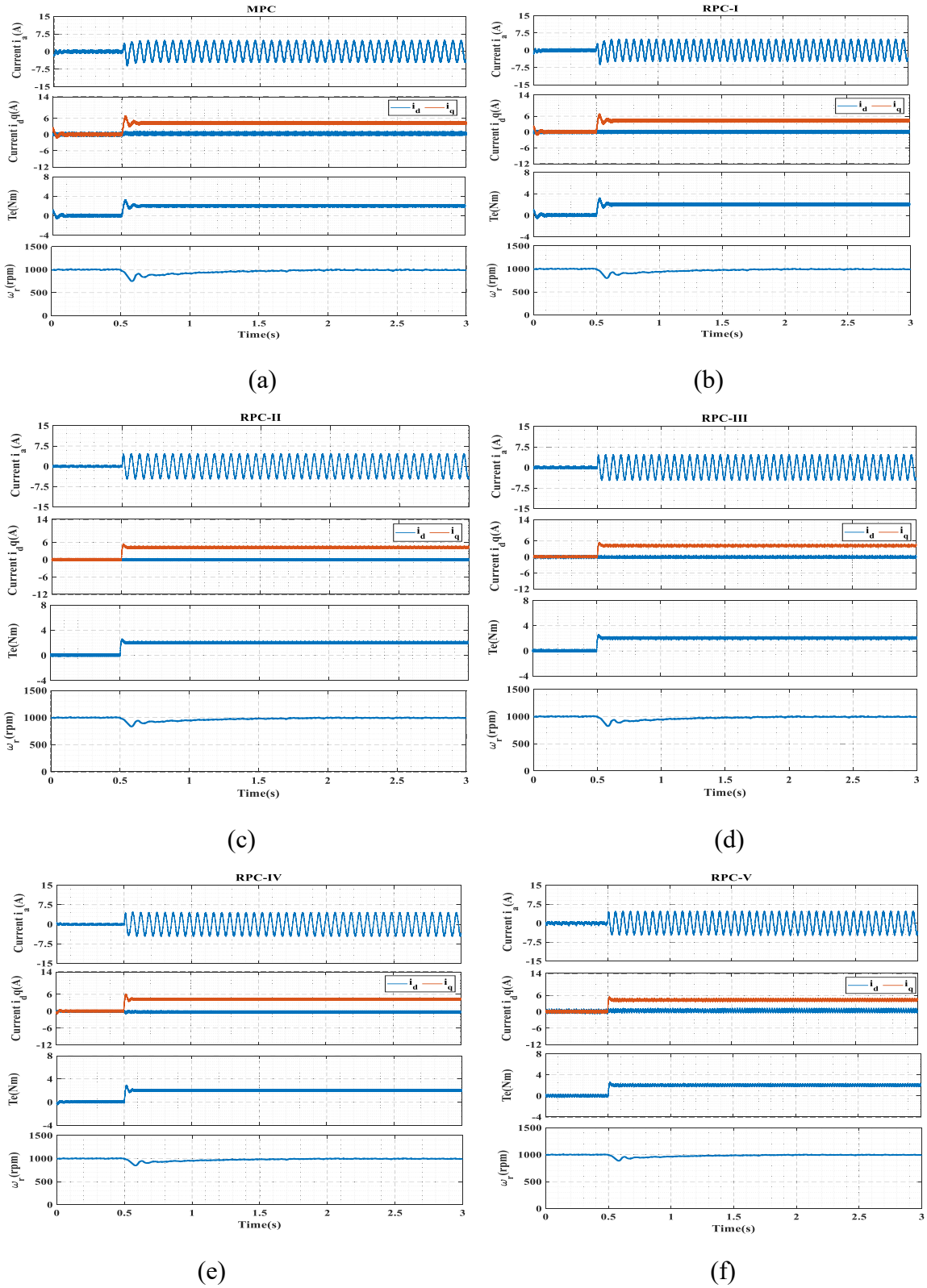


Fig. 4.9. Experimental steady-state response under 2 Nm load torque for (a) MPC, (b)RPC-I, (c) RPC-II, (d) RPC-III, (e) RPC-IV, and (f) RPC-V.

4.6.2 Quantitative Analysis

To further assess the performance against changes in operating conditions and parameter uncertainties, the above control methods are evaluated numerically by measuring the torque and current tracking performances at the steady state. At different operating conditions and several parameters variation, the current and torque ripples are computed using (3.52) and (3.53)[4.29].

In this evaluation, two cases are considered. In the first case, MPC and the five RPC methods are evaluated with different speeds and load torques at nominal machine parameters. Under no load, 1 Nm, and 2 Nm load torque conditions, five different speeds (200,400,600,800 and 1000 rpm) are applied. At each test, the torque ripples and stator current ripples are computed. Fig. 4.10 shows the torque ripples of MPC and RPC methods at no load, 1 Nm, and 2 Nm loads, and five different speeds. Similarly, the current ripples are presented in Fig. 4.11 under no load, 1 Nm, and 2 Nm load torque.

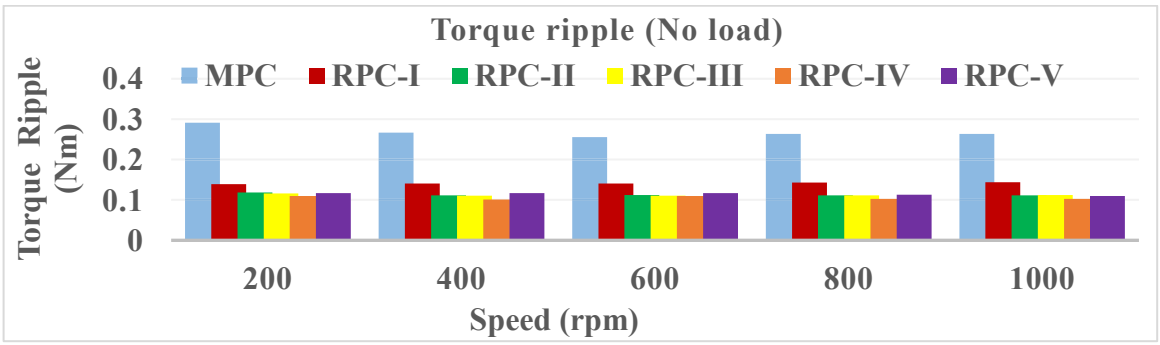
In the second case, the motor operates at 1000 rpm and under 2 Nm load torque, and then the machine parameters vary in ranges according to Table 4.1. The torque and current ripples are computed with PMSM parameters changing in the range from -Δ% to Δ% of their nominal values. The torque and current ripples of MPC and RPC methods with variations of L_q,L_d,R_s and Ψ_{pm} are presented in Figs 4.12- 4.15. L_q and L_d are varied from -70% to 40% of the nominal value, while other PMSM parameters are kept at their nominal values. R_s is varied from -20% to 80% of the nominal value, while other PMSM parameters are kept at their nominal values. Ψ_{pm} is varied from -30% to 20% of the nominal value, while other PMSM parameters are kept at their nominal values.

From the obtained data of torque and current ripples at different torque and speed conditions, it can be observed that MPC has the highest torque and current ripples under different load conditions and at different speed operations. RPC methods vary in their response to load and speed variation. RPC-IV achieved the best torque ripples compared to other RPC methods. Compared to MPC and at 1000 rpm, RPC-IV reduced the torque ripples by 61.1%, 58.4 %, and 58.7% under no load, half load, and full load conditions, respectively. Conversely, RPC-V achieved the smallest current ripples compared to MPC and other RPC methods. At 1000 rpm and no load, half load, and full load conditions, RPC-V reduced the current ripples by 55.4%, 53.2 %, and 63%, respectively, compared to MPC.

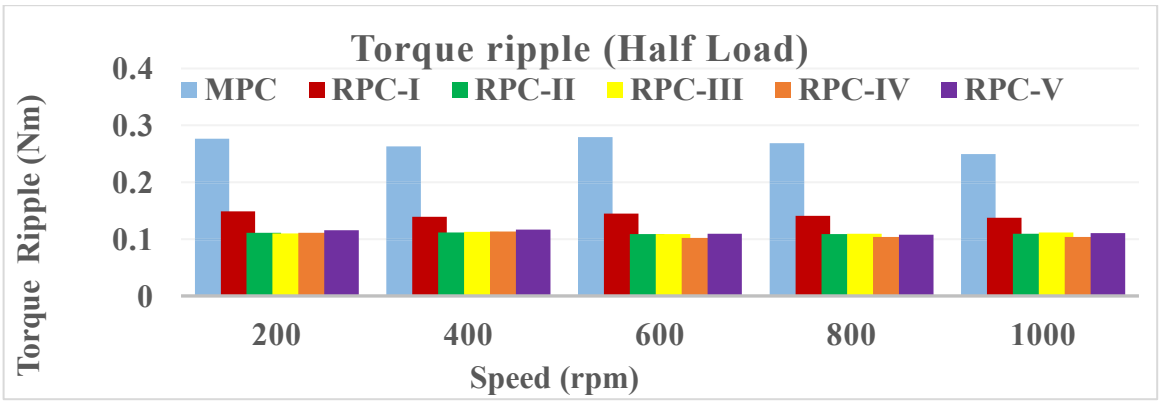
With different parameters variation, higher effects are produced with the variation of inductances (L_d, L_q) particularly at low values, where very high ripples are recorded. While the variation of permanent magnet flux linkage (Ψ_{pm}) has less effect compared to the variations of inductance, and the high ripples are recorded with high values of (Ψ_{pm}). The variation of stator resistance (R_s) does not have much effect on the current and torque ripples. The highest torque and current ripples are recorded when L_d and L_q are decreased by 70%. When L_q is reduced by 70%, the torque and current ripples respectively increased by 321.8% and 179% for MPC, 280.1% and 205% for RPC-I, 243.9% and 91.4% for RPC-II, 190.1% and 120.8% for RPC-III, 160.1% and 127.2% for RPC-IV, and 153.7% and 100.1% for RPC-V higher than the values recorded at nominal (L_q). However, compared to MPC, the torque and current ripples respectively reduced by 17.6% and 20.8% for RPC-I, 35.2% and 44.6% for RPC-II, 49.5% and 56.8% for RPC-III, 55.9% and 39.5% for RPC-IV, and 57.4% and 61.7% for RPC-V. Similarly, with -70% mismatch in the *d*-axis inductance (L_d), the torque and current ripples respectively increased by 244.5% and 230% for MPC, 155% and 236.5%

for RPC-I, 141.8% and 128.8% for RPC-II, 137.4% and 156.1% for RPC-III, 132.6% and 110.9% for RPC-IV, and 123.7% and 121.7% for RPC-V higher than the values recorded at nominal (L_q)**.** Compared to MPC, the torque and current ripples respectively reduced by 16.5% and 26.1% for RPC-I, 35.8% and 63.2% for RPC-II, 41.6% and 57.6% for RPC-III, 44.5% and 52.5% for RPC-IV, and 47.4% and 65.1% for RPC-V**.** The torque and current ripples significantly increase as the machine inductances (L_d, L_q) decrease from their nominal values. The torque and current ripples are slightly affected as the inductances increase from their nominal values.

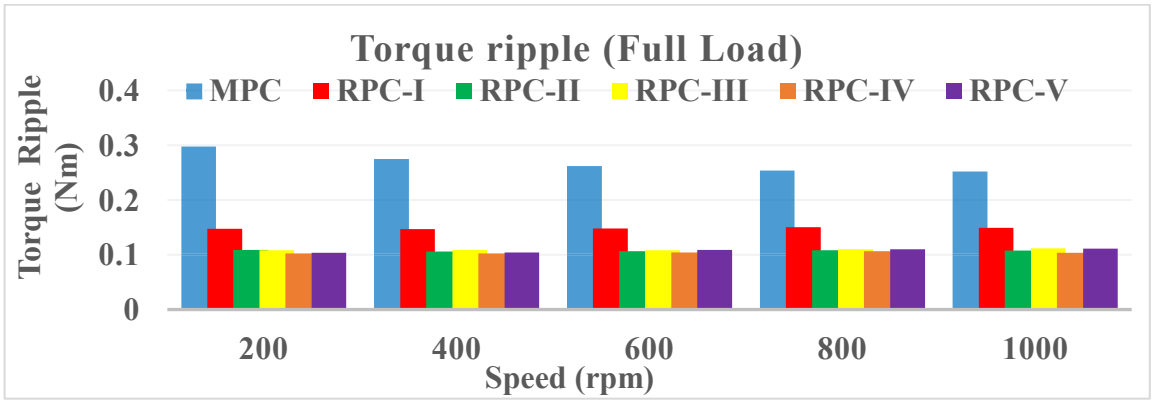
In addition, the torque ripples of all controllers increase steadily as the value of Ψ_{pm} increases. The highest values are generated at 20% increase in Ψ_{pm} , in which increments from the values generated at nominal Ψ_{pm} are recorded as 22.9%, 21%, 19.5%, 18.3%, 19.8%, and 19.6% for MPC, RPC-I, RPC-II, RPC-III, RPC-IV, and RPC-IV, respectively. The current ripples vary non-linearly with the variation of Ψ_{pm} MPC produces the worst current ripples, and RPC-V and RPC-III recorded the lowest ones. At 20% increase in Ψ_{pm} , RPC-V and RPC-III recorded 46.2% and 44.7% less current ripples than MPC. Moreover, the torque and current ripples are slightly influenced by the variation of stator resistance (R_s). MPC produced the worst torque and current ripples with the variation of R_S. While RPC-IV showed the best torque ripples and RPC-V showed the best current ripples. Overall, all RPC methods showed good response compared to MPC at different operating conditions (load, speed). RPC-V has the best response regarding current ripples and maintaining a trade-off between torque and current ripples.



(a)

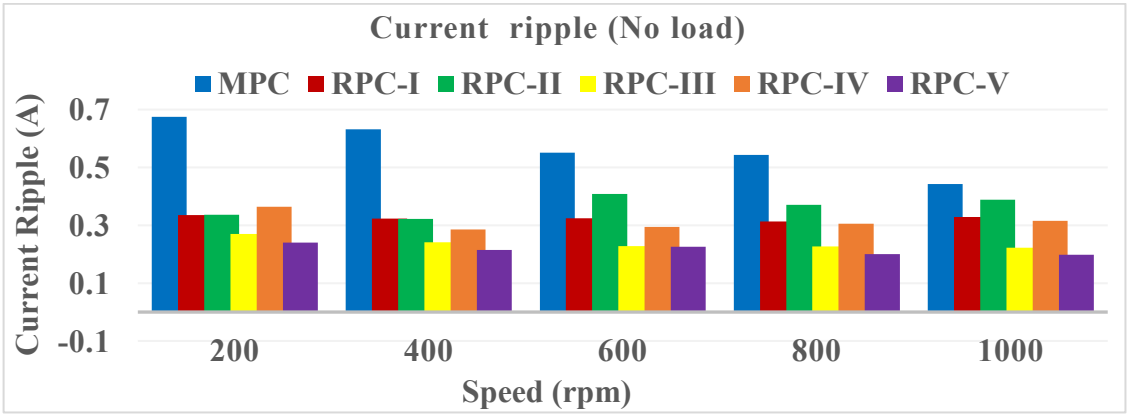


(b)

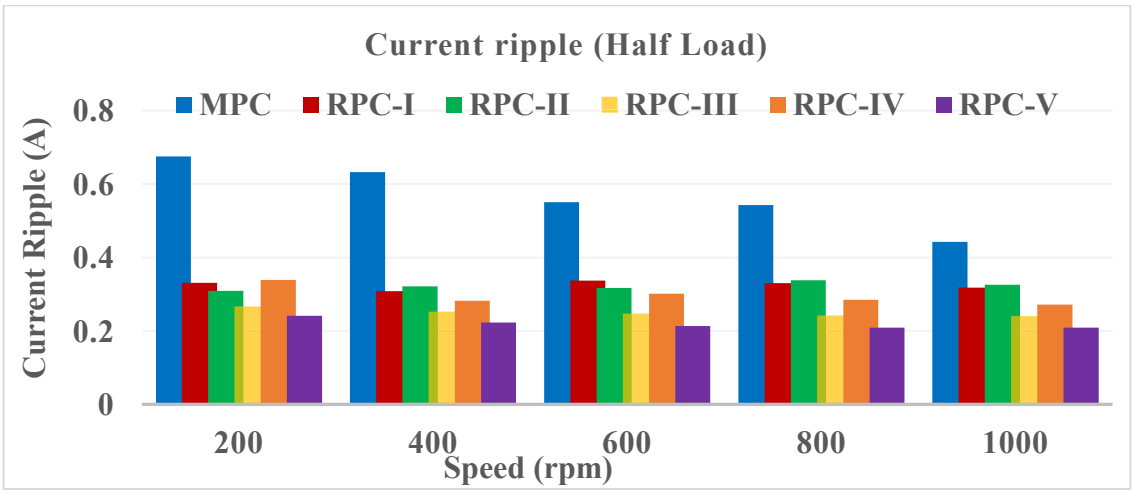


(c)

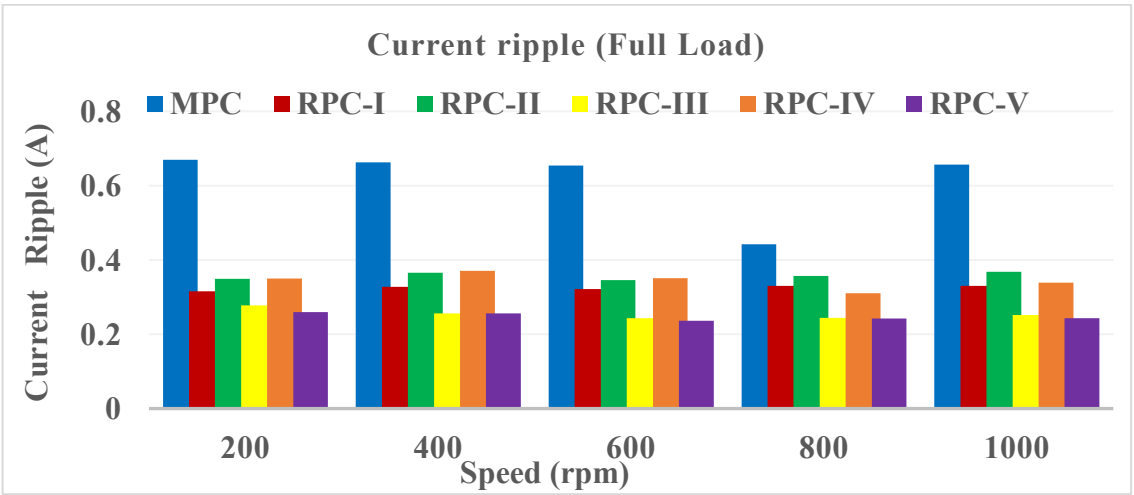
Fig. 4.10. Comparison of torque ripples in different control methods under (a) no load, (b)half load, and (c) full load conditions.



(a)

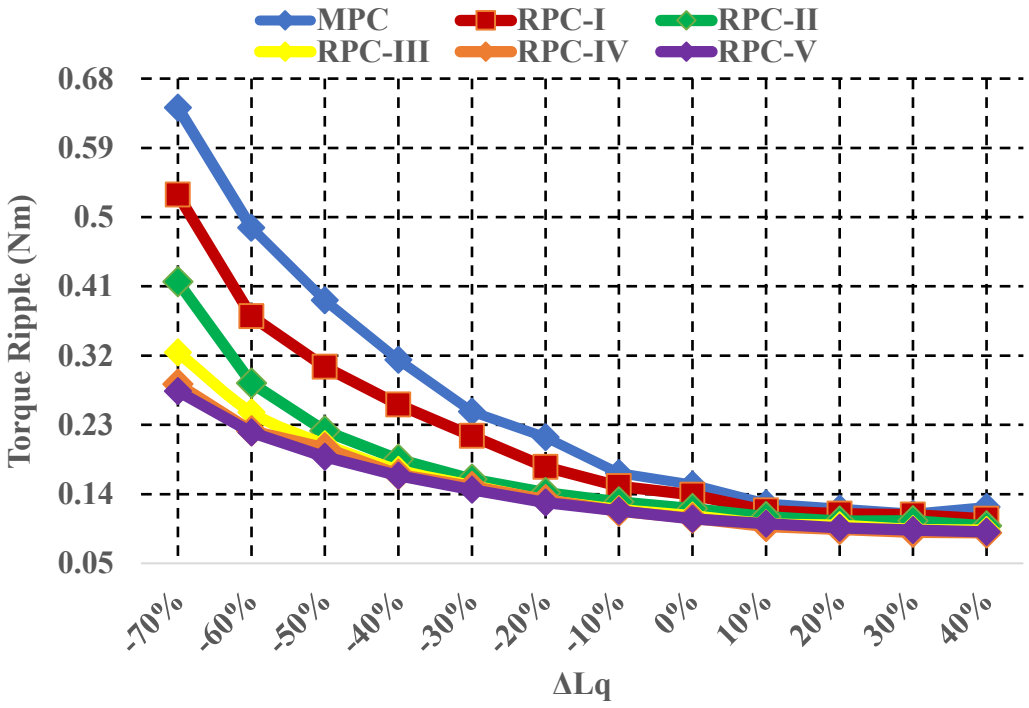


(b)

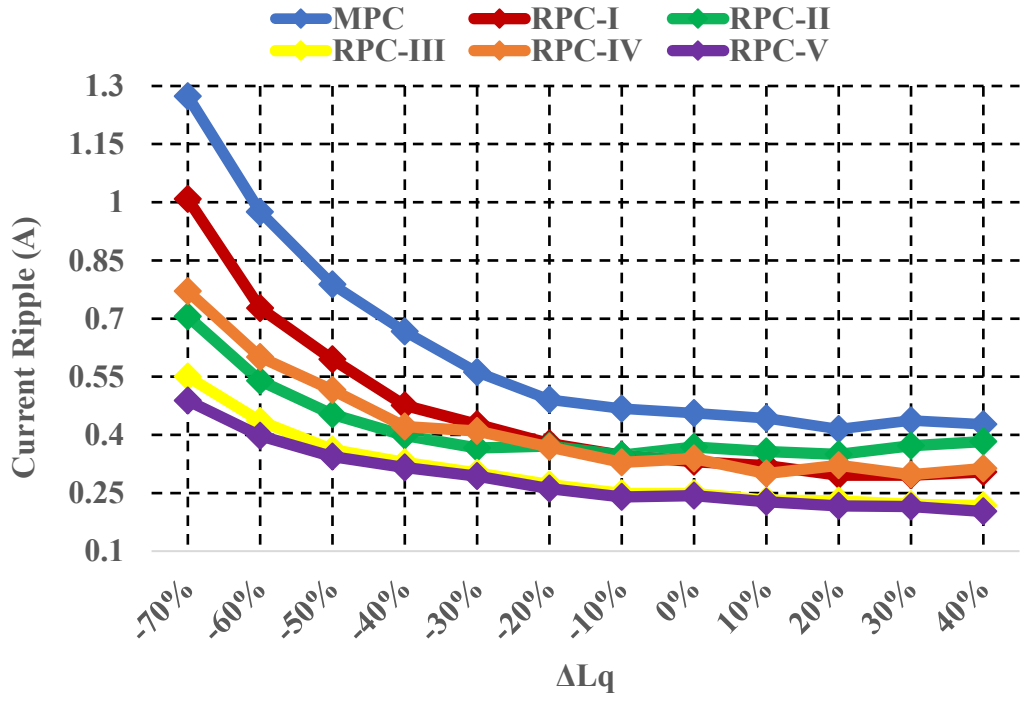


(c)

Fig. 4.11. Comparison of stator current (i_a) ripples in different control methods under (a) no load, (b)half load, and (c) full load conditions.

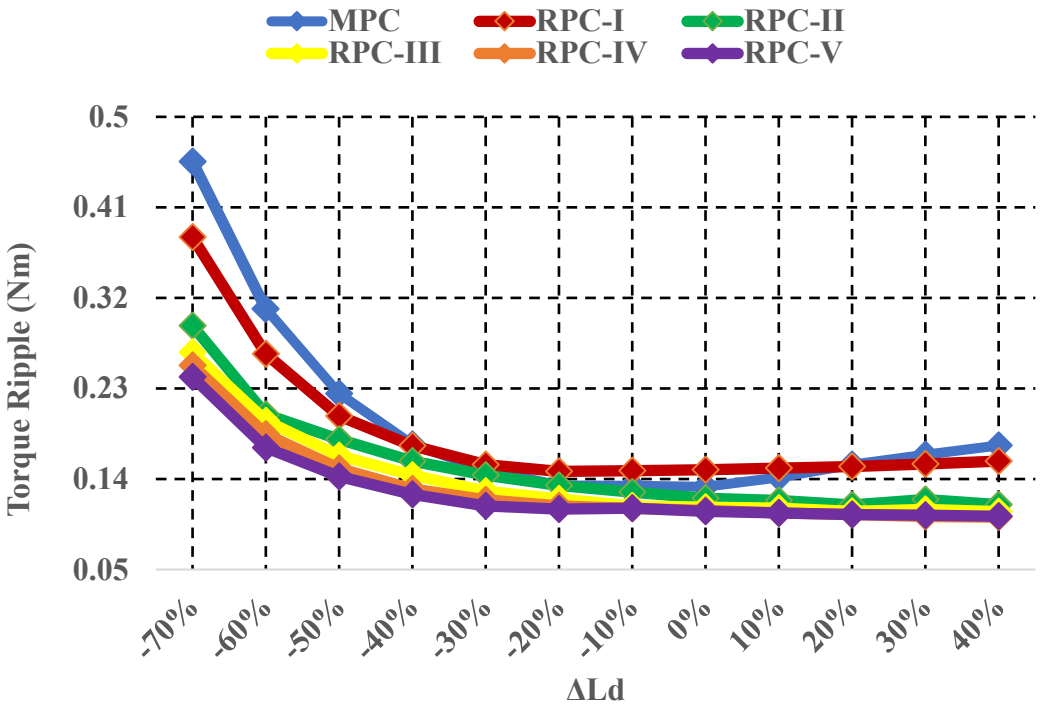


(a)

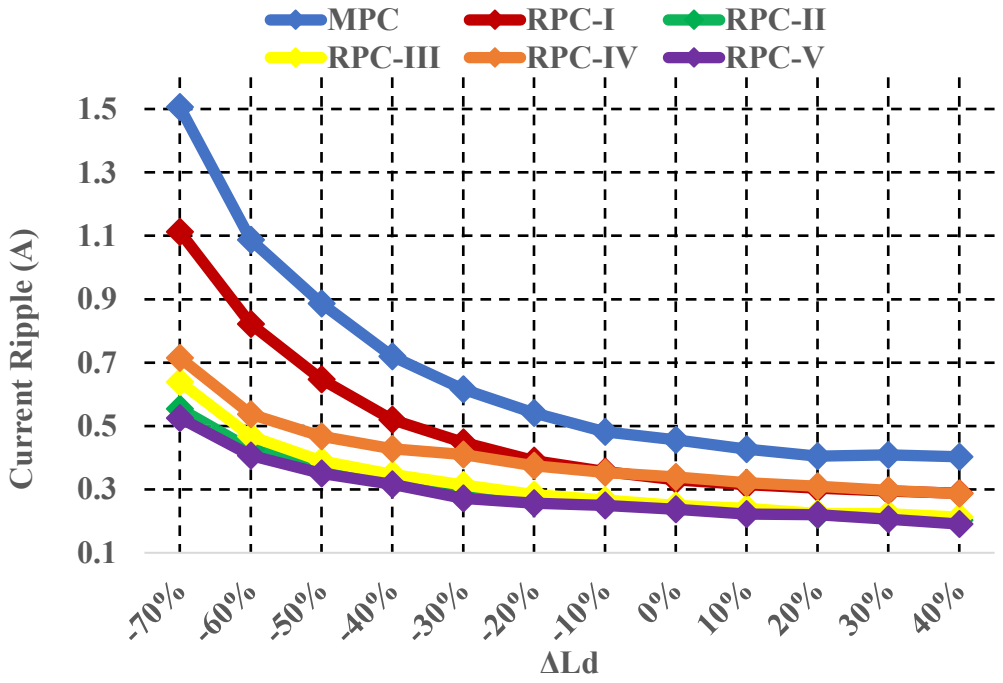


(b)

Fig. 4.12. Comparison of different control methods under the variation of machine inductance (L_q), (a) Torque ripples, (b) stator current (i_a) ripples.

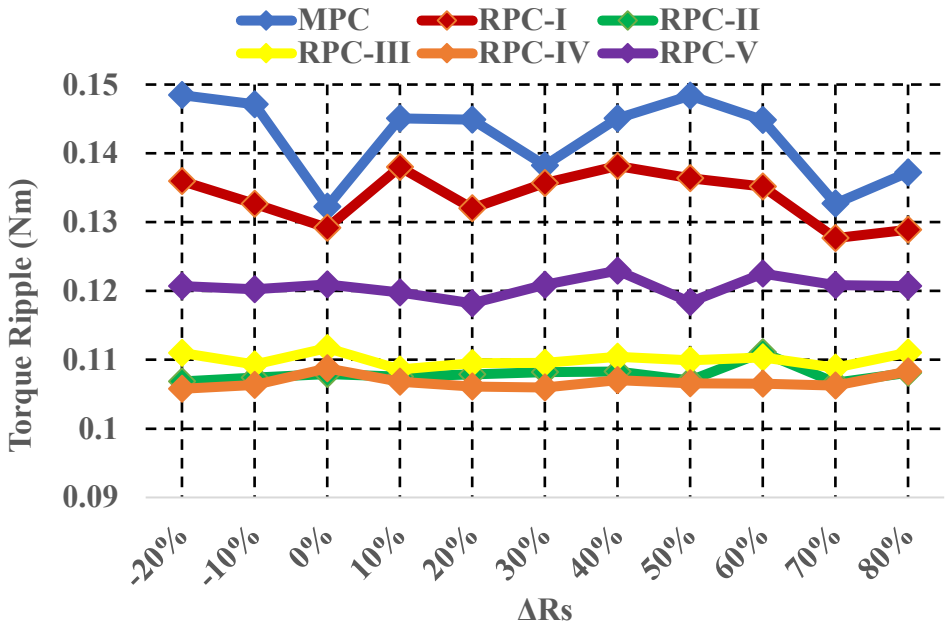


(a)

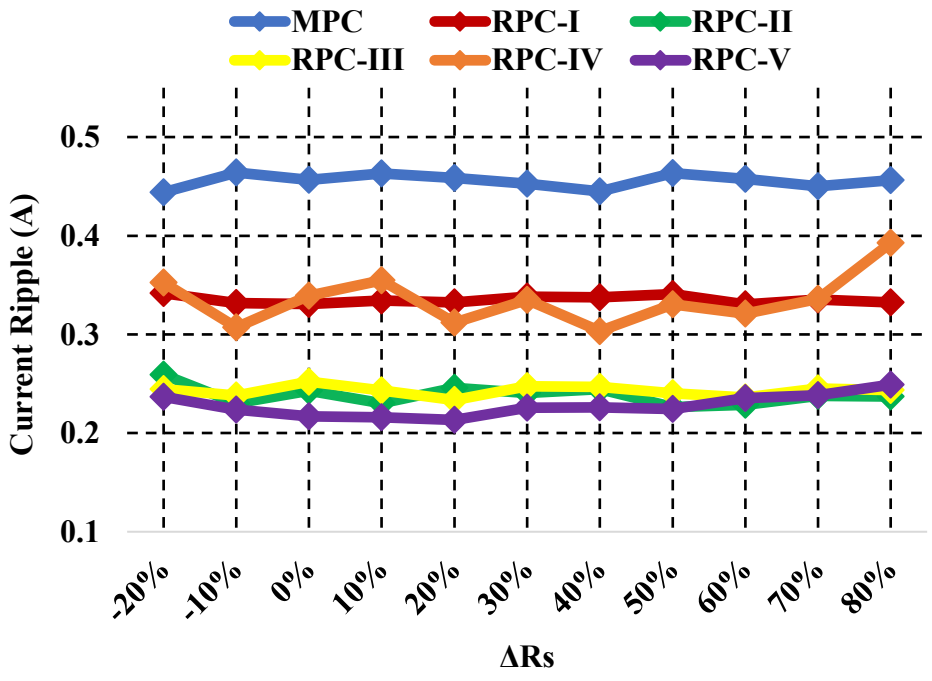


(b)

Fig. 4.13. Comparison of different control methods under the variation of machine inductance (L_d), (a) Torque ripples, (b) stator current (i_a) ripples.

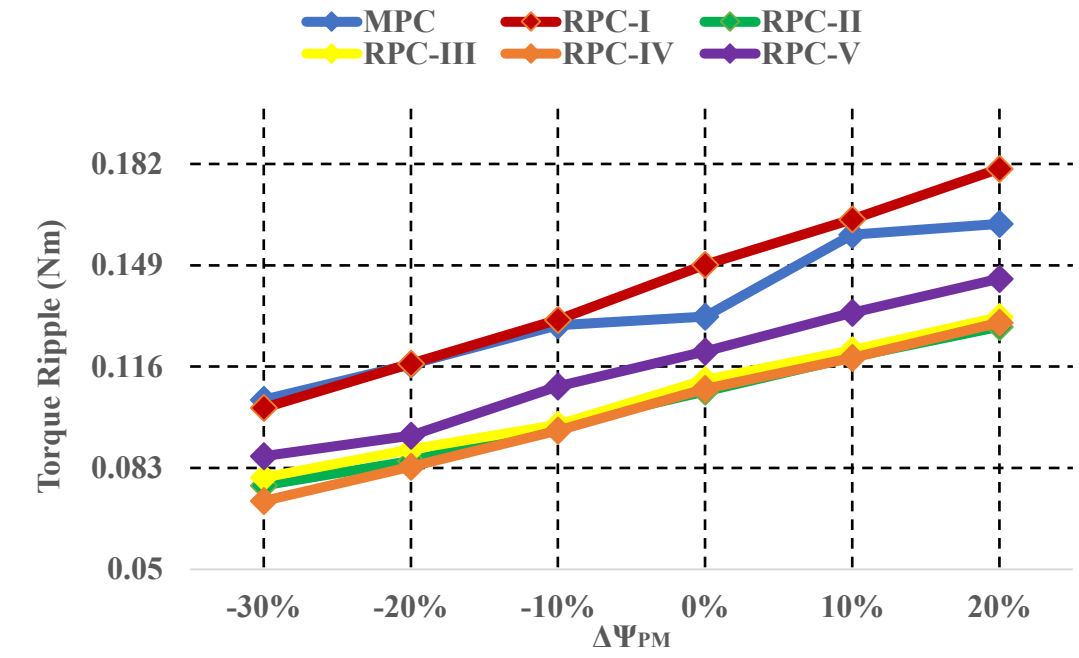


(a)

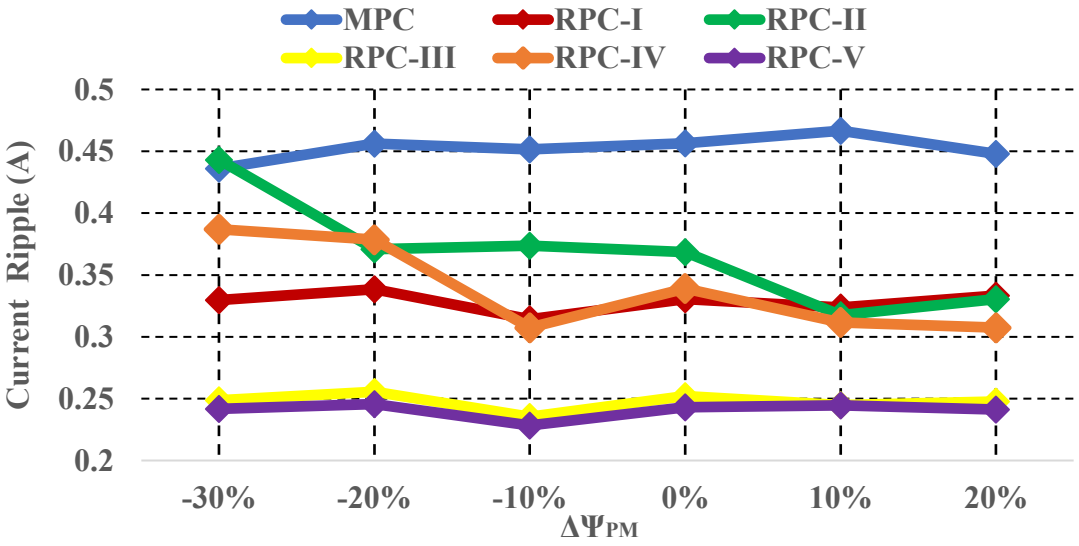


(b)

Fig. 4.14. Comparison of different control methods under the variation of stator Resistance (R_s), (a) Torque ripples, (b) stator current (i_a) ripples.



(a)



(b)

Fig. 4.15. Comparison of different control methods under the variation of permanent flux linkage (Ψ_{PM}), (a) Torque ripples, (b) stator current (i_a) ripples.

4.6.3 Parameter Sensitivity Analysis

Sensitivity analysis is the study of how uncertainty in the output of a model can be attributed to different sources of uncertainty in the model input[4.30]. In other words, sensitivity analysis is a technique used to assess the influence of changes in input parameters on the output of a system or model [4.31]. In the case of PMSM control, it aims to evaluate how variations in specific motor parameters affect its performance characteristics. There are two main types of sensitivity analysis: local and global. In the context of PMSM control, sensitivity analysis can be used to study how variations in the control parameters affect the motor's performance. Local sensitivity analysis involves varying one machine parameter at a time while keeping all other parameters constant. Local sensitivity analysis is relatively easy to perform and can provide insights into the relative importance of different control parameters. However, it does not account for interactions between parameters and may miss significant effects that arise from simultaneous variations of multiple parameters.

In contrast, global sensitivity analysis involves varying multiple control parameters simultaneously over their entire range of possible values. For example, one could vary all the machine parameters over a range of values and then study how the motor's performance varies across this range. Global sensitivity analysis can account for interactions between parameters and identify critical nonlinear effects that may be missed by local sensitivity analysis. This section presents the global sensitivity analysis using Monte Carlo techniques to study the effect of varying machine parameters on the machine drive performance. Various performance indicators are evaluated across a global set of samples to explore the design space.

Various machine parameters are used for this analysis, including the stator resistance R_s, d- and q-axes inductances L_d, L_q, permanent magnet flux in the rotor Ψ_{pm}, the momentum of inertia J, and viscous friction coefficient B. These parameters are varied simultaneously over a range of values a specific range, and then various performance indicators are evaluated, including settling time (T_S), overshoot (OS), root mean square error ($RMSE$) of speed, torque ripples, and current ripples. The sensitivity analysis is carried out based on MPCC of PMSM drives with nominal parameters presented in Table 3.1 and their variation range of possible values at maximum operating conditions in Table 4.1. The workflow of sensitivity analysis of PMSM drives is described as follows:

i. Sample the machine parameters using experimental design principles. For each parameter, generate multiple values (1000 values) based on the parameter's possible range (Table 4.1). The parameter sample space is defined based on normal probability distributions for each parameter.
ii. Performance indicators are defined by creating a design requirement on the model signals for each indicator.
iii. The performance indicators (design requirement) are evaluated using Monte Carlo simulations at each combination of parameter values. The indicator outputs and parameters are normalized in a range [0 1], then plotted to analyze trends visually.
iv. The relation between the evaluated requirement and the samples is evaluated based on correlation and standardized regression.

Parameter sample space (normal probability distribution) is created by generating random samples of each machine's parameters (1000 samples) based on their maximum possible variation range (Table 4.1), as presented in Fig. 4.16. Each performance indicator is

evaluated, normalized, and visually plotted against the parameter samples (normalized) to identify trends. Scatter plot (Fig. 4.17) displaying the evaluated performance indicator normalized values as a function of each parameter in the parameter sample space. This plot provides visual intuition about how the various parameters affect the performance indicators. A linear fit line is added to the scatter plot to identify the effect of each parameter in the indicators. The best-fit line indicates that the parameter has a lot of influence on the indicators. Furthermore, the parameters and indicators data are normalized to [0, 1] range for better and more meaningful visual representation. By normalizing data to the [0, 1] range, the minimum value in the dataset is mapped to 0, and the maximum value is mapped to 1. The values in between are linearly scaled based on their relative positions within the minimum and maximum values. Normalizing the data brings them to a uniform scale, which simplifies the process of comparing values and recognizing patterns or trends.

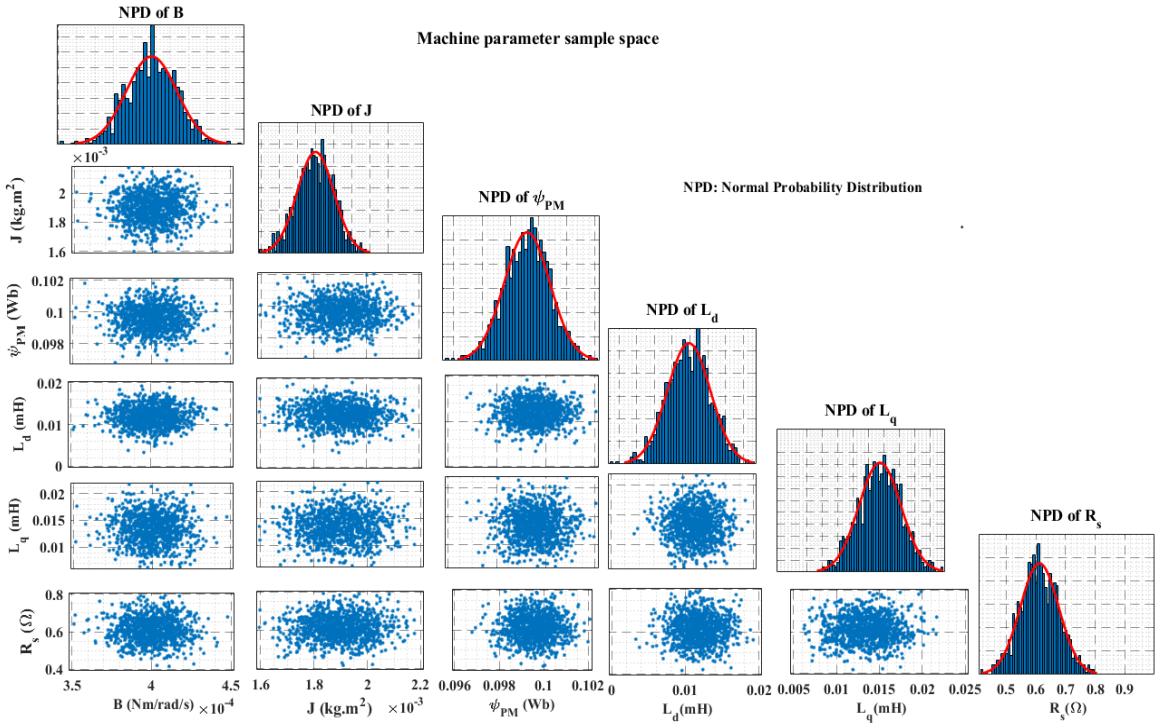
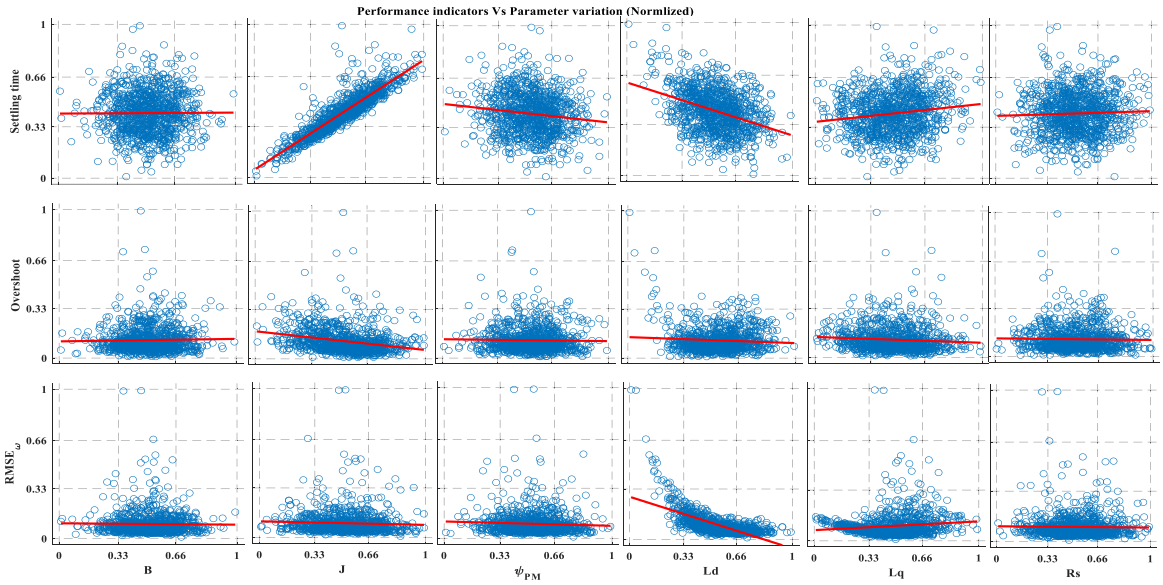


Fig. 4.16. Machine parameter sample space.

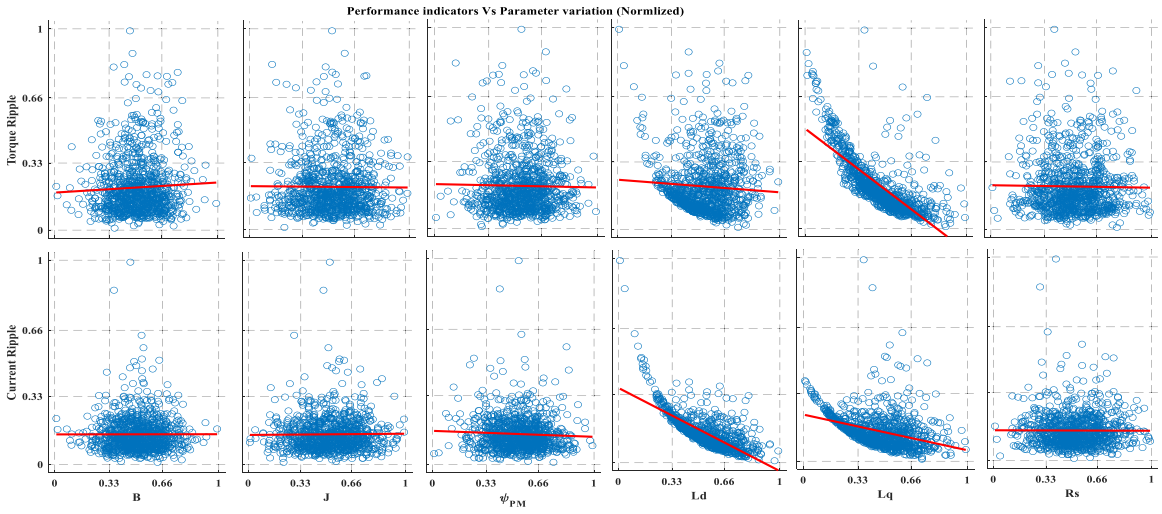
The plot in Fig. 4.17 shows that the settling time is directly proportional to the variation of machine inertia (J); as the inertia increases, the settling time increase. The maximum settling time is 24.5% higher than the mean value of the settling time data. No other proper trends can be found to show how the settling time changes with the variations of other parameters. Furthermore, no specific appropriate trends can be found to show how machine parameter variation affects the overshoot and RMSE speed.

The torque ripples increase monotonically with the decrease in the values of d-axis inductance (L_d). No other trends between changes in torque ripples and variations of other machine parameters. The highest torque ripple is 34.9% higher than the mean value of the torque ripple. The current ripples are found to be inversely proportional to the variation of d-axis and q-axis inductances (L_d, L_q) as the values of L_d and L_q decrease, higher current ripples are obtained. The highest current ripple is 29.6% higher than the mean value of the current ripples.

In addition to visually analyzing the effect of machine parameter variations on the performance indicators, the relation between varying parameters and indicators can be statistically quantified by computing the correlation and standardized regression. Parameter influences in each indicator are obtained based on correlation and regression coefficients, as illustrated by the tornado plot shown in Fig. 4.18. The coefficients are plotted in order of the influence of parameters on the performance indicator. The parameter with the most significant influence on the indicator is displayed on the top, giving the plot a tornado shape. A negative coefficient indicates an inversely proportional relationship between the performance indicator and the parameter, while a positive coefficient implies a directly proportional relationship between them.



(a)



(b)

Fig. 4.17. Scatter plot of performance indicators against parameter variation (normalized), (a) settling time, overshoot, and RMSE of speed, (b) torque and current ripples.

The coefficient values are in the [-1, 1] range, and high parameter influences are indicated by the values approaching 1 or -1. Correlation and regression coefficient values

close to 1 imply a (direct) linear relation between the performance indicator and the respected parameter. While values close to -1 mean an (inverse) linear relation between them.

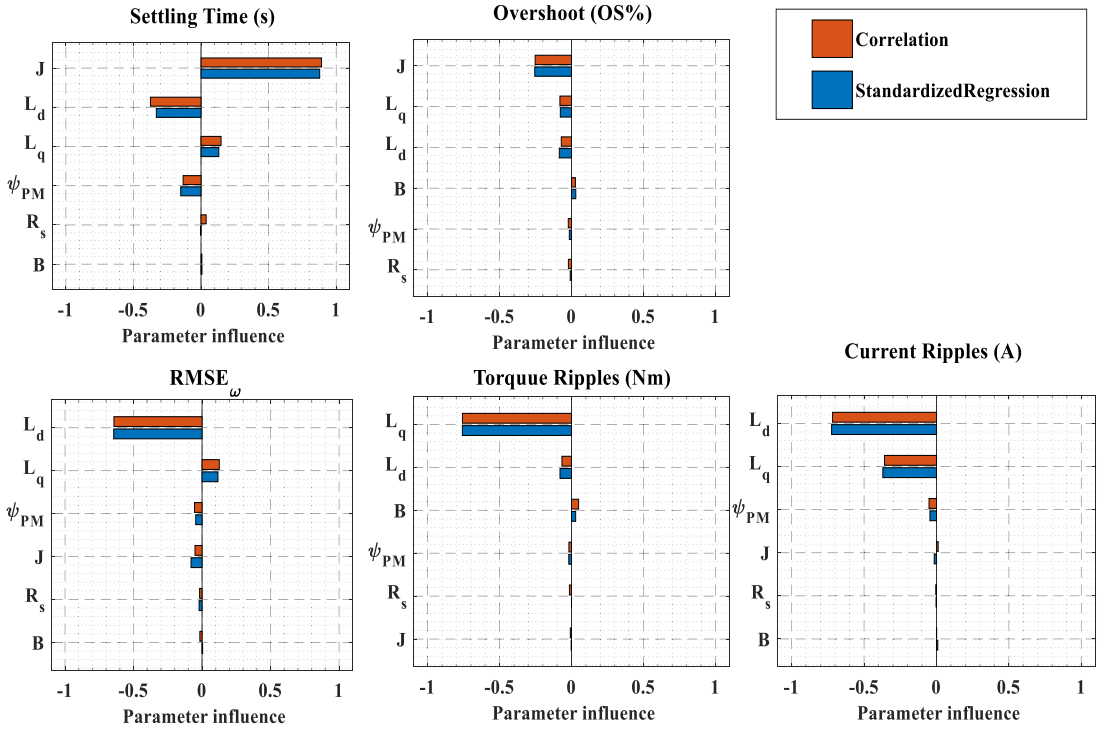


Fig. 4.18. Tornado plot of parameter influences on performance indicators.

The parameter influences presented in Fig. 4.18 indicate that settling time and overshoot indicators are highly sensitive to the variation of machine inertia (J). RMSE of speed is sensitive to d-axis inductance (L_d). Torque ripple is significantly influenced by the q-axis inductance (L_q) variation. The current ripple is highly sensitive to the variation in the d-axis inductance (L_d), followed by the variation of q-axis inductance (L_q).

In summary, the machine inductances (L_d, L_q) and inertia (J) are the most parameters that influence the performance indicators. Transient performance indicators are significantly affected by the variation in machine inertia, while steady-state indicators are highly affected by the variation in machine inductance. However, the impact in transient indicators (e.g.,

settling time and overshoot) is relatively small compared to that in steady-state indicators (e.g., torque and current ripples). For instance, the highest settling time and overshoot values are 30.5% and 37.45% than those obtained at nominal parameters. In contrast, the highest torque and current ripples are 269.2% and 325.5% higher than those obtained at nominal parameters.

4.6.4 Robustness Evaluation

While sensitivity analysis provides valuable insights into the impact of parameter variations on a PMSM control method's performance, it alone is insufficient to indicate the control system's robustness to motor parameter variations. Sensitivity analysis focuses on quantifying the influence of parameters on the system's performance indicators and identifying the parameters with significant impact. Thus, the simulation, experimental, quantitative, and sensitivity analyses of MPC and RPC methods give a general overview of the robustness of these controllers. However, the simulation, experimental, and quantitative analyses only consider single uncertainties cases (deterministic approach), and sensitivity analysis only shows which parameters highly influence the performance. These analyses do not precisely determine how well (robustness level) a drive system performed (compared to other methods) in the presence of uncertainties. Hence, the proposed robustness evaluation method based on Six Sigma numerically identifies the drive system's robustness.

PMSM drive robustness can be divided into two types: stability and performance. Stability is necessary for performance robustness, meaning the stability robustness level is much larger than the performance robustness level. Therefore, only performance robustness is considered in this evaluation. To evaluate the performance robustness of PMSM drives, indicators that essentially indicate the performance are to be identified. Some of these

indicators are the transient response characteristics such as settling time (T_S) and overshoot (OS), and steady-state characteristics such as root mean square error ($RMSE$) of speed, torque, and current ripples. Then, each indicator's robustness acceptance level (USL) needs to be defined. For instance, a torque ripple indicator with USL of 0.4 means the torque ripples of a controller must not exceed 0.4 Nm over different uncertainties to be considered robust. To select appropriate USLs for all indicators, the performance requirement of a specific PMSM drive application is considered.

PMSM drives can be used in several applications, such as water-pumping systems, EVs, aircraft flight control, radar systems, and satellites. Each application has different performance requirements; for example, water pumping can operate with low PMSM drive performance, EVs may require moderate drive performance, and applications like radar systems require high drive performance. Thus, the performance indicators K_i with specification limits considering applications with low requirements (Application-I), moderate requirements (Application-II), and high requirements (Application-III) are listed in Table 4.6. Therefore, a robustness evaluation model of PMSM drive with K_i performance indicators and their USL_i can be defined as follows:

$$\mathrm{K} = \begin{bmatrix} K_1 \\ K_2 \\ K_3 \\ K_4 \\ K_5 \end{bmatrix} = \begin{bmatrix} Ts(X_c) \\ OS(X_c) \\ RMSE_{\omega}(X_c) \\ T_{rip}(Xc) \\ i_{rip}(Xc) \end{bmatrix} \leq \mathrm{USL} \tag{4.29}$$

where $X_c = [R_{s_n}, L_{d_n}, L_{q_n}, \psi_{PM_n}, J_i, B_i]$ represent the machine parameters with($i = 1 \ldots . N$) variation samples. From Table 4.6, USLs are:

$$USL = \begin{bmatrix} 0.2 \\ 0.05 \\ 0.003 \\ 0.8 \\ 1.2 \end{bmatrix} \tag{4.30}$$

for Application-I,

$$USL = \begin{bmatrix} 0.15 \\ 0.03 \\ 0.002 \\ 0.6 \\ 0.7 \end{bmatrix} \tag{4.31}$$

for Application-II, and

$$USL = \begin{bmatrix} 0.1 \\ 0.02 \\ 0.001 \\ 0.4 \\ 0.5 \end{bmatrix} \tag{4.32}$$

for Application-III.

Table 4.6 Performance requirements of different PMSM drive applications.

Indicator (K)	Specification limits (USL)		
	Application-I	Application-II	Application-III
Ts	≤ 0.2	≤ 0.15	≤ 0.1
OS	$\leq 5\%$	$\leq 3\%$	$\leq 2\%$
$RMSE_{\omega}$	≤ 0.003	≤ 0.002	≤ 0.001
T_{rip}	≤ 0.8	≤ 0.6	≤ 0.4
i_{rip}	≤ 1.2	≤ 0.7	≤ 0.5

The conventional MPC and the five RPC methods are evaluated for performance robustness based on the proposed six-sigma robustness evaluation method. The evaluation is done based on the PMSM drive with nominal parameters (Table 3.1) and their uncertainties ranges (Table 4.1) with 10000 (*N*) samples of parameter variations and based on three different application requirements (Table 4.6). First, machine parameter variation samples are generated for two ranges (rated and maximum conditions). The machine operates from a standstill to 1000 rpm (rated speed), and then 2 Nm load torque is applied during steady-state. The six control methods (MPC, RPC-I, RPC-II, RPC-III, RPC-IV, and RPC-V) are evaluated by computing the settling time (T_s), overshoot (OS), RMSE of speed ($RMSE_\omega$), torque ripples (T_{rip}), and stator current i_a ripples (i_{rip}) at every parameter variation sample. The robustness evaluation process is illustrated by the flowchart presented in Fig. 4.19.

The Z values of settling time (Z_{T_s}), overshoot (Z_{OS}), RMSE of speed (Z_ω), torque ripple ($Z_{T_{rip}}$), and current ripple ($Z_{i_{rip}}$) performance indicators, and the sigma level (n_{sys}) and the probability of failure of system performance (POF_{sys}) are computed as in Table 4.7. As can be seen from the obtained n_{sys} in Table 4.7, at the rated parameter uncertainty (variation) ranges and application with low requirement (Application-I), both controllers, including MPC, achieved 6σ. With a moderate application requirement (Application-II), MPC and RPC-I only achieved 2.2σ and 3.6σ, respectively, while other RPC methods achieved 6σ. As for applications with high requirements (Application-III), no controller achieved 6σ, and the highest system sigma level is 2.8σ achieved by RPC-V compared to 0σ, 2.0σ , 2.2σ 2.1σ and 2.5σ achieved by MPC, RPC-I, RPC-II, RPC-III, and RPC-IV, respectively.

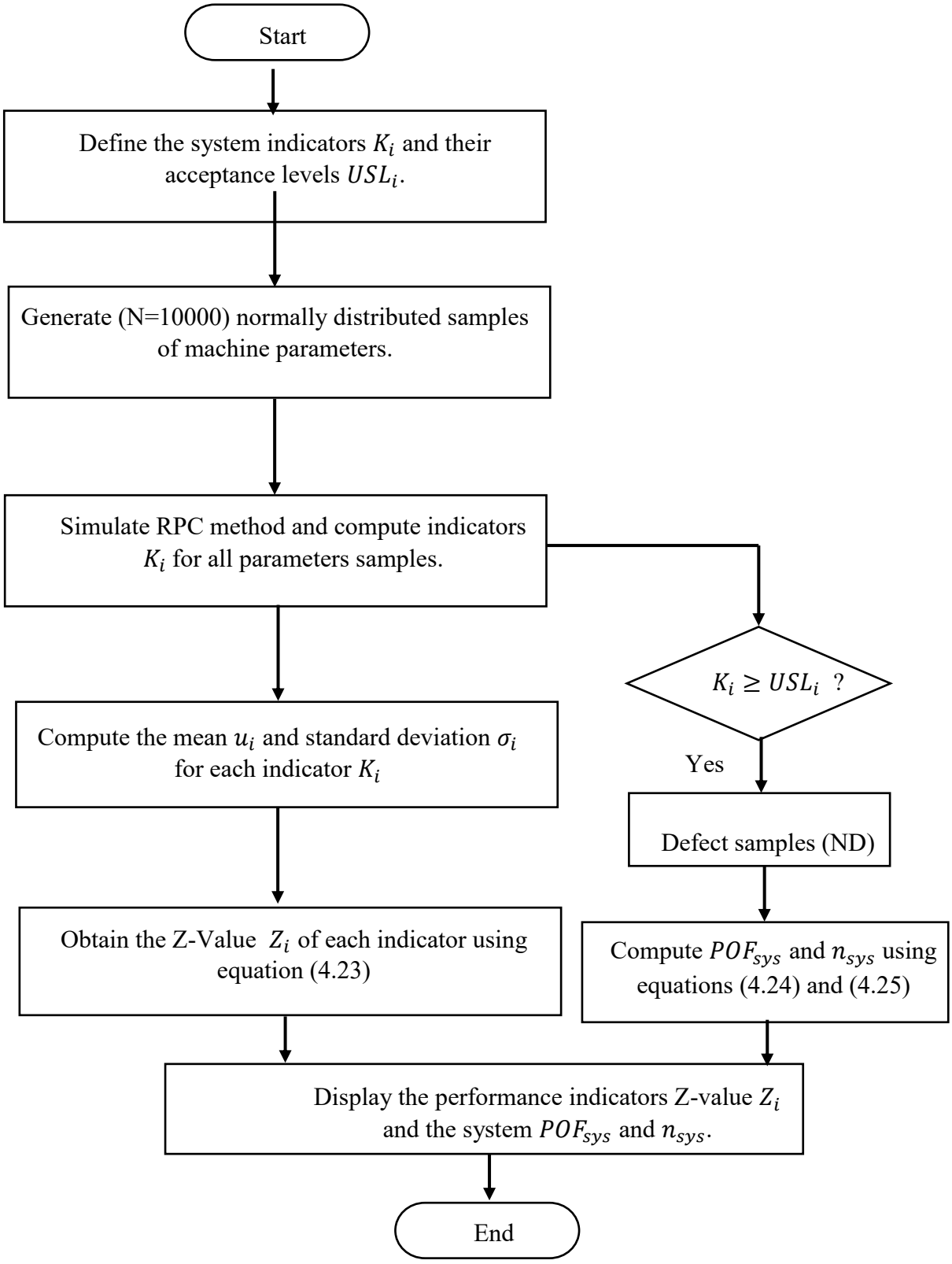


Fig. 4.19. Flow chart of robustness evaluation process.

Table 4.7 Robustness evaluations of MPC, RPC-I, RPC-II, RPC-III, RPC-IV, and RPC-V with two uncertainty ranges based on three applications' requirements.

Indicator	Z_{T_s}	Z_{OS}	Z_{ω}	$Z_{T_{ripp}}$	$Z_{i_{ripp}}$	n_{sys}	POF
Controller	Rated condition uncertainties (Application-I)						
MPC	63.3	66.2	54.6	16.7	9.1	6.0	0
RPC-I	73.4	207.8	173.9	18.6	14.7	6.0	0
RPC-II	73.7	162.3	122.2	28.8	24.6	6.0	0
RPC-III	73.7	42.4	113.3	27.5	23.8	6.0	0
RPC-IV	49.9	12.7	132.8	37.7	18.4	6.0	0
RPC-V	82.2	53.6	50.9	35.5	25.9	6.0	0
	Maximum condition uncertainties (Application-I)						
MPC	46.8	54.4	30.7	10.2	4.9	3.1	0.19%
RPC-I	55.6	163.3	111.0	11.5	8.1	3.6	0.03%
RPC-II	55.8	105.7	83.0	18.6	15.6	6.0	0
RPC-III	55.8	36.3	71.7	17.3	14.6	6.0	0
RPC-IV	35.2	12.2	94.0	22.9	11.8	6.0	0
RPC-V	61.4	52.1	43.9	21.0	14.3	6.0	0
	Rated condition uncertainties (Application-II)						
MPC	44.4	38.5	35.1	11.4	2.2	2.2	2.78%
RPC-I	51.9	120.9	113.7	12.7	5.5	3.6	0.03%
RPC-II	52.1	95.8	79.2	17.0	8.5	6.0	0
RPC-III	52.1	25	73.1	16.2	8.2	6.0	0
RPC-IV	35.2	6.9	86.7	26.6	7.1	6.0	0
RPC-V	58.7	31.6	31.4	24.8	10.0	6.0	0
	Maximum condition uncertainties (Application II)						
MPC	32.7	31.6	19.6	6.9	0.9	1.5	13.36%
RPC-I	39.2	95.0	72.6	7.8	2.8	2.4	1.64%
RPC-II	39.4	62.3	53.8	10.9	5.2	3.3	0.10%
RPC-III	36.4	21.4	46.2	10.0	4.8	3.2	0.14%
RPC-IV	24.8	6.7	61.3	16.0	4.3	3.1	0.19%

RPC-V	43.8	30.7	27.0	14.6	5.3	3.8	0.01%
	Rated condition uncertainties (Application-III)						
MPC	25.6	24.7	15.7	6.1	-0.6	0.4	70.85%
RPC-I	30.4	77.5	53.6	6.8	1.8	2.0	4.55%
RPC-II	30.5	62.5	36.1	5.1	2.1	2.2	2.78%
RPC-III	30.4	16.2	32.8	4.8	2.0	2.1	3.573%
RPC-IV	20.5	4.1	40.6	15.5	2.5	2.5	1.24%
RPC-V	35.2	20.6	11.9	14.1	3.7	2.8	0.51%
	Maximum condition uncertainties (Application-III)						
MPC	18.7	20.2	8.6	3.6	-0.7	0.3	76.65%
RPC-I	22.8	60.8	34.1	4.4	0.7	1.4	16.15%
RPC-II	22.9	40.6	24.4	3.1	1.0	1.5	13.36%
RPC-III	22.9	13.8	20.6	2.8	0.9	1.5	13.36%
RPC-IV	14.4	4.0	28.6	9.3	1.3	1.7	8.91%
RPC-V	26.1	20.0	10.2	8.2	1.7	2.0	4.55%

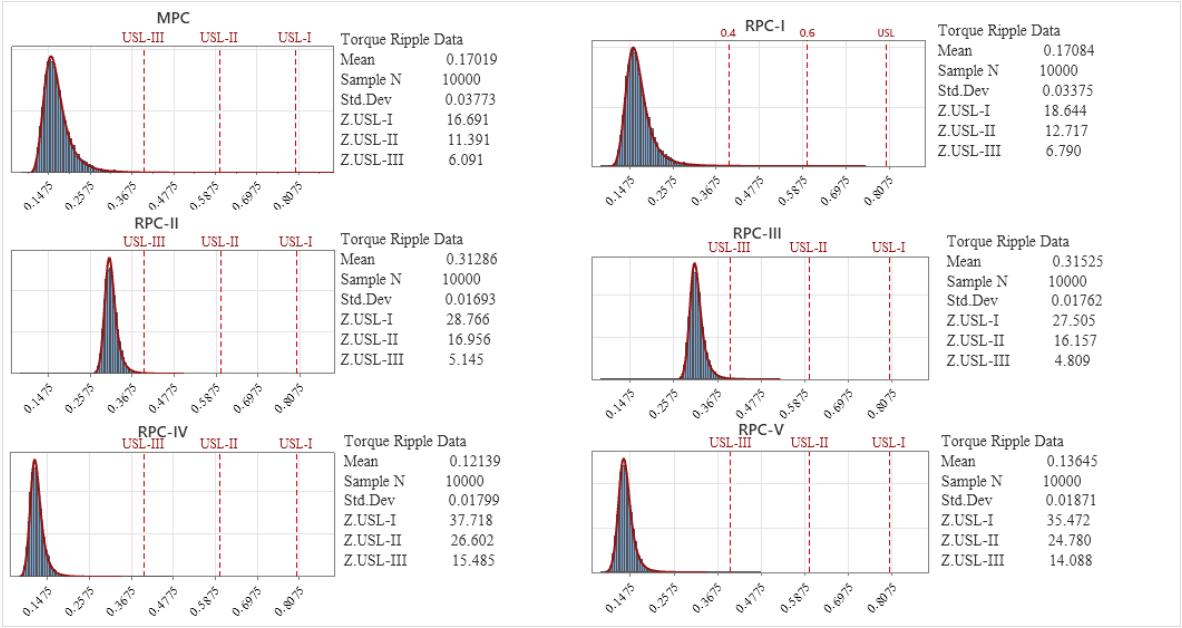
At the maximum parameter uncertainty ranges, MPC and RPC-I only achieved 3.1σ and 3.6σ, respectively, and other RPCs achieved 6σ for Application-I. No controller achieved 6σ at maximum parameter uncertainty ranges for moderate and high application requirements, and RPC-V achieved the highest sigma level with 3.8σ and 2.0σ for Application-II and Application-III, respectively. The increase in uncertainty ranges critically influences the robustness, especially for applications with high requirements. For instance, for Application-III, at the maximum uncertainty ranges, low system performance robustness was recorded for MPC, RPC-I, RPC-II, and RPC-III with 100%, 16.15%,13.36%, and 13.36% POF, respectively. RPC-V achieved the best sigma level and low POF at various uncertainty ranges and for different application requirements.

The current and torque ripples are the most critical performance indicators and the

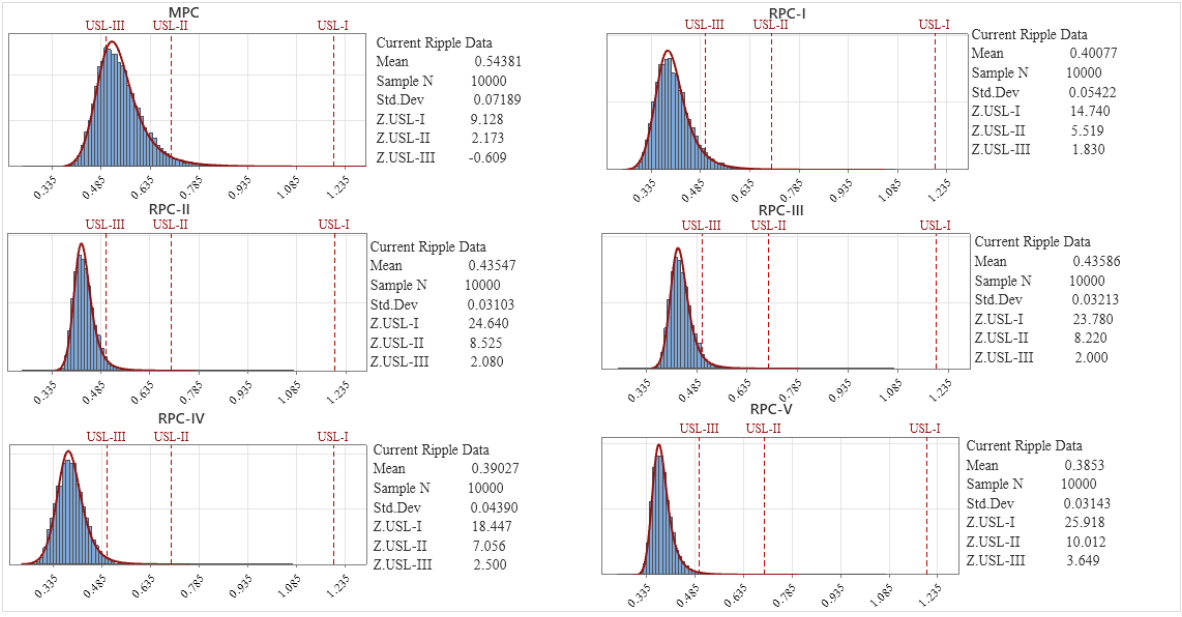
main factor for dropping $n\sigma$ for the overall system. In other words, although some controllers have good robustness for most of the performance indicators, their low robustness levels of torque and/or current ripples result in low $n\sigma$ for overall system performance. In contrast, controllers which maintain trade-offs among torque and current ripples and other performance indicators achieved a good overall system sigma level.

In addition, the Z-value of individual performance indicators is used to show the robustness difference for different controllers, especially when multiple controllers achieve a similar system's sigma level. The Z-value describes how far the specification limits of each application are from the average value of the N-sample data of each performance indicator. The Z values of torque and current ripples are the most critical indicators for a controller's robustness. Thus, to illustrate the concept of Z-value, the process capability of the torque and current ripple indicators with rated and maximum uncertainty ranges are shown in Figs 4.20 and 4.21. USL-I, USL-II, and USL-III, and Z.USL-I, Z.USL-II, and Z.USL-III are the specification limits (Table 4.6) and Z-values for applications I, II, and III, respectively.

The process capability plots show how far the specification limits positions of different applications from the mean (μ) of torque and current ripples of each controller. The dispersion of torque and current ripples around the mean (μ) shows how good each controller is in maintaining minimum torque or current ripples with parameter variations. For example, the process capability plots of torque ripples show RPC-IV's effectiveness in minimizing torque ripples. Similarly, the current ripple capability plots illustrate how good RPC-V is in maintaining low current ripples over different parameter variations. Thus, RPC-IV produced the highest Z-values of torque ripples, and RPC-V produced the highest Z-values of current ripples at all parameter variation ranges.

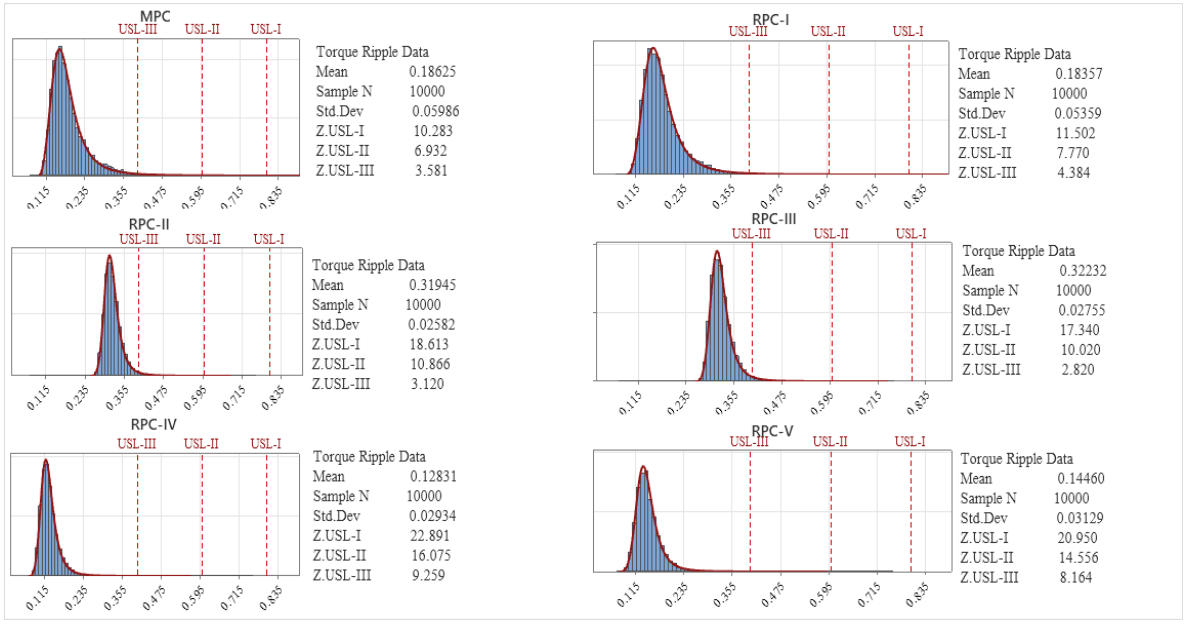


(a)

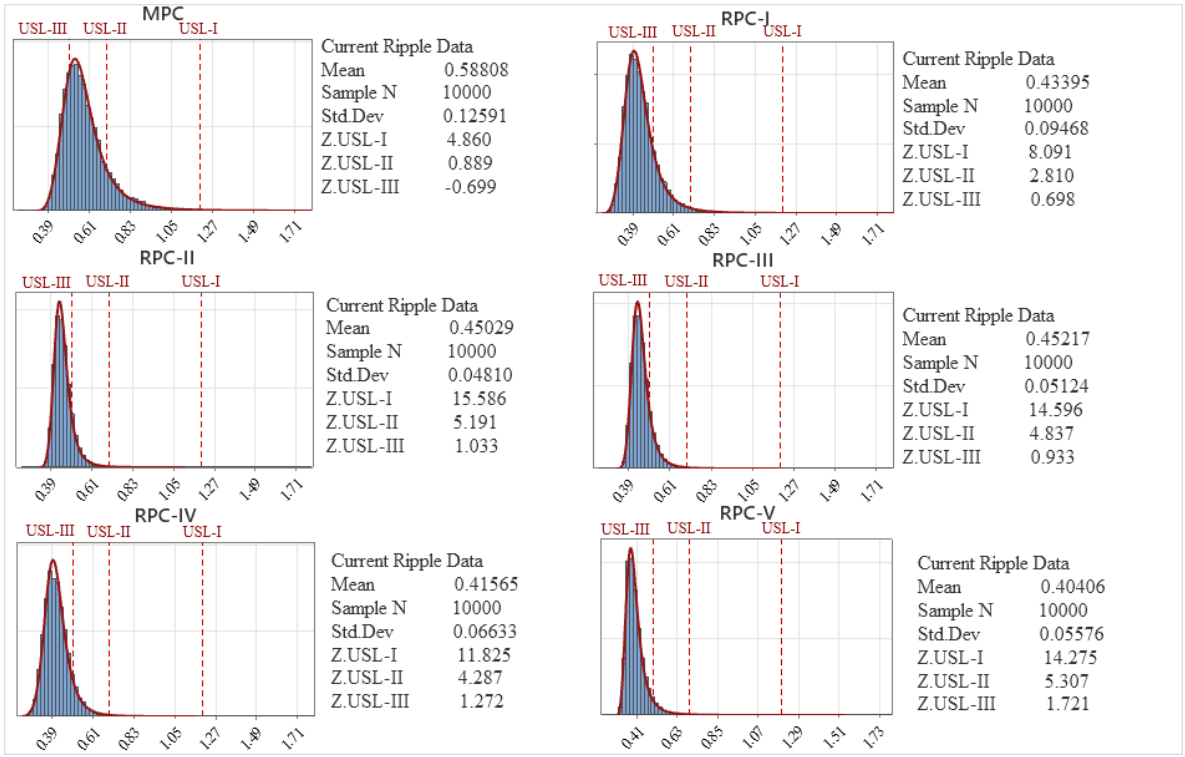


(b)

Fig. 4.20. Process capability plot with rated parameters uncertainties range for MPC, RPC-I, RPC-II, RPC-III, RPC-IV, and RPC-V, (a) torque ripple(T_{rip}), (b) current ripple (i_{rip}).



(a)



(b)

Fig. 4.21. Process capability plot with maximum parameters uncertainties range for MPC, RPC-I, RPC-II, RPC-III, RPC-IV, and RPC-V, (a) torque ripple(T_{rip}), (b) current ripple (i_{rip}).

The Z values of different performance indicators show the strengths and weaknesses of different controllers to specific indicators and which controller can maintain a robustness trade-off among all the indicators. For instance, RPC-I has the highest Z-values of overshoot (Z_{OS}) and RMSE of speed (Z_{ω}), but their Z-values of torque and current ripples decrease significantly as the uncertainties increase. RPC-IV also has the best Z-values of torque ripples, but their Z-values of overshoot are the worst. Hence, achieving good robustness for all performance indicators is essential to obtain good overall system robustness. For example, MPC and RPC-I have achieved good speed performance robustness ($T_s, OS, RMSE_{\omega}$), but they were unable to maintain good torque and current (T_{rip}, i_{rip}) robustness over different parameter uncertainty ranges. On the other hand, RPC-IV and RPC-V may have achieved less overshoot robustness than MPC and RPC-I, but they maintained a robustness trade-off with other indicators, thus achieving higher system sigma levels.

The proposed six-sigma robustness evaluation method offers a simple and reliable robustness evaluation tool, which can be used to assess any system's robustness to uncertainties. This means the robustness level of a control system can be determined by evaluating its quality indicators against defined acceptance levels (specification limits) for N samples of uncertainties. Therefore, when considering a specific application requirement (e.g., EV), the best controller that is more robust to uncertainties can be selected effectively. The control methods and the application requirements used in this research are just examples to illustrate the proposed six-sigma robustness evaluation method. However, other control methods for PMSM drives, systems, and application performance requirements can be used. Hence, the proposed method is a quality measure of a control system that can be used to numerically assess the robustness of any control system to uncertainties.

4.7 Summary

Various robust PMSM drive methods, including RPC, have been introduced to deal with the different drive uncertainties. Most of these methods lack discussion on uncertainties modelling, robustness definition, and evaluations. Thus, in this research, a novel robustness evaluation method based on the Six Sigma concept is proposed and used to evaluate control system robustness numerically. Based on the proposed method, five RPC methods of PMSM drive and conventional MPC are assessed at two uncertainty ranges and considering three different application requirements. Different robustness levels are obtained for each method in the presence of uncertainties. Besides, considering different application requirements, the robustness levels of RPC methods differ accordingly. With such robustness evaluations, selecting the best RPC method that fits the required applications is more accessible.

REFERENCES

[4.1] M. Siami, D. A. Khaburi, A. Abbaszadeh, and J. Rodríguez, "Robustness Improvement of Predictive Current Control Using Prediction Error Correction for Permanent-Magnet Synchronous Machines," *IEEE Transactions on Industrial Electronics,* vol. 63, no. 6, pp. 3458-3466, 2016.

[4.2] F. Wang, K. Zuo, P. Tao, and J. Rodriguez, "High Performance Model Predictive Control for PMSM by using Stator Current Mathematical Model Self-regulation Technique," *IEEE Transactions on Power Electronics,* 2020.

[4.3] S.-W. Kang, J.-H. Soh, R.-Y. Kim, K.-J. Lee, and S.-I. Kim, "Robust predictive current control for IPMSM without rotor flux information based on a discrete-time disturbance observer," *IET Electric Power Applications,* vol. 13, no. 12, pp. 2079-2089, 2019.

[4.4] M. Yang, X. Lang, J. Long, and D. Xu, "Flux immunity robust predictive current control with incremental model and extended state observer for PMSM drive," *IEEE Transactions on Power Electronics,* vol. 32, no. 12, pp. 9267-9279, 2017.

[4.5] F. Wang, L. He, and J. Rodriguez, "FPGA based Continuous Control Set Model Predictive Current Control for PMSM System Using Multi-step Error Tracking Technique," *IEEE Transactions on Power Electronics,* 2020.

[4.6] Y. Zhou, H. Li, and H. Zhang, "Model-free deadbeat predictive current control of a surface-mounted permanent magnet synchronous motor drive system," *Journal of Power Electronics,* vol. 18, no. 1, pp. 103-115, 2018.

[4.7] C. Lin, T. Liu, J. Yu, L. Fu, and C. Hsiao, "Model-Free Predictive Current Control for Interior Permanent-Magnet Synchronous Motor Drives Based on Current Difference Detection Technique," *IEEE Transactions on Industrial Electronics,* vol. 61, no. 2, pp. 667-681, 2014.

[4.8] R. Errouissi, M. Ouhrouche, W.-H. Chen, and A. M. Trzynadlowski, "Robust nonlinear predictive controller for permanent-magnet synchronous motors with an optimized cost function," *IEEE Transactions on Industrial Electronics,* vol. 59, no. 7, pp. 2849-2858, 2011.

[4.9] M. Preindl and S. Bolognani, "Model predictive direct torque control with finite control set for PMSM drive systems, Part 1: Maximum torque per ampere operation," *IEEE Transactions on Industrial Informatics,* vol. 9, no. 4, pp. 1912-1921, 2013.

[4.10] W. Xie *et al.*, "Finite-control-set model predictive torque control with a deadbeat solution for PMSM drives," *IEEE Transactions on Industrial Electronics,* vol. 62, no. 9, pp. 5402-5410, 2015.

[4.11] Z. Zhou, C. Xia, Y. Yan, Z. Wang, and T. Shi, "Disturbances attenuation of permanent magnet synchronous motor drives using cascaded predictive-integral-resonant controllers," *IEEE Transactions on Power Electronics,* vol. 33, no. 2, pp. 1514-1527, 2017.

[4.12] Y. Liu, S. Cheng, B. Ning, and Y. Li, "Robust model predictive control with simplified repetitive control for electrical machine drives," *IEEE Transactions on Power Electronics,* vol. 34, no. 5, pp. 4524-4535, 2018.

[4.13] G. E. Dullerud and F. Paganini, *A course in robust control theory: a convex approach*. Springer Science & Business Media, 2013.

[4.14] K.-Z. Liu and Y. Yao, *Robust control: theory and applications*. John Wiley & Sons, 2016.

[4.15] R. Fales, "Uncertainty modeling and predicting the probability of stability and performance in the manufacture of dynamic systems," *ISA transactions,* vol. 49, no. 4, pp. 528-534, 2010.

[4.16] T. M. Jahns and W. L. Soong, "Pulsating torque minimization techniques for permanent magnet AC motor drives-a review," *IEEE transactions on industrial electronics,* vol. 43, no. 2, pp. 321-330, 1996.

[4.17] T. Sebastian, "Temperature effects on torque production and efficiency of PM motors using NdFeB magnets," *IEEE Transactions on Industry Applications,* vol. 31, no. 2, pp. 353-357, 1995.

[4.18] D. Leggate and R. J. Kerkman, "Pulse based dead time compensator for PWM voltage inverters," in *Proceedings of IECON'95-21st Annual Conference on IEEE Industrial Electronics*, 1995, vol. 1: IEEE, pp. 474-481.

[4.19] J. Yang, W.-H. Chen, S. Li, L. Guo, and Y. Yan, "Disturbance/Uncertainty Estimation and Attenuation Techniques in PMSM Drives—A Survey," *IEEE Transactions on Industrial Electronics,* vol. 64, no. 4, pp. 3273-3285, 2017, doi: 10.1109/tie.2016.2583412.

[4.20] R. Yang, M.-Y. Wang, L.-Y. Li, C.-M. Zhang, and J.-L. Jiang, "Robust predictive current control with variable-gain adaptive disturbance observer for PMLSM," *IEEE Access,* vol. 6, pp. 13158-13169, 2018.

[4.21] S. Li and Z. Liu, "Adaptive speed control for permanent-magnet synchronous motor system with variations of load inertia," *IEEE transactions on industrial electronics,* vol. 56, no. 8, pp. 3050-3059, 2009.

[4.22] Z.-H. Liu, H.-L. Wei, X.-H. Li, K. Liu, and Q.-C. Zhong, "Global identification of electrical and mechanical parameters in PMSM drive based on dynamic self-learning PSO," *IEEE Transactions on Power Electronics,* vol. 33, no. 12, pp. 10858-10871, 2018.

[4.23] E. M. Tofighi, A. Mahdizadeh, and M. R. Feyzi, "Real-time estimation and tracking of parameters in permanent magnet synchronous motor using a modified two-stage particle swarm optimization algorithm," in *2013 IEEE International Symposium on Sensorless Control for Electrical Drives and Predictive Control of Electrical Drives and Power Electronics (SLED/PRECEDE)*, IEEE, pp. 1-7, 2013.

[4.24] C. Lai, G. Feng, K. Mukherjee, and N. C. Kar, "Investigations of the influence of PMSM parameter variations in optimal stator current design for torque ripple minimization," *IEEE Transactions on Energy Conversion,* vol. 32, no. 3, pp. 1052-1062, 2017.

[4.25] P. N. Koch, R.-J. Yang, and L. Gu, "Design for six sigma through robust optimization," *Structural and Multidisciplinary Optimization,* vol. 26, no. 3, pp. 235-248, 2004.

[4.26] G. Lei, T. Wang, J. Zhu, Y. Guo, and S. Wang, "System-level design optimization method for electrical drive systems—Robust approach," *IEEE Transactions on Industrial Electronics,* vol. 62, no. 8, pp. 4702-4713, 2015.

[4.27] C.-S. Jun and B.-I. Kwon, "Process capability control procedure for electrical machines by using a six-sigma process for achieving six-sigma quality level," *IET Electric Power Applications,* vol. 11, no. 8, pp. 1466-1474, 2017/09/01 2017, doi: https://doi.org/10.1049/iet-epa.2017.0011.

[4.28] X. Zhang, L. Zhang, and Y. Zhang, "Model Predictive Current Control for PMSM Drives With Parameter Robustness Improvement," *IEEE Transactions on Power Electronics,* vol. 34, no. 2, pp. 1645-1657, 2019, doi: 10.1109/TPEL.2018.2835835.

[4.29] Y. Zhang and J. Zhu, "Direct Torque Control of Permanent Magnet Synchronous Motor With Reduced Torque Ripple and Commutation Frequency," *IEEE Transactions on Power Electronics,* vol. 26, no. 1, pp. 235-248, 2011, doi: 10.1109/TPEL.2010.2059047.

[4.30] A. Saltelli, S. Tarantola, F. Campolongo, and M. Ratto, *Sensitivity analysis in practice: a guide to assessing scientific models*. Wiley Online Library, 2004.

[4.31] E. Borgonovo and E. Plischke, "Sensitivity analysis: A review of recent advances," *European Journal of Operational Research,* vol. 248, no. 3, pp. 869-887, 2016.

CHAPTER 5

ADAPTIVE MODEL-FREE PREDICTIVE CURRENT CONTROL OF PMSM DRIVES

5.1 Introduction

Model-free predictive current control (MFPCC) has recently emerged as a promising alternative to robust MPC methods. MFPCC eliminates the prediction dependency on a simplified parametric machine model by developing a prediction model independent of the machine model and parameters [5.1]. MFPCC can be achieved using an ultra-local model [5.2], where an ultra-local model replaces a complex system model with one or two unknowns that can be estimated based on the system's measured data [5.3-5.4].

On the other hand, MFPCC can be achieved by solely using the system's measured input and/or output data and their variations. In such methods, the current differences due to the possible voltage vectors are stored and employed for predicting the future current. The current differences due to the applied voltages over the past one, two, or three control cycles are typically used to predict the current in the next control cycle. These current differences are updated continuously as new voltage vectors are applied [5.5-5.7]. However, when the same voltage vector is applied for an extended control period, the update mechanism is corrupted, and stagnation can occur, resulting in an inaccurate prediction and performance degradation.

Most existing MFPCCs utilize the measured current and applied voltage variations to compensate for the effect of parameter variations. However, parameter inaccuracies influence the reference current when a speed control loop is used. This results in suboptimal

tracking performance and high current ripple. Thus, the tracking error must be considered in the prediction stage to account for the effect of parameter variations in the speed control loop.

This research proposes an adaptive MFPCC (A-MFPCC) with a modified current difference updating technique for PMSM drives. First, an incremental prediction model with two lumped parameters is derived. Then, using the recursive least square (RLS) algorithm to estimate these parameters, a model-free current prediction can be achieved in a similar manner to MFPCC based ultra-local model. To avoid the additional RLS computation requirement and account for tracking error, the derived incremental prediction model and tracking error variations are used to establish a prediction model independent of the machine model and parameters. Thus, the measured and reference currents deviations due to parameter mismatching can be eliminated. With a reference voltage vector generated based on the reference current vector position and the tracking error, the current difference is obtained due to the applied and reference voltage vector. Using a reference voltage vector accounts for the tracking error in each control cycle and avoids stagnation by constantly updating the current differences.

In comparison to existing methods, the proposed A-MFPCC does not require two or three successive applied voltage vectors to be different. Thanks to the generated reference voltage vector, which adaptively updates the current differences despite the successively applied voltage vectors being similar. With the proposed A-MFPCC, the stagnation effect is eliminated, and tracking error is considered in the prediction stage. Thus, effective current prediction and better tracking performance can be achieved. The effectiveness of the proposed method is validated by comparison with two other MFPCC schemes based on simulation, experimental results, and robustness evaluation.

5.2 The Proposed Adaptive Model-Free Predictive Current Control (A-MFPCC)

The currents dynamic equations of PMSM in dq-frame are presented in Chapter 3 and rewritten here as follows:

$$\frac{di_d}{dt} = -\frac{R_s}{L_d} i_d + \frac{L_q}{L_d}\, \omega i_q + \frac{1}{L_d} v_d \tag{5.1}$$

$$\frac{di_q}{dt} = -\frac{R_s}{L_q} i_q - \frac{L_d}{L_q}\, \omega i_d - \frac{\omega\, \Psi_{pm}}{L_q} + \frac{1}{L_q} v_q \tag{5.2}$$

The currents i_{dq} at time step $(k+1)$ can be predicted by discretizing (3) and (4) at sampling time T_s as follows:

$$i_d(k+1) = \left(1 - \frac{R_s T_s}{L_d}\right) i_d(k) + \frac{L_q}{L_d} T_s\, \omega(k) i_q(k) + \frac{T_s}{L_d} v_d(k) \tag{5.3}$$

$$i_q(k+1) = \left(1 - \frac{R_s T_s}{L_q}\right) i_q(k) - \frac{L_d}{L_q} T_s\, \omega(k) i_d(k) - T_s \omega(k) \frac{\Psi_{pm}}{L_q} + \frac{T_s}{L_q} v_q(k) \tag{5.4}$$

The accuracy of the prediction model in (5.3) and (5.4) is highly dependent on both the machine parameters and the mathematical model of PMSM. However, machine parameters vary due to machine structure and changes in operating conditions. To account for parameters variation, the prediction model is rewritten considering parameters mismatching as follows:

$$i_d(k+1) = \left(1 - T_s \frac{R_s + \Delta R_s}{L_d + \Delta L_d}\right) i_d(k) + T_s \frac{L_q + \Delta L_q}{L_d + \Delta L_d} \omega(k) i_q(k) + \frac{T_s}{L_d + \Delta L_d} v_d(k) \tag{5.5}$$

$$\begin{aligned} i_q(k+1) \;=& \left(1 - T_s \frac{R_s + \Delta R_s}{L_q + \Delta L_q}\right) i_q(k) - \frac{L_d + \Delta L_d}{L_q + \Delta L_q} T_s\, \omega(k) i_d(k) \\ & - T_s \omega(k) \frac{\Psi_{pm} + \Delta\, \Psi_{pm}}{L_q + \Delta L_q} + \frac{T_s}{L_q + \Delta L_q} v_d(k) \end{aligned} \tag{5.6}$$

Based on (5.5) and (5.6), it is evident the machine parameters and their deviations (ΔR_s , ΔL_q, ΔL_d, and $\Delta \Psi_{pm}$) play a significant part in deciding the accuracy of the predicted currents. A mismatch or uncertainty in one of the machine parameters leads to prediction inaccuracy and degrades the control performance. The conventional prediction model in (5.3) and (5.4) achieve current prediction using a one-time step kT_s data. With two-time steps data $(k-1)T_s$ and kT_s, an incremental prediction model can be obtained [5.8]. Based on the predicted currents at time step *k*-1 in (5.3) and (5.4), the current at step k is predicted as follows:

$$i_d(k) = \left(1 - \frac{R_s T_s}{L_d}\right) i_d(k-1) + \frac{L_q}{L_d} T_s\, \omega(k-1) i_q(k-1) + \frac{T_s}{L_d} v_d(k-1) \tag{5.7}$$

$$i_q(k) = \left(1 - \frac{R_s T_s}{L_q}\right) i_q(k-1) - \frac{L_d}{L_q} T_s\, \omega(k-1) i_d(k-1) - T_s \omega \frac{\Psi_{pm}}{L_q} + \frac{T_s}{L_q} v_q(k-1) \tag{5.8}$$

The mechanical speed ω can be assumed constant over a few control cycles because the mechanical time constant τ_m is much larger than the electrical time constants $\tau_d = \frac{L_d}{R_s}, \tau_q = \frac{L_q}{R_s}$. Thus, by subtracting (5.7) and (5.8) from (5.3) and (5.4), the current incremental prediction model at *k+1* can be expressed as:

$$i_d(k+1) = i_d(k) + \left(1 - \frac{R_s T_s}{L_d}\right)\left(i_d(k) - i_d(k-1)\right)$$

$$+ \frac{L_q}{L_d} T_s \omega \left(i_q(k) - i_q(k-1)\right) + \frac{T_s}{L_d}\left(v_d(k) - v_d(k-1)\right) \tag{5.9}$$

$$i_q(k+1) = i_q(k) + \left(1 - \frac{R_s T_s}{L_q}\right)\left(i_q(k) - i_q(k-1)\right)$$

$$- \frac{L_d}{L_q} T_s \omega \left(i_d(k) - i_d(k-1)\right) + \frac{T_s}{L_q}\left(v_q(k) - v_q(k-1)\right) \tag{5.10}$$

With a short enough sampling time T_s, much smaller than electrical time constants τ_d

and τ_q, the current difference deviations $\Delta\delta i_d$ and $\Delta\delta i_d$, as shown in [5.7] and [5.9], can be approximated by:

$$\begin{bmatrix}\Delta\delta i_d \\ \Delta\delta i_q\end{bmatrix} = \begin{bmatrix}\frac{T_s}{L_d}\delta v_d \\ \frac{T_s}{L_q}\delta v_q\end{bmatrix} \tag{5.11}$$

$$\begin{bmatrix}\big(i_d(k+1)-i_d(k)\big) & -\big(i_d(k)-i_d(k-1)\big) \\ \big(i_q(k+1)-i_q(k)\big) & -\big(i_q(k)-i_q(k-1)\big)\end{bmatrix} = \begin{bmatrix}\frac{T_s}{L_d}\big(v_d(k)-v_d(k-1)\big) \\ \frac{T_s}{L_q}\big(v_q(k)-v_q(k-1)\big)\end{bmatrix} \tag{5.12}$$

Based on (5.11) and (5.12), the incremental prediction model in (5.9) and (5.10) can be simplified as follows:

$$i_d(k+1) = i_d(k) + \left(1-\frac{R_sT_s}{L_d}\right)\big(i_d(k)-i_d(k-1)\big) + \frac{T_s}{L_d}\big(v_d(k)-v_d(k-1)\big) \tag{5.13}$$

$$i_q(k+1) = i_q(k) + \left(1-\frac{R_sT_s}{L_q}\right)\big(i_q(k)-i_q(k-1)\big) + \frac{T_s}{L_q}\big(v_q(k)-v_q(k-1)\big) \tag{5.14}$$

For simplicity, (5.13) and (5.14) can be rewritten as:

$$i_s(k+1) = i_s(k) + \alpha\big(i_s(k)-i_s(k-1)\big) + \beta\big(v_s(k)-v_s(k-1)\big) \tag{5.15}$$

where $i_s(k+1) = \big[i_d(k+1)\ \ i_q(k+1)\big]^T$, $i_s(k) = \big[i_d(k)\ \ i_q(k)\big]^T$,

$v_s(k) = [v_d(k)\ v_d(k)]^T$, $\alpha = \begin{bmatrix}1-\frac{R_sT_s}{L_d} & 1-\frac{R_sT_s}{L_q}\end{bmatrix}^T$, and $\beta = \begin{bmatrix}\frac{T_s}{L_d} & \frac{T_s}{L_q}\end{bmatrix}^T$.

5.2.1 MFPCC Based on Recursive Least Square (MFPCC-I)

Based on (5.15), an incremental prediction model is established with two unknown parameters α and β. The currents $i_s(k+1)$ can be predicted by estimating α and β at every

time step kT_s based on the system input and output data. Different methods can be used to estimate α and β, such as sliding mode observer, extended state observer, and other estimation techniques. However, these methods are computationally intensive and incorporate various coefficients to be tuned, and their selections influence the estimation accuracy. Here, an effective and less computational recursive least square (RLS) algorithm is used to estimate α and β based on the current and past measurement of currents and voltages. At every time step *k*, the RLS algorithm estimates α and β which are used to predict the current in the next step $i_s(k+1)$. In general, the RLS algorithm is defined as follows:

$$y(k) = x^T(k)\theta(k) \tag{5.16}$$

where $y(k)$ is the observed output, $\theta(k)$ the vector of unknown parameters, and $x^T(k)$ the input vector. Based on (5.16), the prediction model in (5.15) is rewritten in linear regression form as follows:

$$\underbrace{i_s(k+1) - i_s(k)}_{y(k)} = \underbrace{[i_s(k) - i_s(k-1) \quad v_s(k) - v_s(k-1)]}_{x^T(k)} \underbrace{\begin{bmatrix}\alpha \\ \beta\end{bmatrix}}_{\theta(k)} \tag{5.17}$$

The unknown parameter estimation, $\hat{\theta}(k)$, is computed for every time step *k* based on observed output data $y(k)$ and measured input data $x^T(k)$ as follows:

$$\hat{\theta}(k), = \hat{\theta}(k-1), + K(k)\left[y(k) - x^T(k)\,\hat{\theta}(k-1)\right] \tag{5.18}$$

$$K(k) = \frac{P(k-1)x(k)}{\lambda + x^T(k)P(k-1)x(k)} \tag{5.19}$$

$$P(k) = \frac{1}{\lambda}\left[P(k-1) - K(k)x^T(k)P(k-1)\right] \tag{5.20}$$

where λ is the forgetting factor (a value between 0 and 1 determining how much weight to give to the older data); $K(k)$ is the gain matrix, and $P(k)$ the covariance matrix. The estimation of the unknown $\hat{\theta}$ using the RLS can be summarized in Algorithm 5.1.

Algorithm 5.1: Recursive Least Square

Initialization

$$\hat{\theta}(k-1) = \hat{\theta}(0)$$
$$P(k-1) = P(0)$$

For each time step k=1,2….

- obtain input $x^T(k)$,and estimate output $y(k)$
- Compute the estimation error:

$$\varepsilon(k) = y(k) - x^T(k)\hat{\theta}(k-1)$$

- Compute the gain vector (5.19).
- Update $\hat{\theta}(k)$(5.18), and $P(k)$ (5.20).
- Predict the output for next time step $k+1$

End For

The current at time step k+2 can be predicted as follows:

$$i_s(k+2) = i_s(k+1) + \hat{\alpha}(k+1)\big(i_s(k+1) - i_s(k)\big) + \hat{\beta}(k+1)\big(v_s(k+1) - v_s(k)\big) \quad (5.21)$$

The optimum switching vector is selected by minimizing a cost function as follows:

$$g = \left(i_s^{ref} - i_s^{k+2}\right)^2 \quad (5.22)$$

where i_s^{ref} is the reference current obtained from the speed control loop. The overall working principle of the MFPCC-I-based PMSM drive is illustrated by the block diagram shown in Fig. 5.1 and the flow chart presented in Fig. 5.2.

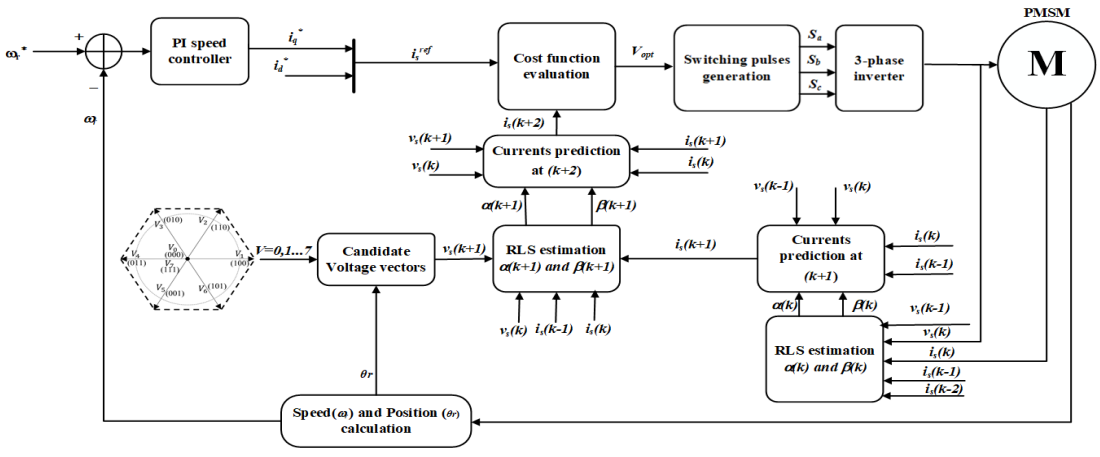


Fig. 5.1. A block diagram of MFPCC-based RLS (MFPCC-I).

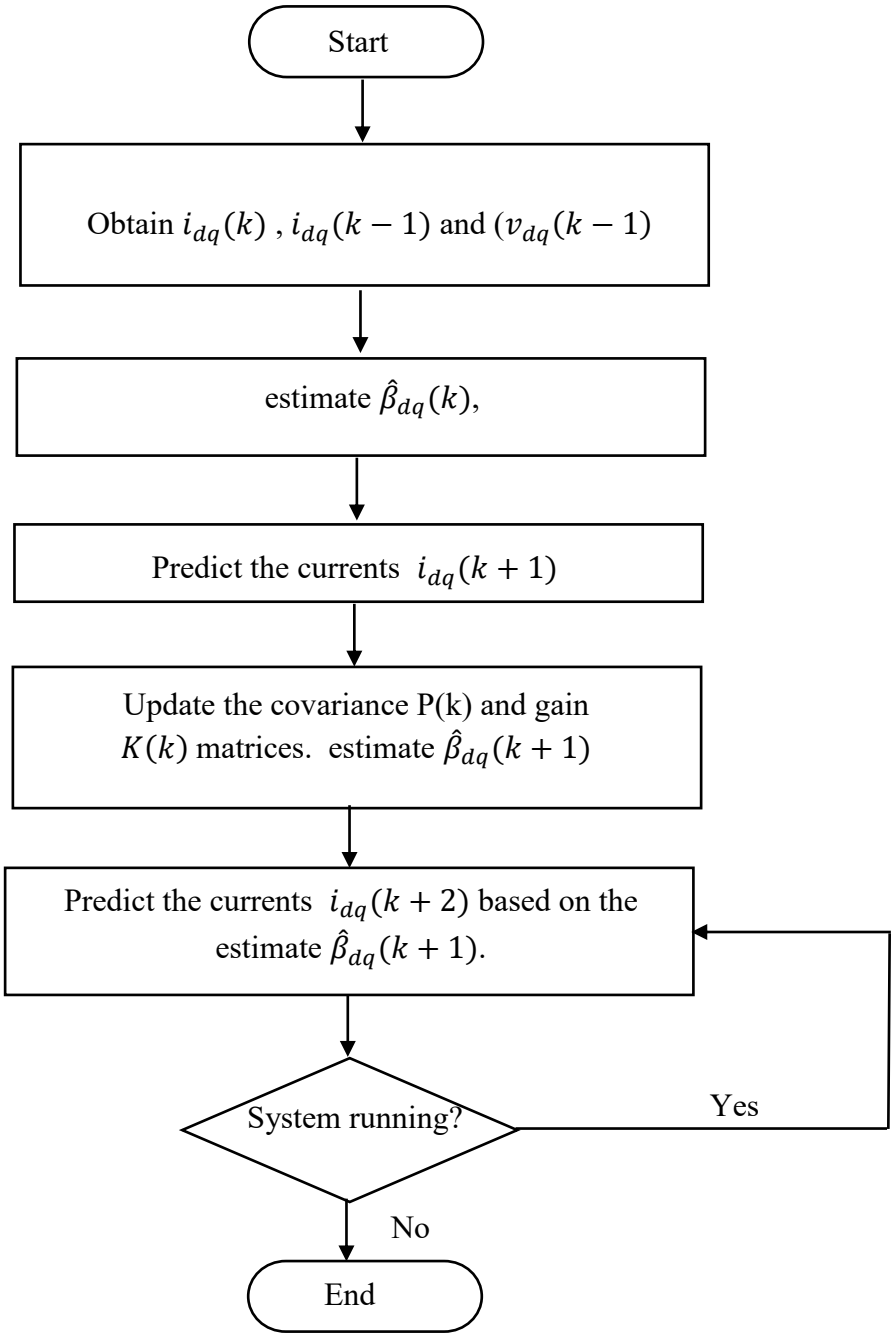


Fig. 5.2. Flow chart of MFPCC based RLS (MFPCC-I).

5.2.2 MFPCC-Based current difference update (MFPCC-II)

The current difference variation due to two successive applied voltage vectors can be expressed based on (5.15) as follows[5.7]:

$$\big(i_s(k) - i_s(k-1)\big) \ - \big(i_s(k-1) - i_s(k-2)\big) = \beta\big(v_s(k) - v_s(k-1)\big) \tag{5.23}$$

For the possible voltage vectors $v_{s_j}, j \in \{0,1,2..7\}$, (5.23) can be written as:

$$\big(i_s(k) - i_s(k-1)\big)_j \ - \big(i_s(k-1) - i_s(k-2)\big) = \beta\left(v_s(k)_j - v_s(k-1)\right) \tag{5.24}$$

With a short enough sampling time (T_s), the machine inductance can be assumed constant over two adjacent sampling instants. Thus, β can be eliminated by combining (5.23) and (5.24) as follows:

$$\frac{\big(i_s(k) - i_s(k-1)\big)_j \ - \big(i_s(k-1) - i_s(k-2)\big)}{\big(i_s(k) - i_s(k-1)\big) \ - \big(i_s(k-1) - i_s(k-2)\big)} = \frac{\left(v_s(k)_j - v_s(k-1)\right)}{\big(v_s(k) - v_s(k-1)\big)} \tag{5.25}$$

Then, the current difference for the possible voltage vector $v_{s_j}, j \in \{0,1,2..7\}$ can be estimated by constantly updating two successive current differences in each control cycle as follows:

$$\big(i_s(k) - i_s(k-1)\big)_j = \big(i_s(k-1) - i_s(k-2)\big)$$

$$+\frac{\left(v_s(k)_j - v_s(k-1)\right)\Big(\big(i_s(k) - i_s(k-1)\big) - \big(i_s(k-1) - i_s(k-2)\big)\Big)}{v_s(k) - v_s(k-1)} \tag{5.26}$$

Then, the current at step time *k+1* is predicted as follows:

$$i_s(k+1)_j = i_s(k) + \big(i_s(k) - i_s(k-1)\big)_j \tag{5.27}$$

where $i_s(k)$ is the measured current at kT_s; $i_s(k+1)_j$ is the estimated current at $(k+1)T_s$ under the possible voltage vector $v_s(k)_j$ applied in kT_s, and $\left(i_s(k) - i_s(k-1)\right)_j$ the current difference caused by the v_{s_j}. Then, the current at $(k+2)T_s$ is predicted as follows:

$$i_s(k+2)_j = i_s(k+1)_j + \left(i_s(k+1) - i_s(k)\right)_j \tag{5.28}$$

where $i_s(k+2)_j$ is the predicted current at$(k+2)T_s$ due to the applied voltage vector $v_s\ (k+1)_j$, and $\left(i_s(k+1) - i_s(k)\right)_j$ is the current difference caused by $v_s(k+1)_j$.

Based on (5.27), (5.28), and Fig. 5.3, it can be seen that the current differences due to the applied voltage vectors are essential for MFPCC-II, as their accuracy can directly affect the current estimations and current predictions. However, for this method to work, two successive voltage vectors cannot be the same. Thus, the update process can be corrupted when only two vectors are applied for a few control cycles, and stagnation may occur.

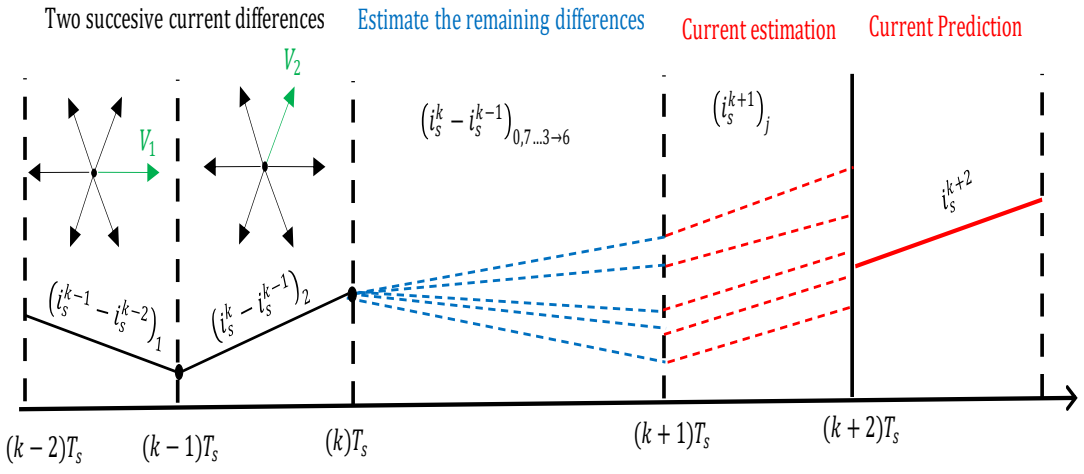


Fig. 5.3. Current difference update technique in MFPCC-II.

5.2.3 MFPCC-Based Adaptive Reference Vector (A-MFPCC)

The measured current and applied voltage variations over two successive control cycles can be used to estimate the effect of parameter variations on the measured currents. However, parameter inaccuracies influence the reference current when a speed control loop is used. Thus, high parameter variations and speed measurement errors produce suboptimal tracking performance and high current ripple. Therefore, it is important to consider current tracking improvement within the prediction stage so that the predicted switching vector minimizes the tracking error.

The tracking performance improvements are not considered in MFPCC-I and MFPCC-II. This may result in high current ripples for these methods with parameter mismatching. Additionally, MFPCC-I utilizes the RLS algorithm, which includes some estimation errors and can be computationally intensive. In MFPCC-II, the current difference estimation requires the last two successive applied voltage vectors to be different, which may activate only two vectors for an extended control interval, resulting in stagnation. Long stagnation significantly produces inaccurate predictions and degrades performance.

In this research, a reference voltage vector is generated based on the reference current vector position and the tracking error and used to obtain the current difference due to the applied and reference voltage vector. Then, the current difference due to the possible voltage vector is estimated without using any machine parameter. The tracking performance is improved by considering the tracking error in the prediction stage, and the reference vector adaptively updates the current difference to avoid stagnation. A block diagram of the proposed A-MFPCC is presented in Fig. 5.4.

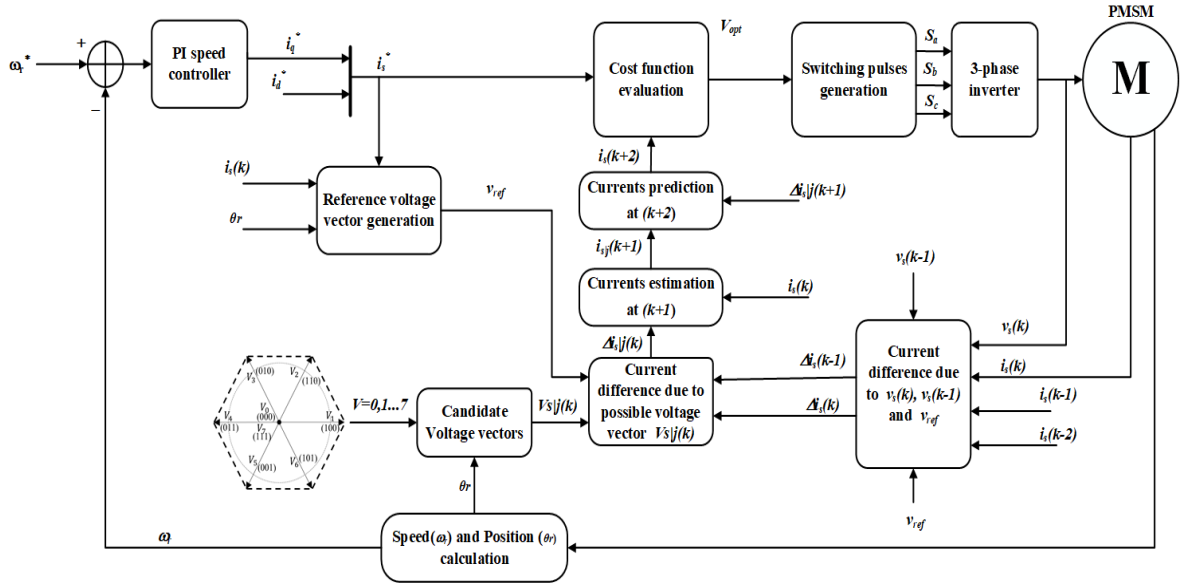


Fig. 5.4. A block diagram of the proposed A-MFPCC for PMSM drives.

A. Adaptive current difference estimation

During each control cycle kT_s, the current is evaluated for eight possible voltage vectors $v_{s_j}, j \in \{0,1,2..7\}$. Each possible voltage vector, when applied, results in different current variations and tracking error variations. Thus, it is essential to investigate the relationship between the applied voltage vector and corresponding current and tracking error variations. From (5.15), the relationship between two applied voltage vectors and the current variation over two consecutive control cycles can be expressed as:

$$\alpha\big(\Delta i_s(k) - \Delta i_s(k-1)\big) = \beta(v_s(k) - v_s(k-1)) \tag{5.29}$$

where $\Delta i_s(k) = i_s(k) - i_s(k-1)$ and $\Delta i_s(k-1) = i_s(k-1) - i_s(k-2)$.

From (5.24), the current difference variation between two successive control cycles is equivalent to the corresponding voltage deviation. To consider the relationship between

tracking error and applied voltage, the current tracking error is first defined as follows:

$$e_{i_s}(k) = i_s^*(k) - i_s(k) \tag{5.30}$$

Based on (5.29), the tracking error variation over two consecutive control cycles can be represented by the difference between a reference voltage vector v_s^* and input voltage vector v_s, as follows:

$$\Delta e_{i_s}(k) - \Delta e_{i_s}(k-1) = \beta\big(v_s^*(k) - v_s(k)\big) \tag{5.31}$$

where $\Delta e_{i_s}(k) = e_{i_s}(k) - e_{i_s}(k-1)$, $\Delta e_{i_s}(k-1) = e_{i_s}(k-1) - e_{i_s}(k-2)$, and v_s^* is the reference voltage vector.

Initially, the reference vector, v_s^*, can be obtained using the deadbeat solution, where the current $i_s(k+2)$ is considered the reference current $i_s^*(k)$ to achieve fast tracking at the start of k+2. By solving $i_s(k+2) = i_s^*(k)$, the reference voltage vector at k can be obtained as:

$$v_s^*(k) = \frac{1}{\beta}\Big(i_s^*(k) - i_s(k) - \alpha\big(i_s(k) - i_s(k-1)\big)\Big) + v_s(k-1) \tag{5.32}$$

However, in this research, v_s^* is determined based on the current reference vector $\overline{i_s^*}$ position and the current error e_{i_s}. A detailed explanation of the generation of the reference vector v_s^* is presented in the following section.

From (5.29) and (5.31), the relationship between tracking error and current difference variations is expressed as follows:

$$\Big(\Delta e_{i_s}(k) - \Delta e_{i_s}(k-1)\Big) - \alpha\big(\Delta i_s(k) - \Delta i_s(k-1)\big)$$
$$= \beta\big((v_s^*(k) - v_s(k)) - (v_s(k) - v_s(k-1))\big) \tag{5.33}$$

At steady state, the current reference, i_s^*, is constant between two successive control cycles, and thus:

$$\left(\Delta e_{i_s}(k) - \Delta e_{i_s}(k-1)\right) - \left(\Delta i_s(k) - \Delta i_s(k-1)\right) = 2\left(\Delta i_s(k) - \Delta i_s(k-1)\right) \quad (5.34)$$

$$\alpha\left(\Delta i_s(k) - \Delta i_s(k-1)\right) = \frac{\beta}{2}\left((v_s^*(k) - v_s(k)) - (v_s(k) - v_s(k-1))\right) \quad (5.35)$$

During each control cycle kT_s, the voltage vector $v_s(k)$ can be one of the possible vectors, $v_s = v_{s_j}, j \in [0,1,2 \ldots .7]$. Considering the eight possible vectors, (5.35) is updated as follows:

$$\alpha\left(\Delta i_s(k)_j - \Delta i_s(k-1)\right) = \frac{\beta}{2}\left(\left(v_s^*(k) - v_s(k)_j\right) - \left(v_s(k)_j - v_s(k-1)\right)\right) \quad (5.36)$$

Hence, by using the current variation and voltage deviation over the past two successive control cycles, the current difference due to each possible voltage vector, $\Delta i_s(k)_j$, can be estimated by combining (5.35) and (5.36) as follows:

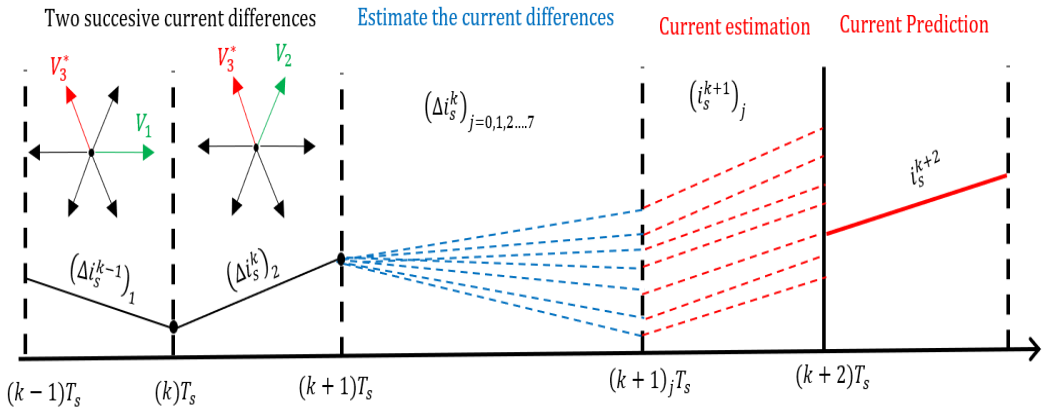
$$\frac{\Delta i_s(k)_j - \Delta i_s(k-1)}{\Delta i_s(k) - \Delta i_s(k-1)} = \frac{\left(v_s^*(k) - v_s(k)_j\right) - \left(v_s(k)_j - v_s(k-1)\right)}{(v_s^*(k) - v_s(k)) - (v_s(k) - v_s(k-1))} \quad (5.37)$$

$$\Delta i_s(k)_j = \Delta i_s(k-1) + \frac{\left(\left(v_s^*(k) - v_s(k)_j\right) - \left(v_s(k)_j - v_s(k-1)\right)\right)(\Delta i_s(k) - \Delta i_s(k-1))}{(v_s^*(k) - v_s(k)) - (v_s(k) - v_s(k-1))} \quad (5.38)$$

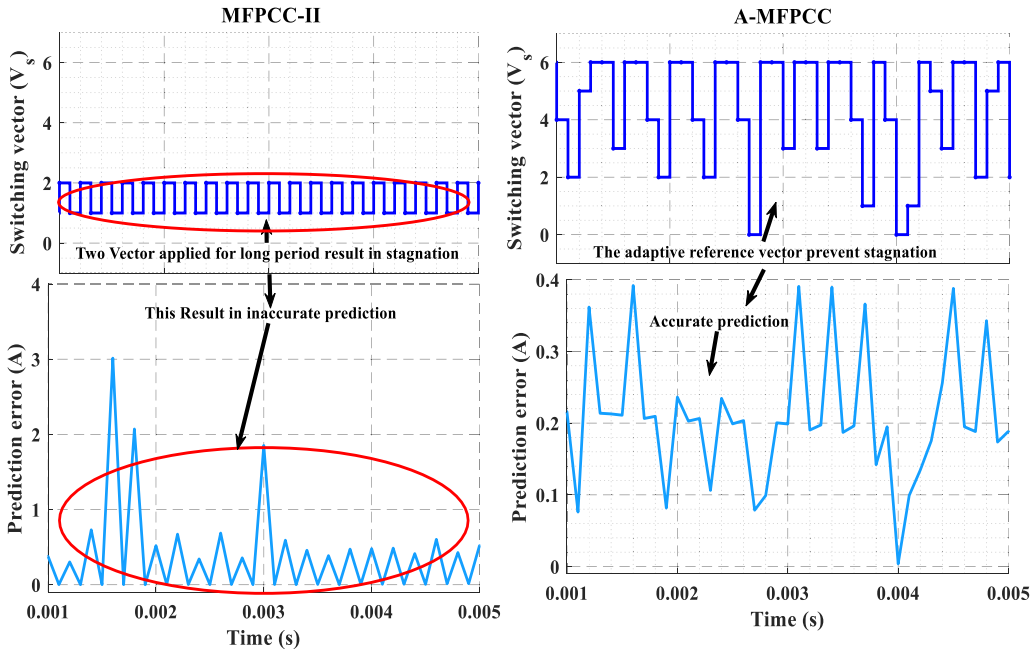
Based on (5.38), the parameters α and β are eliminated, and the current at time steps (k+1) and *(k+2)* can be predicted as follows:

$$i_s(k+1) = i_s(k) + \Delta i_s(k)_j \quad (5.39)$$

$$i_s(k+2) = i_s(k+1) + \Delta i_s(k+1)_j \quad (5.40)$$



(a)



(b)

Fig. 5.5. The proposed A-MFPCC, (a) Current prediction and current differences estimation, and (b) stagnation elimination compared to MFPCC-II.

Thus, the current prediction is accomplished in a model-free approach without using the machine model, parameter, or any estimation method. In addition, the prediction process account for the tracking error during each sampling instant kT_s, by adaptively improving the prediction accuracy based on a reference voltage vector v_s^* . Unlike MFPCC-II, the proposed

A-MFPCC does not require two successive voltage vectors to be different, which avoids applying only two vectors for an extended period. The adaptive reference vector constantly forces updating the applied voltage to minimize tracking error and prevent stagnation.

Fig. 5.5 shows the current difference estimation, prediction, and stagnation elimination of the proposed A-MFPCC. The reference voltage vector improves the selection of an optimum switching vector, reducing the tracking error. In addition, it eliminates stagnation by constantly updating the current differences, improving the prediction accuracy.

B. Reference Vector Generation

The reference voltage vector v_s^* is determined based on the reference current vector position. The reference current vector, $\overline{i_s^*}$, is expressed as:

$$\overline{i_s^*} = i_\alpha^* + j i_\beta^* = I^* \angle \theta_{sec}^* \tag{5.41}$$

where I^* is the reference current magnitude, θ_{sec}^* the reference angle that defines which sector the reference current vector $\overline{i_s^*}$ is in.

The reference current trajectory in the space phasor is shown in Fig. 5.6. Based on the obtained θ_{sec}^*, the sector of the reference current vector $\overline{i_s^*}$ can be determined. To force the measured currents i_s to follow the reference current i_s^*, a reference voltage vector v_j^*, $(j = 0,1,2, \dots 7)$ can be obtained based on the current tracking deviation $(\Delta i_s^* = i_s^* - i_s)$. The current deviation, Δi_s^*, direction and magnitude are used to select an appropriate vector (v_s^*) that can minimize the tracking error.

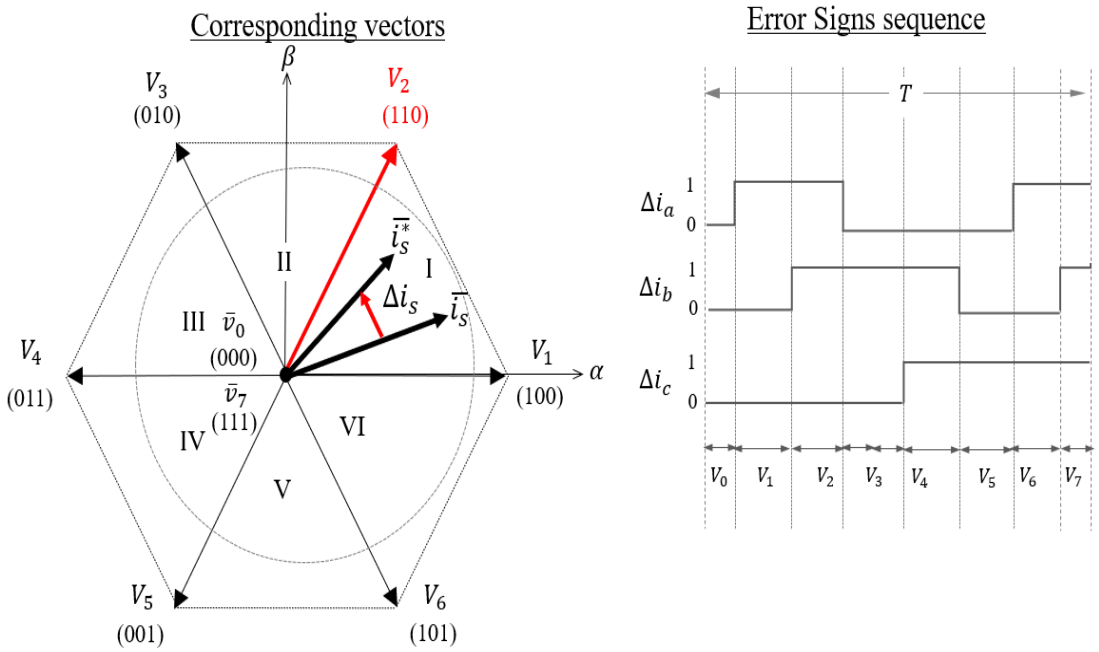


Fig. 5.6. Reference current trajectory and current error sign sequence.

Considering the difference between the three-phase currents, and reference currents ($i^*_{abc} - i_{abc}$) and obtained sector from the position of the reference current vector, the reference vector v^*_s can be determined as depicted in Fig. 5.6. Based on the error sign, the state is determined as:

$$sign(\Delta i^*_s) = \begin{cases} 1, & \Delta i^*_s > 0 \\ 0, otherwise \end{cases} \tag{5.42}$$

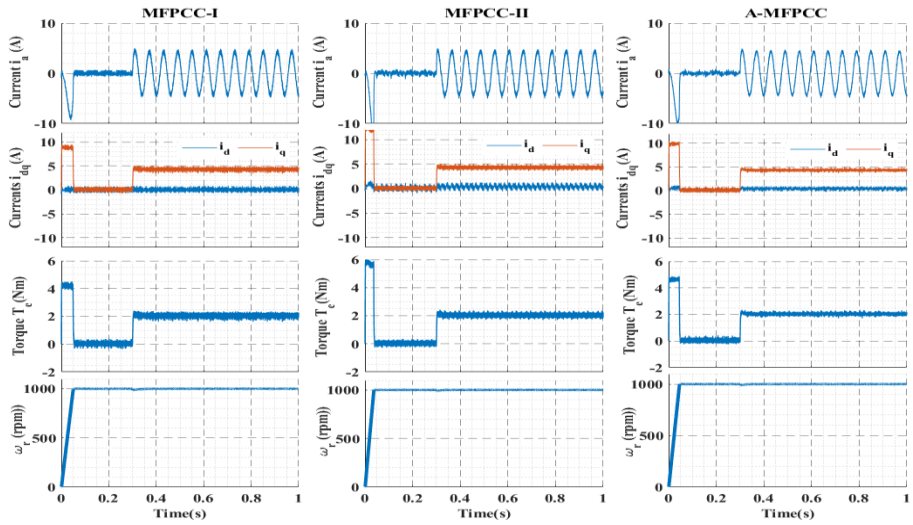
where $\Delta i^*_s = i^*_{abc} - i_{abc}$.

5.3 Numerical Simulation

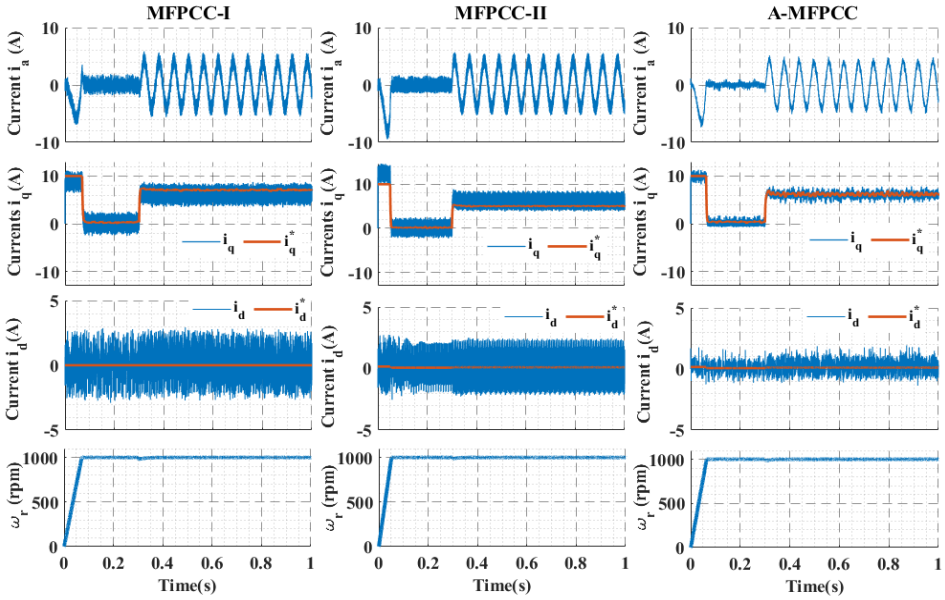
Numerical simulations using Matlab/Simulink are performed to demonstrate the effectiveness of the proposed A-MFPCC for PMSM drive at different operating conditions. Two MFPCC schemes are considered for comparison, namely, MFPCC-I and MFPCC-II, discussed in Sections 5.2.1 and 5.2.2. MFPCC-I uses the ultra-local model with two unknowns estimated by using RLS. MFPCC-II considers the current differences due to two successive voltage vectors to estimate the differences for the remaining vectors. The PMSM drive system is designed with a proportional-integral (PI) controller for the outer speed control loop and MFPCC-I, MFPCC-II, or the proposed A-MFPCC for the inner current control loop. The same drive parameters listed in Table 3.1 are used for evaluating the three controllers.

To assess the effectiveness of MFPCC-I, MFPCC-II, and the proposed A-MFPCC under various operating conditions, simulations are conducted at three different speeds (1000, 600, and 200 rpm) and with the rated load torque (2 Nm). In addition, the controllers are evaluated against parameter variations by testing them with both nominal and mismatched machine parameters, expressly set as: $0.8R_s, 0.3L_d, 0.3L_q,$ and $0.7\psi_{PM}$. The simulation results capture the controllers' transient and steady-state responses from 0 rpm up to the desired speed, with a load torque of 2 Nm, applied at 0.3s. At the rated speed (1000 rpm), the performance of MFPCC-I, MFPCC-II, and the proposed A-MFPCC with both nominal and mismatched parameters are presented in Fig. 5.7. Similarly, the performance comparison at 600 and 200 rpm with nominal and mismatched parameters are presented in Figs 5.8 and 5.9, respectively.

Each of Figs 5.7 and 5.8 comprises three parallel columns illustrating the responses of MFPCC-I, MFPCC-II, and the proposed A-MFPCC, respectively. The curves in each column, from top to bottom, correspond to the phase a stator current, i_a, dq-axis currents, estimated torque, and motor speed, respectively.

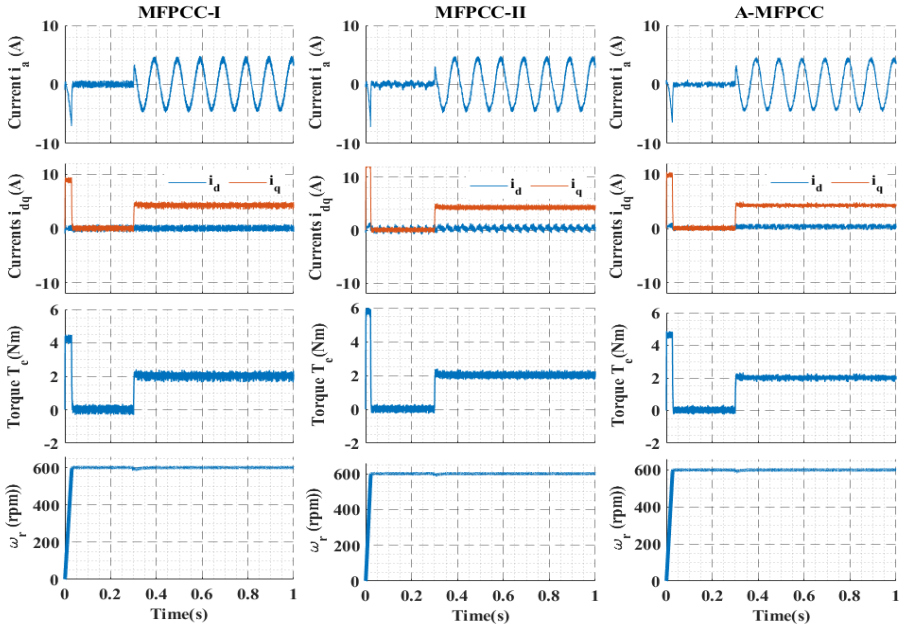


(a)

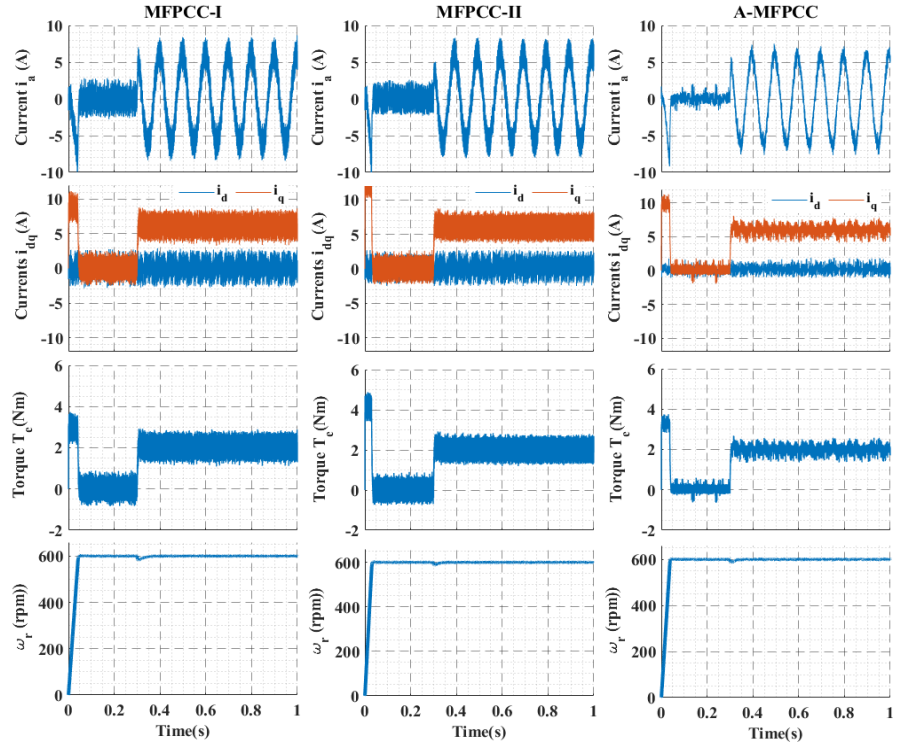


(b)

Fig. 5.7. Performance of MFPCC-I, MFPCC-II, and the proposed A-MFPCC at rated speed (1000 rpm) and torque (2 Nm) with (a) nominal and (b) mismatched parameters.

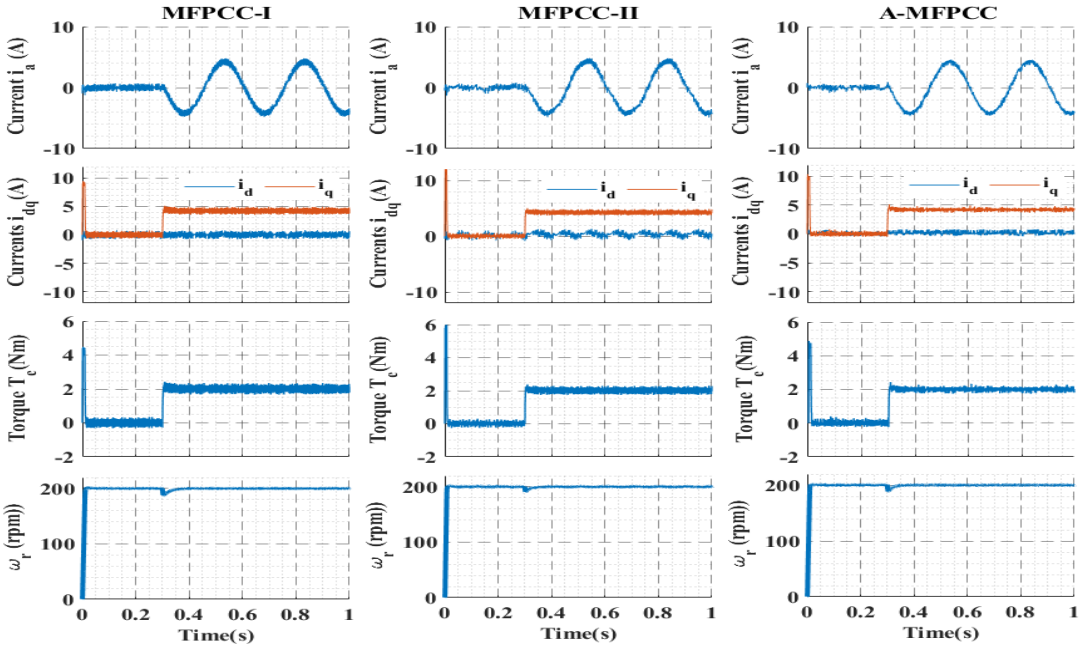


(a)

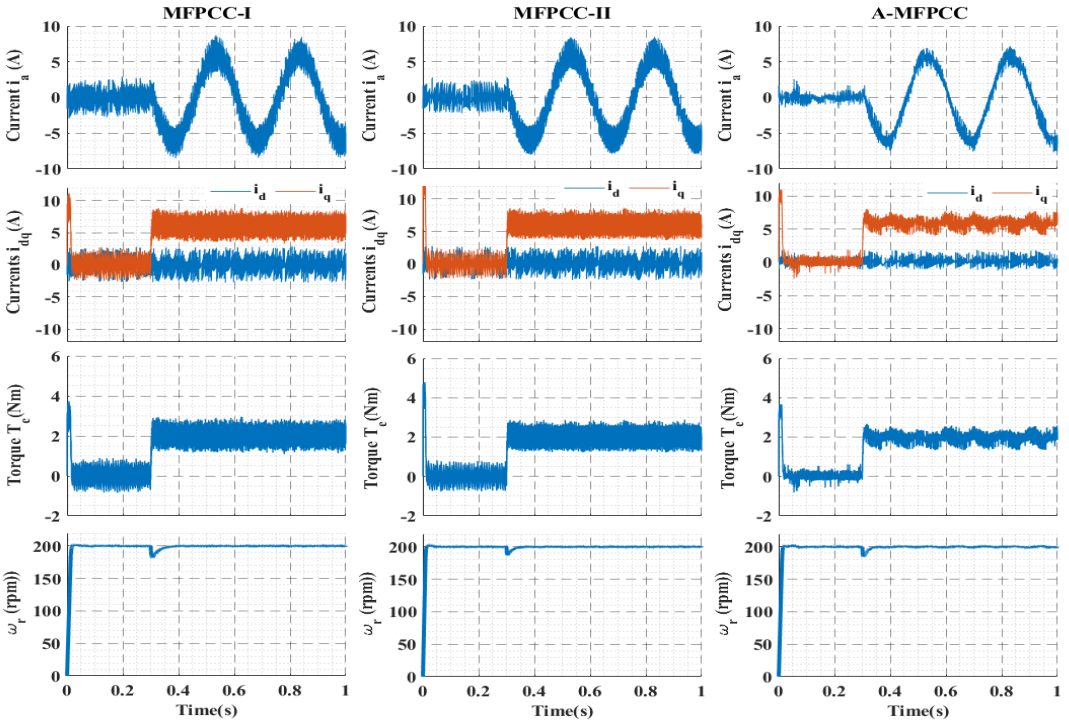


(b)

Fig. 5.8. Performance of MFPCC-I, MFPCC-II, and the proposed A-MFPCC at rated speed (600 rpm) and torque (2 Nm) with (a) nominal and (b) mismatched parameters.



(a)

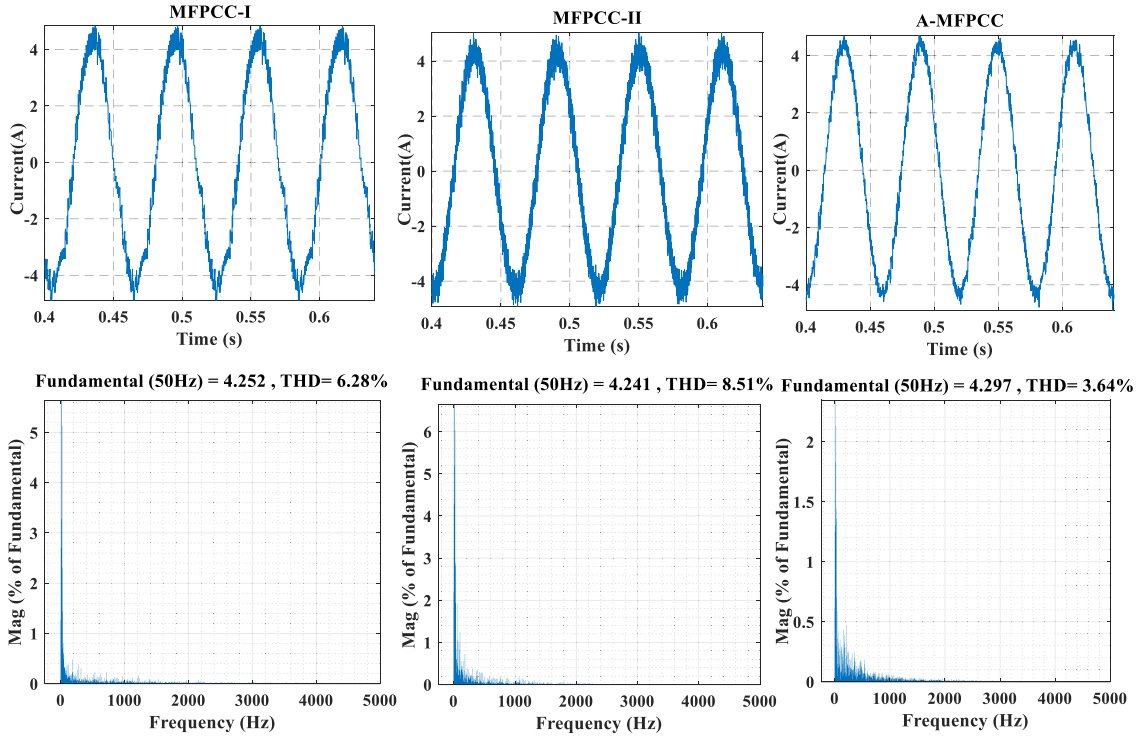


(b)

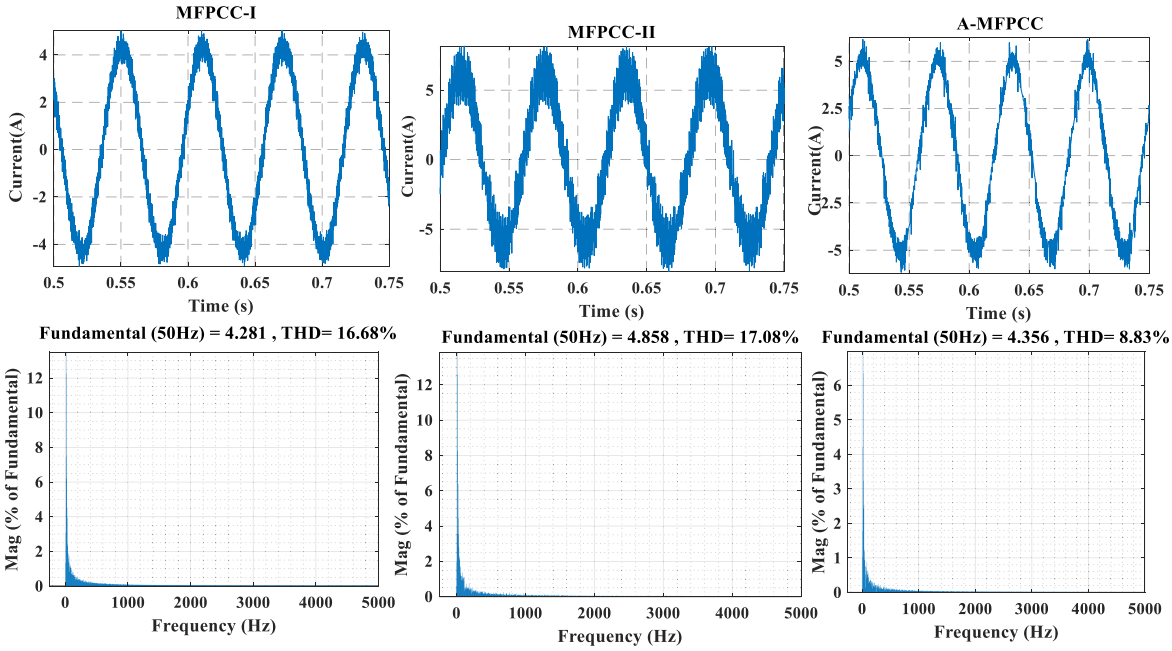
Fig. 5.9. Performance of MFPCC-I, MFPCC-II, and the proposed A-MFPCC at rated speed (200 rpm) and torque (2 Nm) with (a) nominal and (b) mismatched parameters.

With nominal machine parameters, all controllers exhibit satisfactory performance with a fast dynamic response and good steady-state tracking capability. Notably, the proposed A-MFPCC demonstrates better current performance than the other controllers, producing smoother and less ripple current signals. With mismatched machine parameters, MFPCC-I and MFPCC-II show significant performance deterioration. Conversely, the proposed A-MFPCC, despite being affected by parameter mismatching, maintained an excellent current performance with low distortion and fluctuations compared to the other controllers. This can be attributed to the adaptive reference voltage vector employed by A-MFPCC, which continuously updates the current difference based on the tracking error, thereby maintaining good tracking performance even in the presence of parametric uncertainties.

To gain a deeper insight into the impact of parametric uncertainties on current performance, the total harmonic distortion (THD) of stator current i_a is calculated up to 5000 Hz. Fig. 5.10 shows the harmonic spectra of the steady state stator currents i_a at 1000 rpm with a 2 Nm load for MFPCC-I, MFPCC-II, and the proposed A-MFPCC with the nominal and mismatched parameters. The results indicate that the proposed A-MFPCC achieves the lowest stator current THDs of 3.64% and 8.83% for nominal and mismatched machine parameters, respectively. In contrast, MFPCC-I and MFPCC-II record higher stator current THDs of 6.28% and 8.51% for nominal parameters and 16.68% and 17.08% for mismatched parameters, respectively. These findings demonstrate the superior performance of the proposed A-MFPCC in maintaining low THD levels even in the presence of parametric uncertainties.



(a)



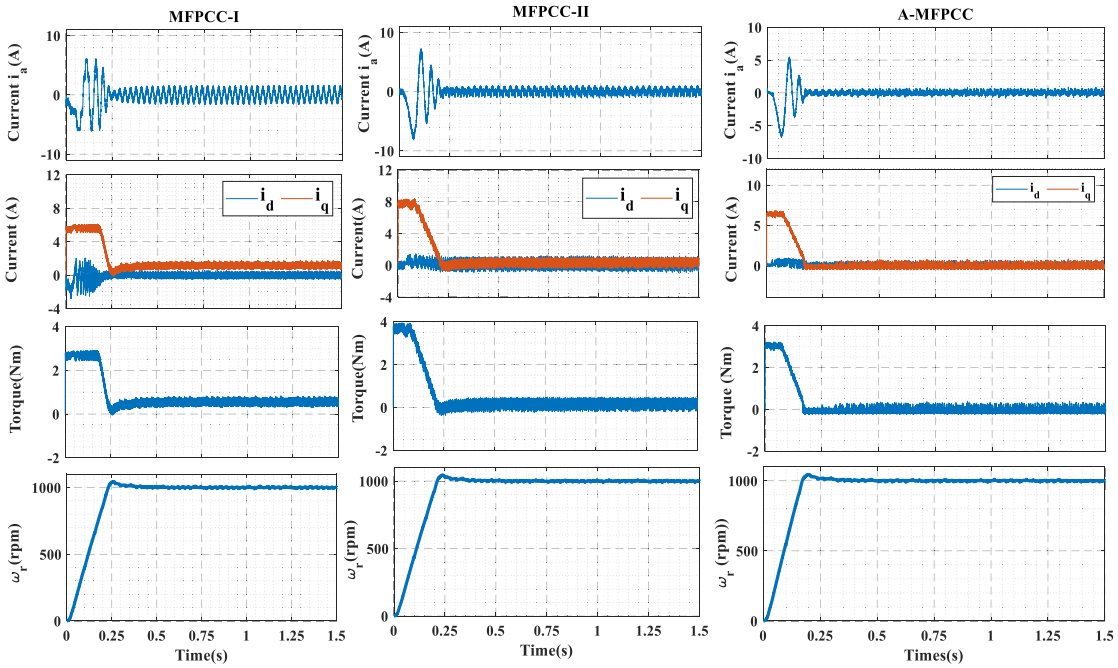
(b)

Fig. 5.10. Harmonic spectra of stator currents i_a of MFPCC-I, MFPCC-II and the proposed MF-APCC at (a) nominal and (b)mismatched parameters.

5.4 Experimental Validation

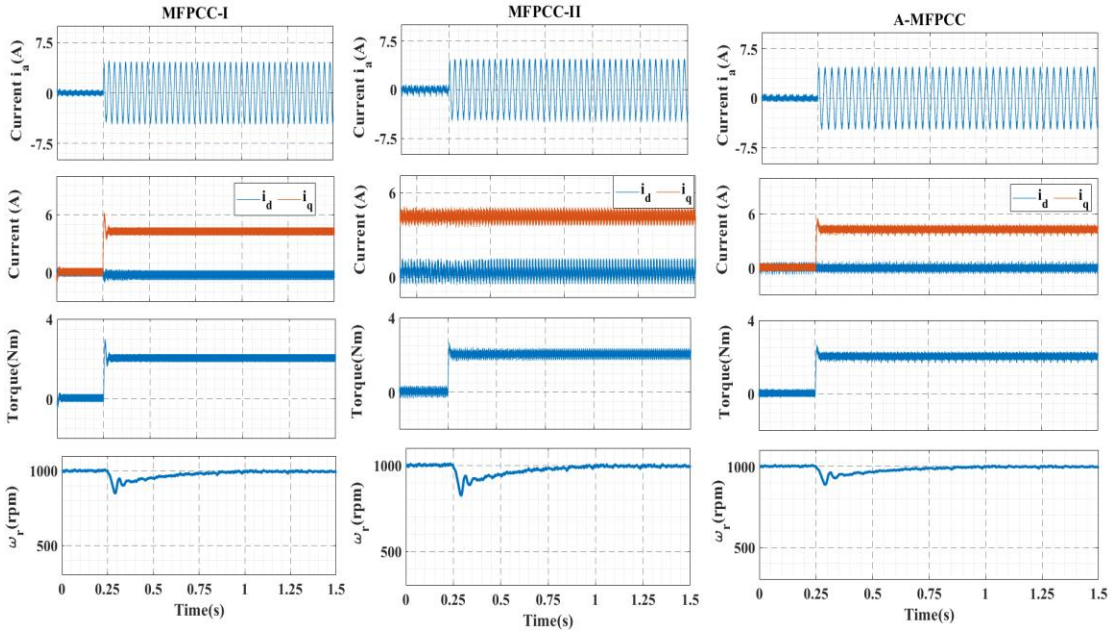
The proposed A-MFPCC is experimentally validated based on a PMSM drive system with a two-level inverter, as shown in Fig. 3.10. The control and motor parameters used in the experiment are the same as those presented in Table 3.1. Matlab/Simulink software interfaced with a dSPACE DS1104 PPC/DSP control board is employed for the real-time implementation of the control algorithm. This board serves as the platform for executing the algorithm. As for the inverter, an Insulated-Gate Bipolar Transistor (IGBT) integrated with a three-phase intelligent power module is utilized. The gating pulses required by the inverter are generated by the algorithm and subsequently transmitted through DS1104. To apply the load during the experiment, a programmable dynamometer controller DSP6000 is employed. Furthermore, the motor speed is measured using an interior 2500-pulse incremental encoder. To facilitate the monitoring, real-time control, and recording of all experimental results, ControlDesk is utilized.

In line with the simulation analysis, the performance of the A-MFPCC is evaluated experimentally and compared against MFPCC-I and MFPCC-II. The experimental results encompass a comparative analysis of the three controllers under various operating conditions, including start-up, load disturbance, and steady-state operations. Specifically, during the start-up tests, the controller performances are assessed as the motor speed progressively accelerated from a standstill to the rated speed of 1000 rpm with a slight overshoot. The corresponding responses of the controllers to the load disturbance are obtained by subjecting the motor, operating at a steady state of 1000 rpm, to a load torque of 2 Nm.



(a) (b) (c)

Fig. 5.11. Experimental performance comparison during start-up from standstill to rated speed (1000 rpm) for (a) MFPCC-I, (b) MFPCC-II, and (c) the proposed A-MFPCC.

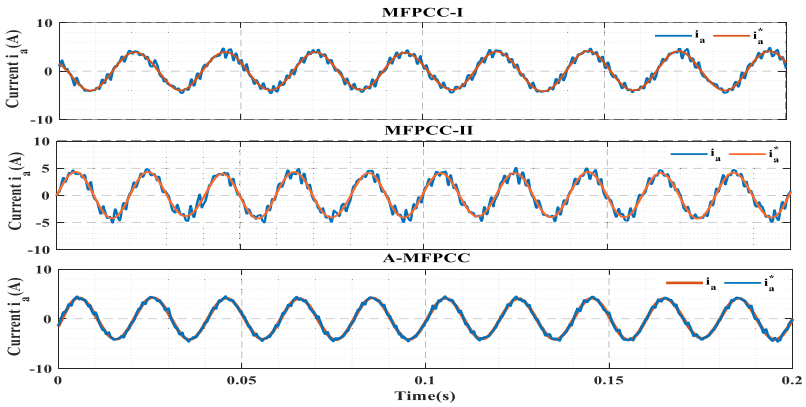


(a) (b) (c)

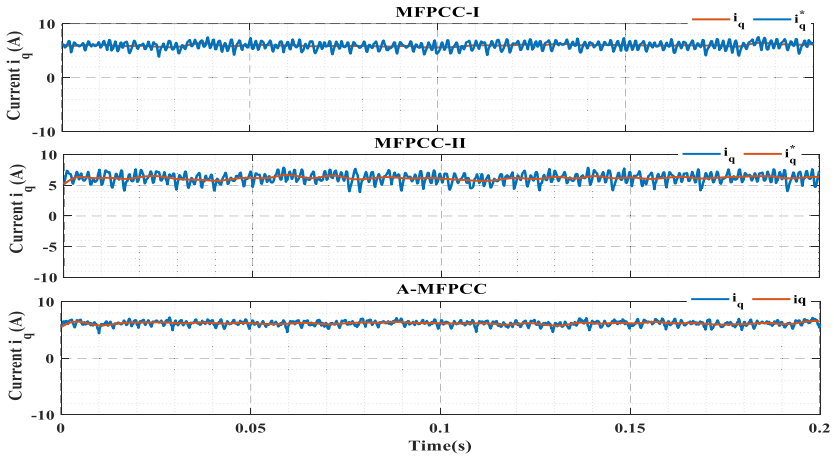
Fig. 5.12. Experimental performance comparison to 2 Nm load disturbance at a steady-state of 1000 rpm for (a) MFPCC-I, (b) MFPCC-II, and (c) the proposed A-MFPCC.

The experimental results, presented in Figs 5.11 and 5.12, correspond to the stator current i_a, q-axis current, d-axis current, estimated torque, and motor speed. All controllers exhibited exceptional dynamic performance during the start-up tests, with minimal overshot and rapid reference tracking. Additionally, the controllers displayed excellent disturbance rejection capabilities during the load disturbance tests. All controllers recovered from the load disturbance and resumed their steady-state conditions. However, the proposed A-MFPCC exhibits superior performance under certain circumstances. For instance, the current fluctuations observed during start-up and load disturbance tests are significantly lower than MFPCC-I and MFPCC-II. Furthermore, A-MFPCC exhibits faster recovery, within 0.3125s, from the load disturbance with a marginal speed drop of 113 rpm compared to MFPCC-I, which takes 0.35s with a speed drop of 150 rpm, and MFPCC-II, which takes 0.375s with a speed drop of 175 rpm.

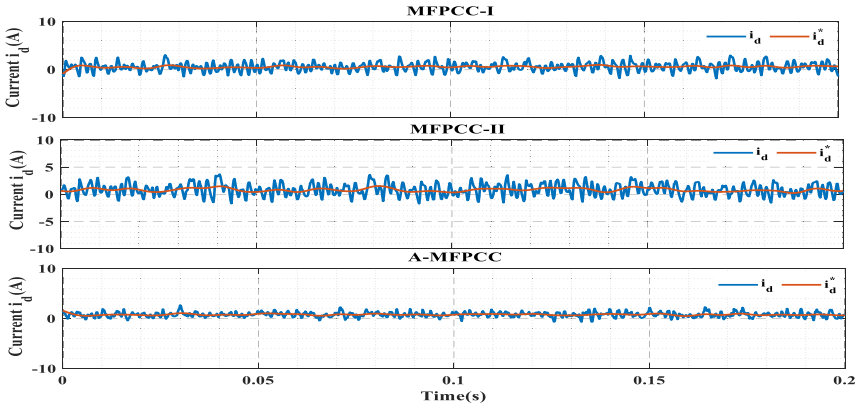
In this research, the controllers' steady state current tracking responses are assessed at 1000, 600, and 200 rpm with a 2 Nm load applied to the motor shaft. The responses of measured stator current i_a, q-axis current i_q, the d-axis current i_d, and their respective reference signals are captured for MPCC-I, MPCC-II, and A-MFPCC. Fig. 5.13 shows the steady state stator current i_a, q-axis current i_q, d-axis current i_d, and their corresponding reference signals at the rated speed (1000 rpm), while the responses at 600 rpm and 200 rpm are presented in Figs 5.14 and 5.15, respectively. The experimental results demonstrate the superiority of the proposed A-MFPCC compared to MPCC-I and MPCC-II regarding the steady state current tracking performance. This is attributed to the adaptive reference voltage vector employed by A-MFPCC. The current difference is updated based on the tracking error between measured and reference currents.



(a)

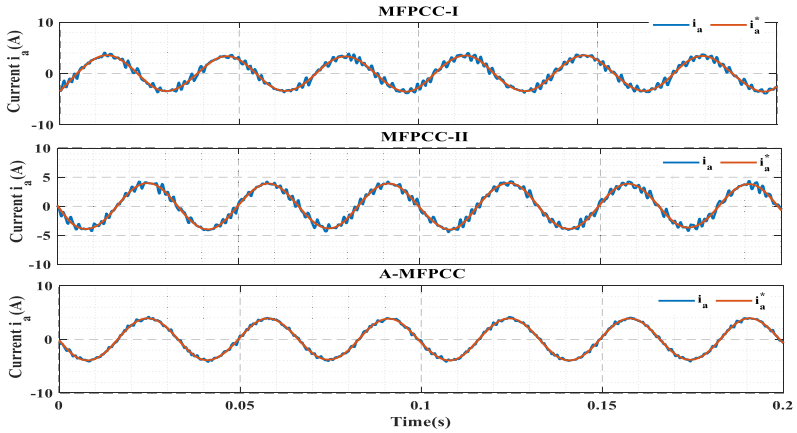


(b)

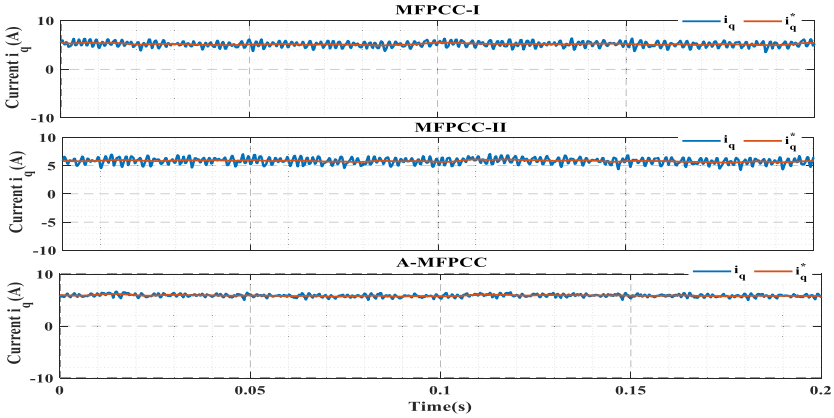


(c)

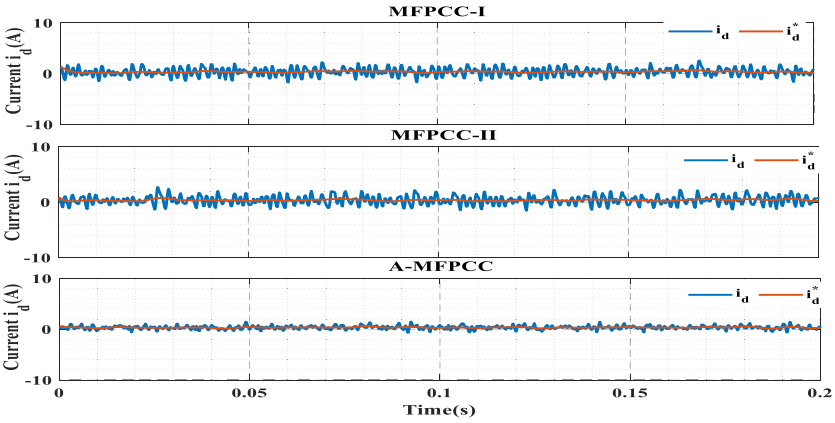
Fig. 5.13. Steady-state current responses of MFPCC-I, MFPCC-II, and A-MFPCC at 1000 rpm, (a) stator current i_a, (b) q-axis current i_q, and (c) d-axis current i_d .



(a)

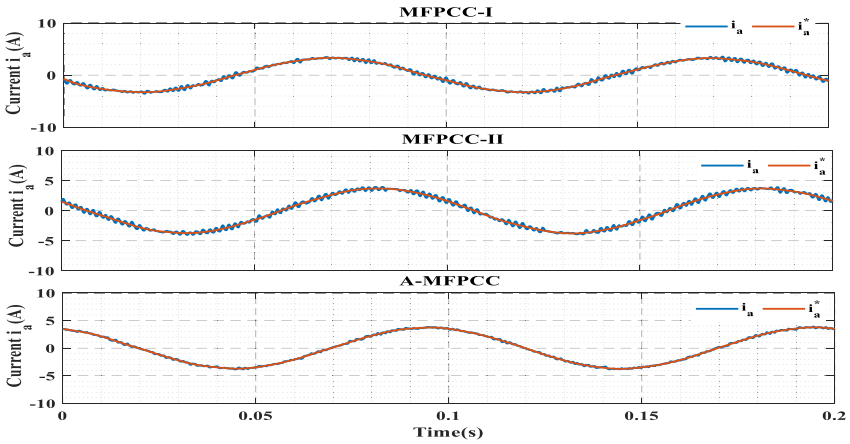


(b)

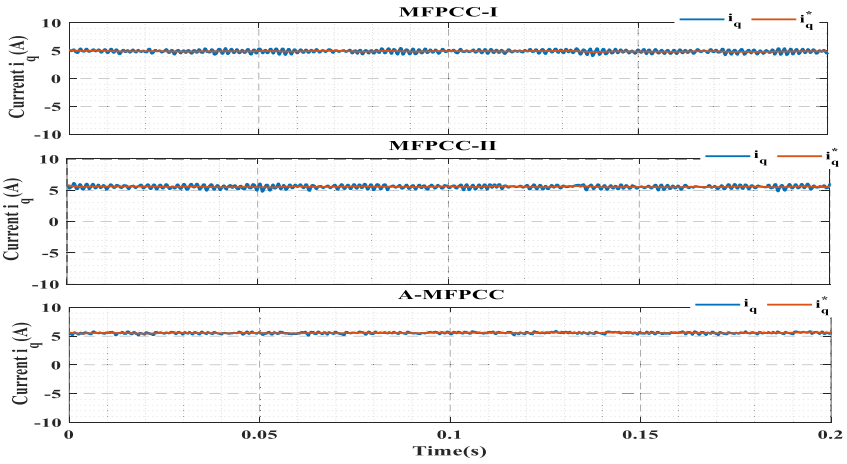


(c)

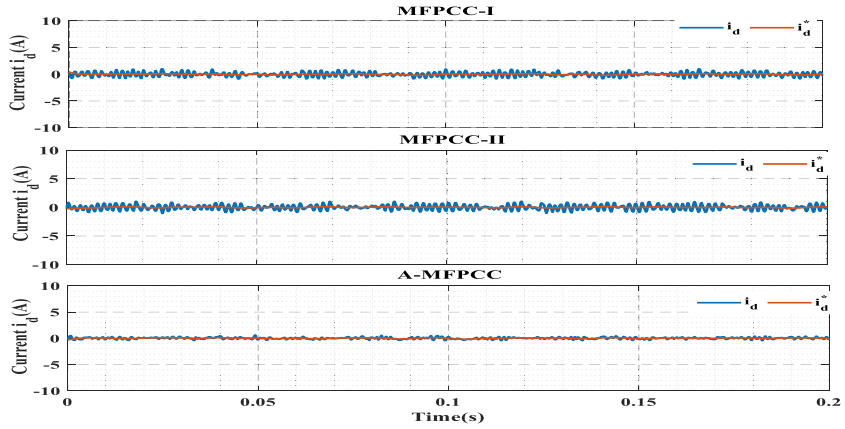
Fig. 5.14. Steady-state current responses of MFPCC-I, MFPCC-II, and A-MFPCC at 600 rpm, (a) stator current i_a, (b) q-axis current i_q, and (c) d-axis current i_d .



(a)



(b)



(c)

Fig. 5.15. Steady-state current responses of MFPCC-I, MFPCC-II, and A-MFPCC at 200 rpm, (a) stator current i_a, (b) q-axis current i_q, and (c) d-axis current i_d .

5.5 Quantitative Analysis and Comparison

This section presents a quantitative analysis of MFPCC-I, MFPCC-II, and proposed A-MFPCC at different operating conditions and against parameter mismatching. The three controllers are parameter-free control, which makes it challenging to evaluate their performance against parametric uncertainties in real-time experiments where actual machine parameters are unavailable. Nonetheless, the controllers' performances are evaluated under various operating conditions. Therefore, the controllers are quantitatively evaluated against parameter variations based on numerical simulations and against speed and load torque variation based on experimental tests.

5.5.1 Variation of Machine Parameters

Parameter variations are the most severe uncertainties for PMSM drive, significantly degrading the drive system performance. The ability of a control to maintain a good drive performance in the presence of parameter uncertainties is a critical indicator of its robustness. In this section, the motor parameters (R_s, L_d, L_q, ψ_{PM}) are varied to their permissible range ($\pm\Delta$) of the nominal values based on Table 4.1, then the proposed A-MFPCC, MFPCC-I, and MFPCC-II are tested with these variations accordingly. For instance, the value of stator resistance R_s varied between -20% and 80% of the nominal R_s and other motor parameters are kept as nominal values. The same process is repeated with the variations of other parameters. For each parameter variation, the three control methods are tested, and the numerical values of the settling time T_s, overshoot OS, torque ripples T_{rip}, and current ripples $i_{a_{rip}}$ are recorded.

The performance of MFPCC-I, MFPCC-II, and A-MFPCC in terms of settling time, overshoot, torque, and current ripples with the variations of stator resistance R_s are shown in

Fig. 5.16. The performances are slightly affected by the variations of R_s. A-MFPCC recorded the best torque and current ripple robustness, and MFPCC-I recorded the best settling time robustness. For the overshoot, both controllers recorded inconsistent performance.

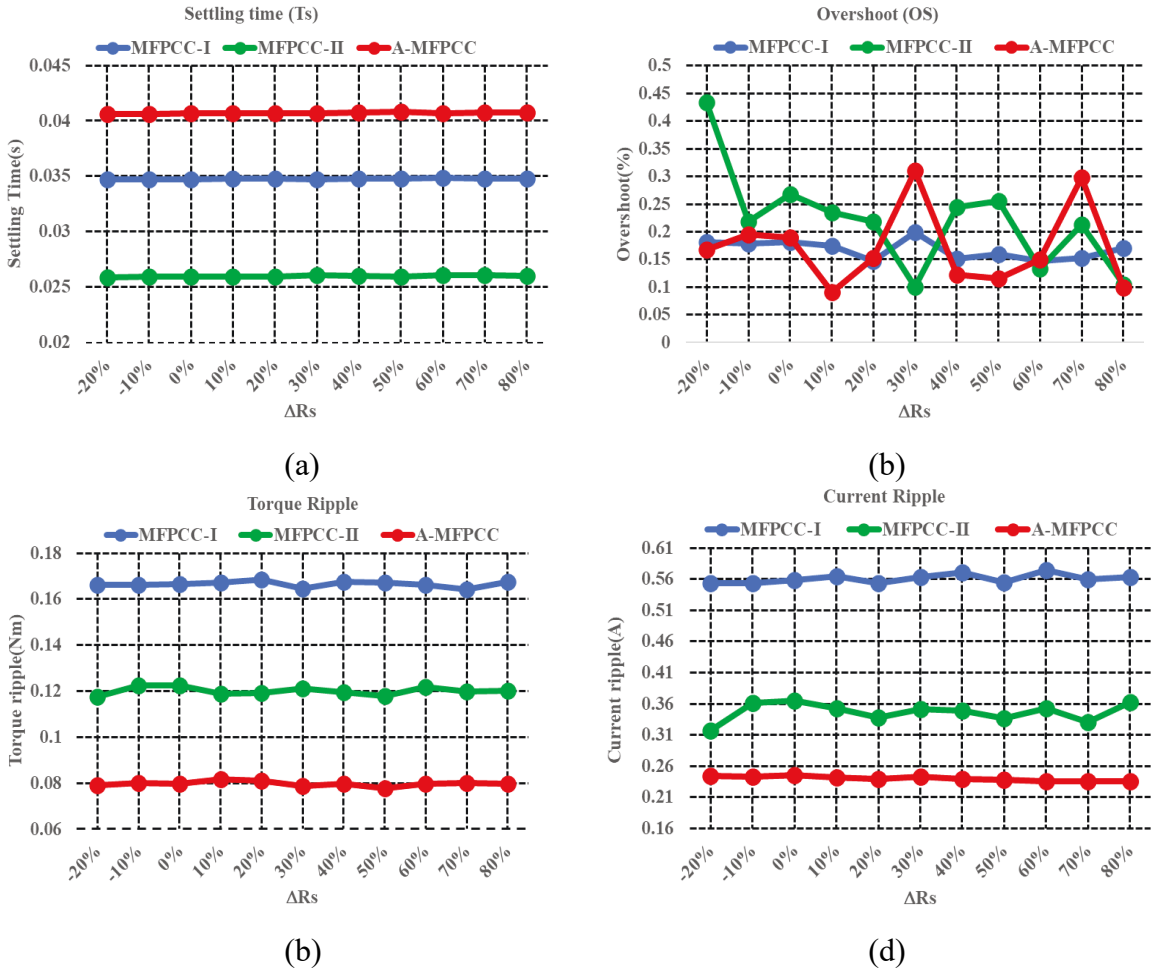


Fig. 5.16. Performance comparisons of three controllers with the variation of (R_s), (a) settling time (Ts), (b) Overshoot (OS), (c) Torque ripple, and (d) current ripple.

The performances of MFPCC-I, MFPCC-II, and A-MFPCC in terms of settling time, overshoot, torque, and current ripples with the variations of d-axis machine inductance L_d are shown in Fig. 5.17. The variations of L_d significantly influence the drive performance, especially at low values ($-\%\Delta$) below the nominal value, while at high values ($+\%\Delta$) above the nominal value, the performance is almost the same as the nominal value. MFPCC-II

records the best settling time robustness, while A-MFPCC records the best torque and current ripple robustness. It is worth noting that A-MFPCC showed better robustness only in terms of torque and current ripples. However, its robustness to other indicators is comparable to the other controllers.

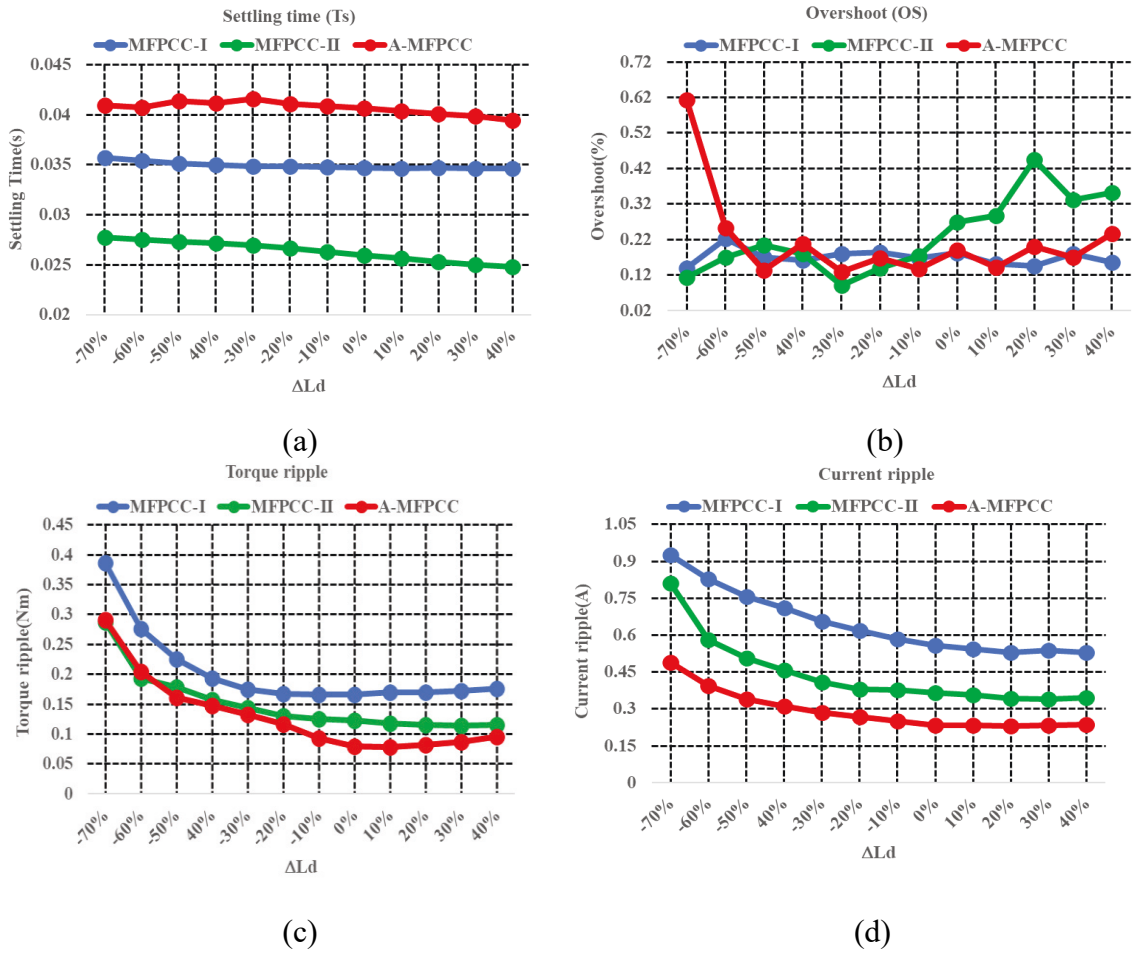


Fig. 5.17. Performance comparisons of the three controllers with the variation of inductance (L_d), (a) settling time (Ts), (b) Overshoot (OS), (c) Torque ripple, and (d) current ripple.

The performances of MFPCC-I, MFPCC-II, and A-MFPCC in terms of settling time, overshoot, torque, and current ripples with the variations of q-axis machine inductance L_q are shown in Fig. 5.18. Similar to L_d, the variations of L_q significantly influence the drive

performance, especially at low values. The proposed A-MFPCC has shown the best robustness regarding current ripples and comparable and almost similar robustness with MFPCC-II regarding torque ripples. MFPCC-I has shown the best robustness in terms of settling time and overshoot. Overall, the proposed A-MFPCC showed good robustness with the variation of L_q, and maintained a trade-off between a good transient response (settling time, overshoot) and a steady state response (torque and current ripples), leading to good overall system robustness.

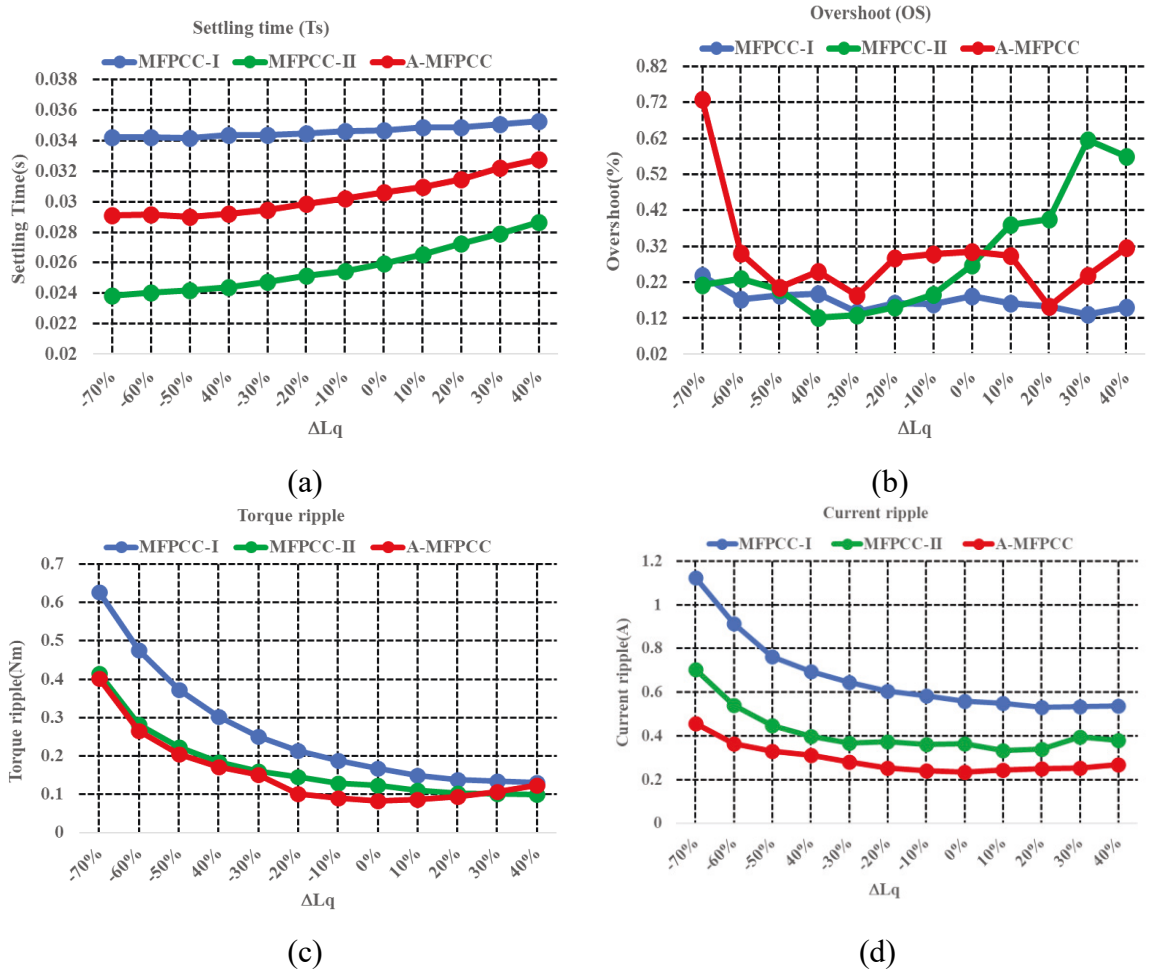


Fig. 5.18. Performance comparisons of the three controllers with the variation of inductance (L_q), (a) settling time (Ts), (b) Overshoot (OS), (c) Torque ripple, and (d) current ripple.

The performances of MFPCC-I, MFPCC-II, and A-MFPCC in terms of settling time, overshoot, torque, and current ripples with the variations of permanent magnet flux Ψ_{PM} are shown in Fig. 5.19. Unlike L_d and L_q, Ψ_{PM} affects the drive performance with high values (+%Δ) above the nominal value, particularly for torque ripples. The proposed A-MFPCC has shown the best robustness with the variation of Ψ_{PM} in terms of overshoot, torque ripples, and current ripples. In contrast, the best settling time robustness is achieved by MFPCC-I but at the cost of the lowest torque and current ripple robustness.

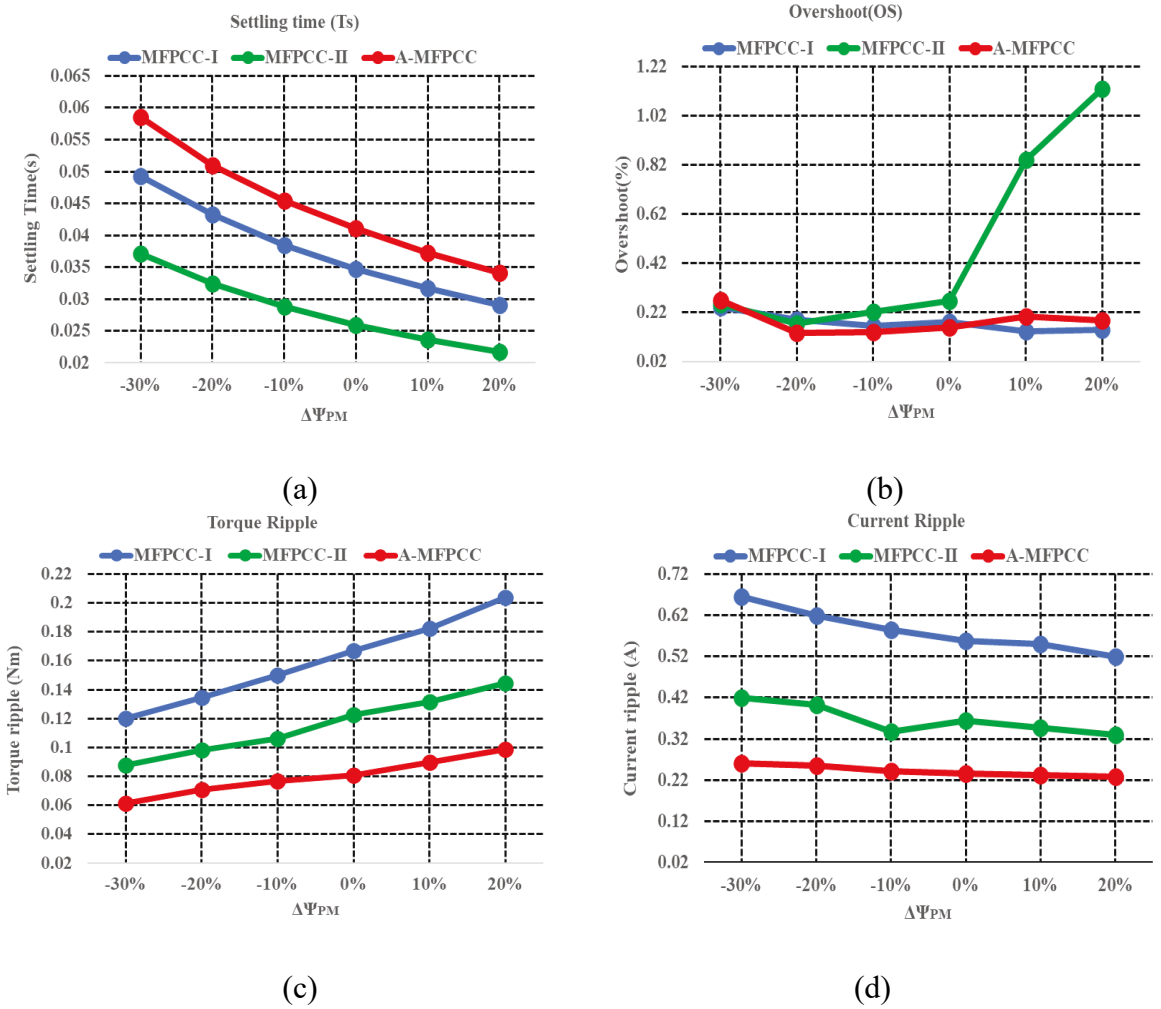


Fig. 5.19. Performance comparisons of the three controllers with the variation of permanent magnet flux (Ψ_{PM}), (a) settling time (Ts), (b) Overshoot (OS), (c) Torque ripple, and (d) current ripple.

The variations of machine parameters significantly impact the drive performance reflected by high torque and current ripples, high overshot, and longer settling times. The variations of stator resistance R_s and permanent magnet flux Ψ_{PM} have slight impacts on the performance compared to the variations of machine inductances L_d and L_q. For instance, the worst current ripples with the variation of R_s are 0.5%, 1.6%, and 1.2% higher than the values recorded at nominal R_s for A-MFPCC, MFPCC-I, and MFPCC-II, respectively. Similarly, the worst current ripples with the variation of Ψ_{PM} are 10.4%,19%, and 15% higher than the values recorded at nominal Ψ_{PM} for A-MFPCC, MFPCC-I, and MFPCC-II, respectively. On the other hand, variations of L_d and L_q, the worst current ripple values are higher than the values at nominal L_d and L_q by 108% and 94.2% for A-MFPCC, 122.4% and 93% for MFPCC-II, and 126.7% and 101.4% for MFPCC-I.

In summary, the variations of machine inductances of L_d and L_q have the highest impact on the drive performance with all three controllers. The proposed A-MFPCC has shown good robustness to parameter uncertainties, particularly in terms of torque and current ripples. The proposed A-MFPCC may have sacrificed some of its robustness in the transient state (settling time and overshoot) but showed strong robustness in the steady state (torque and current ripples), thus, maintaining good overall robustness of the drive system. The excellent tracking performance of A-MFPCC is attributed to the adaptive reference vector generated based on the position reference current and tracking error. This vector is applied each control cycle in the prediction stage; thus, the switching vector that minimizes the current tracking error is selected. Moreover, the reference vector constantly updates the current difference, preventing switching stagnation as in MFPCC-II.

5.5.2 Variations of Operating Conditions (Speed and Load Torque)

MFPCC-I, MFPCC-II, and A-MFPCC are optimized for optimal performance. Therefore, it is difficult to differentiate their performance solely based on graphical results. Hence, a quantitative comparison is conducted between them based on torque and stator current i_a ripples at different speed operations and load conditions. The torque and stator current i_a ripples for the three controllers are computed at a steady state for 0.5s. The motor operates at a desired speed, and a desired load torque is applied, after which the ripples are calculated for 5000 sampling instants (0.5s). For instance, the motor operates from a standstill to the desired speed (1000 rpm), and when it reaches the desired speed, a desired load torque (2 Nm) is applied, and the ripples are computed for a duration of 0.5s. This test is repeated with different load torque values ranging from 0 to 2 Nm and for various speed operations (200, 400, 600, and 800 rpm). The torque ripples of MFPCC-I, MFPCC-II, and A-MFPCC at different speed operations and load conditions are presented in Tables 5.1-5.3. Similarly, the stator current i_a ripples are presented in Tables 5.4-5.6. Torque and current ripples are significant measures that indicate the controller performance, showing how well a controller tracks the desired performance in various operating conditions. The torque ripples T_{rip} and current ripples $i_{a_{rip}}$ are computed using (3.52) and (3.53).

Table 5.1 Torque ripples of MFPCC-I at different speeds and load conditions (Unit: Nm).

$T_{rip}(Nm)$		Speed (rpm)				
		200	**400**	**600**	**800**	**1000**
Torque (Nm)	**0**	0.152	0.155	0.158	0.153	0.157
	0.25	0.147	0.153	0.151	0.16	0.157
	0.5	0.155	0.164	0.158	0.16	0.15
	0.75	0.156	0.159	0.158	0.157	0.156
	1	0.152	0.153	0.159	0.154	0.157
	1.25	0.152	0.157	0.157	0.158	0.165
	1.5	0.158	0.156	0.162	0.157	0.16
	1.75	0.154	0.154	0.156	0.156	0.163
	2	0.159	0.158	0.162	0.16	0.162

Table 5.2 Torque ripples of MFPCC-II at different speeds and load conditions (Unit: Nm).

$T_{rip}(Nm)$		Speed (rpm)				
		200	**400**	**600**	**800**	**1000**
Torque (Nm)	**0**	0.115	0.116	0.117	0.119	0.119
	0.25	0.116	0.116	0.12	0.12	0.12
	0.5	0.116	0.113	0.12	0.117	0.118
	0.75	0.116	0.115	0.118	0.117	0.12
	1	0.115	0.115	0.119	0.117	0.119
	1.25	0.115	0.117	0.117	0.117	0.12
	1.5	0.116	0.117	0.12	0.12	0.121
	1.75	0.116	0.116	0.116	0.119	0.12
	2	0.114	0.116	0.118	0.116	0.122

Table 5.3 Torque ripples of proposed A-MFPCC at different speeds and load conditions (Unit: Nm).

$T_{rip}(Nm)$		Speed (rpm)				
		200	**400**	**600**	**800**	**1000**
Torque (Nm)	**0**	0.097	0.089	0.09	0.104	0.11
	0.25	0.123	0.123	0.123	0.123	0.124
	0.5	0.06	0.076	0.081	0.086	0.084
	0.75	0.058	0.058	0.065	0.07	0.074
	1	0.057	0.065	0.066	0.071	0.075
	1.25	0.06	0.061	0.069	0.071	0.077
	1.5	0.059	0.065	0.068	0.073	0.077
	1.75	0.06	0.065	0.07	0.075	0.077
	2	0.061	0.068	0.074	0.076	0.08

Table 5.4 Current ripples of MFPCC-I at different speeds and load conditions (Unit: A).

$i_{rip}(A)$		Speed (rpm)				
		200	**400**	**600**	**800**	**1000**
Torque (Nm)	**0**	0.371	0.355	0.353	0.354	0.365
	0.25	0.362	0.372	0.364	0.369	0.379
	0.5	0.377	0.378	0.385	0.383	0.39
	0.75	0.387	0.397	0.401	0.405	0.414
	1	0.403	0.416	0.417	0.437	0.44
	1.25	0.435	0.444	0.444	0.455	0.467
	1.5	0.461	0.465	0.478	0.491	0.498
	1.75	0.491	0.506	0.502	0.523	0.527
	2	0.533	0.515	0.524	0.55	0.552

Table 5.5 Current ripples of MFPCC-II at different speeds and load conditions (Unit: A).

$i_{rip}(A)$		Speed (rpm)				
		200	**400**	**600**	**800**	**1000**
Torque (Nm)	**0**	0.334	0.337	0.383	0.377	0.425
	0.25	0.325	0.332	0.361	0.37	0.37
	0.5	0.326	0.322	0.322	0.339	0.36
	0.75	0.314	0.316	0.314	0.336	0.351
	1	0.302	0.327	0.302	0.34	0.323
	1.25	0.31	0.31	0.3	0.309	0.326
	1.5	0.321	0.307	0.335	0.339	0.345
	1.75	0.32	0.334	0.332	0.329	0.36
	2	0.341	0.336	0.337	0.322	0.364

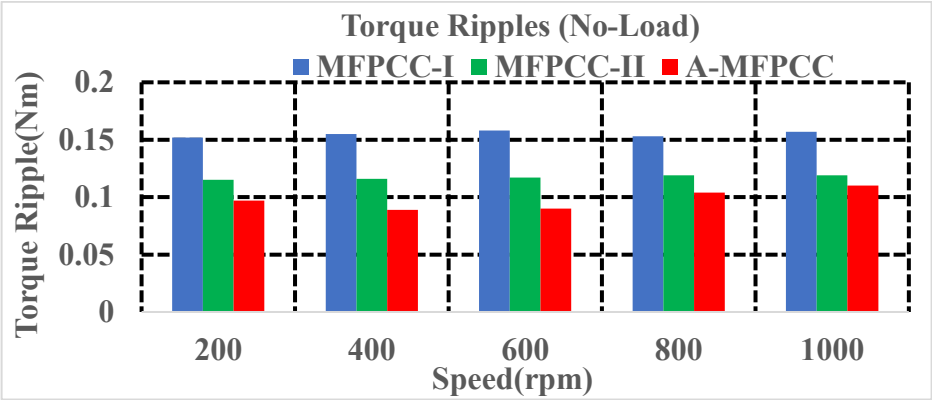
Table 5.6 Current ripples of proposed A-MFPCC at different speeds and load conditions (Unit: A).

$i_{rip}(A)$		Speed (rpm)				
		200	**400**	**600**	**800**	**1000**
Torque (Nm)	**0**	0.272	0.258	0.265	0.325	0.349
	0.25	0.299	0.317	0.315	0.325	0.331
	0.5	0.216	0.242	0.257	0.266	0.243
	0.75	0.189	0.213	0.216	0.227	0.216
	1	0.186	0.206	0.215	0.22	0.22
	1.25	0.177	0.209	0.213	0.226	0.229
	1.5	0.19	0.209	0.218	0.226	0.228
	1.75	0.189	0.216	0.223	0.235	0.235
	2	0.184	0.215	0.231	0.24	0.239

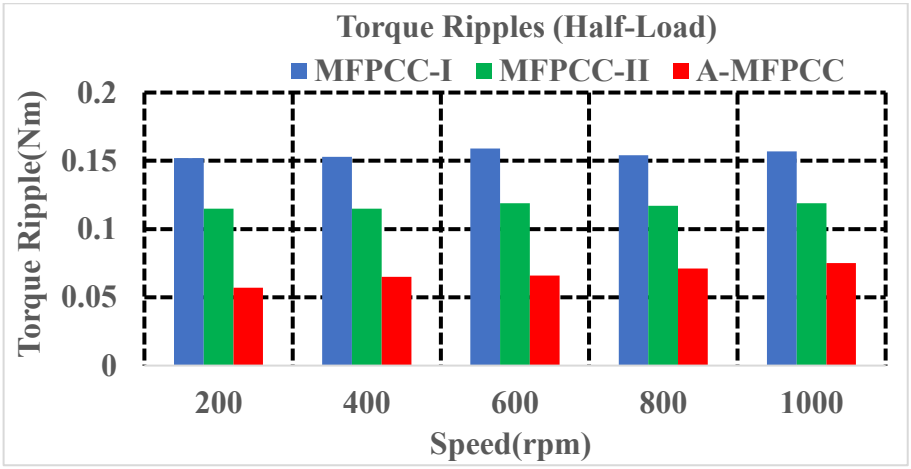
To visualize the differences in torque and current ripples among the three controllers, Figs. 5.20 and 5.21 present graphical representations of torque and current ripples for MFPCC-I, MFPCC-II, and A-MFPCC at different speeds (200, 400, 600, 800, and 1000 rpm) and load conditions (no-load, half-load, and full-load). These figures provide a clear overview of the variations in torque and current ripples.

The proposed A-MFPCC has demonstrated superior torque and current ripples performance among the three controllers (MFPCC-I, MFPCC-II, and A-MFPCC), revealing that the A-MFPCC consistently outperforms the other two controllers in terms of torque and current ripples at all operating conditions. For instance, at 1000 rpm with 2 Nm, the A-MFPCC recorded 0.08 Nm torque ripples and 0.239 A current ripples, while the MFPCC-II and MFPCC-I recorded 0.122 Nm and 0.162 Nm torque ripples and 0.364 A and 0.552 A current ripples, respectively. This demonstrates a significant improvement in torque ripples by 34.4% and 50.6% and current ripples by 34.3% and 56.7% for the A-MFPCC, respectively, compared to MFPCC-I and MFPCC-II.

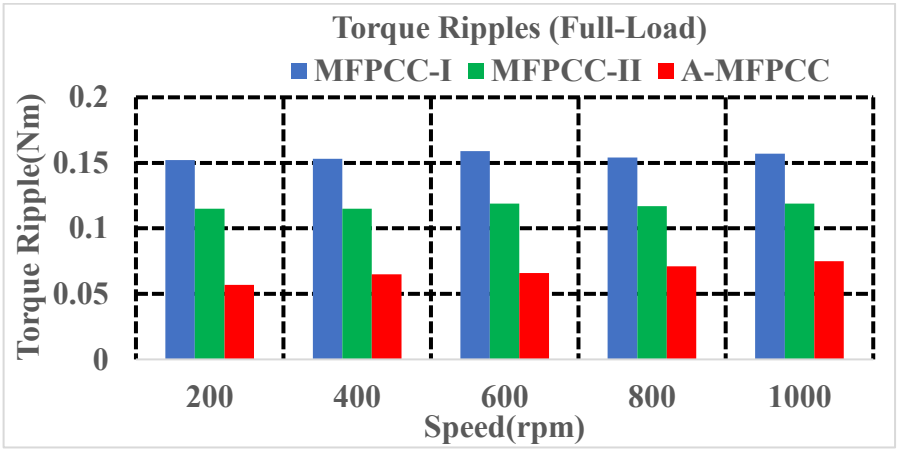
The quantitative analysis of torque and current ripples at various operating conditions provides strong evidence supporting the superiority of the proposed A-MFPCC over MFPCC-I and MFPCC-II. Additionally, the comparison between MFPCC-I and MFPCC-II demonstrates that MFPCC-II performs better, particularly in terms of current ripples, where MFPCC-I shows higher values. In conclusion, the A-MFPCC outperforms MFPCC-I and MFPCC-II in terms of torque and current ripples across different operating conditions, with MFPCC-II performing better than MFPCC-I.



(a)

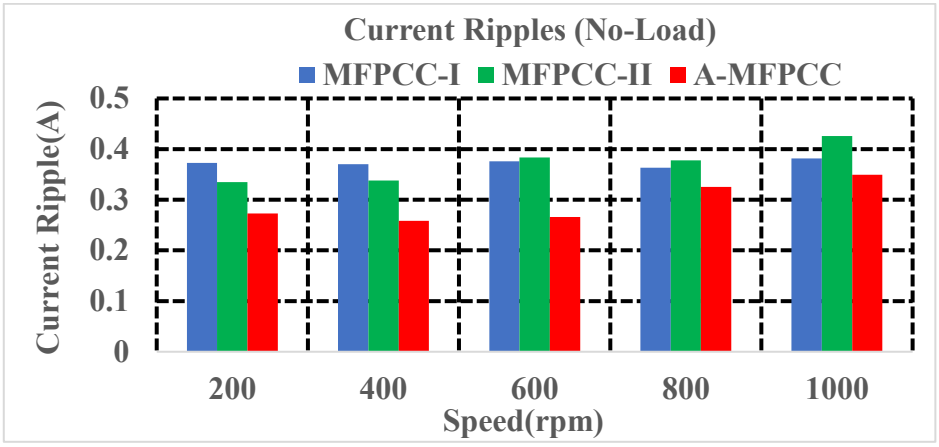


(b)

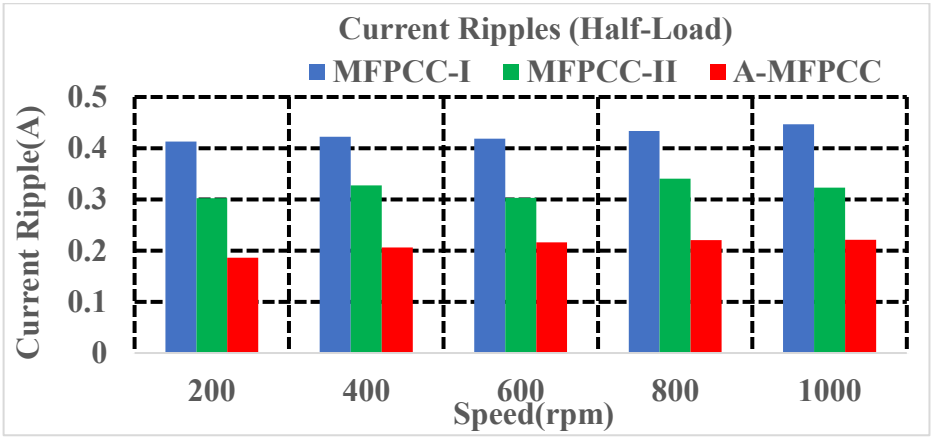


(c)

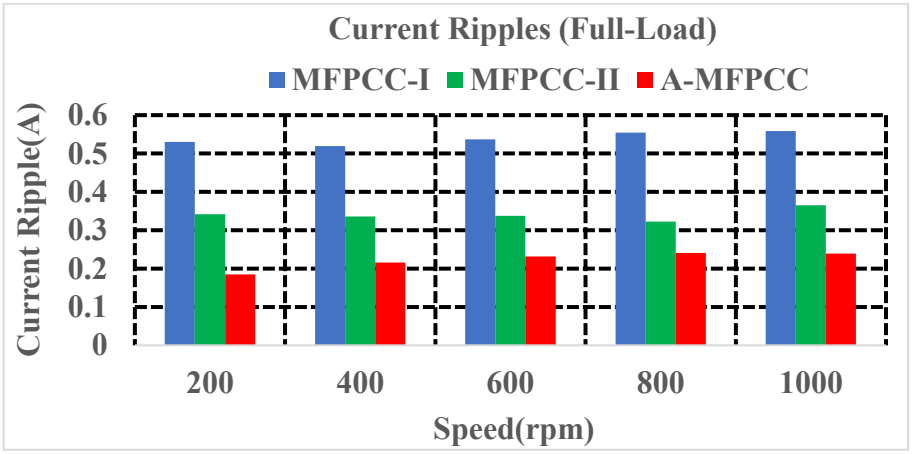
Fig. 5.20. Torque ripples for different control methods under various speed operations, (a) no load, (b) half-load, and (c) full load.



(a)



(b)



(c)

Fig. 5.21. Stator current i_a ripples for different control methods various speed operations, (a) no load, (b) half-load, and (c) full-load.

5.6 Robustness Evaluation

The simulation, experimental, and quantitative evaluations provide an initial assessment of the proposed A-MFPCC, MFPCC-I, and MFPCC-II controllers' performance robustness under varying operating conditions and parametric uncertainties. The results show that both controllers perform satisfactorily across a specific range of speed operations, torque conditions, and parameter mismatches. However, it is essential to note that these evaluations only test the controllers' performance under specific operating conditions and deterministic sets of single-parameter uncertainties. In practice, a broader range of parametric uncertainties can affect the controllers' robustness, and it is essential to consider stochastic sets of bounded range uncertainties to perform a comprehensive robustness evaluation. To this end, the Six Sigma robustness evaluation method introduced in Chapter 4 is used to numerically determine the controllers' robustness indexes (sigma level). This approach involves the use of large parameter sets and encompasses multiple performance indicators to ensure a comprehensive assessment.

A set of performance indicators K_i that can reflect the system's robustness to parameter variations are defined. This includes both transient and steady-state response characteristics, such as settling time (T_s), overshoot (OS), root mean square error of speed ($RMSE_\omega$), torque ripples (T_{rip}), and current ripples (i_{rip}). Each indicator is crucial in designing a PMSM drive for a specific application. Depending on the application requirements, each performance indicator (K_i) has a corresponding *USL*. For example, a racing car would necessitate a faster settling time (e.g., $T_s \leq 0.1s$), whereas a passenger car might allow for a slower settling time (e.g., $T_s \leq 0.2s$). Therefore, the upper specification limit for each performance indicator (K_i) varies based on the intended application.

The evaluation considers low, medium, and high-performance applications to cater to various performance requirements. Low-performance applications like water pumping systems have relatively lenient specification limits. Medium-performance applications, such as electric vehicles, have moderate limits. On the other hand, high-performance applications, such as radar systems and CNC machines, demand stringent and precise limits. Table 5.7 presents the performance indicators (K_i) and their respective USL for each application category (low, medium, and high performance).

Table 5.7 PMSM drive Performance indicators and limits considering three different applications requirements.

Performance Indicator (K_i)	upper Specification limits (USL)		
	Application-I	Application-II	Application-III
Ts	≤ 0.2	≤ 0.15	≤ 0.1
OS	$\leq 5\%$	$\leq 3\%$	$\leq 2\%$
$RMSE_\omega$	≤ 0.1	≤ 0.08	≤ 0.06
T_{rip}	≤ 0.8	≤ 0.6	≤ 0.4
$i_{a_{rip}}$	≤ 0.9	≤ 0.7	≤ 0.5

With the changes in operating conditions, PMSM parameters vary within a specific range [-%, +%] of their nominal values. Therefore, the possible variation ranges of PMSM electrical parameters considering rated and maximum operating conditions are presented in Table 4.1. The evaluation process involves generating 10,000 samples of normally distributed data of machine parameters ($R_s, L_d, L_q, \psi_{PM}, J, B$) between a bounded range (Table 4.1). The controllers are then simulated, and the resulting performance indicators, K_i,

are computed for each sample. Specifically, the motor runs from a standstill to the rated speed of 1000 rpm, with a steady state load torque of 2 Nm. The resulting 10,000 data points of each K_i are used to calculate the mean μ_i and standard deviation σ_i . The Z-value Z_i of each performance indicator, the system sigma level, n_{sys} and system probability of failure POF_{sys} are computed as discussed in Chapter 4.

The robustness evaluation process based on Six Sigma involves a few steps. The first step is to define a set of performance indicators, K_i, which can reflect the system's performance robustness. Then, set a robustness limit for each indicator $USLs$ such that any values exceeding this limit are non-robust. The machine parameters are generated based on two bounded ranges (Table 4.1) with (N=10000) normally distributed samples. Using Matlab/Simulink environment, each controller is simulated with all the samples, and the indicators are computed, resulting in 10000 data for each indicator. The indicators are computed by running the motor from a standstill to the rated speed (1000 rpm) and applying 2 Nm load torque at a steady state. The settling time T_s and overshoot OS performance indicators are computed during the transient state, while $RMSE_\omega$, T_{rip} and T_{rip} are computed during a steady state for a duration of 0.5 s.

Table 5.8 presents the results of the robustness evaluation for the proposed A-MFPCC, MFPCC-I, and MFPCC-II controllers, including their Z-values, system sigma levels, and POFs for three different application requirements and two parameter-variation ranges. The obtained system sigma levels n_{sys} clearly demonstrate the superiority of the proposed A-MFPCC, which achieved 6σ for four out of six cases. Under maximum condition uncertainties, A-MFPCC achieved 4.6σ and 3.9σ for Applications II and III, respectively, while for the other cases, it achieved 6σ. In comparison, MFPCC-I and MFPCC-II only

achieved 6σ in three cases. At the rated condition uncertainties, MFPCC-I and MFPCC-II achieved 2.5σ and 2.8σ for Application III, respectively. Under maximum condition uncertainties, MFPCC-I achieved 3.1σ and 1.7σ for Applications II and III, while MFPCC-II achieved 3.8σ and 2σ, respectively.

The Z-values of the performance indicators show the strengths and weaknesses of each control method and their ability to maintain a trade-off between all indicators. For instance, MFPCC-I showed good robustness in terms of overshot and RMSE speed but at the cost of torque and current ripples. Specifically, MFPCC-I showed low torque and current ripples Z-values under high uncertainties and application requirements, resulting in low system sigma levels despite their high Z-values of overshoot and RMSE. On the other hand, MFPCC-II's weakness is current ripples, but it showed strengths in terms of torque ripples and settling time. The proposed A-MFPCC, it may have sacrificed some robustness for specific indicators but maintained a trade-off between all performance indicators, resulting in a higher system sigma level. For example, the proposed A-MFPCC exhibits lower Z-values for overshoot Z_{OS} compared to MFPCC-I and lower Z-values for RMSE of speed Z_{ω}compared to MFPCC-II. This suggests that the A-MFPCC controller prioritizes a trade-off between torque and current ripple indicators, potentially at the expense of overshoot and RMSE performance indicators.

Table 5.8 Robustness evaluations of MFPCC-I, MFPCC-II, and A-MFPCC with two parameter uncertainty ranges based on three applications' requirements.

Indicator	Z_{T_s}	Z_{OS}	Z_{ω}	$Z_{T_{ripp}}$	$Z_{i_{ripp}}$	n_{sys}	POF
Controller	Rated condition uncertainties (application-I)						
MFPCC-I	61.5	192.5	593.5	18.6	12.7	6.0	0
MFPCC-II	82.2	26.1	349.6	35.5	16.4	6.0	0
A-MFPCC	68.6	51.6	331.4	27.3	29.4	6.0	0
	Maximum condition uncertainties (application I)						
MFPCC-I	46.2	170.2	312.6	11.5	7.0	6.0	0
MFPCC-II	61.4	25.4	234.1	21.0	8.9	6.0	0
A-MFPCC	50.2	44.1	167.1	13.9	22.0	6.0	0
	Rated condition uncertainties (application-II)						
MFPCC-I	43.0	113.0	366.2	12.6	8.2	6.0	0
MFPCC-II	58.7	15.1	321.4	24.8	10	6.0	0
A-MFPCC	49.3	28.6	264.8	19.5	20.1	6.0	0
	Maximum condition uncertainties (application II)						
MFPCC-I	32.2	99.7	261.3	7.8	4.4	3.1	0.1935%
MFPCC-II	43.8	14.7	212.4	14.6	5.3	3.8	0.0144%
A-MFPCC	35.3	24.3	133.4	9.8	14.9	4.6	≈ 0
	Rated condition uncertainties (application-III)						
MFPCC-I	24.4	73.2	138.7	6.5	3.7	2.5	1.2419%
MFPCC-II	35.2	9.6	70.3	14.1	3.6	2.8	0.5110%
A-MFPCC	28.2	17.1	63.2	11.7	10.8	6.0	0
	Maximum condition uncertainties (application-III)						
MFPCC-I	18.2	64.5	79.3	3.9	1.7	1.7	8.9130%
MFPCC-II	26.1	9.4	61.0	8.2	1.7	2.0	4.5500%
A-MFPCC	20.5	14.5	51.8	5.7	7.9	3.9	0.0096%

To illustrate the concept of the Z-value, capability plots of performance indicators, K_i (settling time, overshoot, torque, and current ripples), are presented for MFPCC-I, MFPCC-II, and the proposed A-MFPCC in Figs 5.22-5.35. These plots depict the position of specification limits (USL-I, USL-II, and USL-III) relative to each indicator's mean (μ_i) for the three controllers, considering two uncertainty ranges. The capability plots provide insights into the data distribution of each performance indicator and their proximity to the application's specification limits. These plots are particularly useful in evaluating the distance between the performance indicators and the specified limits for different applications (I, II, and III). The number of data samples exceeding the specification limits can impact the achieved sigma levels. For example, under maximum condition uncertainties, MFPCC-I surpasses the limits of application-III in 891 samples, resulting in an overall system sigma level of 1.7σ.

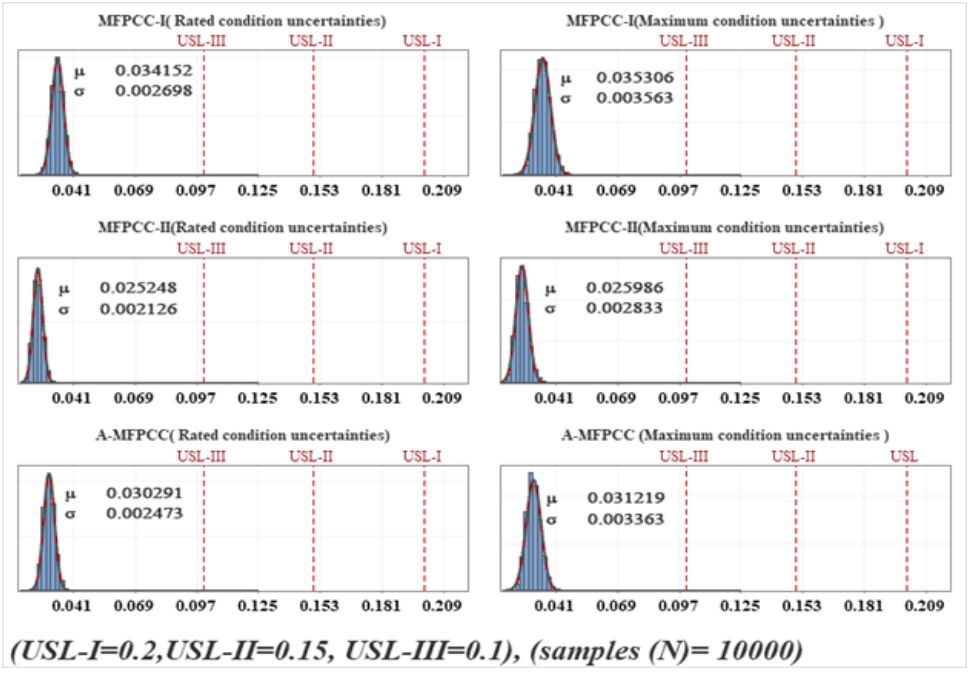


Fig. 5.22. Capability plot of settling time (T_s) at two uncertainty ranges for MFPCC-I, MFPCC-II, and the proposed A-MFPCC.

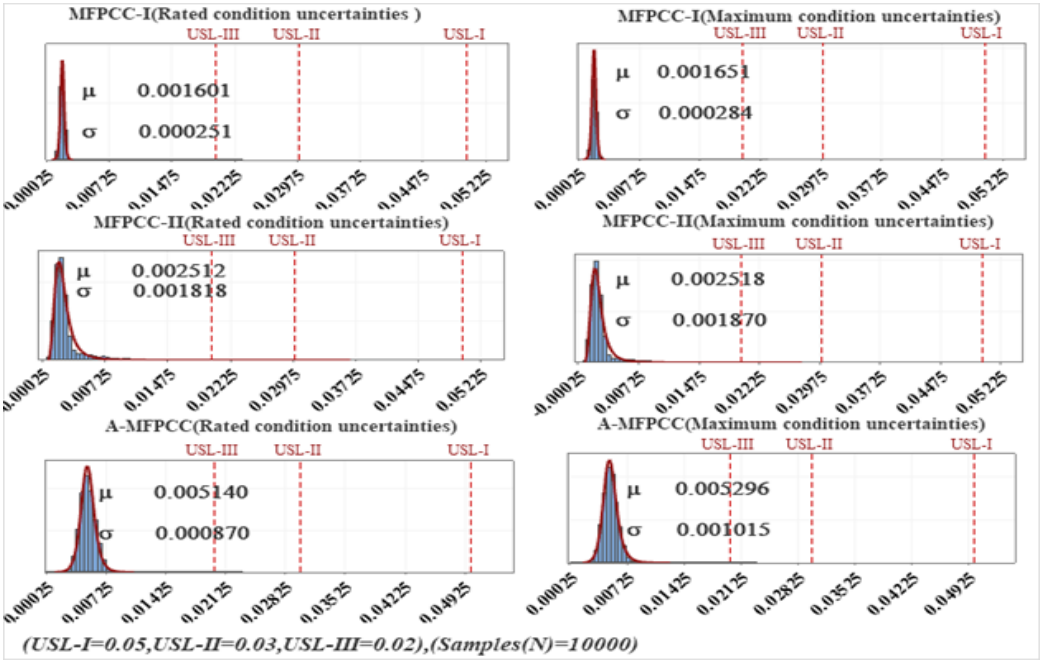


Fig. 5.23. Capability plot overshoot at two uncertainty ranges for MFPCC-I, MFPCC-II, and the proposed A-MFPCC.

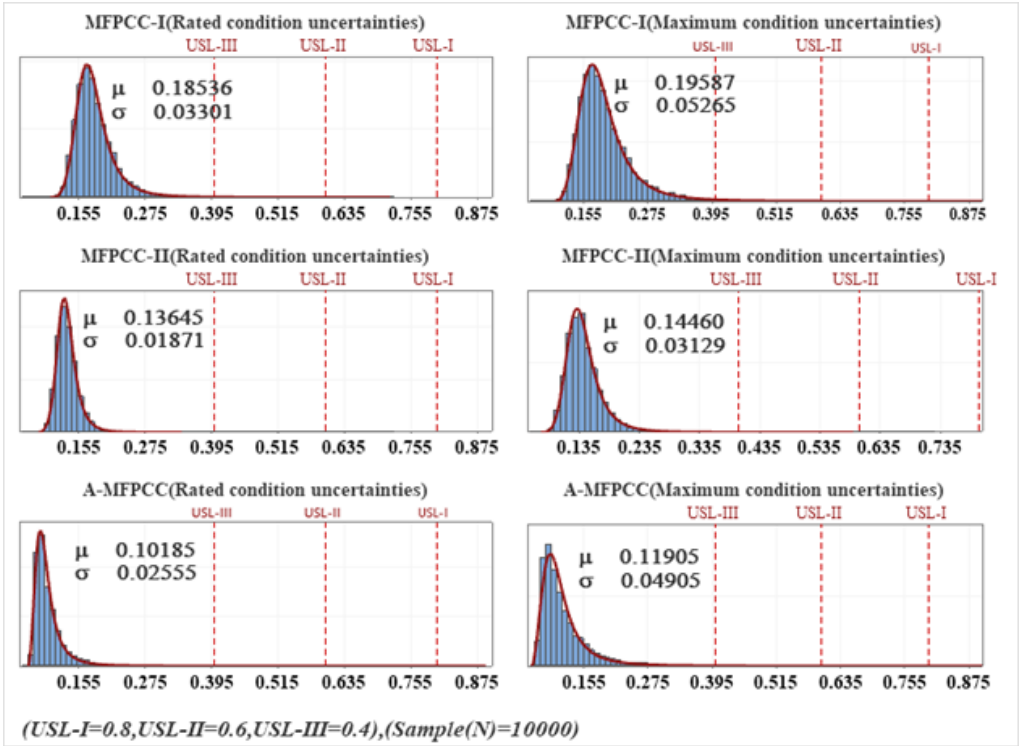


Fig. 5.24. Capability plot torque ripple at two uncertainty ranges for MFPCC-I, MFPCC-II, and the proposed A-MFPCC.

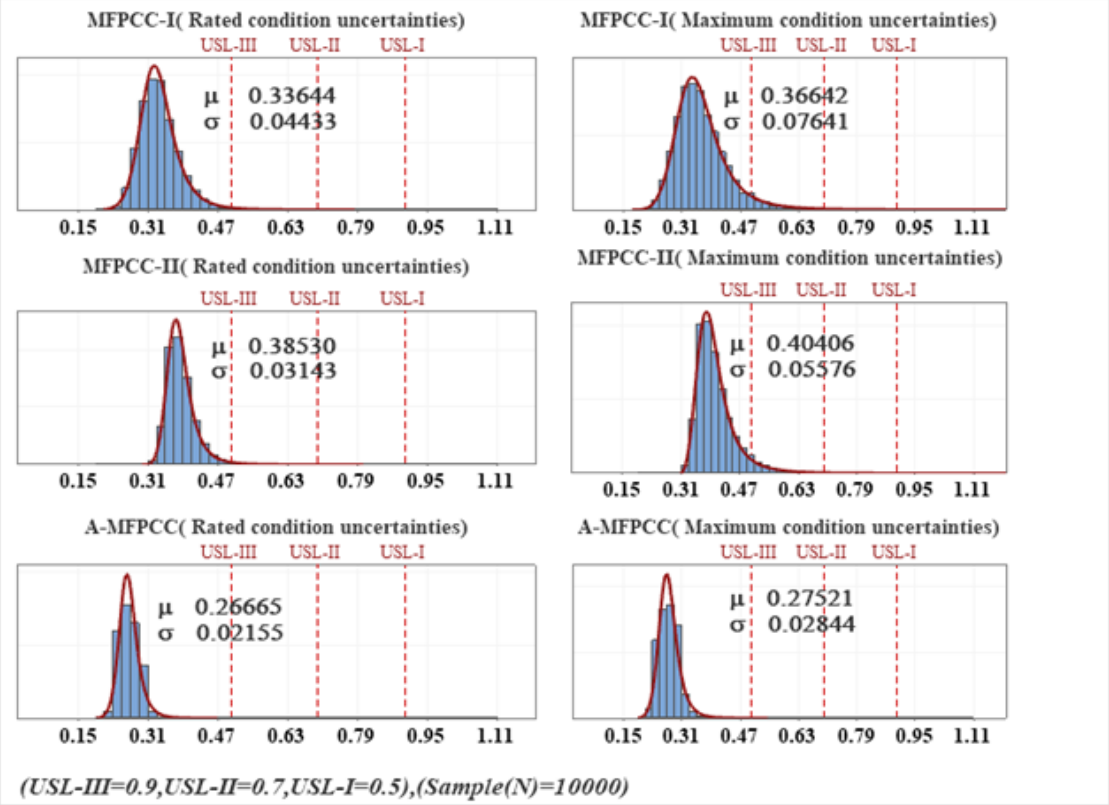


Fig. 5.25. Capability plot of current ripple at two uncertainties ranges for MFPCC-I, MFPCC-II, and the proposed A-MFPCC.

5.7 Summary

The proposed A-MFPCC eliminates the prediction based on a simplified parametric machine model by developing a model-free prediction model utilizing a modified current difference updating technique with the inclusion of the reference tracking error. The work presented in this chapter can be summarized into the following:

- An incremental prediction model with two lumped parameters is derived and used to develop two conventional MFPCC methods (MFPCC-I, MFPCC-II).
- A reference voltage vector was generated based on the reference current vector position and the tracking error and used to obtain the current difference due to the applied and

reference voltage vector. Then, the current difference due to the possible voltage vector was derived without using any machine parameter (A-MFPCC).

- The proposed A-MFPCC effectiveness was validated by comparison with two MFPCC methods (MFPCC-I, MFPCC-II) based on simulation and experimental results.
- The proposed A-MFPCC showed superior performance in terms of different measures with various parameter variations and operating conditions.
- The robustness evaluation showed that the proposed A-MFPCC achieved a sigma level much higher than the other two methods.

Apart from that, model-free approaches including MFPCC-I,MFPCC-II and A-MFPCC are data-driven controllers that rely on the measured current and voltage to predict the future behavior of the motor current. Sensor noise can introduce errors in these measurements, leading to inaccuracies in the predicted current. This can result in suboptimal control performance. However, the proposed A-MFPCC generate the switching pulses based on measured current differences, voltage deviation and a generated reference voltage. This reference voltage is generated based on the error between the measured current and the reference current. Theferoe, the sensor noise impact on the prediction accuracy is minimized, because the the swithcing pulse are selected such that minimum current error is produced.

Furthermore, the proposed A-MFPCC require the calculation of a reference voltage vector and need the current difference and voltage deviation over two concective sampling cycles to perform current prediction. This require faster sampling frequency for better performance . The various calculation algorithms and faster sampling requirement can be a limitation in terms of computational resources and real-time processing.

REFERENCES

[5.1] M. Khalilzadeh, S. Vaez-Zadeh, J. Rodriguez, and R. Heydari, "Model-Free Predictive Control of Motor Drives and Power Converters: A Review," *IEEE Access,* vol. 9, pp. 105733-105747, 2021, doi: 10.1109/access.2021.3098946.

[5.2] M. Fliess and C. Join, "Model-free control," *International Journal of Control,* vol. 86, no. 12, pp. 2228-2252, 2013/12/01 2013, doi: 10.1080/00207179.2013.810345.

[5.3] Z. Sun, Y. Deng, J. Wang, T. Yang, Z. Wei, and H. Cao, "Finite Control Set Model-Free Predictive Current Control of PMSM With Two Voltage Vectors Based on Ultralocal Model," *IEEE Transactions on Power Electronics,* vol. 38, no. 1, pp. 776-788, 2023, doi: 10.1109/TPEL.2022.3198990.

[5.4] X. Li, Y. Wang, X. Guo, X. Cui, S. Zhang, and Y. Li, "An Improved Model-Free Current Predictive Control Method for SPMSM Drives," *IEEE Access,* vol. 9, pp. 134672-134681, 2021, doi: 10.1109/ACCESS.2021.3115782.

[5.5] C. Lin, T. Liu, J. Yu, L. Fu, and C. Hsiao, "Model-Free Predictive Current Control for Interior Permanent-Magnet Synchronous Motor Drives Based on Current Difference Detection Technique," *IEEE Transactions on Industrial Electronics,* vol. 61, no. 2, pp. 667-681, 2014.

[5.6] P. G. Carlet, F. Tinazzi, S. Bolognani, and M. Zigliotto, "An Effective Model-Free Predictive Current Control for Synchronous Reluctance Motor Drives," *IEEE Transactions on Industry Applications,* vol. 55, no. 4, pp. 3781-3790, 2019, doi: 10.1109/TIA.2019.2910494.

[5.7] C. Ma, H. Li, X. Yao, Z. Zhang, and F. D. Belie, "An Improved Model-Free Predictive Current Control With Advanced Current Gradient Updating Mechanism," *IEEE Transactions on Industrial Electronics,* vol. 68, no. 12, pp. 11968-11979, 2021, doi: 10.1109/TIE.2020.3044809.

[5.8] X. Zhang, L. Zhang, and Y. Zhang, "Model Predictive Current Control for PMSM Drives With Parameter Robustness Improvement," *IEEE Transactions on Power Electronics,* vol. 34, no. 2, pp. 1645-1657, 2019, doi: 10.1109/TPEL.2018.2835835.

[5.9] F. M. L. De Belie, P. Sergeant, and J. A. Melkebeek, "A Sensorless Drive by Applying Test Pulses Without Affecting the Average-Current Samples," *IEEE Transactions on Power Electronics,* vol. 25, no. 4, pp. 875-888, 2010, doi: 10.1109/tpel.2009.2036617.

CHAPTER 6

ROBUST MODEL-FREE REINFORCEMENT LEARNING-BASED CURRENT CONTROL OF PMSM DRIVES UNDER MULTIPLE UNCERTAINTY SETS

6.1 Introduction

Model uncertainties and parameter mismatching are unavoidable in PMSM drives, and they significantly affect the performance and robustness of model-dependent control approaches [6.1]. In response to this challenge, data-driven control approaches are utilized to eliminate the dependency on machine models and parameters. Model-free predictive control (MFPC) is a well-known data-driven control in PMSM drives. However, inaccurate estimation and stagnation can occur depending on the technique used to achieve MFPC (such as current difference and ultra-local model), and higher computational effort may be required [6.2-6.3].

Recently, reinforcement learning (RL) has emerged as a promising approach for achieving data-driven control in PMSM drives. An optimal control action can be obtained by training an RL agent with appropriate rewards based on measured PMSM data. This results in a computationally efficient controller optimized offline during training [6.4-6.5]. Unlike MFPC, which requires continuous optimization during online control and can be computationally intensive, RL-based controllers offer the advantage of reduced computational effort.

The effectiveness of RL-based controllers depends on the amount and quality of the data used for training. In the standard RL-based controller, an agent learns an optimal

policy that maximizes its expected cumulative reward over a single training task with specific operating conditions and a single parameter set. Thus, new operating conditions and different parameter sets (due to parameter mismatching) can lead to poor performance, robustness, or instability in the controlled system [6.6]. A robust RL-based controller is often employed to deal with this issue, in which an agent learns an optimal policy that can perform well with worst-case uncertainty [6.7]. However, it may limit the flexibility of the learned policy or result in an overly conservative policy that performs poorly with all other cases, including nominal cases.

This research proposes a multi-set robust RL (MSR-RL) based current control of PMSM drive. The MSR-RL aims to learn a single optimal policy robust to several different parameter sets. This is done by leveraging the multi-task RL setting to optimize a policy that can generalize to and provide good worst-case performance with respect to new parameter sets. Instead of learning a policy over a single training task with a single parameter set, the proposed MSR-RL learns a single policy over multiple training tasks with various parameter sets. The resultant policy can be robust to new parameter sets and generalized to the remaining ones.

6.2 Reinforcement Learning Basics

RL is a machine learning technique where an agent learns to make decisions in an environment to maximize a reward signal. The agent interacts with the environment by taking action and receiving feedback through rewards or punishments. RL aims to learn a policy that maps states to actions, maximizing the expected cumulative reward [6.8].

RL can be applied in control systems to learn control policies that optimize objectives, such as minimizing energy consumption or maximizing performance. It can

also be used to learn control policies for systems that are challenging to model or have complex dynamics, such as robots or autonomous vehicles [6.9]. Incorporating RL into control enables more flexible and adaptive control systems that can improve their performance through experiential learning. However, using RL in control presents challenges such as balancing exploration and exploitation, addressing partial observability, and managing environmental uncertainty [6.10].

The learning process relies on a set of decisions aimed at maximizing the cumulative reward, which is predetermined. The agent's input signals consist of observations and rewards. Observations encompass a predefined set of signals that characterize the process, while rewards measure the success of the resulting action signal. The control quantities of the controlled process represent the actions, while observations encompass the signals visible to the agent and take the form of measured signals, their rate of change, and associated errors [6.11].

Fig. 6.3 shows the RL process's general block diagram, which includes an agent, environment, action, observations, and rewards. At each time step k, the agent executes an action, a_k, and receives observations O_k and rewards R_k. The environment receives an action, a_k , and emits observation O_{k+1} and scalar reward R_{k+1}. A reward R_k is a scalar feedback signal which indicates how well the agent is doing at time step k. RL is based on the reward hypothesis, which is defined such that the maximization of expected cumulative reward can describe all goals: $\max E[\sum_{i=0}^{\infty} R_{k+i+1}]$. The agent's job is to maximize the return g_k which is the discounted future rewards:

$$g_k = \sum_{i=0}^{\infty} \gamma^i R_{k+i+1} \tag{6.1}$$

Another essential part of the RL process is the history and state. History is the sequence of observations, actions, and rewards expressed as the following:

$$H_k = O_1, R_1, a_1, \ldots, a_{k-1}, O_k, R_k \tag{6.2}$$

The state is the information used to determine what happens next; formally, it is a function of history: $S_k = f(H_k)$. There are two types of states, the environmental state, S_k^e, whatever data the environment uses to pick the next observation/reward. Agent state S_k^a, whatever information the agent uses to pick the next action.

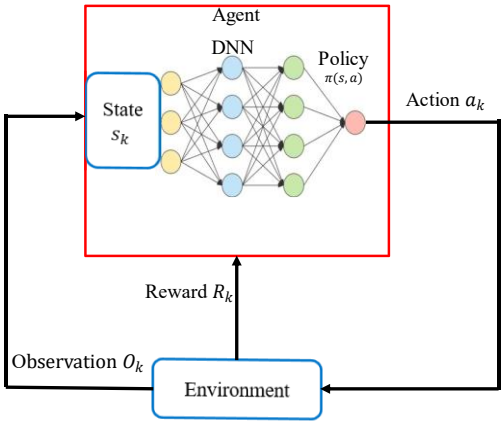


Fig. 6.1. General block diagram reinforcement learning process.

The main components of an RL agent include the policy and the value function. The policy represents the agent's behaviour as a mapping from the state to the action. The value function is a prediction of future rewards, which is used to evaluate the goodness or badness of states and, therefore, to select actions. RL agents can be further categorized into model-based and model-free RL. In model-based RL, the agent uses a model to represent the environment, predict the next observations, and execute an action, regardless of the consequences. On the other hand, in model-free RL (MFRL), the agent does not use a model but learns through experience, taking into account the consequences of applying an action [6.12].

In general, RL is a fundamental machine learning paradigm typically formulated as a Markov decision process (MDP), represented by a tuple (S, A, P, R, γ). In this formulation, the environment defines the state space (S), while the agent holds the action

space (A). The agent interacts with the environment to update its policy (π), which maps environment states to actions. During each iteration, the agent selects an action ($a_k \in A$) based on its policy (π). Subsequently, the environment generates the next state (s_{k+1}) according to its transition probability (P). This probability function takes the current state and action as inputs and produces a distribution over the possible next states. The environment also provides instantaneous feedback to the agent in the form of a reward (R). This iteration continues until the agent discovers the optimal policy (π^*) that maximizes the infinite horizon discounted reward $J(\pi)$. The optimization problem for finding the optimal policy can be expressed as follows:

$$\pi^* \in \arg\max_{\pi} J(\pi) = E \sum_{k=1}^{\infty} \gamma^k \; R\,(s_k, a_k) \tag{6.3}$$

where γ represents the discounting factor, and $J(\pi)$ denotes the infinite horizon discounted reward, E denotes the expectation over possible sequences of states and rewards generated by following policy π. The optimal policy ensures that the agent accumulates the maximum possible reward from the environment.

In the context of Model-Free RL (MFRL), both the action space (A) and state space (S) can be either continuous or discrete. MFRL can be categorized into two main types: value-based and policy-based agents. Policy-based agents estimate the desirability of a state s by using a state value function $V(s)$. Examples of such agents include deep deterministic policy gradient (DDPG) and actor-critic (AC) algorithms. On the other hand, value-based agents estimate the quality of state s by utilizing the state-action value function $Q(s, a)$, as demonstrated in the deep Q-learning network (DQN) agent. Initially, Q-values in Q-learning were stored in a Q-table [6.13]. However, due to the inefficiency of the table-based approach, a deep neural network (DNN) with powerful approximation capabilities was introduced [6.14]. This neural network maps states to Q-values more

efficiently, effectively replacing the Q-table. Consequently, this approach became known as DQN RL.

DQN RL agents are well-suited for discrete control tasks such as an inverter (on/off control). Thus, in this research, a model-free DQN RL algorithm is employed. The state-action value function of DQN $Q_\pi(s,a)$ defines the value of being in state s, taking an action a, and subsequently following a policy π. The Bellman optimality equation for Q^* is expressed as follows:

$$Q^*(s_k,a_k) = r(s_k,a_k) + \gamma \max_{a_{k+1}} Q^*(s_{k+1},a_{k+1}) \tag{6.4}$$

In DQN RL, a learning rate α is utilized to fine-tune the optimization process and regulate the adjustment of neural network weights to minimize the loss function. The learning rate is a parameter that falls within the range of 0 to 1 and plays a crucial role in determining the step size for each episode. The new Q-value for a specific state-action pair, $Q^{new}(s_k,a_k)$ is expressed as follows:

$$Q^{new}(s_k,a_k) = (1-\alpha)\overbrace{Q(s_k,a_k)}^{old} + \alpha\overbrace{\left(r(s_k,a_k) + \gamma \max_{a_{k+1}} Q^*(s_{k+1},a_{k+1})\right)}^{learned\ value} \tag{6.5}$$

Thus, the new Q-value is equal to a weighted sum of our old value and the learned value. Based on the obtained new Q-value , the policy π is improved by computing a better policy as follows:

$$\pi_{k+1}(s_k,a_k) = arg\max_{a_{k+1}} Q_k^\pi(s_k,a_k) \tag{6.6}$$

This process is repeated until convergence, i.e., when the policy cannot be improved anymore. The goal of the DQN is to meet the convergence of Q-value; thus, (6.4) can be further expressed as the cost function $J(\theta)$:

$$J(\theta) = (Q(s_k, a_k; \theta) - r(s_k, a_k) + \gamma \, max \; Q(s_{k+1}, a_{k+1}; \theta))^2 \tag{6.7}$$

where $Q(s_k, a_k; \theta)$ represents approximation function, $r(s_k, a_k)$ represents previous rewards, $\gamma \, max \; Q(s_{k+1}, a_{k+1}; \theta)$ denotes the immediate and future rewards, and θ represents the network weights for training. The lower the cost function, the lower the difference between the predicted and target Q values.

6.3 Multi-Set Robust RL-Based Current Control of PMSM Drives

In standard RL, an agent learns an optimal policy that maximizes its expected cumulative reward over a single training task with specific operating conditions and a single parameter set. Thus, new operating conditions and different parameter sets (due to parameter mismatching) can lead to poor performance, robustness, or instability in the controlled system. A common solution to this issue is using a robust RL-based controller in which an agent learns an optimal policy that can perform well with worst-case uncertainty [6.7]. However, the worst case incorporates a single parameter set applied for a single training task which may limit the flexibility of the learned policy. Using a large uncertainty case can result in an overly conservative policy that performs poorly with all other cases, including nominal cases.

The optimal learned policy varies for different parameter sets, making it difficult to generalize and adapt a learned policy to new operating conditions with new parameter sets. To generate a policy that can adapt to new tasks (e.g., new motors), meta-RL [6.15] is used to learn a policy that can adapt to new tasks more efficiently and quickly by leveraging prior experience on similar tasks. With a data set of different motor parameters, the environment of each motor data was pictured as a partially observable Markov decision process (POMDP), where the environment state is not fully available to

the agent. Then, additional contexts (variables) containing information about the momentary environment are included in the environment state. However, these contexts can be static within each measurement, and incorporating them into the state creates a larger COMDP and reduces the learned policy's generalizing power [6.16].

Furthermore, Meta-RL can be computationally intensive since it requires much data to learn the meta-policy and adapt to new tasks efficiently. Additionally, there is a risk of overfitting to the training tasks, where the agent memorizes the training tasks and cannot adequately generalize to test tasks. This can lead to poor performance on unseen tasks or tasks significantly different from the training tasks [6.17].

This research proposes a multi-set robust RL (MSR-RL) based current control of PMSM drive. The MSR-RL aims to learn a single optimal policy robust to several different parameter sets. This is done by leveraging the multi-task RL setting to optimize a policy that can generalize to and provide good worst-case performance with respect to new parameter sets. Instead of learning a policy over a single training task with a single parameter set, the proposed MSR-RL learns a single policy over multiple training tasks with various parameter sets. The resultant policy can be robust to new parameter sets and generalized to the remaining ones.

The parameter sets are referred to as contexts, and each context's environment is represented as a Contextual Markov decision process (CMDP). The objective is to learn a policy that maximizes the accumulated reward over all contexts. Each CMDP represents a training task during the training, and tasks with shared information are clustered into models. These models are exploited to create a robust unified policy for all the clustered models and new unseen models. To illustrate the concept of the proposed MSR-RL and

how it is different from standard single-task RL and meta-RL, their general frameworks are presented in Fig. 6.2.

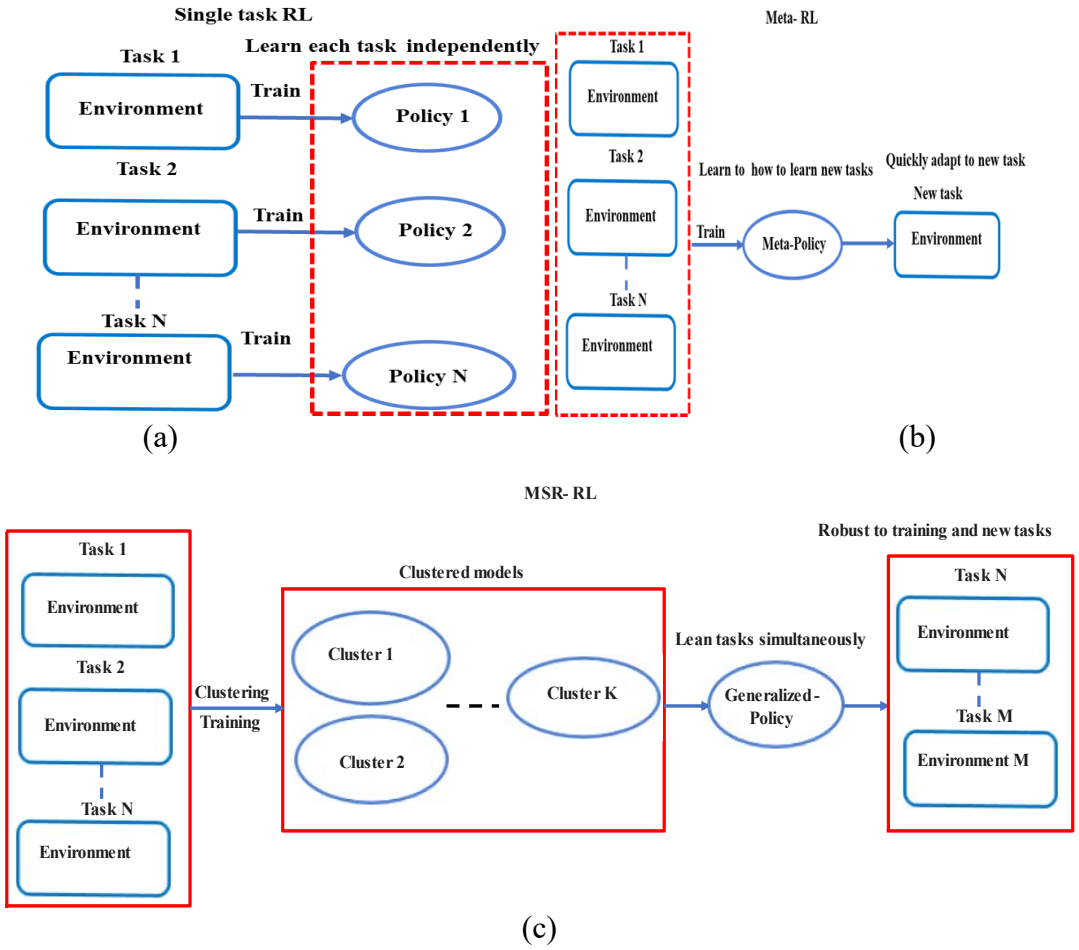


Fig. 6.2. Reinforcement learning frameworks, (a) single task RL, (b) meta-RL, and (c) multi-set robust RL.

6.3.1 Controller Design

The environment in MSR-RL is represented by CMDP with C context space (parameter sets). The CMDP is defined by a tuple $(C;\ S;\ A;\ M(c))$ where C is the context space; S and A are the state and action space, and M is function mapping any context $c \in C$ to an MDP $M(c) = (S, A, p(s,a), r(s,a))$.

With $H = \{h_1, h_2 \ldots . h_n\}$ training episodes, the environment applies a context

(set of PMSM parameters) $c \in C$ at the beginning of each training episode. Then, an initial state is generated based on the initial policy, π_0^*, and the agent interacts with the environment as in the standard RL. With a finite context space C, the goal is maximizing cumulative rewards for all contexts. To learn a policy that is robust to all context space C (uncertainty sets) and can be generalized to a new uncertainty set, the regret concept R is used. Regret measures the difference between cumulative rewards generated with the current policy and the best possible policy in hindsight. Mathematically, the regret R is expressed as the difference between the rewards of potential action and the action that has been taken as the following:

$$R_h = Q_\pi^*(s_h, a_h) - \sum_{k=1}^{K} r(s_k, a_k) \tag{6.8}$$

where $Q_\pi^*(s_h, a_h)$ is expected discounted rewards for applying optimal policy π^* and $r(s_k, a_k)$ is the reward at time step k in a training episode h with context $c \in C$.

The goal is to learn a policy that bounds the cumulative regret, R_n, i.e., the sum of the regrets for the number of training episodes, n, converges to a small value ϵ, as the following:

$$R_n = \sum_{h=1}^{n} Q_\pi^*(s_h, a_h) - \sum_{h=1}^{n} \sum_{k=1}^{K} r(s_k, a_k) \tag{6.9}$$

A good policy is expected to strike a balance between the exploration of unvisited action spaces and the exploitation of visited action spaces. In standard RL training, exploration is typically performed within a single context. However, in the proposed MSR-RL, multiple contexts need to be explored. To achieve this, a finite context space C is considered, where each context $c \in C$ corresponds to a distinct parameter set and is referred to as a task. Each task is represented by MDP model, thus with C tasks, multiple

models $(M_1, M_2, \ldots, M_C)$ are created. The model's information $(M_1(c), M_2(c), \ldots, M_C(c))$ are partitioned into mini-batches. Then, in each training episode, previously observed models are clustered together to form a single representative model, facilitating information consolidation. Concurrently, new models are explored to acquire additional knowledge and expand the context space.

Let M_1 and M_2 to be represented by two MDPs with the same action and state space, and the same context $c \in C$, the ϵ-approximated model relationship can be defined as follows:

$$M_1(c) = \big(S, A, p_1(s,a), r(s,a)\big) \tag{6.10}$$

$$M_2(c) = \big(S, A, p_2(s,a), r(s,a)\big) \tag{6.11}$$

M_2 is said to be ϵ-approximated model of M_1 if the state action pairs (s, a) meet:

$$|p_1(s,a) - p_2(s,a)| \leq \epsilon \tag{6.12}$$

Through the clustering and exploration process, a set of N models are identified, encompassing a diverse range of contexts. These models are then employed for exploitation, wherein a unified policy is generated to accommodate all the formed models and adapt to new ones. By leveraging the identified models, the policy aims to maximize performance across different contexts, minimizing regret during learning. The proposed MSR-RL learning framework shown in Fig. 6.3 utilizes an additional DNN for clustering, maximizing the generalization power of the learned policy and avoiding the creation of large models. In contrast to meta-RL approaches based on POMDPs, which incorporate a hidden context into each model state S regardless of previous models. By avoiding the creation of large models with distinct and unrelated dynamics, the learned policy can be effectively generalized to diverse models.

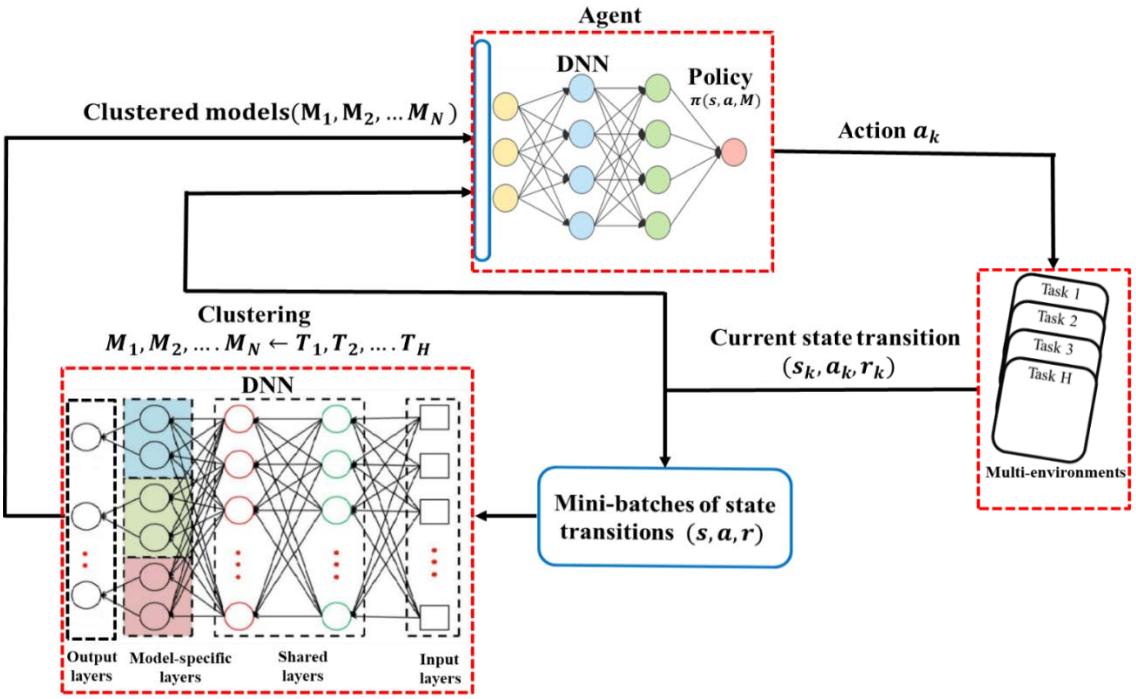


Fig. 6.3. The Proposed MSR-RL learning frameworks.

6.3.2 The RL Environment

The RL environment serves as a platform for the interaction between an RL agent and its surroundings, enabling the agent to learn and optimize its decision-making process. In the case of PMSMs, the RL environment represents the operational context in which the motor operates. It encompasses various components and factors that influence motor behaviour, including the physical characteristics of the motor, such as electrical and mechanical parameters, as well as its internal dynamics and external factors affecting performance. These factors include load torque, rotor position, stator current, and electrical and mechanical constraints. In general, RL-based current control of PMSM drives is trained offline using a simulation model of the PMSM drive. This model is a mathematical representation that describes the dynamics of the PMSM and is expressed in the *dq*- coordinates as follows:

$$v_d = R_s i_d + L_d \frac{di_d}{dt} - \omega L_q i_q \tag{6.13}$$

$$v_q = R_s i_q + L_q \frac{di_q}{dt} + \omega L_d i_d + \omega \Psi_{pm} \tag{6.14}$$

$$\frac{di_d}{dt} = -\frac{R_s}{L_d} i_d + \frac{L_q}{L_d} \omega i_q + \frac{1}{L_d} v_d \tag{6.15}$$

$$\frac{di_q}{dt} = -\frac{R_s}{L_q} i_q - \frac{L_d}{L_q} \omega i_d - \frac{\Psi_{pm}}{L_q} \omega + \frac{1}{L_q} v_q \tag{6.16}$$

However, this model is a simplified representation that neglects several real-world PMSM dynamics. Additionally, the model heavily relies on machine parameters, which can vary due to machine structure and changes in operating conditions. Training an RL agent solely based on this model using a single parameter set, such as nominal or worst-case parameters, can limit the flexibility of the learned policy. Consequently, parameter mismatching can lead to poor performance, lack of robustness, or instability in the controlled system. The proposed MSR-RL method addresses this challenge, which trains the RL agent based on multiple parameter sets. By considering various parameter sets, the agent learns a single optimal policy that is robust to these sets and adaptive to new ones. This approach aims to mitigate the adverse effects of parameter mismatching and enhance the performance, robustness, and stability of the controlled PMSM system.

6.3.3 Observations, Rewards, and Action

The selection of appropriate observations, rewards, and actions is a crucial factor in determining the effectiveness of the RL-based controller. For current control of PMSM drives, standard observations are measured and reference *dq* currents, measured and reference speed, measured position, and *dq* voltages, expressed as:

$$o^k = \left[\omega^*, \omega, \theta, i_d^*, i_d^k, i_q^*, i_q^k, v_d^{k-1}, v_q^{k-1}\right]^T \tag{6.17}$$

In various MFPCs, the effect of parameter variations is compensated for by estimating the current gradient. Thus, to help the RL agent learn better and account for

parameter mismatching, the current gradient is included in the observation space as follows:

$$o^k = [\omega^*, \omega, \theta, i_d, i_d^k, \Delta i_d^{k-1}, i_q, i_q^k, \Delta i_q^{k-1}, v_d^{k-1}, v_q^{k-1}]^T \tag{6.18}$$

where $\Delta i_d^{k-1} = i_d^k - i_d^{k-1}$ and $\Delta i_q^{k-1} = i_q^k - i_q^{k-1}$ are the *dq* current gradients.

Rewards are an essential part of the RL learning process; it tells the agent how good or bad the selected action is. Therefore, rewards must be appropriately calculated to help the agent learn an optimal policy. In current control, the objective is to minimize the current error, and the quadric objective function can be written as:

$$g^k = \left(i_d^* - i_d^k\right)^2 + \left(i_q^* - i_q^k\right)^2 \tag{6.19}$$

The objective function g^k, along with constraint violation penalty and past action, are employed as a reward signal for RL agent as follows:

$$r^k = -\left(w_1 g^k + w_2 P^k + w_3 \sum_j u_j^{k-1} \times u_j^{k-1}\right) \tag{6.20}$$

where w_1, w_2 and w_3 are the reward gains, and u_j^{k-1} is the past control action; P^k is a penalty term to ensure safe operation and discourage overcurrent region during the training by penalizing the agent when the measured current $i_s^k = \sqrt{\left(i_d^k\right)^2 + \left(i_q^k\right)^2}$ exceeds the nominal current i_n, and can be calculated as the following:

$$P^k = \begin{cases} (i_n - i_s^k)^2 \ , i_s^k > i_n \\ 0 \ , otherwise \end{cases} \tag{6.21}$$

The PMSM drive is characterized by a discrete action space (switching vectors. s_j); the DQN RL agent is utilized in this research. Based on the observation's signals o^k, and rewards r^k, the RL agent will select an action a^k. The action is a switching vector to

operate the inverter. With a two-level three-phase inverter, eight possible vectors can be applied. Thus, the RL action can be expressed as:

$$a^k = s_j \in \{0,1,2 \dots \dots 7\} \tag{6.18}$$

RL algorithms use function approximators, such as neural networks, to estimate the values or policies. These networks can be sensitive to the scale of the input data, and significant differences in scales may lead to difficulties in training. Normalizing the observations can mitigate issues such as vanishing or exploding gradients, which can hinder learning and slow down convergence. Therefore, the observations are normalized to a range of [-1, 1].

The working principle of the proposed MSR-RL-based current control of PMSM drives can be illustrated as follows. First, finite parameter sets (context) are created. Second, at the beginning of each training episode, a context $c \in C$ is randomly selected by the environment, where each context is represented by MDP (M_C). Afterwards, an initial state is chosen according to an initial state distribution, π_0^*. Finally, the agent interacts with the chosen M_c for K time steps (k=1,2, 3,…,K) as in the standard RL. In the next training episode, new contexts are selected, similar models are clustered, and new models are explored. Thus, N-clustered models are identified, and then a single policy is generated to accommodate all the formed models and adapt to new ones. The principle and workflow of the proposed MSR-RL-based current control of PMSM drive is illustrated by **Algorithm 6.1** and the block diagram in Fig. 6.4.

A h 6. : MSR-RL

Create parameter sets (context space) *C.*

F *h*=1,2…. n training episode

Initialize policy π_h^0

Initialize Q-function $Q_h^0(s,a)$

Select a task with a context $c \in C$.

Generate initial state $p_h^0(s,a)$ based on π_h^0

Generate discounted long-term rewards $Q_{\pi_h^0}^*(s,a)$

F *k*=1,2… K steps

Execute an action a^k based on the initial state $p_h^0(s,a)$.

and policy π_h^0.

Observe the next state $p_h^k(s,a)$ and reward r^k

Update Q-function $Q_h^k(s,a)$

Update policy π_h^k

Compute the cumulative rewards $r^h = r^k + r^{k+1}$

Return the optimal policy π_h^*.

F

Update discounted long-term rewards $Q_{\pi_h^*}^*(s,a)$

Partition each task info into mini-batches.

Cluster tasks into ϵ-approximated models M_N.

Learn a policy by exploiting formed models M_N

Minimize the cumulative regret R_H.

If the cumulative regret $R_n = \sum_{h=1}^{n} R_h > \epsilon$

Continue training.

Terminate the training.

Return the robust policy π_n^*.

F

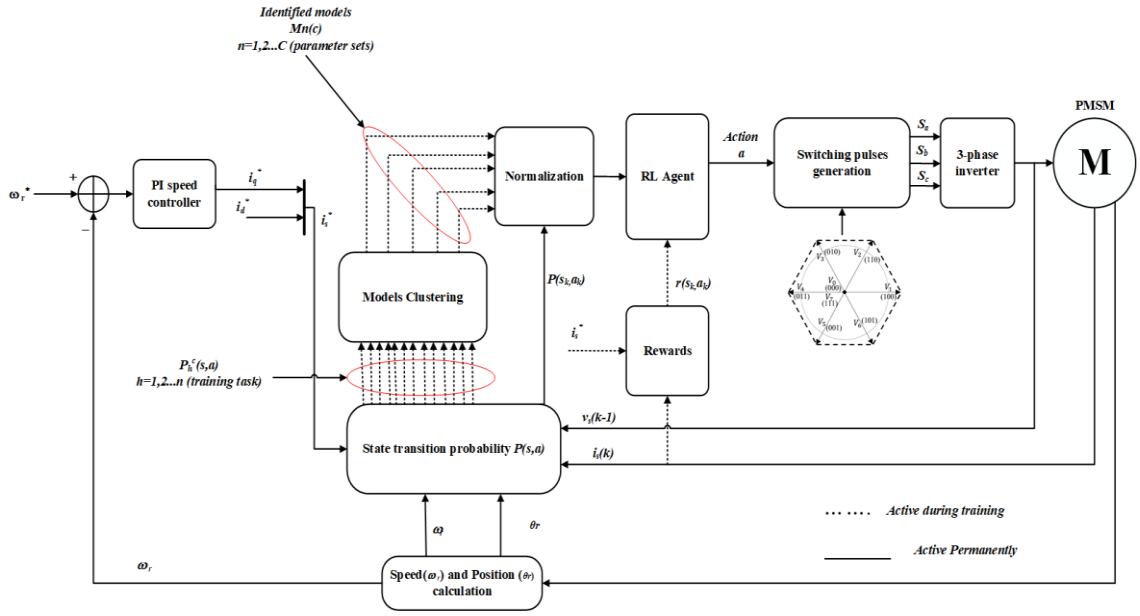


Fig. 6.4. The proposed MSR-RL-based current control of PMSM drives.

6.4 Training and Simulation Analysis

This section outlines the methodology employed to develop, train, and validate the proposed MSR-RL approach for the PMSM drive system using Matlab/Simulink. MSR-RL aims to train the agent with a finite set of parameter contexts, enabling the acquisition of an optimal policy that can be applied to new parameter sets. The training process utilizes the nominal machine parameters specified in Table 3.1 and their respective ranges of potential variations.

Subsequently, the learned policy obtained from the MSR-RL training is tested using different parameter sets, and its performance is evaluated under various operating conditions. A comparative analysis is conducted against standard RL, represented by a single-task DQN-RL agent trained using a single parameter set. This comparison encompasses numerical performance investigations that evaluate the system performance with nominal and mismatched machine parameters across diverse operating conditions.

6.4.1 Uncertainties and Safety Considerations

PMSM drive systems inherently contend with a multitude of uncertainties, encompassing load variations, noise, and sensor errors. These uncertainties present a formidable challenge when applying standard RL algorithms, often leading to difficulties in achieving effective policy adaptation in response to real-world environmental changes. Moreover, the utilization of RL algorithms, primarily trained on simulation data, is constrained by practical system limitations and safety considerations, which may impede their capacity for large-scale data sampling, thereby affecting algorithm efficiency and overall performance.

However, the proposed MSR-RL offers a novel approach to address these challenges. Unlike conventional RL methods that learn policies from single training tasks with specific parameter sets, MSR-RL endeavors to cultivate a single comprehensive policy over multiple training tasks characterized by diverse parameter sets. The outcome of this approach is a policy endowed with robustness, capable of adapting to new parameter sets and exhibiting generalization to a broader spectrum of operating conditions.

The MSR-RL framework is designed to be resilient against uncertainties, specifically calibrated to handle variations in parameter sets and load conditions. In practice, MSR-RL is subjected to extensive training episodes, totaling 10,000 or more, to ensure comprehensive exploration of potential operating conditions within the system. This rigorous training regime, coupled with the application of clustering techniques, empowers MSR-RL to acquire a unified policy that deftly navigates through diverse operating conditions, effectively addressing the practical uncertainties inherent in PMSM drives.

Furthermore, to ensure the proposed MSR-RL can safely operate in real-time application after training using simulation data, various safety measures are considered during the design and training of MSR-RL. These safety measures include:

Safety Constraints: Safety constraints are implemented and incorporated into the reward functions, such as current and voltage limit, to guide the RL agent to learn a safe control policy. These constraints, exemplified by current and voltage limits, play a pivotal role in guiding the RL agent towards the acquisition of safe control policies. Actions that transgress these predefined safety limits are met with substantial penalties, emphasizing the importance of compliance with safety boundaries.

Reward Function Augmentation: The reward function within MSR-RL is augmented to penalize deviations from desired system behavior stemming from uncertainties. This reinforcement encourages the agent to prioritize safety and robustness in its learned policies, thereby enhancing its capacity to navigate uncertain environments.

Safe Exploration Strategies: MSR-RL incorporates safe exploration strategies, such as epsilon-greedy with adaptive exploration rates. These strategies are instrumental in ensuring that the RL agent refrains from exploring actions that could potentially lead to hazardous conditions during the learning process. By implementing these safe exploration methods, MSR-RL further enhance its ability to operate safely in real-time applications.

In summary, the incorporation of safety considerations and the strategic handling of uncertainties within the MSR-RL framework provdes a solid foundation for the practical deployment of this innovative control strategy in PMSM drive systems. Through rigorous training, safety-conscious reward structures, and safe exploration strategies, MSR-RL emerges as a robust and adaptable solution capable of addressing the various challenges posed by real-world applications.

6.4.2 RL Hyperparameters Tuning

RL algorithms come with a set of critical settings called hyperparameters, including the learning rate, discount factor, and exploration strategy. These hyperparameters significantly impact the algorithm's performance, particularly in motor control tasks. However, finding the optimal combination of these settings often involves time-consuming trial-and-error processes, introducing complexity during algorithm design and debugging. To address this challenge, several methods are available to optimize hyperparameters effectively:

1. **Automated Hyperparameter Optimization**

To simplfiy the hyperparameter tuning process and alleviate the burden of manual adjustment, automated hyperparameter optimization tools can be used,including :

Bayesian Optimization: This technique employs probabilistic models to predict which hyperparameter configurations are most likely to lead to improved RL performance. It iteratively evaluates different hyperparameter settings, focusing on promising areas of the configuration space. Bayesian optimization efficiently narrows down the search for optimal hyperparameters.

Grid Search: Grid search systematically explores a predefined range of hyperparameter values, evaluating the performance of the RL algorithm for each combination. While it is more computationally expensive than Bayesian optimization, grid search provides a comprehensive view of the hyperparameter landscape.

2. **Online Hyperparameter Adaptation**

Online hyperparameter adaptation is a dynamic approach that adjusts hyperparameters during RL training based on the agent's real-time performance. This adaptation helps reduce the reliance on manual tuning and enhances algorithm efficiency.

It begins with the initialization of the RL algorithm with initial hyperparameter values. As training unfolds, the algorithm continuously monitors the agent's interactions with the environment, assessing its learning progress and overall performance. Periodic performance evaluations are conducted, employing predefined criteria to gauge the agent's efficiency. Based on these evaluations, the algorithm adjusts hyperparameters on the fly to enhance the learning process. This iterative approach repeats until the RL algorithm achieves its desired performance or meets predefined convergence criteria. By adapting hyperparameters to the agent's real-time experience, Online Hyperparameter Adaptation improves efficiency and robustness, making RL algorithms more adaptable in dynamic environments while reducing the need for manual intervention.

3. Parameter Sweeping Technique

Parameter sweeping is a systematic method that involves varying hyperparameters within predetermined ranges. This technique explores a set of hyperparameter values to assess their impact on RL performance. Parameter sweeping provides valuable insights into the influence of different hyperparameter settings on the algorithm's behavior and allows for the identification of optimal configurations.

In summary, tuning hyperparameters in RL algorithms is crucial for achieving optimal performance in motor control and other applications. To streamline this process and reduce the complexity of manual tuning, automated optimization methods like Bayesian optimization and grid search are employed. Online hyperparameter adaptation further enhances efficiency by adjusting settings based on real-time performance. Additionally, the parameter sweeping technique systematically explores the effects of different hyperparameter values to fine-tune the RL algorithm. These methods collectively contribute to the refinement and effectiveness of RL in practical applications.

6.4.3 Training and Parameter Sets (Contexts)

Through a trial-and-error process, RL training empowers agents to acquire intelligent decision-making capabilities in complex and dynamic environments. The agent explores the environment, learns from its experiences, and gradually enhances its decision-making skills. In general, the training process of the RL agent involves several key steps. Initially, the agent is initialized to set up its initial state. Then, the environment is reset for each episode to prepare for the interaction. The agent observes the environment and computes an action using its current policy. This action is applied to the environment, leading to the next observation and a corresponding reward. The agent learns from this experience, updating its knowledge based on the observed state-action-reward transitions. Subsequently, the agent computes the next action based on the updated policy and repeats the process, iteratively interacting with the environment and refining its decision-making strategy. The training continues until a specific termination condition is met, indicating the completion of the training process. Throughout this process, the agent explores different actions, receives feedback from the environment, and adjusts its policy to optimize its decision-making capabilities.

However, conventional RL methods treat each training task independently without considering potential relations between tasks. In the proposed MSR-RL, a multi-task RL training framework is employed to simultaneously learn related tasks by extracting and utilizing shared information across them. Shared representations can be used effectively by creating models for tasks with similar structures based on the underlying task structure. In MSR-RL, tasks with similar information are clustered together, forming models that accommodate these tasks while exploring unseen tasks. The agent interacts with each task within a cluster, partitioning the information into mini-batches, and the process iterates to generate multiple clustered models. These models are

then exploited, enabling the agent to generate a policy that adapts to the formed models and new, unseen tasks, thereby enhancing overall performance.

MSR-RL-based current control of PMSM drives incorporates finite parameter sets, known as contexts, to enhance training. At the start of each training episode, a task with a random parameter set is selected from the available sets. Tasks with similar parameters or approximately close structures are clustered and learned simultaneously. The parameter sets are generated such that PMSM parameters vary within a specific range of their nominal values. Table 6.1 presents the possible variation ranges of PMSM parameters considering maximum operating conditions.

Table 6.1 PMSM parameter variation ranges

Parameter	Unit	Increase (+)	Decrease (-)
R_s	Ω	80%	-20%
L_d	mH	40%	-70%
L_q	mH	40%	-70%
Ψ_{PM}	Wb	20%	-30%
J	kgm^2	32%	-19%
B	Nm/rad/s	20%	-10%

Using Matlab/Simulink, the proposed MSR-RL with DQN for the current control of the PMSM drive is implemented and trained using 100 parameter sets obtained from Table 6.1, each representing six different parameters. The training process involves a maximum of 10,000 episodes, each with a maximum of 10,000 steps. The termination condition for training is based on the average rewards obtained, while the reference q-axis current determines episode termination. To ensure sufficient training episodes and convergence of the episode reward to the long-term reward, the termination average reward value is set at -50. In each episode, a random parameter set is selected from the 100 sets, and with the help of an additional DNN, episodes with similar parameter sets or

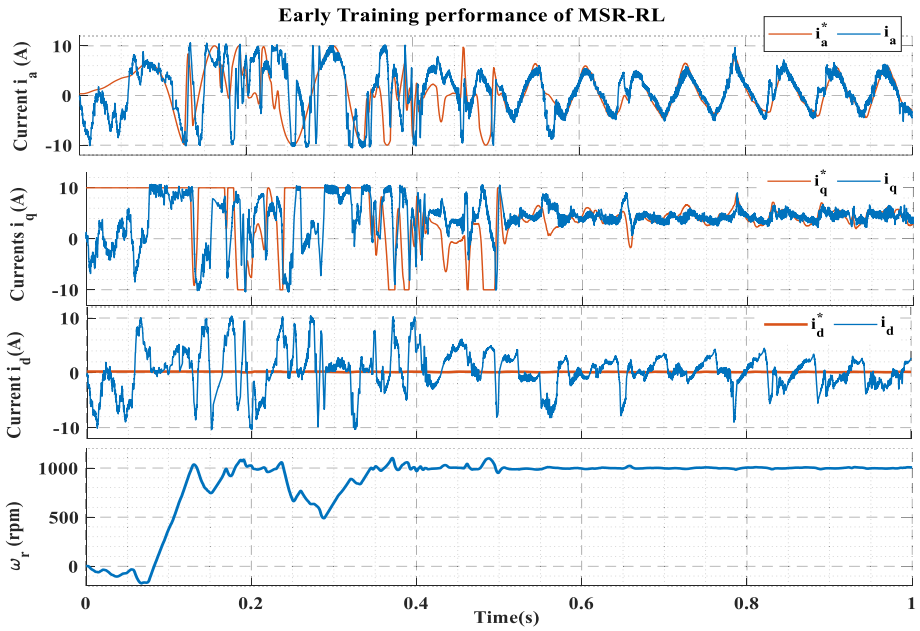
structures are clustered into models. The number of clustered models can be equal to or less than the number of parameter sets.

The training process incorporates various operating conditions, such as different speed operations and load conditions, to ensure robustness. Performance metrics like cumulative, average, and discounted long-term rewards are recorded throughout the training process to monitor training progress. Over the course of 10,000 episodes, N-clustered models are identified and utilized to learn a policy that maximizes cumulative rewards while minimizing accumulated regret. The training statistics presented in Fig. 6.5 indicate that the cumulative reward reached a maximum value of -59, closely approaching the discounted long-term reward with a slight offset due to the discount factor.

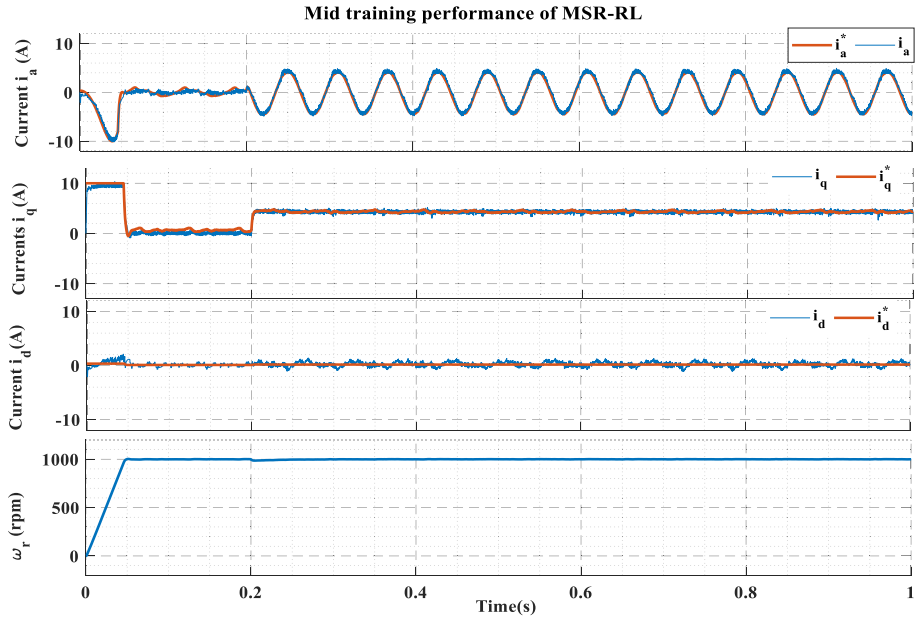
Fig. 6.5. Training stats of the proposed MSR-RL-based current control of PMSM drives.

The RL agent interacts with the environment, gathering knowledge through these interactions. Initially, the agent's actions may lead to suboptimal performance, but as it receives rewards, it learns to avoid actions with low rewards and improves the

performance of the PMSM drive system. The performance of the PMSM drive system is analyzed at different stages of training, specifically during the early training (episode 1) and mid-training (episode 5000), as shown in Fig. 6.6.



(a)



(b)

Fig. 6.6. PMSM drives performance during (a) early training (episode 1) and (b) mid-training (episode 5000).

The agent's progressive improvement in performance over the training episodes can be seen through the training stats in Fig. 6.5. This improvement is achieved by

leveraging the relationships shared among the training episodes to form N-clustered models, each encompassing a set of related tasks. Across the 10,000 training episodes, these models are effectively utilized to learn a unified optimal policy capable of generalizing across all clustered and new, unseen models. This approach enhances the agent's decision-making abilities and enables it to achieve high performance across various task scenarios.

To validate the robustness of the learned policy, it is tested with 1000 different parameter sets, combining sets used in training with new, unseen sets. For each parameter set (iteration), the accumulated and average rewards are computed over 10,000-time steps and compared with the rewards obtained using the optimal policy. The results of these iterations and average rewards, shown in Fig. 6.7, demonstrate the learned policy's ability to maintain rewards that closely align with the final optimal training rewards depicted in Fig. 6.5. This indicates the effectiveness of the learned policy in maintaining performance across diverse parameter sets.

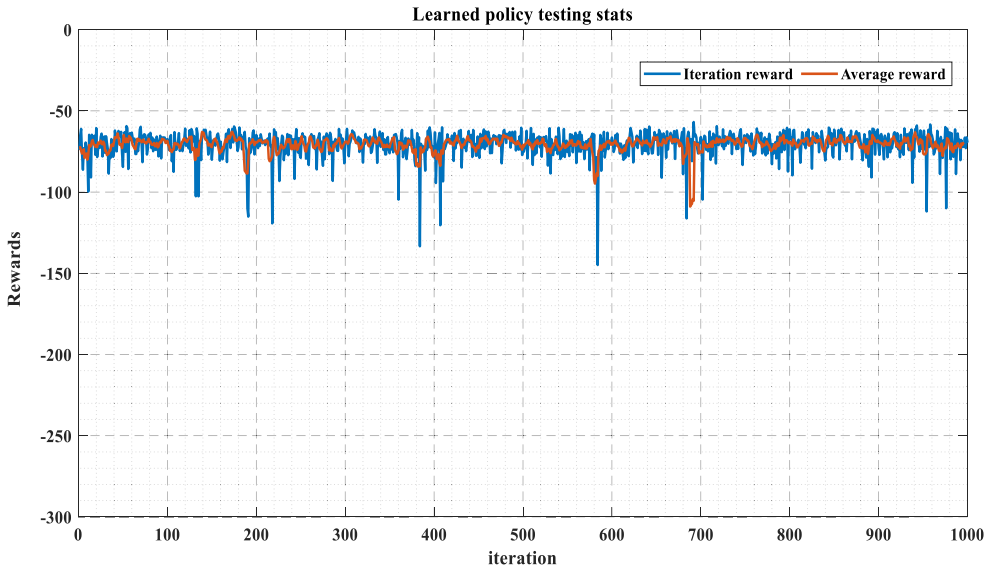


Fig. 6.7. Average and cumulative rewards for testing the learned policy of MSR-RL over 1000 iterations.

6.4.4 Deployment and Numerical Validation

The evaluation of the learned policy demonstrates the agent's impressive ability to adapt to new, unseen tasks and paves the way for its practical deployment. With the knowledge gained through extensive training, the trained MSR-RL agent can be effectively utilized by generating a policy function that can be employed for both simulation and experimental validation purposes. The deployment of the trained agent represents the final stage in the RL workflow, marking the transition from the training setup to the actual utilization of the learned policy. This deployment process involves replacing the RL training framework with a dedicated policy block, which takes observations from the environment and generates corresponding actions. Fig. 6.8 visually represents the learning and deployment stages, illustrating the seamless transition from training to deployment. Notably, the deployed policy operates without the need for explicit rewards. Instead, it solely relies on incoming observations to inform its decision-making process and generate optimal actions.

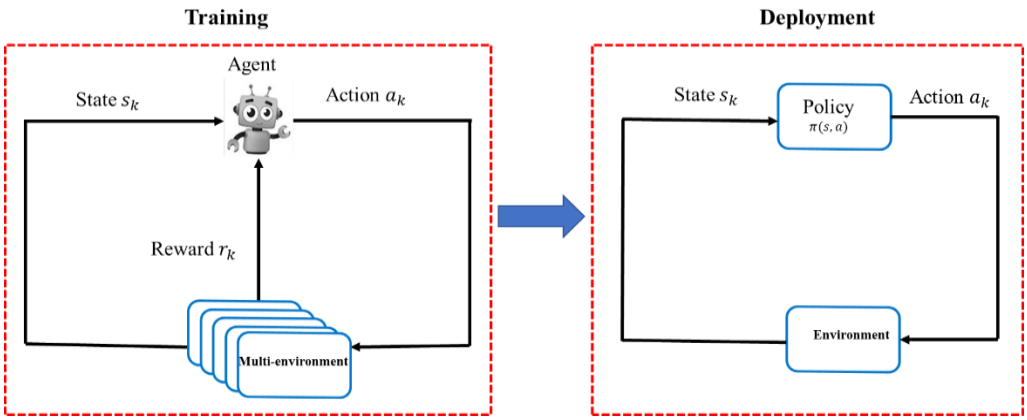


Fig. 6.8. RL agent learning and deployment.

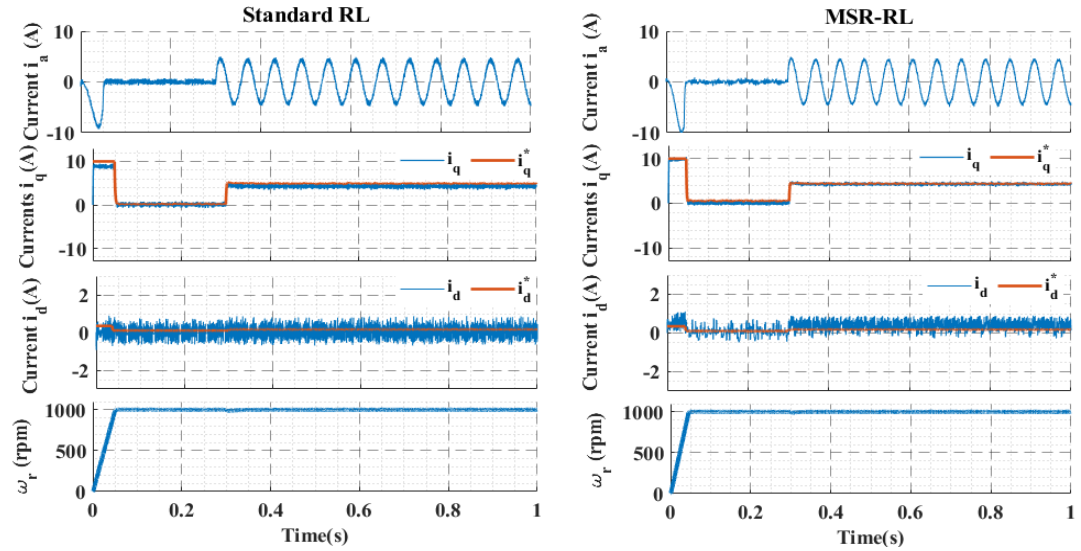
To demonstrate the effectiveness of the proposed MSR-DQN RL approach, a comprehensive performance comparison with the standard DQN RL method is conducted. The standard RL agent is trained using a single parameter set and subsequently

deployed for validation and comparison against the proposed MSR-RL. The performance comparison is made through numerical simulations under various operating conditions, considering nominal and mismatched machine parameters. Firstly, the performances of the MSR-RL and standard RL are evaluated at the rated speed of 1000 rpm and torque of 2 Nm, using both the nominal machine parameters (Table 3.1) and the mismatched parameters, specifically $0.8R_s, 0.3L_d, 0.3L_q, and\ 0.7\Psi_{PM}$. The corresponding results are illustrated in Fig. 6.9. Moreover, the performances at 600 rpm and 200 rpm under a 2 Nm load are also examined to analyze the agent's capabilities comprehensively. These results are presented in Figs. 6.10 and 6.11, respectively. The top-to-bottom curves in these figures represent the phase *a* stator current, *d*- and *q*-axis currents, and motor speed.

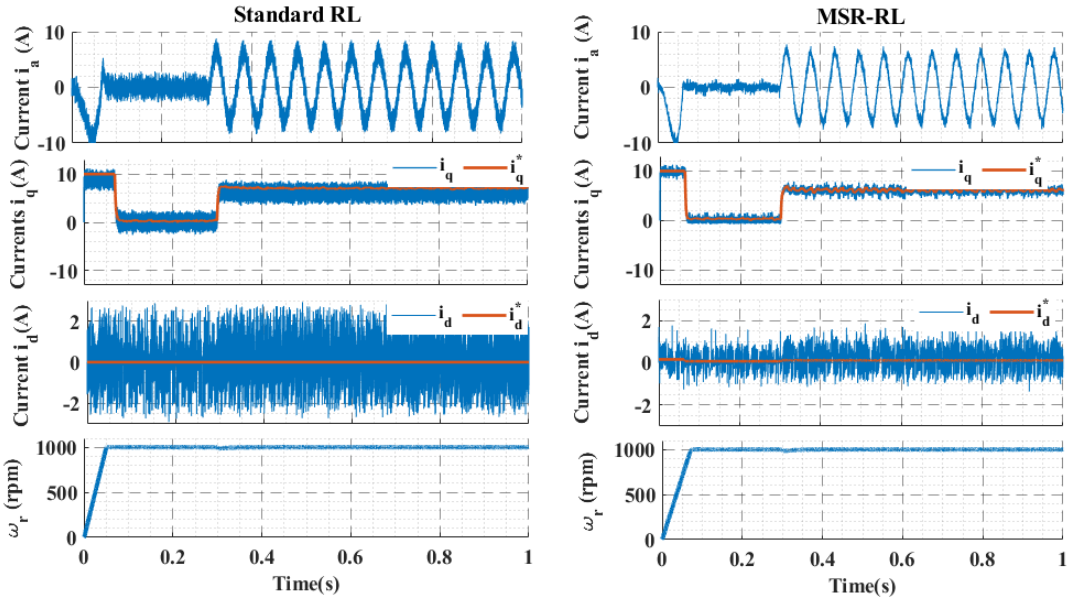
The performance comparison between the proposed MSR-RL and the standard RL across various operating conditions and machine parameters unequivocally demonstrates the superior effectiveness of the proposed MSR-RL. Notably, the standard RL exhibits significant performance degradation, particularly when faced with mismatched machine parameters, while the MSR-RL consistently performs well across diverse operating conditions. Examining the behaviour of stator currents and motor speed reveals that the MSR-RL exhibits rapid dynamic response and exceptional tracking performance, regardless of whether the nominal or mismatched parameters are used.

On the other hand, the standard RL delivers acceptable performance with the nominal parameters but experiences high current distortion and poor tracking when subjected to mismatched parameters. Unlike the proposed MSR-RL, which sustains an excellent performance of the drive system due to its robust policy optimized through various parameter sets.

Moreover, the performances of the proposed MSR-RL and the standard RL are numerically compared based on the total harmonic distortion (THD) spectrum of the phase *a* current under nominal and mismatched parameters. The current signals are captured during the steady state of 1000 rpm with a torque of 2 Nm applied. The captured current signals are subsequently analyzed to extract the THD spectrum, which enables the computation of the THD up to a frequency of 5 kHz, encompassing a broad range of harmonics.

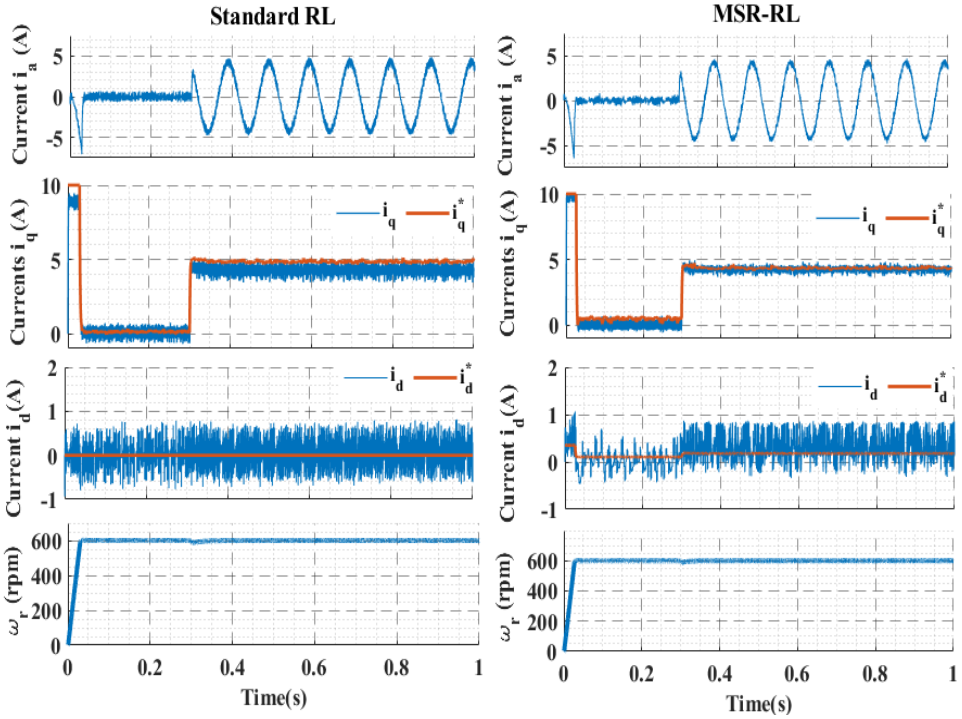


(a)

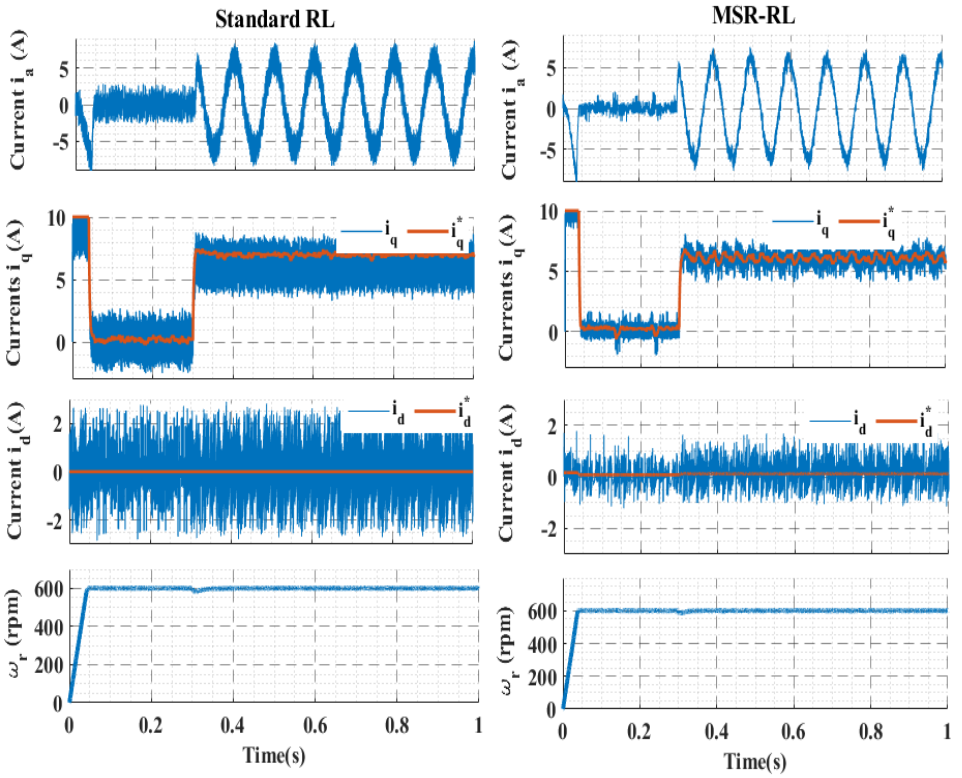


(b)

Fig. 6.9. Performance comparison of proposed MSR-RL and standard RL at 1000 rpm and 2 Nm load torque, (a) nominal and (b) mismatched parameters.

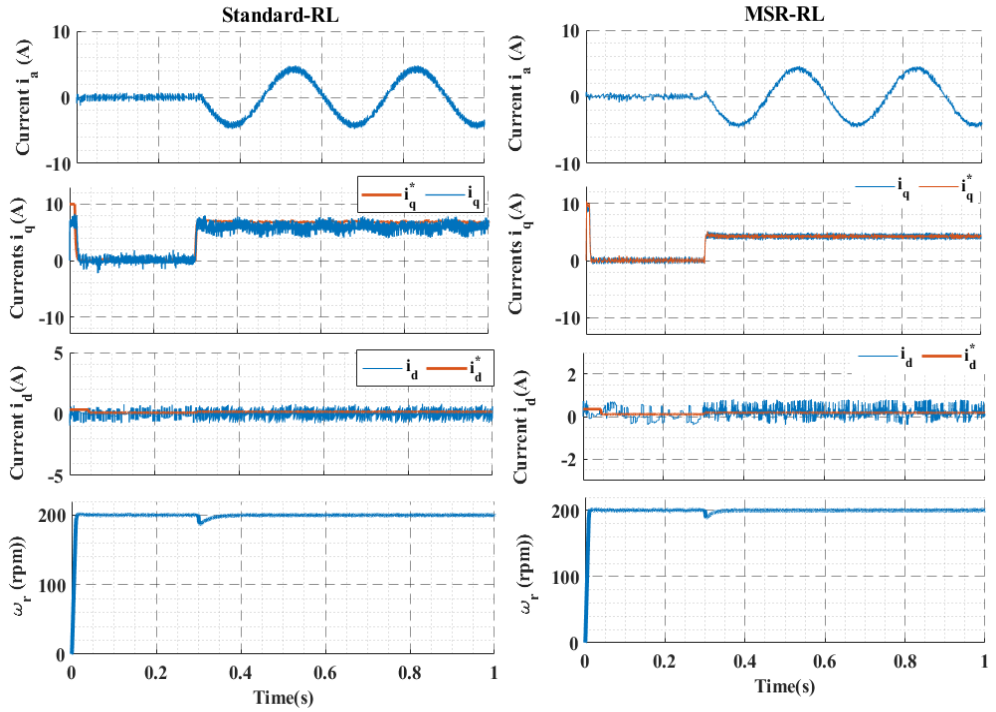


(a)

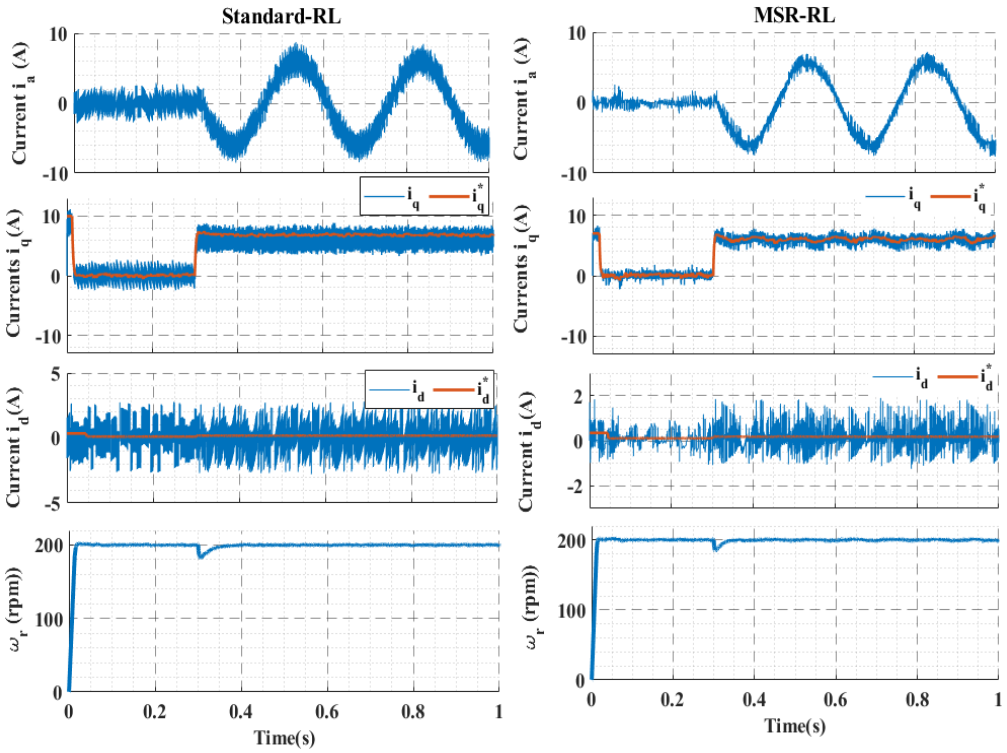


(b)

Fig. 6.10. Performance comparison of proposed MSR-RL and standard RL at 600 rpm and 2 Nm load torque, (a) nominal and (b) mismatched parameters.



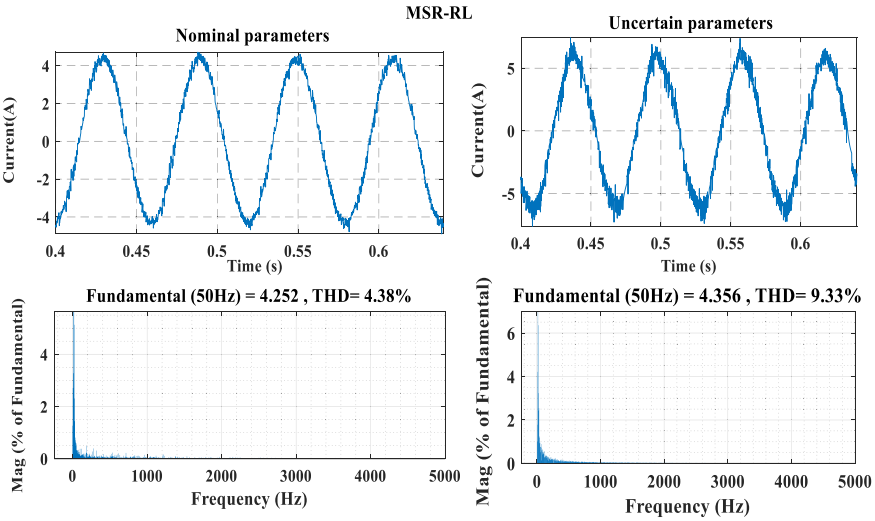
(a)



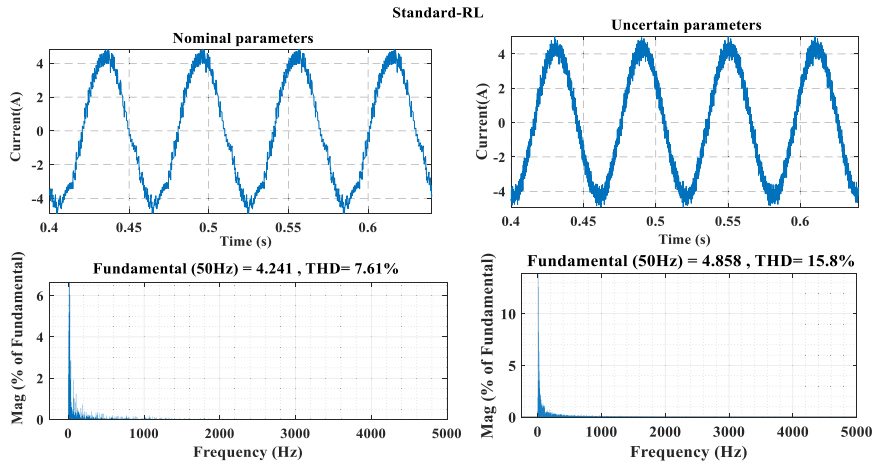
(b)

Fig. 6.11. Performance comparison of proposed MSR-RL and standard RL at 200 rpm and 2 Nm load torque, (a) nominal and (b) mismatched parameters.

Fig. 6.12 visually presents the outcomes of this comparison, illustrating the THD spectra for the proposed MSR-RL and the standard RL under nominal and mismatched parameters. Remarkably, the THD achieved by the MSR-RL is substantially lower than that of the standard RL, irrespective of whether nominal or mismatched parameters are considered. Specifically, the MSR-RL records THD values of 4.38% and 9.33% compared to 7.61% and 15.6% for the standard RL under nominal and mismatched parameters, respectively.



(a)



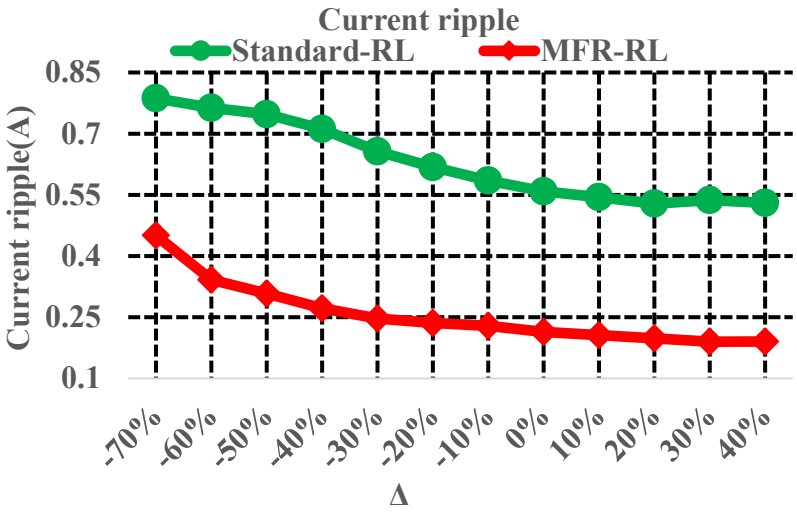
(b)

Fig. 6.12. The proposed MSR-RL and standard RL THD spectrum, (a) nominal and (b) mismatched parameters.

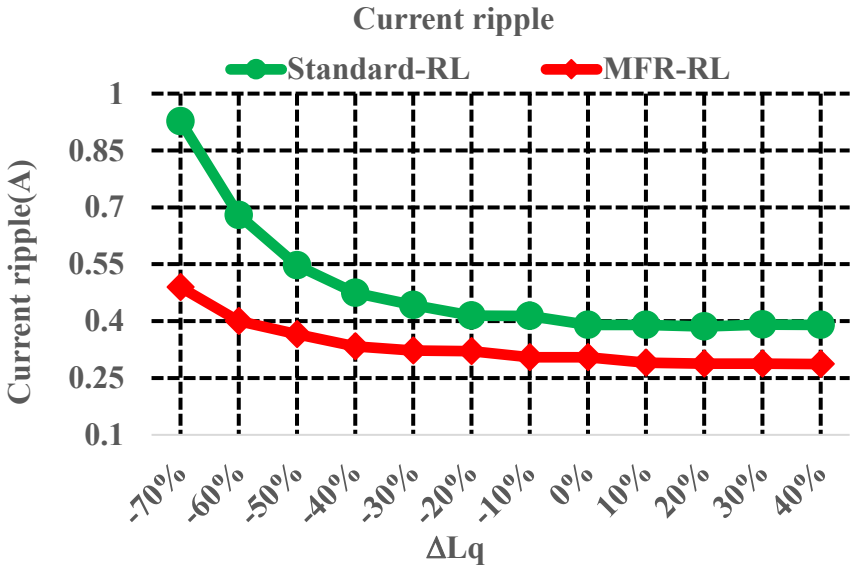
This comprehensive assessment, encompassing motor performance and harmonic analysis, facilitates a comprehensive understanding of the effectiveness and superiority of the proposed MSR-RL approach in controlling PMSM drives, particularly in the presence of uncertainties. The results obtained from these evaluations provide compelling evidence of the MSR-RL's potential to adapt to new environments, maintain exceptional tracking performance, and effectively mitigate undesired harmonic components in the current waveform.

To further validate the robustness of the proposed MSR-RL against parameter mismatching, a quantitative analysis is conducted to evaluate the tracking performance of MSR-RL and standard RL against parameter variations. The proposed MSR-RL and standard RL tracking capabilities are evaluated by computing the current ripples with the variations of L_d and L_q, as shown in Fig. 6.13. Different values of L_d and L_q are generated within a bounded range according to Table 6.1, and the current ripples are computed.

The quantitative analyses demonstrate MSR-RL's superior performance compared to the standard RL, mainly with variations in L_d and L_q. The standard RL, trained with a single parameter set, exhibited substantial performance degradation with mismatched parameters. In contrast, MSR-RL shows consistent and robust performance with mismatched parameters. This difference highlights the adaptability and generalizability of the learned policy within the MSR-RL framework, enabling it to handle diverse environmental conditions effectively.



(a)



(b)

Fig. 6.13. Current ripples of MSR-RL and standard RL with the variation of (a) L_d, (b) L_q.

6.5 Experimental Validation

To validate the workability and assess the effectiveness of MSR-RL, experimental tests are conducted on a PMSM drive system utilizing a two-level inverter and a dSPACE DS1104 PPC/DSP controller, as shown in Fig. 6.14. The experimental setup can be divided into two main components: the software part, which encompasses

Matlab/Simulink and dSPACE ControlDesk, and the hardware part, consisting of the motor, inverter, and DC supply. The software and hardware components are seamlessly integrated through the DS1104 controller. Various measurements, including motor speed, position, and currents, are acquired and fed back to the software for the inverter's processing and generating suitable switching pulses.

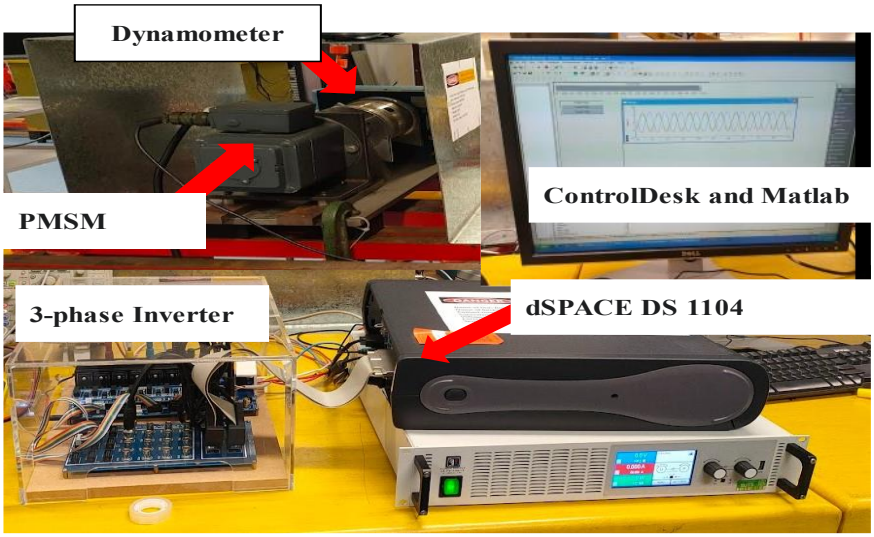


Fig. 6.14. Experimental setup of PMSM drive system.

While the proposed MSR-RL and standard RL have been rigorously tested and validated through numerical simulations across diverse operating conditions, verifying their performance in real-time experiments under different scenarios is crucial. This validation is carried out by subjecting the drive system to start-up, load disturbance, and steady-state tests. In the start-up tests, MSR-RL and standard RL are evaluated by initiating the motor from a standstill and gradually accelerating it to the rated speed of 1000 rpm. The corresponding measurements, including stator current (i_a), dq-currents, and motor speed are recorded and presented in Fig. 6.15. On the other hand, load disturbance tests involve operating the motor at a steady-state speed of 1000 rpm and then applying a 2 Nm load torque to the motor shaft. The resulting motor currents and speed responses are captured and illustrated in Fig. 6.16. In both Figs. 6.15 and 6.16, the plotted

curves, from top to bottom, represent the stator current (i_a), dq-axis measured and reference current (i_{dq}, i_{dq}^*), and measured rotor speed (ω_r).

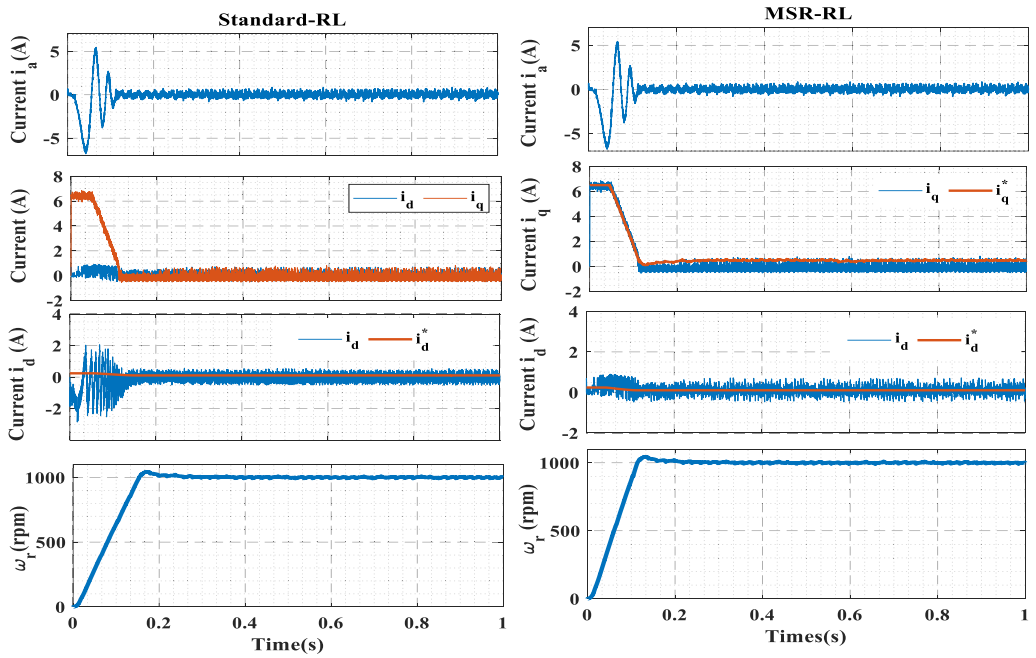


Fig. 6.15. Experimental start-up test of MSR-RL and standard RL from standstill to rated speed (1000 rpm).

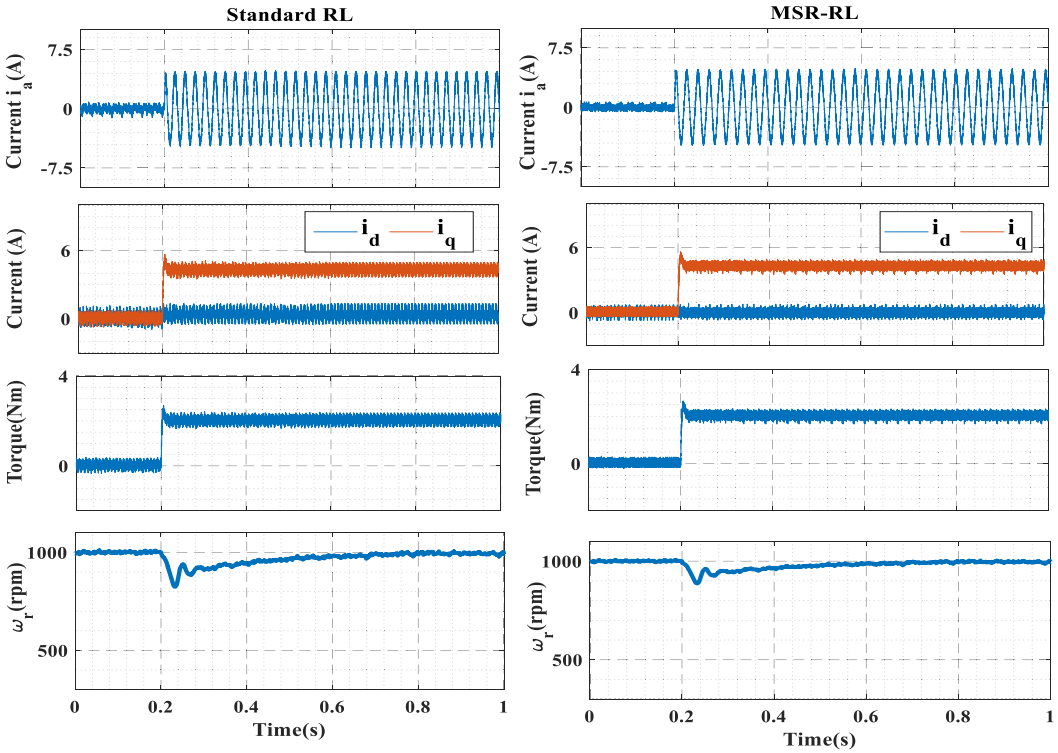


Fig. 6.16. Load disturbance test (2 Nm) of MSR-RL and standard RL at rated speed (1000 rpm).

The proposed MSR-RL demonstrates a superior dynamic response compared to the standard RL approach. In the start-up phase, where the motor initiates from a standstill and accelerates to its rated speed, the MSR-RL rapidly accelerates with a slight overshoot, swiftly reaching the steady-state speed of 1000 rpm. Conversely, the standard RL exhibits a comparatively slower acceleration profile. Moreover, the proposed MSR-RL exhibits robust load disturbance rejection capabilities. When a load disturbance of 2 Nm is applied to the motor shaft at 0.25s, the MSR-RL experiences a modest speed drop of 108 rpm, promptly recovering its steady-state condition within 0.35 s. In contrast, the standard RL encounters a more pronounced speed drop of 171.8 rpm and takes 0.45 s to restore its steady-state condition. Additionally, the steady-state current responses of the MSR-RL exhibit smoother profiles with reduced ripple compared to the standard RL.

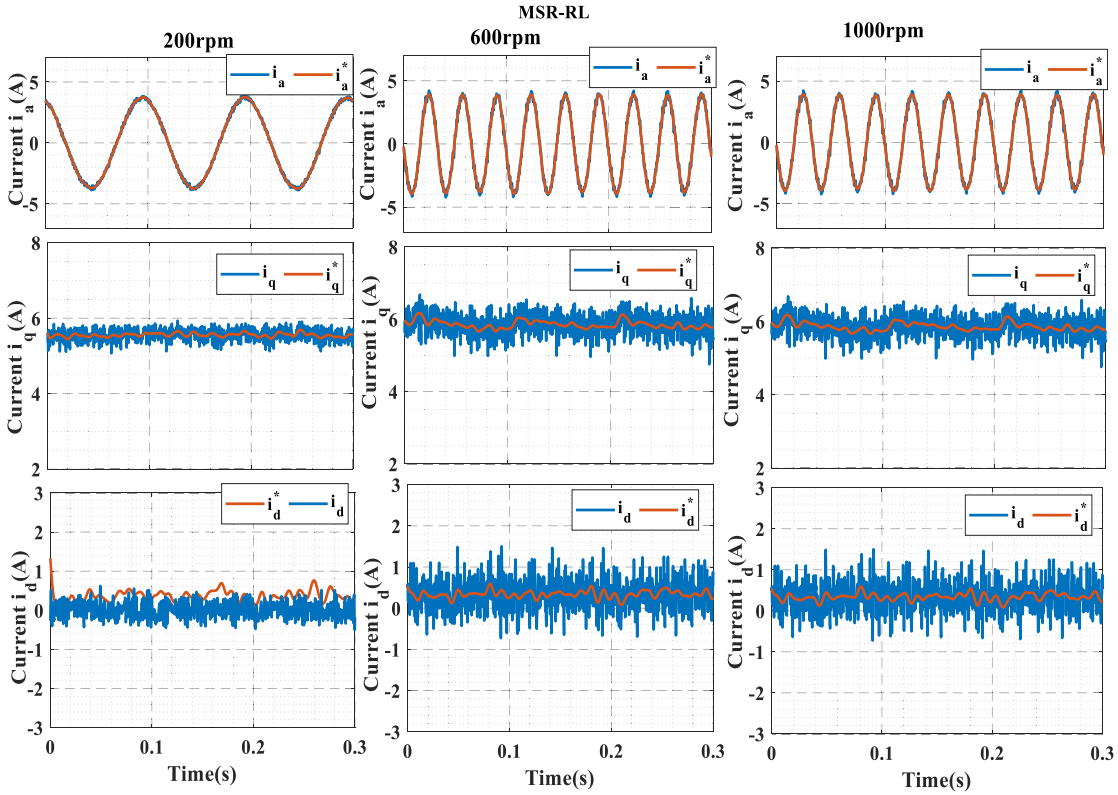
Both the MSR-RL and standard RL models are trained under diverse load conditions. However, it is essential to note that the actual parameters of a real-world PMSM may deviate from the nominal parameters employed during the training process of the standard RL. Consequently, the improved performance of the MSR-RL can be attributed to its training using multiple parameter sets, adapting more effectively to uncertainties inherent in real-world PMSM.

Further experimental investigations are conducted to evaluate the performance of the proposed MSR-RL and the standard RL under steady-state conditions. These investigations focus on observing the motor's measured currents, including i_a, i_q, and i_d, along with their corresponding reference signals, $i_a^*, i_q^*,$ and i_d^*. The steady-state current responses of both the MSR-RL and the standard RL are captured during three different speed operations, namely 200 rpm, 600 rpm, and 1000 rpm. A load of 2 Nm is applied to the motor shaft during these operations.

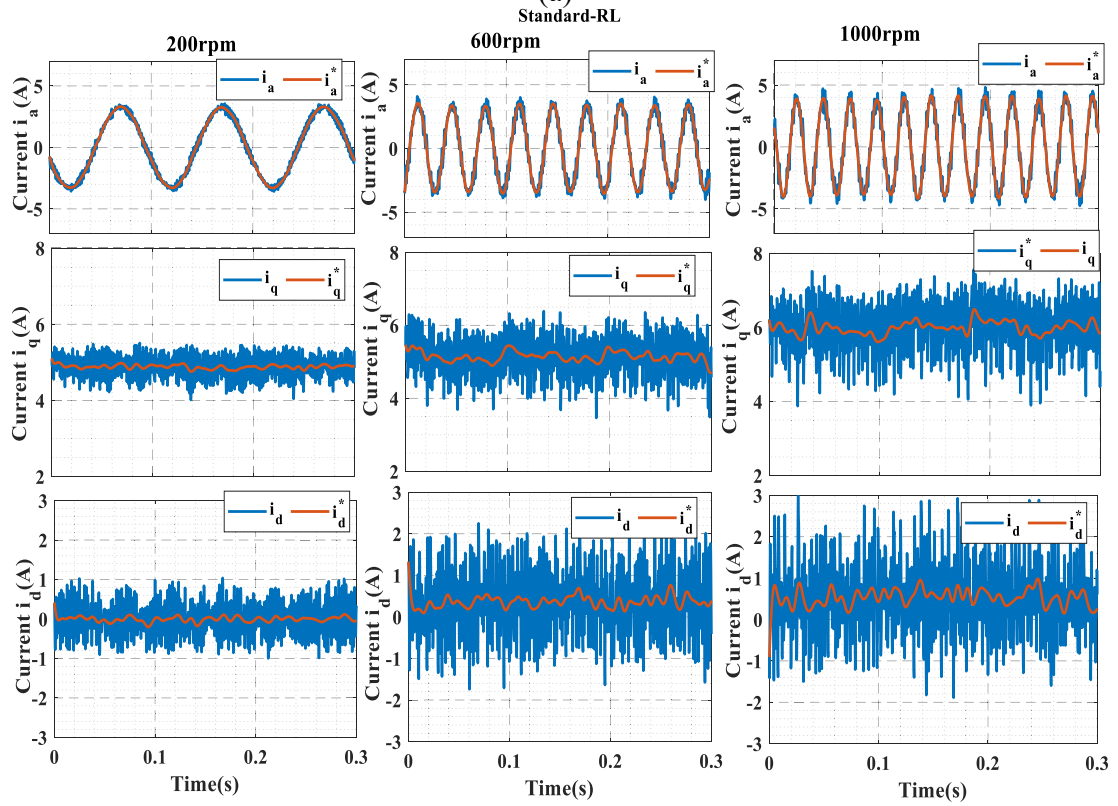
Figs. 6.17 (a) and (b) depict the captured steady-state current responses of the MSR-RL and the standard RL, respectively. The results demonstrate the exceptional tracking performance of the proposed MSR-RL, surpassing that of the standard RL across low, medium, and high-speed operations. Notably, the steady-state current ripples of the MSR-RL are significantly reduced compared to those of the standard RL. This reduction in ripples highlights the robustness of the learned policy of the MSR-RL in adapting to different operating conditions, further affirming its superior performance.

In addition to speed variations, the performance of the MSR-RL is being assessed under varying load conditions, as illustrated in Fig. 6.18. The motor is operated at a steady state of 1000 rpm, and the responses of the estimated torque, stator current, and *dq*-axis currents are observed under three load conditions: no-load, half-load (1 Nm), and full-load (2 Nm). The results showcase the ability of the MSR-RL to effectively handle different load conditions, maintaining stable and accurate torque estimation and consistent stator and *dq*-axis current responses.

Quantitative analyses are conducted at various operating conditions to further analyze the performance numerically since real-time access to machine parameters is unavailable. The evaluation focuses on computing the steady-state stator current ripples, considering different speed and load conditions. Eight load torques are applied at a steady state of 1000 rpm, ranging from minimal to maximum values. Five different speeds are examined while maintaining a fixed load torque of 2 Nm. The current ripples are calculated for each operating condition, yielding valuable insights into the controller's performance, as presented in Fig. 6.19.



(a)



(b)

Fig. 6.17. Steady-state currents at different speeds (200, 600, 1000) rpm, (a) MSR-RL, and (b) standard RL.

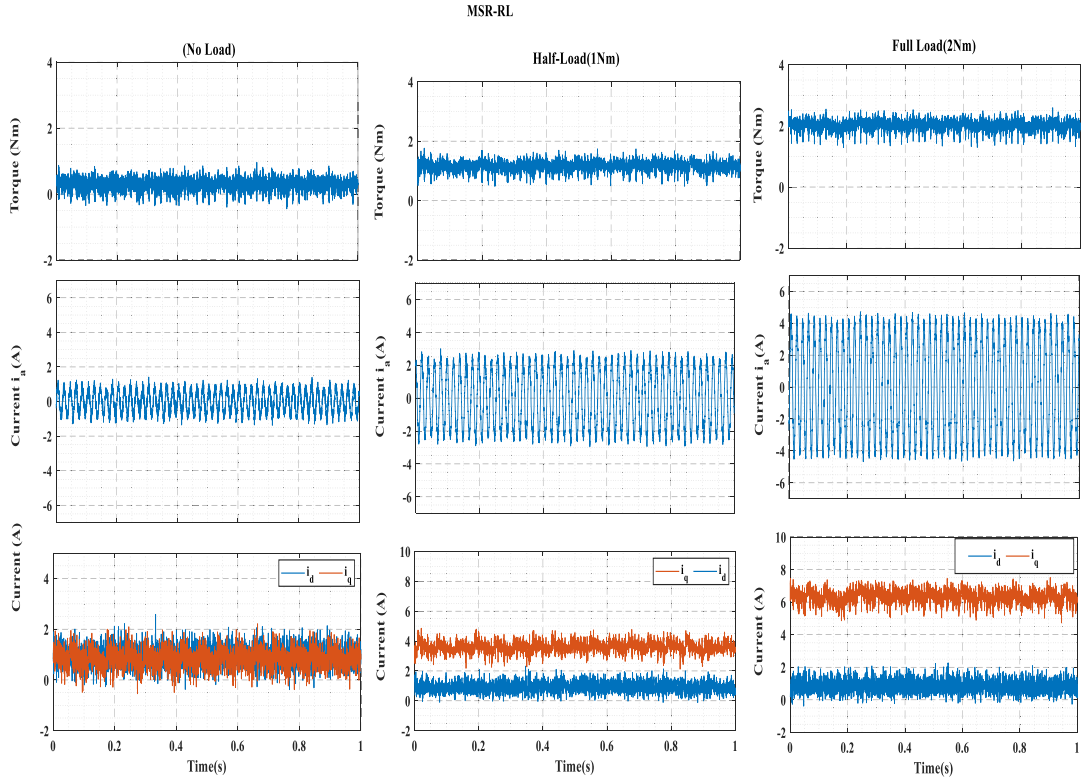
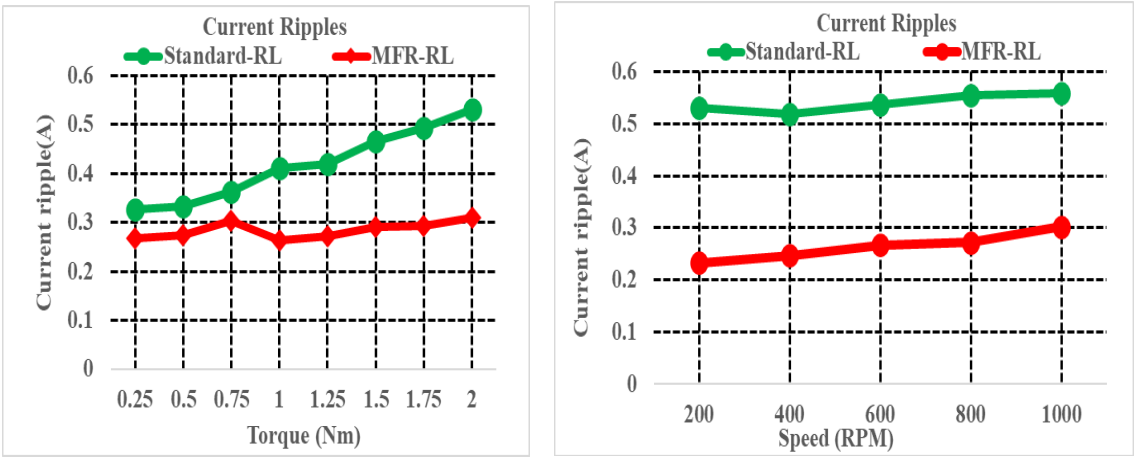


Fig. 6.18. Performance of proposed MSR-RL at steady-state of 1000 rpm with no-load, half-load, and full-load.



(a) (b)

Fig. 6.19. Current ripple comparison of MSR-RL and standard RL with (a) different load torques at rated speed (1000 rpm) and (b) various speeds under rated torque (2 Nm).

The experimental testing conducted across different operating conditions provides compelling evidence supporting the superiority of the proposed MSR-RL compared to the standard RL. The MSR-RL exhibits faster transient responses and less distortion in the steady-state compared to the standard RL, as validated through the start-up, load disturbance, and steady-state tests. These findings highlight the optimality of the learned policy within the MSR-RL framework, enabling robust adaptation to new environments. Moreover, the quantitative analysis considering variations in speed and load further reinforces the superiority of the MSR-RL over the standard RL. The MSR-RL consistently outperforms the standard RL regarding its ability to mitigate steady-state current ripples, indicating its superior control performance and enhanced robustness across different operating conditions.

6.6 Robustness Evaluation

The robustness of the proposed MFR-RL has been demonstrated through simulation and experimental evaluations, which have shown good performance under specific operating conditions and parameter sets. However, it is essential to note that these evaluations do not comprehensively assess the controller's robustness, as they have not considered various other conditions and parameter sets. Additionally, these evaluations do not provide information about the robustness index or how robust the proposed controller compares to other controllers. Therefore, this research employs a six-sigma design robustness evaluation presented in Chapter 4 to thoroughly evaluate the proposed MSR-RL and standard RL. This approach uses large parameter sets and encompasses multiple performance indicators to ensure a comprehensive assessment.

The performance indicators used in the evaluation include both transient and steady-state response characteristics, such as settling time (T_s), overshoot (OS), root mean square error of speed ($RMSE_\omega$), torque ripples (T_{rip}), and current ripples (i_{rip}). Each indicator is crucial in designing a PMSM drive for a specific application. Depending on the application requirements, each performance indicator (K_i) has a corresponding upper USL. For example, a racing car would necessitate a faster settling time (e.g., $T_s \leq 0.1s$), whereas a passenger car might allow for a slower settling time (e.g., $T_s \leq 0.3s$). Therefore, the *USL* for each performance indicator (K_i) varies based on the intended application.

The evaluation considers low, medium, and high-performance applications to cater to a wide range of performance requirements. Low-performance applications like water pumping systems have relatively lenient specification limits. Medium-performance applications, such as electric vehicles, have moderate limits. On the other hand, high-performance applications, such as radar systems and CNC machines, demand stringent and precise limits. Table 6.2 presents the performance indicators and their respective USLs for each application category (low, medium, and high performance).

Table 6.2 Performance indicators and their respective upper specification limits

Indicator (K_i)	Upper Specification limits (USL)		
	Application-I	Application-II	Application-III
Ts	≤ 0.2	≤ 0.15	≤ 0.1
OS	$\leq 5\%$	$\leq 3\%$	$\leq 2\%$
$RMSE_\omega$	≤ 0.03	≤ 0.02	≤ 0.01
T_{rip}	≤ 0.8	≤ 0.6	≤ 0.4
i_{rip}	≤ 0.9	≤ 0.7	≤ 0.5

The evaluation process involves generating 10,000 normally distributed samples of machine parameter variation with a specific range (Table 6.1). The proposed MSR-RL and standard RL are then simulated, and the resulting performance indicators are computed for each sample. Specifically, the motor runs from a standstill to the rated speed of 1000 rpm, with a steady-state load torque of 2 Nm. The resulting 10,000 data points of each K_i are used to calculate the mean μ_i and standard deviation σ_i. The Z-value Z_i of each performance indicator, the system sigma level n_{sys} and system probability of failure POF_{sys} are computed as discussed in Chapter 4.

The Z-values, system sigma levels and POF are calculated for MSR-RL and standard RL based on a dataset of 10,000 samples, as shown in Table 6.3. MSR-RL demonstrates strong robustness compared to standard RL across all performance indicators for the three application requirements. In the case of low requirements (Application-I), standard RL achieves a sigma level of 2.7σ and a POF of 0.73%. However, for medium and high requirements (Applications-II and III), standard RL only achieves sigma levels of 1.9σ (POF of 5.51%) and 1.2σ (POF of 21.51%), respectively. In contrast, MSR-RL achieves a 6σ sigma level for Application-I, while sigma levels of 3.9σ(POF of 0.01%) and 3.2σ (POF of 0.16%) are achieved for Applications-II and III requirements, respectively.

The Z-values of the performance indicators provide insights into the strengths and weaknesses of each control method, as well as their ability to strike a balance among all indicators. For instance, the Z-values of overshoot and root mean square error (RMSE) for standard RL adequately meet the requirements of all three applications. However, the Z-values of torque and current ripples are considerably low, particularly for medium and low-requirement applications. As a result, standard RL yields lower sigma levels and

higher POFs. On the other hand, the proposed MSR-RL maintains a trade-off across all indicators, leading to higher sigma levels for all application requirements.

Table 6.3 Robustness evaluation results of MSR-RL and standard RL.

Indicator	Z_{T_s}	Z_{OS}	Z_{ω}	$Z_{T_{ripp}}$	$Z_{i_{ripp}}$	n_{sys}	POF
Controller	(Application-I)						
S-RL	20.1	44.7	53.0	7.6	3.6	2.7	0.73%
MSR-RL	33.4	61.9	146.5	23.7	17.9	6.0	0
	(Application-II)						
S-RL	13.5	25.9	33.9	5.2	1.9	1.9	5.51%
MSR-RL	22.5	36.2	94.0	17.5	12.3	3.9	0.01%
	(Application-III)						
S-RL	6.7	16.5	21.2	2.8	0.3	1.2	21.51%
MSR-RL	11.7	23.4	59.0	11.3	6.6	3.2	0.16%

It is worth noting that sigma levels below 6σ do not necessarily render a control system unacceptable. In the industry, a sigma level of 3σ is considered acceptable, and 6σ is introduced to account for the long-term shift (approximately 1.5σ) in the mean. In some cases, achieving 6σ is not feasible for specific control methods or application requirements. Consequently, controllers with sigma levels below 6σ can still be reliable. For instance, a sigma level of 3.2σ of MSR-RL for Application-III implies that only 16 parameter combinations out of the total samples (1000) result in any performance indicator exceeding the specification limits of Application-III.

The Z-value is a measure of the distance between USL and the mean (μ_i) of a performance indicator's data. In other words, it is a measure of the number of standard deviations (σ) between USL and the mean (μ_i). This can be illustrated by the capability plots of torque and current ripple performance indicators presented in Figs. 6.20 and 6.21 for standard RL and MSR-RL. The capability plots provide insights into the data

distribution of torque and current ripple performance indicators and their proximity to the application's specification limits (USL-I, USL-II, USL-III). The torque ripple capability plot shows that the standard RL data are widely dispersed around the mean ($\mu = 0.1643$), exceeding Application-III requirements (USL-III=0.4) and a Z-value of 2.8. In contrast, MSR-RL data are narrowly around the mean ($\mu = 0.0334$) with no values exceeding any of the three applications USLs (USL-I, USL-II, USL-III).

Furthermore, the capability plots of current ripples of standard RL and MSR-RL show that the standard RL data has exceeded the USLs of all applications, resulting in low Z-values. Particularly, 2151 samples (out of 10000) of standard RL current ripple data are recorded beyond USL-III. In contrast, MSR-RL current ripples capability plot shows a few data exceeding the USLs of Applications-II and III. Thus, MSR-RL achieves high Z-values of current ripples for three application requirements.

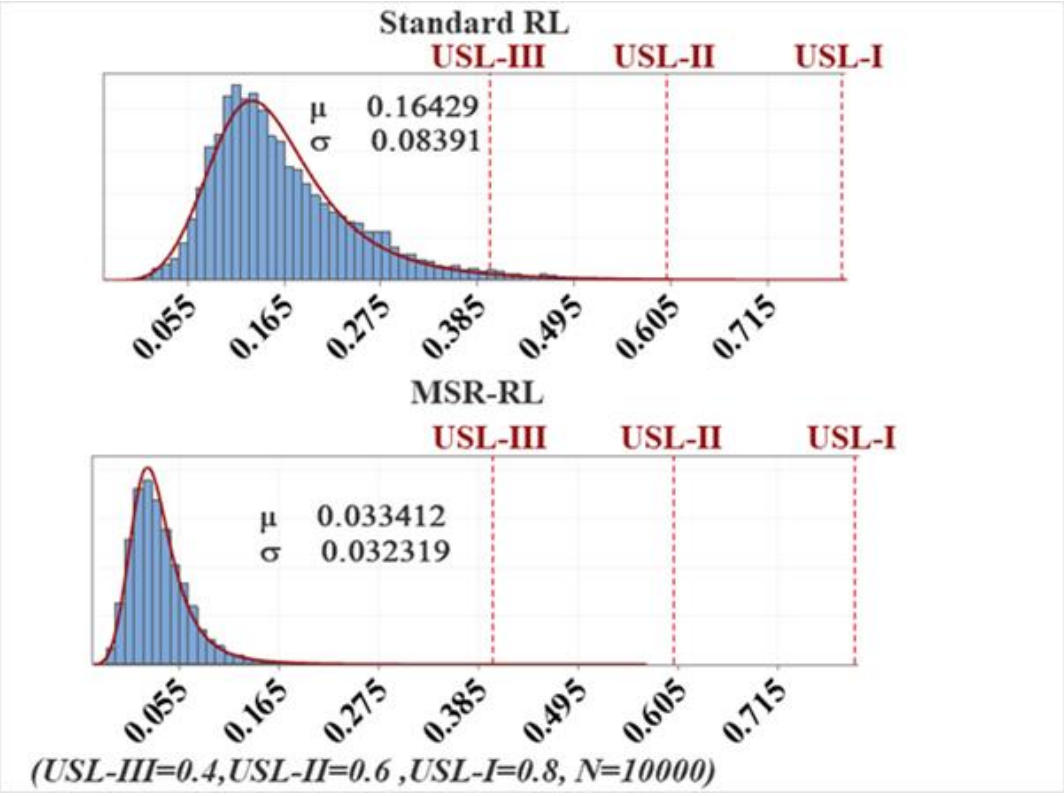


Fig. 6.20. Capability plot of torque ripples for standard RL and proposed MSR-RL.

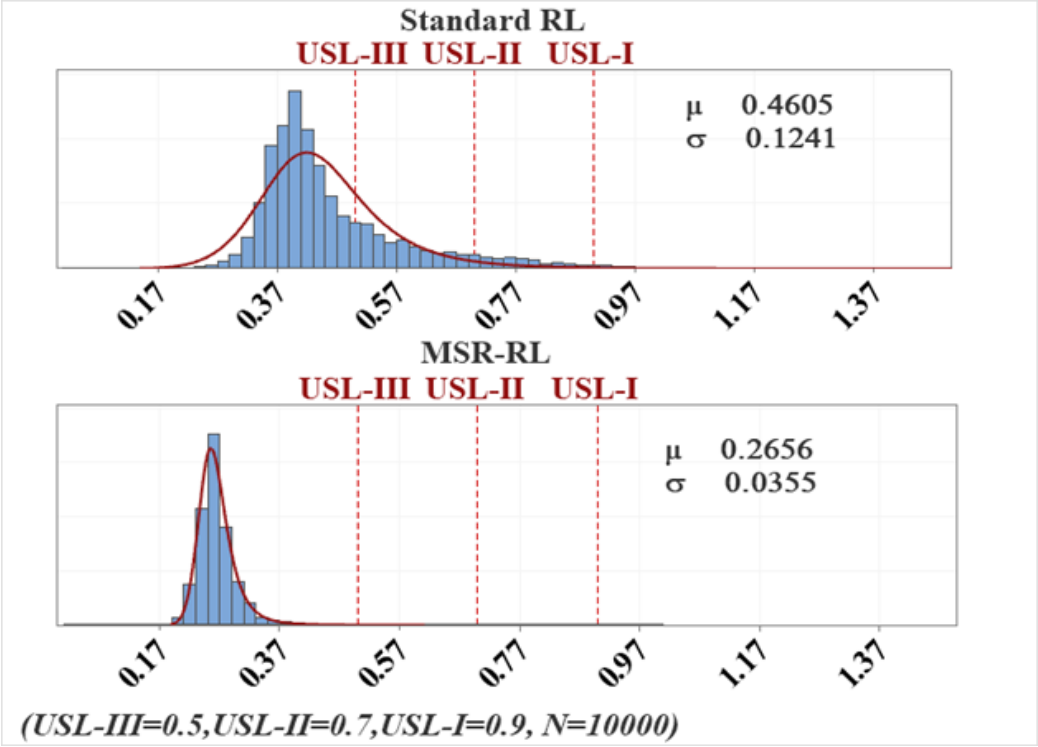


Fig. 6.21. Capability plot of current ripples for standard RL and proposed MSR-RL.

6.7 Comparison of MSR-RL and A-MFPCC

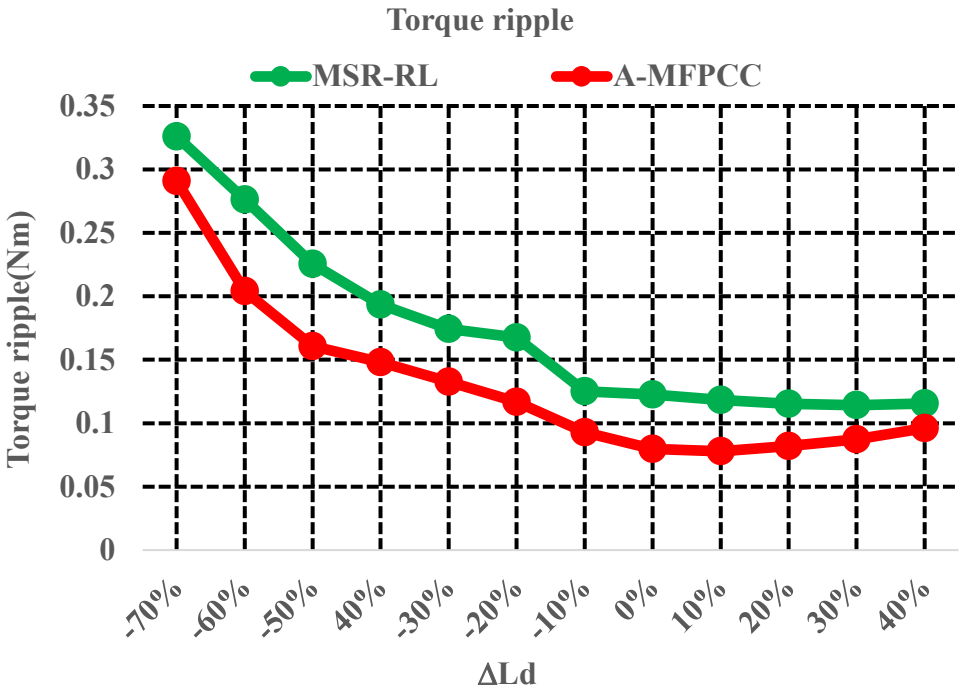
Data-driven MSR-RL based current control of PMSM drives is achieved based on a pre-trained policy, resulting in less computational online control. In contrast, A-MFPCC (Chapter 5) controls the PMSM drives through online optimization, requiring additional computational capacity compared to MSR-RL. Therefore, it is essential to conduct a performance comparison between MSR-RL and A-MFPCC based on PMSM drives to see if a computationally efficient MSR-RL can achieve good performance as A-MFPCC at different operating conditions. MSR-RL and A-MFPCC were evaluated at different operating conditions and showed excellent performance compared to their corresponding conventional methods. MSR-RL and A-MFPCC were optimized to achieve the best performance possible; thus, it can be challenging to differentiate their performance difference based on graphical results. This section presents a quantitative analysis and robustness evaluation of MSR-RL and A-MFPCC.

6.7.1 Quantitative Analysis

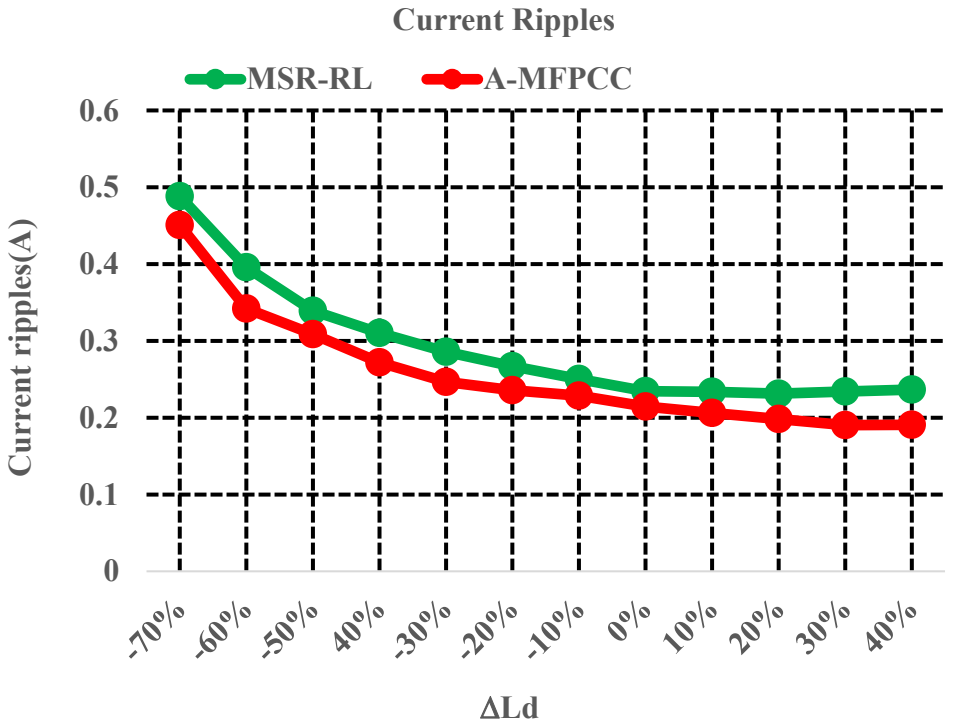
Torque and current ripples are essential performance indicators to judge the performance of a PMSM control method. MSR-RL and A-MFPCC are quantitatively compared against parameter mismatching and changes in operating conditions. Higher performance effects are experienced with the variations of machine inductances (L_d, L_q); thus, both controllers are simulated at different values of L_d and L_q, according to Table 6.2. The corresponding numerical data of torque and current ripples with the variation of L_d and L_q are plotted in Figs. 6.22 and 6.23, respectively.

Furthermore, MSR-RL and A-MFPCC are quantitatively evaluated at different speeds and load conditions. Experimental tests are conducted for both controllers based on PMSM drives at five speeds (200, 400, 600, 800, and 1000 rpm) and under load conditions between 0.25 Nm and 2 Nm. The current ripples are computed for each operating condition, and the resulting quantitative data of MSR-RL and A-MFPCC are presented in Fig. 6.24.

The quantitative analysis against parameter variations and changes in operating conditions has shown that the proposed MSR-RL based current control of PMSM drive can achieve comparable performance to A-MFPCC. Notably, an offset difference in the performance, with A-MFPCC achieving better performance. This suggests that MSR-RL was not trained in all possible conditions or that the parameter sets used for training were not large enough to cover the whole parameter variations range. Furthermore, MSR-RL may have reduced the computational cost compared to A-MFPCC. However, with the availability of a high computation capacity processor (i.e., dSPACE), even a high computational controller can be implemented with a smaller sampling size.

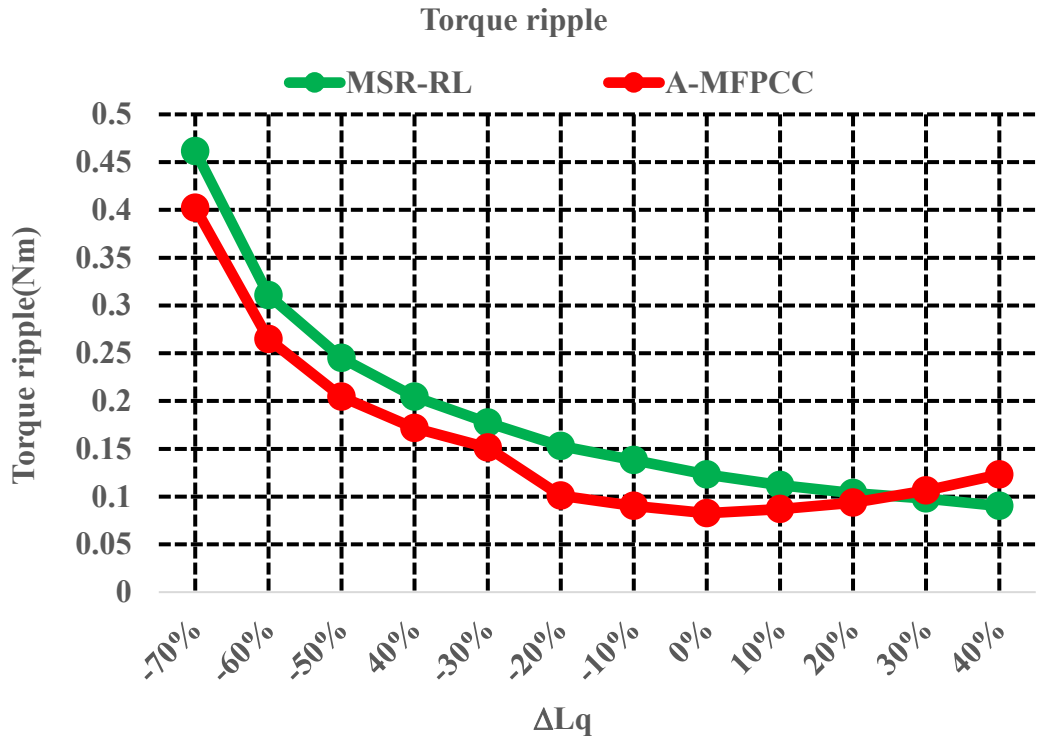


(a)

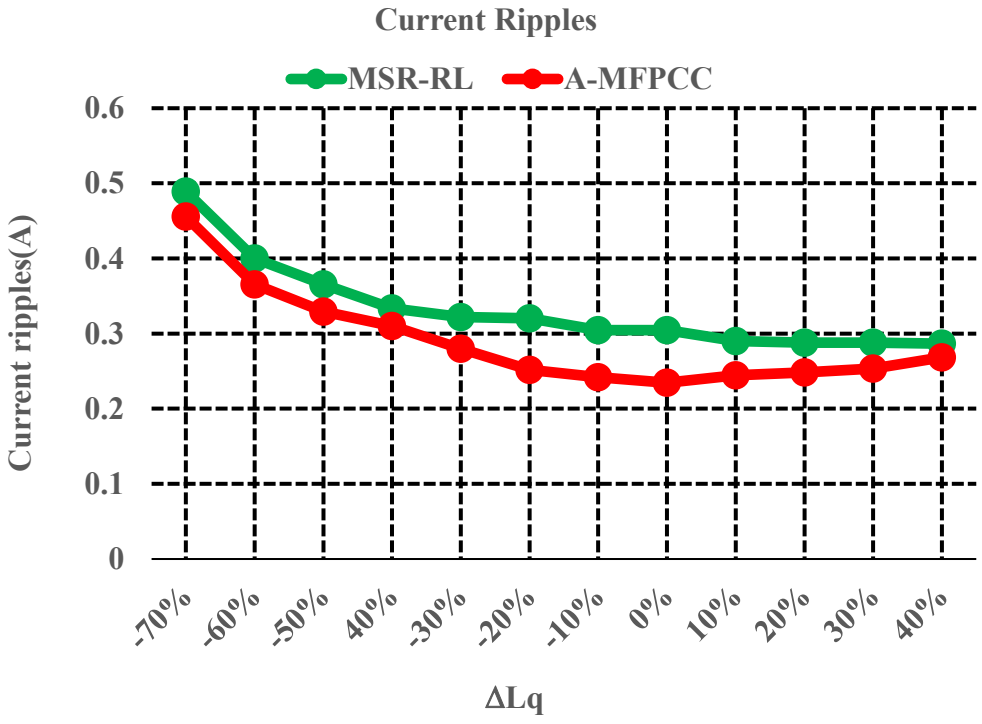


(b)

Fig. 6.22. Quantitative comparison of MSR-RL and A-MFPCC with the variation of inductance (L_d), (a) torque ripples, and (b) current ripples.

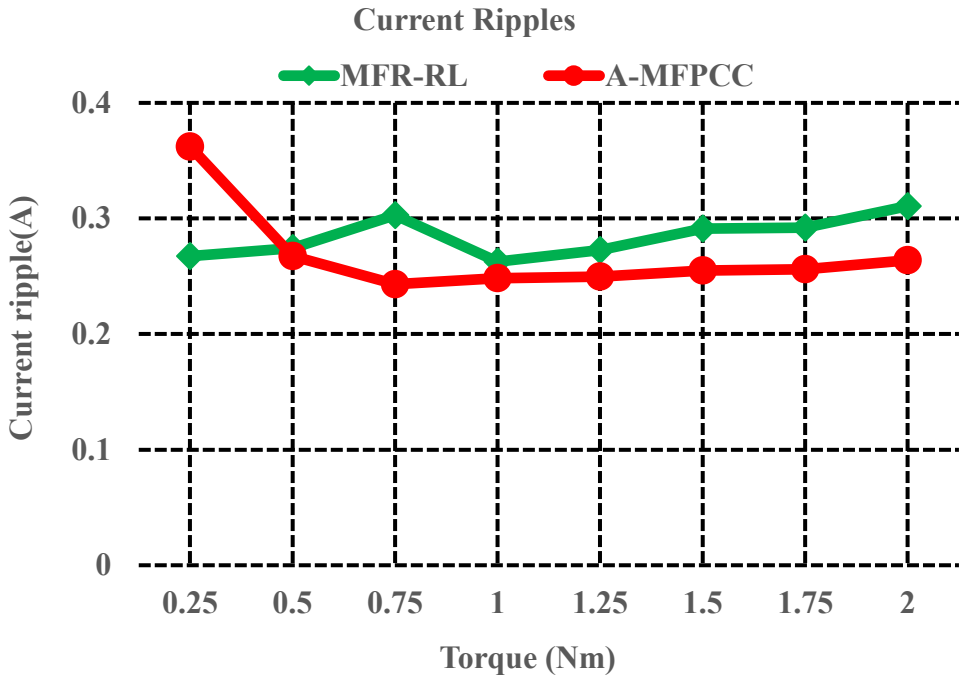


(a)

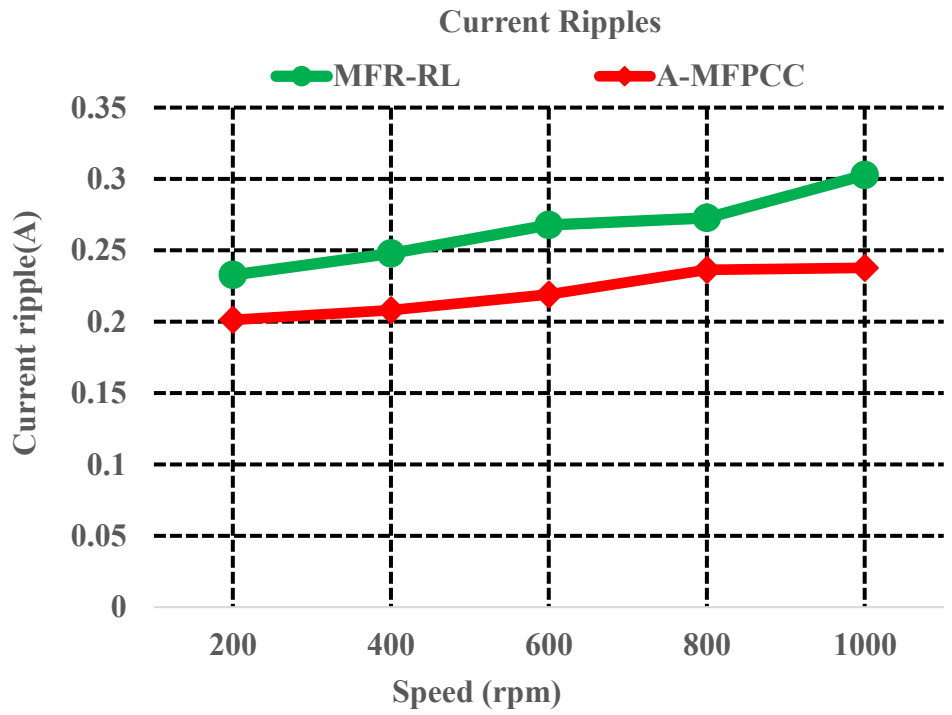


(b)

Fig. 6.23. Quantitative comparison of MSR-RL and A-MFPCC with the variation of inductance (L_q), (a) torque ripples, and (b) current ripples.



(a)



(b)

Fig. 6.24. Current ripple comparison of MSR-RL and A-MFPCC with (a) different load torques at rated speed (1000 rpm) and (b) various speeds under torque (2 Nm).

6.7.2 Robustness Evaluation

Robustness to uncertainties is a primary goal of both MSR-RL and A-MFPCC. Therefore, their robustness is evaluated using the proposed method (Chapter 4). Based on the maximum parameter variations (Table 6.1), the robustness evaluation of MSR-RL and A-MFPCC is conducted, and the corresponding sigma levels and Z-Values are presented in Table 6.5.

Table 6.5: Robustness evaluation results of the MSR-RL and A-MFPCC.

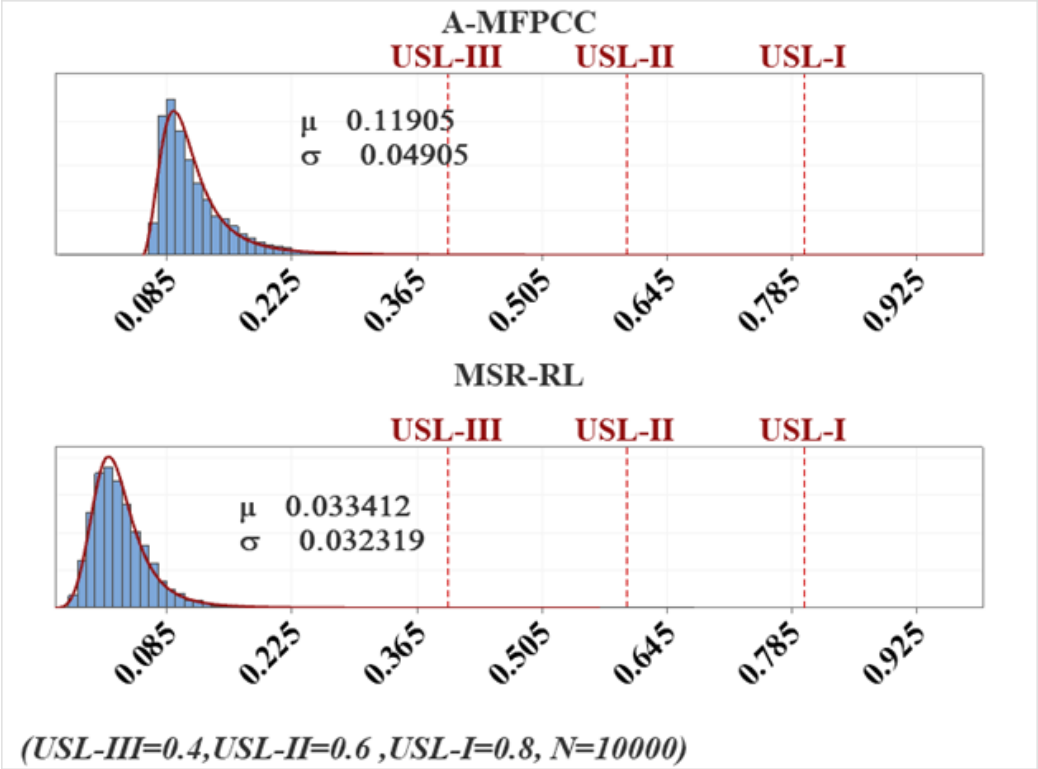
Indicator	Z_{T_s}	Z_{OS}	Z_{ω}	$Z_{T_{ripp}}$	$Z_{i_{ripp}}$	n_{sys}	POF
Controller	(Application-I)						
A-MFPCC	50.2	44.1	167.1	13.9	22.0	6.0	0
MSR-RL	33.4	61.9	146.5	23.7	17.9	6.0	0
	(Application-II)						
A-MFPCC	35.3	24.3	133.4	9.8	14.9	4.6	≈ 0
MSR-RL	22.5	36.2	94.0	17.5	12.3	3.9	0.01%
	(Application-III)						
A-MFPCC	20.5	14.5	51.8	5.7	7.9	3.9	0.01%
MSR-RL	11.7	23.4	43.0	11.3	6.6	3.2	0.16%

The robustness evaluation in Table 6.5 shows that both controllers achieved good robustness regarding three application requirements. A sigma level of 6σ was achieved for both controllers for low requirements (application-I). For Application-II, MSR-RL achieved 3.9σ, while A-MFPCC achieved 4.6σ. As for Application-III, A-MFPCC achieved 3.9σ, while MSR-RL achieved only 3.2σ. MSR-RL achieved sigma levels lower than A-MFPCC even though the Z-values of MSR-RL are higher for the overshoot and torque ripple performance indicators. This is because the sigma levels are computed based on the number of defects, while Z-values are computed based on the mean (μ) and

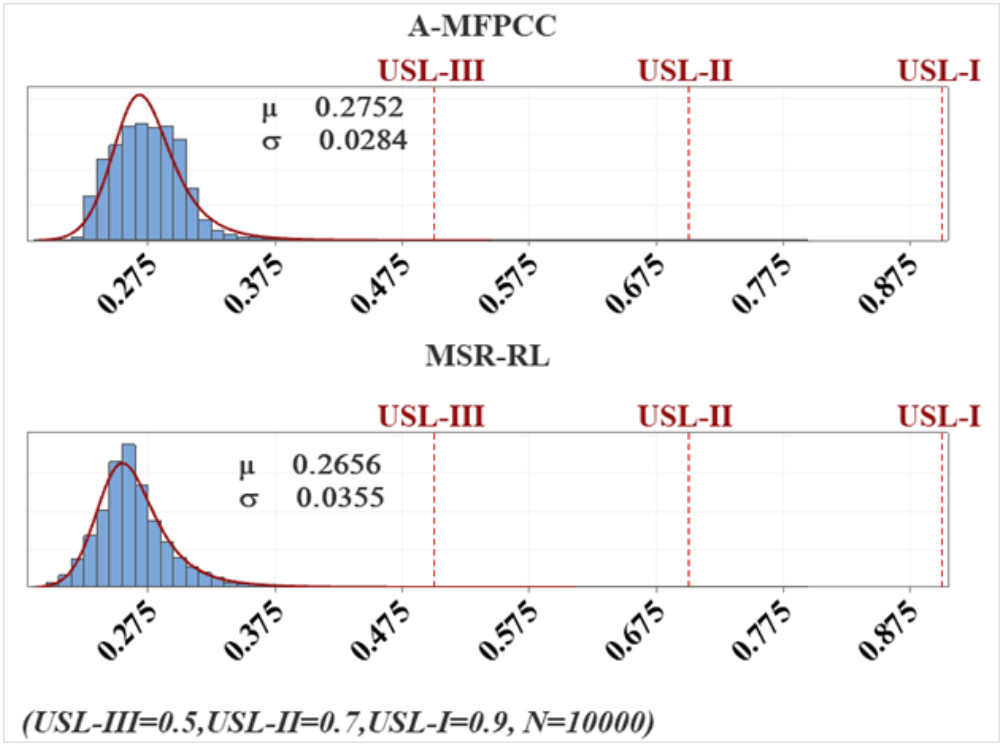
standard deviation (σ). This means MSR-RL may have achieved effective performance for most of the data samples, maintaining low values of the performance indicators far away from the specification limits. However, it exceeds the limit for a few samples, resulting in a low sigma level and high Z-values (because the mean is small compared to the specification limit). This can be illustrated by the capability plot of torque and current ripples of A-MFPCC and MSR-RL presented in Fig. 6.25.

Furthermore, the Z-Values of performance indicators show the weakness of MSR-RL is the settling time robustness. This means MSR-RL dynamic robustness is the main factor affecting overall system robustness. Thus, it is essential to improve the dynamic response of MSR-RL, including the optimization of the speed controller for better settling time robustness and achieving 6σ for all three application requirements.

MSR-RL and A-MFPCC have shown excellent robustness over maximum parameter variations (Table 6.1) with three application requirements (Table 6.3). This indicates the effectiveness of MSR-RL and its ability to perform as the online optimized controller (A-MFPCC).



(a)



(b)

Fig. 6.25. Capability plots of A-MFPCC and MSR-RL for (a) torque ripples and (b) current ripples.

6.8 Summary

This chapter introduced an MSR-RL-based current control of PMSM drives. Standard RL methods often struggle to adapt to new operating conditions and parameter sets, reducing system performance and robustness. To overcome these limitations, the proposed MSR-RL leveraged multi-task RL to train a single policy that can generalize and perform well across a wide range of parameter sets.

MSR-RL achieved strong robustness and improved adaptability to new parameter sets by utilizing multiple training tasks with varying parameters. The parameter sets, represented as contexts in the form of Contextual Markov decision processes (CMDPs), allow for optimizing a policy that maximizes the cumulative reward over all contexts. By clustering tasks with shared information into models, a unified policy is generated, ensuring robustness not only to the clustered models but also to unseen models.

The proposed MSR-RL approach presented several advantages compared to standard RL methods. The need to train separate policies for different parameter sets was eliminated, reducing computational effort and enhancing efficiency. Additionally, the learned policy can adapt to new operating conditions and parameter sets, thereby improving the overall performance and robustness of the system.

The performance comparisons based on simulation, experiments, quantitative, and robustness evaluation have validated the superiority of MSR-RL over standard RL in different operating conditions. Additionally, the comparison of MSR-RL with A-MFPCC indicates the ability of MSR-RL to achieve similar performance with fewer online computational requirements.

REFERENCES

[6.1] J. Yang, W.-H. Chen, S. Li, L. Guo, and Y. Yan, "Disturbance/Uncertainty Estimation and Attenuation Techniques in PMSM Drives—A Survey," *IEEE Transactions on Industrial Electronics,* vol. 64, no. 4, pp. 3273-3285, 2017, doi: 10.1109/tie.2016.2583412.

[6.2] C. Lin, T. Liu, J. Yu, L. Fu, and C. Hsiao, "Model-Free Predictive Current Control for Interior Permanent-Magnet Synchronous Motor Drives Based on Current Difference Detection Technique," *IEEE Transactions on Industrial Electronics,* vol. 61, no. 2, pp. 667-681, 2014.

[6.3] Y. Zhou, H. Li, and H. Zhang, "Model-free deadbeat predictive current control of a surface-mounted permanent magnet synchronous motor drive system," *Journal of Power Electronics,* vol. 18, no. 1, pp. 103-115, 2018.

[6.4] A. Traue, G. Book, W. Kirchgässner, and O. Wallscheid, "Toward a Reinforcement Learning Environment Toolbox for Intelligent Electric Motor Control," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 33, no. 3, pp. 919-928, 2022, doi: 10.1109/TNNLS.2020.3029573.

[6.5] G. Book *et al.*, "Transferring Online Reinforcement Learning for Electric Motor Control From Simulation to Real-World Experiments," *IEEE Open Journal of Power Electronics,* vol. 2, pp. 187-201, 2021, doi: 10.1109/OJPEL.2021.3065877.

[6.6] A. Ez-zizi, S. Farrell, D. Leslie, G. Malhotra, and C. J. H. Ludwig, "Reinforcement Learning Under Uncertainty: Expected Versus Unexpected Uncertainty and State Versus Reward Uncertainty," *Computational Brain & Behavior,* 2023/03/20 2023, doi: 10.1007/s42113-022-00165-y.

[6.7] D. Liu, X. Yang, D. Wang, and Q. Wei, "Reinforcement-Learning-Based Robust Controller Design for Continuous-Time Uncertain Nonlinear Systems Subject to Input Constraints," *IEEE Transactions on Cybernetics,* vol. 45, no. 7, pp. 1372-1385, 2015, doi: 10.1109/TCYB.2015.2417170.

[6.8] K. Nussenbaum and C. A. Hartley, "Reinforcement learning across development: What insights can we draw from a decade of research?," *Developmental cognitive neuroscience,* vol. 40, p. 100733, 2019.

[6.9] S. Meyn, *Control systems and reinforcement learning*. Cambridge University Press, 2022.

[6.10] Z. Ding and H. Dong, "Challenges of reinforcement learning," *Deep Reinforcement Learning: Fundamentals, Research and Applications,* pp. 249-272, 2020.

[6.11] M. A. Wiering and M. Van Otterlo, "Reinforcement learning," *Adaptation, learning, and optimization,* vol. 12, no. 3, p. 729, 2012.

[6.12] A. massoud Farahmand, A. Shademan, M. Jagersand, and C. Szepesvári, "Model-based and model-free reinforcement learning for visual servoing," in *2009 IEEE International Conference on Robotics and Automation*, 2009: IEEE, pp. 2917-2924.

[6.13] J. N. Tsitsiklis, "Asynchronous stochastic approximation and Q-learning," *Machine learning,* vol. 16, pp. 185-202, 1994.

[6.14] V. Mnih *et al.*, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602,* 2013.

[6.15] D. Jakobeit, M. Schenke, and O. Wallscheid, "Meta-Reinforcement-Learning-Based Current Control of Permanent Magnet Synchronous Motor Drives for a Wide Range of Power Classes," *IEEE Transactions on Power Electronics,* vol. 38, no. 7, pp. 8062-8074, 2023, doi: 10.1109/tpel.2023.3256424.

[6.16] A. Modi, N. Jiang, S. Singh, and A. Tewari, "Markov decision processes with continuous side information," in *Algorithmic Learning Theory*, PMLR, pp. 597-618, 2018.

[6.17] T. Yu *et al.*, "Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning," in *Conference on robot learning*, PMLR, pp. 1094-1100, 2020.

CHAPTER 7

CONCLUSION AND FUTURE WORKS

7.1 Conclusion

The main findings and achievements presented in this thesis are summarized as follows:

- A comprehensive literature survey was conducted about PMSMs and their control methods. PMSM drive uncertainties, robust control methods, and data-driven control methods were critically investigated.
- Improved two-vector MPCs were proposed to eliminate the shortcomings of conventional MPC. Fuzzy decision-making criteria were used to eliminate the weighting factors and select an additional switching vector. The proposed methods were validated by simulation, experiment, and quantitative analysis and performed better than the conventional MPC.
- A novel robustness evaluation method was introduced based on the Six Sigma design methodology. Five RPC methods and conventional MPC were numerically evaluated based on the proposed method, and their robustness levels (sigma levels) were obtained. This method was also used to evaluate the robustness of the proposed controllers in the subsequent chapters.
- An adaptive model-free predictive current control (A-MFPCC) with an adaptive current difference updating mechanism was proposed. A reference vector was generated based on the position of the current reference and current error, then was

used to update the current difference and prevent stagnation. The effectiveness of this method was validated by conducting a comparison with two other MFPCC schemes based on simulation, experiment, and robustness evaluation. The proposed method performed better than other methods over different operating conditions.

- Robust data-driven RL-based control was proposed for PMSM drives. The proposed method was trained using multiple parameter sets to obtain a robust policy for all parameter variations. Considering finite parameter sets, the proposed MSR-RL was trained and validated based on PMSM drives. Compared to standard RL, the proposed MSR-RL showed strong robustness to parameter variations and changes in operating conditions. The controller was validated through a comparison with standard RL based on simulation, experimental, and robustness evaluations.
- The proposed MSR-RL and A-MFPCC were evaluated and compared over different operating conditions and parameter variations. The quantitative and robustness analyses have shown the ability of MSR-RL to achieve comparable performance to A-MFPCC with fewer online computational requirements.

7.2 Future Works

This project has investigated various robust control methods for PMSM drives, including model-free and data-driven RL robust control methods. However, there is always room for future improvements. The potential improvements and future perspective of this research are presented as follows:

- Robustness evaluation with other uncertainties, e.g., unmodelled dynamics. Currently, robustness evaluation is performed with parameter variations only. Thus, evaluating

the robustness by considering parameter variations and unmodelled dynamics is essential.

- Considering the stability robustness evaluation and more performance indicators like switching frequency.
- Robust optimization can be implemented to enhance the performance of the proposed controllers. Speed control and other system parameters can be optimized through this technique in the future.
- Exploration of novel methods for reducing the sample size in robustness evaluation, particularly through stratified sampling, bootstrapping, machine learning techniques, statistical methods, Bayesian inference, meta-analysis, sequential testing, simulation and modeling, expert knowledge, and data preprocessing, can enable more efficient evaluations while maintaining data quality.
- Integrating the robust optimization of control systems with the robust optimization of electrical machines.
- Large-scale machine drive setup to be considered, e.g., high-power rating machines suitable for EVs to validate the controller performance.
- Using real-time measurement data for RL training. Currently, RL is trained based on simulation data; thus, in the future, it is essential to collect extensive measurement data of the drive system at various operating conditions and use it for RL training.
- Considering different machine learning algorithms, including different RL agents than the one currently used
- Exploring the possibility of merging different control methods will be considered. his includes combining data-driven control methods such as A-MFPCC and MSR-RL to

generate robust adaptive and computationally efficient control methods. Additionally, six-sigma robust optimization can be used to optimize the proposed A-MFPCC parameters and RL hyperparameters.

APPENDICES

APPENDIX A

Appendix A.1 Impact of Covid-19 on Research Progress

The COVID-19 pandemic significantly influenced the progression of this doctoral research project. Commencing in January 2020, the project was swiftly confronted with the Australian government's travel ban, preventing me from entering the country until late December 2021. My supervisors and I had to adapt our approach throughout this prolonged period, shifting our focus toward theoretical and simulation-based investigations.

By April 2022, when access to the UTS labs was finally reinstated, we commenced building the experimental setup. By the end of that year, experimental results were obtained, leading to the publication of one research article. However, due to the limited time available (only six months) to complete the Ph.D., I had to prioritize writing my thesis while simultaneously working on three additional papers for publication. One of these papers has been successfully completed and submitted for publication, which is now under revision. However, recognizing the extended review process often associated with esteemed journals, we prioritized the thesis's completion before dedicating our efforts to finalizing the remaining papers.

The challenges posed by the COVID-19 pandemic necessitated adaptability and resilience, requiring us to reframe our research approach and maximize available resources. While the initial experimental phase experienced delays, the shift towards comprehensive theoretical and simulation-based investigations ensured that progress was maintained.

Despite the constrained timeframe, completing the Ph.D. thesis became a focal point, underscoring the significance and novelty of this comprehensive body of work, which we believe holds potential for publication in high-quality journals.

Appendix A.2 List of Publication

Published Articles

[1] **N. Farah**, G. Lei, J. Zhu, and Y. Guo, "Two-vector Dimensionless Model Predictive Control of PMSM Drives Based on Fuzzy Decision Making," in *CES Transactions on Electrical Machines and Systems*, vol. 6, no. 4, pp. 393-403, December 2022, doi: 10.30941/CESTEMS.2022.00051.

Articles Under Revision

[1] **N. Farah**, G. Lei, J. Zhu, and Y. Guo, "A Novel Robustness Evaluation Method for Predictive Control of PMSM Drives," **under revision***, " IEEE Transactions on Energy Conver*sion.

[2] **N. Farah**, G. Lei, J. Zhu, and Y. Guo, " Robust Model-Free Reinforcement Learning Based Current Control of PMSM Drives," **under revision,**, IEEE Transactions on Transportation Electrification.

Articles Under Review

[1] **N. Farah**, G. Lei, J. Zhu, and Y. Guo, "Adaptive Model-Free Predictive Current Control of PMSM Drives," **under review,** IEEE Transactions on Industrial electronics.

Appendix B. Quantitative Analysis of the Literature

The various RPC methods of PMSM drives in the literature are divided into categories depending on the type of robust control used. RPC -based prediction error correction, observers, optimized cost function, model-free, and combined or hybrid techniques are some of these categories. A quantitative comparison of several methods of each category is conducted by considering a set of performance measures based on PMSM drives applications. These applications required /preferred PMSM drive performance can be used as indicators to perform the comparison. Different performance indicators can be identified as essential measures to determine how effective a control method of PMSM drive is for a specific application. These indicators include dynamic response, steady-state response, control robustness, and drive efficiency. The level of each indicator for the various RPC methods is determined by considering a quantitative scale (0-5) depending on the effectiveness of each control method. Table B.1 presents four essential performance indicators of the control method of PMSM drive, along with explanations of the quantitative scales for each indicator.

Based on these indicators, several RPC methods of PMSM drives have been quantitatively analysed by rating their effectiveness regarding each indicator with an appropriate scale. Then, the total sum of the scales is calculated to show which robust method has the highest scale and perfectly matches the essential requirements for a specific application. Depending on each application, the above indicators differ in importance. For example, efficiency is more critical in certain applications than dynamic response, while other applications require high dynamic response no matter the efficiency. Therefore, it is

essential to include a weighting factor for each indicator depending on its significance to the application**.**

Table B.1: Performance indicators of PMSM control methods

scale	**Indicator**
	Dynamic Response (DR)
0 (worst)	The system response cannot track the reference at all
1(bad)	The system response takes a very long time to track the reference
2(average)	The system response is slow and has a long settling time
3(good)	The system response tracks the reference with a moderate settling time
4(very good)	The system tracks the reference with a quite good settling time
5(excellent)	The system response is fast, with a faster settling time
	Steady-state response (SSR)
0 (worst)	The steady-state response fluctuates with high ripples.
1(bad)	The response is not tracking the reference properly and has high ripples.
2(average)	The response is tracking the reference but has high ripples.
3(good)	The response is tracking the reference but has moderate ripples.
4(very good)	The response is tracking the reference but has low ripples.
5(excellent)	The response is precisely tracking the reference and has minimum ripples.
	Control Robustness (CR)
0 (worst)	No robustness, and with any uncertainty, system loses performance tracking

1(bad)	Low robustness and high uncertainties, the system loses performance tracking.
2(average)	The system can withstand uncertainties but with very poor performance.
3(good)	The system can withstand uncertainties with acceptable performance.
4(very good)	The system can withstand uncertainties with very good performance but has no robust method for all uncertainties.
5(excellent)	The system has a robust mechanism for all uncertainties and has perfect performance under any uncertainties.
	Efficiency (η)
0 (worst)	The system losses are maximum with the lowest efficiency.
1(bad)	The system has high losses with lower efficiency
2(average)	The system losses are moderate with moderate efficiency.
3(good)	The system losses are low with good efficiency
4(very good)	The system has a loss minimization technique with high efficiency.
5(excellent)	The system has minimum losses with the highest efficiency.

The four indicators are equally crucial for washing machine applications, but the dynamic response is more critical than the efficiency for servo-drive applications. Thus, the importance of these indicators can be weighted with an appropriate weighting factor. For example, if the significance of the efficiency for the servo drive is 80%, a weighting factor of 0.8 is used for the switching frequency with a servo drive application. Using weighting factors for certain indicators will produce different total scores of a specific PMSM drive method. Servo-drives, EVs, and washing machine PMSM applications with the

corresponding weighting factor (γ) for the dynamic response, steady-state response, control robustness, and efficiency performance indicators are presented in Table B.2. Several RPC methods of PMSM drives have been quantitatively analysed based on four performance indicators with three application weighting factors, as shown in Table B.3.

Table B.2 The weighting factor (γ) for performance indicators of three PMSM drive applications

γ Application	Dynamic response (DR)	Steady-state response (SSR)	Control robustness (CR)	efficiency (η)
Servo drives	1	0.8	0.8	0.75
EV	0.7	1	0.8	0.9
Washing machine	1	1	1	1

Table B.1: Quantitative analysis of RPC studies for PMSM drive

Study	RPC based on:	DR	SSR	CR	η	Total		
						Servo drive	EVs	Washing machine
[B.1]	Optimized cost function	3	4	3	3.2	11	11.38	13.2
[B.2]	Observer	3	4	4	3.2	11.8	12.18	14.2

[B.3]	Optimized cost function	4	3	2	2.4	9.8	9.56	11.4
[B.4]	Hybrid techniques	3	4	4	4	12.4	12.9	15
[B.5]	Observer	3	4	4	4	12.4	12.9	15
[B.6]	Prediction error	3	4	4	4	12.4	12.9	15
[B.7]	Observer	4	3	4	4	12.6	12.6	15
[B.8]	Hybrid techniques	5	4	3	4	13.6	13.5	16
[B.9]	Observer	3	4	5	4	13.2	13.7	16
[B.10]	Observer	4	4	4	3	12.65	12.7	15
[B.11]	Hybrid techniques	5	4	5	4	15.2	15.1	18
[B.12]	Observer	4	4	4	3	12.65	12.7	15
[B.13]	Hybrid techniques	4	4	5	4	14.2	14.4	17
[B.14]	Observer	4	4	4	4	13.4	13.6	16
[B.15]	Optimized cost function	3	3	4	3	10.85	11	13
[B.16]	Prediction error	4	4	4	4	13.4	13.6	16
[B.17]	Prediction error	4	3	4	3	11.85	11.7	14
[B.18]	Prediction error	4	3	4	3	11.85	11.7	14
[B.19]	Hybrid techniques	4	4	4	4	13.4	13.6	16
[B.20]	Model-free	4	4	5	4	14.2	14.4	17
[B.21]	Model-free	5	4	5	4	15.2	15.1	18

REFERENCES

[B.1] A. Sivaprakasam and L. N. Ramya, "A new approach to minimize torque ripple and noise in model predictive control of permanent magnet synchronous motor drives," *Journal of Vibration and Control,* p. 1077546320933743, 2020, doi: 10.1177/1077546320933743.

[B.2] R. Errouissi, M. Ouhrouche, W.-H. Chen, and A. M. Trzynadlowski, "Robust cascaded nonlinear predictive control of a permanent magnet synchronous motor with antiwindup compensator," *IEEE Transactions on Industrial Electronics,* vol. 59, no. 8, pp. 3078-3088, 2011.

[B.3] M. Preindl and S. Bolognani, "Model predictive direct speed control with finite control set of PMSM drive systems," *IEEE Transactions on Power Electronics,* vol. 28, no. 2, pp. 1007-1015, 2012.

[B.4] W. Xie *et al.*, "Finite-Control-Set Model Predictive Torque Control With a Deadbeat Solution for PMSM Drives," *IEEE Transactions on Industrial Electronics,* vol. 62, no. 9, pp. 5402-5410, 2015.

[B.5] S.-C. Carpiuc and C. Lazar, "Fast real-time constrained predictive current control in permanent magnet synchronous machine-based automotive traction drives," *IEEE Transactions on Transportation Electrification,* vol. 1, no. 4, pp. 326-335, 2015.

[B.6] M. Siami, D. A. Khaburi, A. Abbaszadeh, and J. Rodríguez, "Robustness Improvement of Predictive Current Control Using Prediction Error Correction for Permanent-Magnet Synchronous Machines," *IEEE Transactions on Industrial Electronics,* vol. 63, no. 6, pp. 3458-3466, 2016.

[B.7] M. Yang, X. Lang, J. Long, and D. Xu, "Flux immunity robust predictive current control with incremental model and extended state observer for PMSM drive," *IEEE Transactions on Power Electronics,* vol. 32, no. 12, pp. 9267-9279, 2017.

[B.8] Z. Zhou, C. Xia, Y. Yan, Z. Wang, and T. Shi, "Disturbances attenuation of permanent magnet synchronous motor drives using cascaded predictive-integral-resonant controllers," *IEEE Transactions on Power Electronics,* vol. 33, no. 2, pp. 1514-1527, 2017.

[B.9] Y. Jiang, W. Xu, C. Mu, and Y. Liu, "Improved deadbeat predictive current control combined sliding mode strategy for PMSM drive system," *IEEE Transactions on Vehicular Technology,* vol. 67, no. 1, pp. 251-263, 2017.

[B.10] R. Yang, M.-Y. Wang, L.-Y. Li, C.-M. Zhang, and J.-L. Jiang, "Robust predictive current control with variable-gain adaptive disturbance observer for PMLSM," *IEEE Access,* vol. 6, pp. 13158-13169, 2018.

[B.11] H. T. Nguyen and J. Jung, "Finite Control Set Model Predictive Control to Guarantee Stability and Robustness for Surface-Mounted PM Synchronous Motors," *IEEE Transactions on Industrial Electronics,* vol. 65, no. 11, pp. 8510-8519, 2018.

[B.12] X. Zhang, L. Zhang, and Y. Zhang, "Model Predictive Current Control for PMSM Drives With Parameter Robustness Improvement," *IEEE Transactions on Power Electronics,* vol. 34, no. 2, pp. 1645-1657, 2019, doi: 10.1109/TPEL.2018.2835835.

[B.13] Y. Liu, S. Cheng, B. Ning, and Y. Li, "Robust model predictive control with simplified repetitive control for electrical machine drives," *IEEE Transactions on Power Electronics,* vol. 34, no. 5, pp. 4524-4535, 2018.

[B.14] S. Huang, G. Wu, F. Rong, C. Zhang, S. Huang, and Q. Wu, "Novel predictive stator flux control techniques for PMSM drives," *IEEE Transactions on Power Electronics,* vol. 34, no. 9, pp. 8916-8929, 2018.

[B.15] X. Liu, L. Zhou, J. Wang, X. Gao, Z. Li, and Z. Zhang, "Robust Predictive Current Control of Permanent-Magnet Synchronous Motors with Newly Designed Cost Function," *IEEE Transactions on Power Electronics,* 2020.

[B.16] F. Wang, K. Zuo, P. Tao, and J. Rodriguez, "High Performance Model Predictive Control for PMSM by using Stator Current Mathematical Model Self-regulation Technique," *IEEE Transactions on Power Electronics,* 2020.

[B.17] L. Yongfei, Y. Li, and Q. Wang, "Robust Predictive Current Control with Parallel Compensation Terms against Multi-parameter Mismatches for PMSMs," *IEEE Transactions on Energy Conversion,* 2020.

[B.18] F. Niu *et al.*, "Current Prediction Error Reduction Method of Predictive Current Control for Permanent Magnet Synchronous Motors," *IEEE Access,* vol. 8, pp. 124288-124296, 2020.

[B.19] L. Yan, F. Wang, P. Tao, and K. Zuo, "Robust Predictive Torque Control of Permanent Magnet Synchronous Machine Using Discrete Hybrid Prediction Model," *IEEE Transactions on Energy Conversion,* 2020.

[B.20] Y. Zhou, H. Li, and H. Zhang, "Model-free deadbeat predictive current control of a surface-mounted permanent magnet synchronous motor drive system," *Journal of Power Electronics,* vol. 18, no. 1, pp. 103-115, 2018.

[B.21] C. Lin, T. Liu, J. Yu, L. Fu, and C. Hsiao, "Model-Free Predictive Current Control for Interior Permanent-Magnet Synchronous Motor Drives Based on Current Difference Detection Technique," *IEEE Transactions on Industrial Electronics,* vol. 61, no. 2, pp. 667-681, 2014.