Imperial College London

Doctoral Thesis

Imperial College London, Department of Aeronautics

Data-driven reduced-order modelling of the plasma systems dynamics

Candidate:
Farbod Faraji

Supervisor:
Dr. Aaron Knoll

Submitted in partial fulfilment of the requirements for Doctor of Philosophy (PhD) degree in Aeronautics at Imperial College London

September 2024

Statement of Originality

I, Farbod Faraji, hereby certify that this thesis and the materials therein are my own work and are derived from my own PhD research.

I confirm that the literature is appropriately cited, and that no illustration has been reproduced from prior published materials of other researchers without permission and proper credit line.

Parts of the present doctoral thesis are already published in journal articles and/or presented as conference papers. I have included at the end of this statement a cross-referenced list that indicates the correspondence between the chapters of this thesis and the papers in the List of Publications (LOP) directly or indirectly originated from my research during PhD studies.

In the case of co-authored papers, I confirm that any materials potentially reused in this thesis constitute my original contributions to those publications.

Chapter 1: LOP journal articles no. 5, 6, 8, and 7, and conference papers no. 5 and 6.

Chapter 2: LOP journal articles no. 1 - 4, and conference papers no. 3, 4, 7, and 8

Chapter 3: LOP journal article no. 1 and conference paper no. 4

Chapter 4: LOP journal article no. 2 and conference paper no. 4 and 8

Chapter 5: LOP journal article no. 3 and conference paper no. 4

Chapter 6: LOP journal article no. 4 and conference papers no. 3, 4, and 6

Chapter 7: LOP conference papers no. 1 - 3, 7, and 8

Copyright Declaration

The copyright of this thesis rests with the author. Unless otherwise indicated, its contents are licensed under a Creative Commons Attribution-Non Commercial 4.0 International License (CC BY-NC).

Under this license, you may copy and redistribute the material in any medium or format. You may also create and distribute modified versions of the work. This is on the condition that: you credit the author and do not use it, or any derivative works, for a commercial purpose.

When reusing or sharing this work, ensure you make the license terms clear to others by naming the license and linking to the license text. Where a work has been adapted, you should indicate that the work has been changed and describe those changes.

Please seek permission from the copyright holder for uses of this work that are not included in this license or permitted under UK Copyright Law.

Abstract

Despite the increasing need for computationally efficient, predictive, and low-dimensional (low-order) plasma models, these have remained elusive so far. The crucial need for such models stems, in part, from the breakthroughs they can enable in scientific research into yet-unresolved plasma phenomena. Furthermore, these models can boost the efforts toward realizing fusion energy and can revamp the development of advanced plasma-based technologies. Machine-learning (ML) and data-driven (DD) methods can play a transformative role toward achieving predictive, generalizable, and interpretable reduced-order models (ROMs) for plasma systems.

Motivated by the above, the overarching objective of this thesis is to explore the landscape of ML/DD techniques for physics modelling and to examine a subset of approaches among those with the highest promise to realize predictive plasma ROMs.

Toward this goal, a critical literature review is performed. Accordingly, two categories of DD algorithms are down-selected. The algorithms in the first category learn and predict the time evolution of *global* structures (modes) underlying the data. From this category, the Optimized Dynamic Mode Decomposition (OPT-DMD) method is chosen.

The second selected category comprises *local* dynamics discovery algorithms, where the methods discover the differential equations governing the dynamics. A novel method belonging to this category, termed “Phi Method”, is established and extensively assessed.

The predictive performance of ROMs from OPT-DMD and Phi Method is evaluated across a wide array of test cases with varying dynamics complexity. Both approaches demonstrate remarkable potential for forecasting the plasma state. However, by the merit of its local dynamics learning, Phi Method presents a higher degree of flexibility for ROM development. Moreover, the Phi Method ROMs accurately discover parametric dependencies of the dynamics, a crucial aspect for any fully generalizable plasma model.

To conclude, the next advancement steps on Phi Method and some promising future research directions in ML-enabled plasma modelling are outlined.

Acknowledgements

As I stand at the threshold of completing this PhD journey, I am reminded of a sentiment I penned with Maral (Maryam), my personal and professional companion, within our joint master's thesis in 2017: "The 'END' does not exist, at least in absolute terms." Today, I find that thought resonates even more deeply. This PhD is not an end but a continuation of the journey, a chapter closing only to allow the next to begin.

At this moment of reflection over the past 3 and a half years, first and foremost, I would like to express my deepest gratitude to my supervisor, **Dr. Aaron Knoll**. Your guidance, mentorship, and unwavering support have been invaluable. You have not only imparted knowledge and granted me scientific independence but also guided me on how to think critically, challenge assumptions, and approach problems with both rigor and creativity. I will always remember the group dinners we shared, filled with as much laughter as insightful discussions. The witty anecdotes you shared have left a lasting impression, and I hope to avoid some of those more precarious moments you warned us about! For all these experiences and more, I am forever grateful.

I would also like to extend my heartfelt thanks to **Professor Nathan Kutz**, who, though not my formal advisor, has played a crucial role in my academic development. Our collaborations have been a source of great insight and learning, and your encouragement and willingness to share your wisdom have been a tremendous support. I will always cherish the memory of the amazing talks we had during our lunch before the end of your sabbatical in London. I hope that this is the first of many such shared memories. Thank you for being a mentor and a friend.

To **my family**, whose love and support have been the foundation upon which all my successes are built, thank you. Your steadfast belief in me, your patience, and your encouragements have been my pillars of confidence and strength. This achievement is as much yours as it is mine. I am eternally grateful for your presence in my life; your constant care and devotion have been the guiding force behind all my endeavors. Without you, none of this would have been possible.

Lastly, to **Maral**, my wife, my colleague, my partner in every sense of the word—thank you. Together, more than 12 years ago, we started a journey, and together we have grown, faced challenges, and celebrated successes. Your unending support, love, and understanding have been my constant sources of drive and resilience. You are the best partner anyone could ever hope for, both in life and in profession. I am grateful for the endless nights of brainstorming, the shared excitement over new discoveries (oh, when the codes worked…!), and the quiet moments of reflection. Our shared aspirations, mutual respect, and deep love have been the most profound part of our exciting joint journey, which extends far before the PhD studies. Thank you for standing by my side, for believing in me, and for being the remarkable, brilliant person you are. I look forward to all the pages we have yet to write together.

And so, this PhD is not an end but a dot in the story we continue to write. There are many more blank pages, many more chapters, and I am excited about every one of them.

Thank you all.

"*What I wonder is how this thesis will be viewed in the year 2050…*

I deeply hope that it would be seen as a primitive, yet earnest, endeavour towards the solutions that will have been ultimately devised for what are, as of 2024, the long-standing scientific and technological challenges in plasma physics."

F.F.

Contents

STATEMENT OF ORIGINALITY **2**

COPYRIGHT DECLARATION **3**

ABSTRACT **4**

ACKNOWLEDGEMENTS **5**

CONTENTS **8**

LIST OF FIGURES **10**

LIST OF TABLES **18**

LIST OF PUBLICATIONS **19**

CHAPTER 1 BACKGROUND AND PROBLEM STATEMENT **21**

1.1. PLASMAS: UNIVERSALLY UBIQUITOUS, CHALLENGING TO UNDERSTAND, PREDICT AND CONTROL 22

1.2. THE NEED FOR PREDICTIVE MODELS OF PLASMA SYSTEMS 29

1.3. STATUS QUO OF THE COMPUTATIONAL APPROACHES TO STUDY PLASMAS 32

1.4. REDUCED-ORDER PARTICLE-IN-CELL SCHEME: A COST-EFFECTIVE KINETIC TOOL FOR HIGH-FIDELITY DATA GENERATION 35

1.5. MACHINE LEARNING FOR COST-EFFICIENT PREDICTIVE MODELLING OF PLASMAS: PROMISES AND CHALLENGES 36

1.6. PROBLEM STATEMENT: RESEARCH MOTIVATIONS AND OBJECTIVES 40

CHAPTER 2 STATE-OF-THE-ART REVIEW ON MACHINE LEARNING FOR PHYSICAL SYSTEMS MODELLING 41

2.1. GENERAL OVERVIEW 42

2.2. ML FOR DIMENSIONALITY REDUCTION AND COORDINATE TRANSFORMATION 53

2.3. DATA-DRIVEN DISCOVERY OF THE DYNAMICS 54

2.4. PRIOR APPLICATIONS OF ML/DD TECHNIQUES IN PLASMA MODELLING 60

CHAPTER 3 PROPER ORTHOGONAL DECOMPOSITION AND DYNAMIC MODE DECOMPOSITION FOR THE PLASMA DATA ANALYSIS AND PATTERN EXTRACTION **63**

3.1. THE THEORY OF THE PROPER ORTHOGONAL/SINGULAR VALUE DECOMPOSITION 63

3.2. DYNAMIC MODE DECOMPOSITION: BASIC FORMULATION, OPTIMIZED DMD, AND PARAMETRIC DMD 65

3.3. Description of the adopted test case: a cross-field plasma discharge in a 2D radial-azimuthal geometry 71
3.4. Application of the POD/SVD to understand the low-dimensional representability of the data . 73
3.5. Basic DMD and OPT-DMD for data reconstruction 79
3.6. Optimized DMD for data-driven identification of dominant plasma instability modes 82
3.7. Chapter conclusions 90
Chapter 4 Optimized dynamic mode decomposition for the plasma state forecasting 93
4.1. Description of the adopted demonstration cases 93
4.2. Performance of OPT-DMD for plasma dynamics forecasting 99
4.3. Sensitivity of the OPT-DMD performance to added Gaussian noise 117
4.4. Chapter conclusions 118
Chapter 5 Data-driven local operator finding for reduced-order modelling of plasmas.. 120
5.1. Phi Method: a novel approach for the discovery of discretized PDEs 120
5.2. Assessment of Phi Method in canonical test problems 124
5.3. Assessment of Phi Method for the prediction of a plasma discharge dynamics 134
5.4. Noise robustness of Phi Method: comparative analysis against OPT-DMD 143
5.5. Chapter conclusions 146
Chapter 6 Phi Method for the data-driven learning of parametric dynamics 149
6.1. Description of the Phi Method's implementation for parametric dynamics learning 150
6.2. Comparative evaluation of Phi Method's performance against parametric OPT-DMD in the canonical problem of a flow past a cylinder 152
6.3. Assessment of the cross-parameter predictive performance of Phi Method in the plasma discharge problem 162
6.4. Chapter conclusions 174
Chapter 7 Conclusions and future work 176
7.1. Overview and summary of the thesis 176
7.2. Future works and research directions 179
Bibliography 187
Appendix 197

List of Figures

FIG. 1.1 DENSITY AND TEMPERATURE CHARACTERISTICS OF TYPICAL PLASMAS; IMAGE CREDIT: LABORATOIRE DE PHYSIQUE DES PLASMAS (LPP) 25

FIG. 2.1 SAMPLE FIGURE FROM THE WORK OF KOCHKOV ET AL. [130] (SUPER-RESOLUTION ACCELERATION OF NUMERICAL SIMULATIONS). THE PLOT SHOWS THE TIME EVOLUTION OF THE VORTICITY FIELD FROM SIMULATIONS WITH A FINE (**TOP ROW**) AND A COARSE GRID (**BOTTOM ROW**) AS WELL AS A COARSE-GRID SIMULATION ENHANCED WITH ML (**MIDDLE ROW**) 48

FIG. 2.2 SCHEMATIC DIAGRAM OF DISCREPANCY MODELING AS AN ALTERNATIVE PATH TO INFORM ML-BASED DYNAMICS DISCOVERY WITH KNOWN PHYSICS SO AS TO FACILITATE LEARNING UNKNOWN PHYSICS AND EFFECTS THAT MAY BE DIFFICULT TO MODEL VIA FIRST-PRINCIPLES METHODS. 49

FIG. 2.3 GRAPH OF THE COMPUTATIONAL COST VS ACCURACY FOR VARIOUS POSSIBLE PLASMA PHYSICS MODELS. THE KINETIC SIMULATIONS REPRESENT FULL ORDER MODELS (FOMS), WHEREAS FLUID MODELS AND ML/DD MODELS ARE REDUCED-ORDER MODELS (ROMS) 52

FIG. 2.4 SCHEMATIC REPRESENTATION OF AN AUTOENCODER NETWORK. THE ENCODER ($\phi(x)$) REDUCES THE DIMENSIONALITY OF THE DATA, FINDING THE LATENT SPACE VARIABLES BELONGING TO A REDUCED SUBSPACE 53

FIG. 2.5 A SCHEMATIC REPRESENTATION OF DEEP KOOPMAN NETWORKS FOR THE SIMULTANEOUS DISCOVERY OF THE LINEAR EMBEDDING OF NONLINEAR DYNAMICS (KOOPMAN TRANSFORMATION) AND THE ASSOCIATED TIME-STEPPING OPERATOR K. . 56

FIG. 2.6 SCHEMATIC VISUALIZATION OF THE SINDY ALGORITHM AS THE DYNAMICS DISCOVERY METHOD IN THE LATENT SPACE OF AN AUTOENCODER. 58

FIG. 2.7 SCHEMATIC VISUALIZATION OF THE SHRED ARCHITECTURE AS APPLIED TO DATA FROM THE SIMULATION OF A 2D PLASMA DISCHARGE REPRESENTATIVE OF A RADIAL-AZIMUTHAL CROSS-SECTION OF A HALL THRUSTER. 59

FIG. 3.1 SCHEMATICS OF THE COORDINATE SYSTEM, THE COMPUTATIONAL PLANE, AND THE APPLIED CONSTANT ELECTRIC AND MAGNETIC FIELDS CORRESPONDING TO THE RADIAL-AZIMUTHAL TEST-CASE SIMULATIONS. 72

FIG. 3.2 THE DISTRIBUTION OF (A) NORMALIZED SINGULAR VALUES AND (B) NORMALIZED CUMULATIVE SUM OF THE FIRST r SINGULAR VALUES IN SVD OF AXIAL ELECTRON CURRENT DENSITY DATA. 74

FIG. 3.3 VARIATION VS RANK NUMBER (r) OF THE AVERAGE LOSS FACTOR BETWEEN THE GROUND-TRUTH SIMULATION DATA (AXIAL ELECTRON CURRENT DENSITY) OF THE RADIAL-AZIMUTHAL BASELINE TEST CASE AND THE RECONSTRUCTED DATA FROM SVD ... 75

FIG. 3.4 TIME EVOLUTION OF THE SPATIALLY AVERAGED AXIAL ELECTRON CURRENT DENSITY FROM THE QUASI-2D PIC SIMULATION OF THE RADIAL-AZIMUTHAL TEST CASE WITH THE BASELINE CONDITIONS (SOLID BLACK LINES) AND THE RECONSTRUCTED SIGNAL FROM THE SVD (DOTTED RED LINES) USING DIFFERENT NUMBER OF RANKS AND TRAINING-TO-TEST DATA RATIOS 75

FIG. 3.5 COMPARISON OF THE RECONSTRUCTED 2D SNAPSHOTS OF THE AXIAL ELECTRON CURRENT DENSITY FROM SVD USING DIFFERENT NUMBER OF RANKS AND Rt RATIOS AGAINST THE CORRESPONDING GROUND-TRUTH SNAPSHOT (LAST COLUMN) FROM THE QUASI-2D PIC SIMULATION OF THE RADIAL-AZIMUTHAL BASELINE TEST CASE. 76

FIG. 3.6 COMPARISON OF CONTRIBUTION OF THE REMAINING SVD MODES OF THE AXIAL ELECTRON CURRENT DENSITY WITH $Rt = 0.5$ (CORRESPONDING TO MIDDLE ROW OF FIG. 3.5), WHICH ARE NOT USED FOR RECONSTRUCTION OF THE DATA AT EACH TRUNCATION RANK IN THE RADIAL-AZIMUTHAL BASELINE TEST CASE. THE SCALE OF THE FIGURES IS NORMALIZED. 77

FIG. 3.7 VISUALIZATION OF SEVEN DISTINCT SVD MODES OF THE GROUND-TRUTH DATA FROM THE RADIAL-AZIMUTHAL BASELINE SIMULATION WHEN USING r = 100 AND Rt = 0.5 FOR THE SVD 77

FIG. 3.8 THE DISTRIBUTION OF (A) NORMALIZED SINGULAR VALUES AND (B) NORMALIZED CUMULATIVE SUM OF THE FIRST r SINGULAR VALUES IN THE SVD OF THE NORMALIZED AXIAL ELECTRON CURRENT DENSITY DATA WITH ADDED GAUSSIAN NOISE OF DIFFERENT STANDARD DEVIATIONS. 78

FIG. 3.9 COMPARISON OF THE RECONSTRUCTED 2D SNAPSHOTS OF THE NORMALIZED AXIAL ELECTRON CURRENT DENSITY FROM THE SVD USING DIFFERENT NUMBER OF RANKS AGAINST THE GROUND-TRUTH SNAPSHOT (LAST COLUMN) CORRESPONDING TO SIGNAL FROM THE QUASI-2D PIC SIMULATION WITH DIFFERENT LEVELS OF ADDED GAUSSIAN NOISE. 78

FIG. 3.10 TIME EVOLUTION OF THE SPATIALLY AVERAGED Jey FOR THE RADIAL-AZIMUTHAL TEST CASE WITH THE BASELINE CONDITIONS. THE ROWS FROM TOP TO BOTTOM CORRESPOND TO DIFFERENT LENGTHS OF THE TRAINING SIGNAL (OR, EQUIVALENTLY, NUMBER OF SNAPSHOTS). THE LEFT AND RIGHT COLUMNS, RESPECTIVELY, CORRESPOND TO TWO DIFFERENT DMD RANKS OF 75 AND 150. GROUND-TRUTH TRACES FROM THE PIC SIMULATION ARE ILLUSTRATED AS SOLID BLACK LINES, SOLID RED LINES SHOW THE OPT-DMD PREDICTED TRACES, AND THE SOLID BLUE LINES ARE THE PREDICTED TRACES FROM THE EXACT DMD. 79

FIG. 3.11 DISTRIBUTION OF THE EIGENVALUES OVER THE COMPLEX FREQUENCY PLANE FROM THE EXACT DMD AND THE OPT-DMD ALGORITHMS 80

FIG. 3.12 TIME EVOLUTION OF THE SPATIALLY AVERAGED AXIAL ELECTRON CURRENT DENSITY FOR THE RADIAL-AZIMUTHAL BASELINE TEST CASE FROM THE PIC SIMULATION (SOLID BLACK LINE) AND THE OPT-DMD MODEL (SOLID RED LINE) WITH VARIOUS NUMBER OF DMD RANKS. 81

FIG. 3.13 VISUALIZATION OF A SAMPLE RECONSTRUCTED SNAPSHOT OF ELECTRON AXIAL CURRENT DENSITY FROM THE RADIAL-AZIMUTHAL BASELINE TEST CASE USING THE OPT-DMD MODEL WITH DIFFERENT NUMBER OF RANKS. 81

FIG. 3.14 VARIATION VS RANK NUMBER OF THE AVERAGE LOSS FACTOR (EQ. 3.25) BETWEEN THE GROUND-TRUTH Jey DATA FROM THE PIC SIMULATION OF THE RADIAL-AZIMUTHAL BASELINE TEST CASE AND THE RECONSTRUCTED Jey DATA FROM THE OPT-DMD MODEL. 82

FIG. 3.15 COMPARISON OF THE TEMPORAL FFT SPECTRA OF THE AZIMUTHAL ELECTRIC FIELD FOR DIFFERENT RADIAL-AZIMUTHAL TEST CASES AGAINST THE DISTRIBUTION OF THE DMD MODES' FREQUENCY 83

FIG. 3.16 VISUALIZATION OF THE DOMINANT DMD MODES OF THE AZIMUTHAL ELECTRIC FIELD DATA FOR SEVERAL RADIAL-AZIMUTHAL TEST CASES. 84

FIG. 3.17 COMPARISON OF THE TEMPORAL FFT SPECTRA OF THE AXIAL ELECTRON CURRENT DENSITY FOR DIFFERENT RADIAL-AZIMUTHAL TEST CASES AGAINST THE DISTRIBUTION OF THE DMD MODES' FREQUENCY 85

FIG. 3.18 VISUALIZATION OF THE DOMINANT DMD MODES OF THE AXIAL ELECTRON CURRENT DENSITY DATA FOR SEVERAL RADIAL-AZIMUTHAL TEST CASES 86

FIG. 3.19 (A) SPATIAL FFT SPECTRUM, AND (B) THE TEMPORAL FFT SPECTRUM OF THE AZIMUTHAL ELECTRIC FIELD FOR THE RADIAL-AZIMUTHAL CASE WITH Ex = 50,000. IN PLOT (B), THE DISTRIBUTION OF THE FREQUENCY OF THE DMD MODES IS COMPARED AGAINST THE TEMPORAL FFT SPECTRUM. 87

FIG. 3.20 VISUALIZATION OF THE DOMINANT DMD MODES OF THE AZIMUTHAL ELECTRIC FIELD DATA FOR A RADIAL-AZIMUTHAL TEST CASE WITH $Ey = 50{,}000$ 88

FIG. 3.21 EFFECT OF THE NUMBER OF RANKS ON THE DMD FREQUENCY SPECTRA AND ON THE SHAPE OF THE SAMPLE DOMINANT FREQUENCY MODES 89

FIG. 3.22 DEMONSTRATION OF THE EFFECT OF DMD RANK INFLATION, AND THE SIMILARITY OF THE SHAPE OF THE DMD MODES AT CLOSE FREQUENCIES. 90

FIG. 4.1 (A) SCHEMATICS OF THE COORDINATE SYSTEM, THE COMPUTATIONAL DOMAIN, AND THE APPLIED ELECTRIC AND MAGNETIC FIELDS CORRESPONDING TO THE AXIAL-AZIMUTHAL TEST-CASE SIMULATIONS. (B) AXIAL PROFILES OF THE IMPOSED MAGNETIC FIELD AND THE IONIZATION SOURCE 94

FIG. 4.2 SCHEMATIC OF THE SETUP OF THE PENNING DISCHARGE SIMULATIONS. 96

FIG. 4.3 TIME EVOLUTION OF THE SPATIALLY AVERAGED AZIMUTHAL ELECTRIC FIELD FOR THE AXIAL-AZIMUTHAL TEST CASES REPRESENTING FOUR DIFFERENT VALUES OF THE TOTAL ION CURRENT DENSITY 100

FIG. 4.4 (**LEFT COLUMN**) TIME EVOLUTION OF THE SPATIALLY AVERAGED AZIMUTHAL ELECTRIC FIELD FOR THE AXIAL-AZIMUTHAL TEST CASE WITH J = 50 AND TWO DIFFERENT LENGTHS OF THE TRAINING WINDOW: (A) LARGER TRAINING SET OF $6 - 24\ \mu s$, (B) SMALLER TRAINING SET OF $6 - 17\ \mu s$. (**RIGHT COLUMN**) COMPARISON BETWEEN THE 2D SNAPSHOTS OF Ez FROM THE GROUND-TRUTH PIC SIMULATION AND THE OPT-DMD MODEL OBTAINED WITH THE LARGER AND SMALLER TRAINING WINDOWS. 101

FIG. 4.5 TIME EVOLUTION OF THE PLASMA PROPERTIES FROM THE PENNING DISCHARGE TEST CASE: (A) ION NUMBER DENSITY AT THE LOCATION OF THE DENSITY PROBE, (B) SPATIALLY AVERAGED RADIAL ELECTRIC FIELD 102

FIG. 4.6 COMPARISON BETWEEN THE 2D SNAPSHOTS OF PLASMA PROPERTIES FROM THE GROUND-TRUTH QUASI-2D SIMULATION OF THE PENNING PROBLEM AND THE OPT-DMD MODELS AT FOUR TIME INSTANTS WITHIN THE SHARED TEST WINDOW FOR BOTH PLASMA PROPERTIES (360 TO 500 $\boldsymbol{\mu}$S); (A) ION NUMBER DENSITY, (B) RADIAL ELECTRIC FIELD. 103

FIG. 4.7 TIME EVOLUTION OF THE SPATIALLY AVERAGED PLASMA PROPERTIES FOR TWO TEST CASES WITH THE AXIAL ELECTRIC FIELD VALUES OF $E = 0.5E0$ (**LEFT COLUMN**), AND $E = 5E0$ (**RIGHT COLUMN**). FIRST-ROW PLOTS REPRESENT THE AZIMUTHAL ELECTRIC FIELD SIGNAL, SECOND-ROW PLOTS SHOW THE AXIAL ELECTRON CURRENT DENSITY, AND THIRD-ROW PLOTS CORRESPOND TO THE RADIAL ELECTRON CURRENT DENSITY 104

FIG. 4.8 COMPARISON OF THE 2D SNAPSHOTS OF PLASMA PROPERTIES FROM THE GROUND-TRUTH RADIAL-AZIMUTHAL SIMULATIONS AND THE OPT-DMD MODELS FOR TWO VALUES OF THE AXIAL ELECTRIC FIELD AT THREE RANDOMLY SELECTED TIME INSTANTS WITHIN THE TEST WINDOW ($\sim 22 - 30\ \mu s$). (**LEFT COLUMN**) AZIMUTHAL ELECTRIC FIELD, (**RIGHT COLUMN**) AXIAL ELECTRON CURRENT DENSITY 105

FIG. 4.9 (**LEFT COLUMN**) TIME EVOLUTION OF THE SPATIALLY AVERAGED PLASMA PROPERTIES FOR THE RADIAL-AZIMUTHAL TEST CASE WITH THE RADIAL MAGNETIC FIELD INTENSITY OF $Bx = 1.5B0$. (**MIDDLE COLUMN**) COMPARISON OF THE 2D SNAPSHOTS OF THE Ez FROM THE GROUND-TRUTH SIMULATION AND THE OPT-DMD MODEL AT THREE RANDOMLY SELECTED TIME INSTANTS WITHIN THE TEST WINDOW ($\sim 22 - 30\ \mu s$). (**RIGHT COLUMN**) COMPARISON OF THE 2D SNAPSHOTS OF THE Jey FROM THE GROUND-TRUTH SIMULATION AND THE OPT-DMD MODEL AT THREE RANDOMLY SELECTED TIME INSTANTS WITHIN THE TEST WINDOW. 106

FIG. 4.10 SPATIALLY AVERAGED AXIAL ELECTRON CURRENT DENSITY SIGNALS FOR THE RADIAL-AZIMUTHAL TEST CASES WITH VARIOUS SETUP PARAMETERS (PLASMA CONDITIONS) 107

FIG. 4.11 COMPARISON BETWEEN THE 2D SNAPSHOTS OF THE AXIAL ELECTRON CURRENT DENSITY FROM THE GROUND-TRUTH PIC SIMULATIONS AND THE OPT-DMD MODELS FOR THREE SETS OF SETUP PARAMETERS (PLASMA CONDITIONS). 108

FIG. 4.12 COMPARISON BETWEEN THE 2D SNAPSHOTS OF THE AXIAL ELECTRON CURRENT DENSITY FROM THE GROUND-TRUTH PIC SIMULATIONS AND THE OPT-DMD MODELS FOR THREE SETS OF SETUP PARAMETERS 108

FIG. 4.13 OPT-DMD MODEL WITH THE LOW-RANK APPROXIMATION (r = 55) TRAINED OVER A SHORT-DURATION DATASET (t = 5-22 μs). (A) TIME EVOLUTIONS OF THE SPATIALLY AVERAGED AXIAL ELECTRON CURRENT DENSITY FROM THE PIC SIMULATION (SOLID BLACK LINE) AND FROM THE OPT-DMD MODEL (SOLID RED LINE). (B) ZOOMED VIEW ON PLOT (A) OVER THE TIME WINDOW OF 5-60 μs. (C) ZOOMED VIEW ON PLOT (A) OVER THE TIME WINDOW OF 60-135 μs. 109

FIG. 4.14 OPT-DMD MODEL WITH THE MODERATE-RANK APPROXIMATION (r = 150) TRAINED OVER A SHORT-DURATION DATASET (t = 5-22 μs). (A) TIME EVOLUTION OF THE SPATIALLY AVERAGED AXIAL ELECTRON CURRENT DENSITY FROM THE PIC SIMULATION (SOLID BLACK LINE) AND FROM THE OPT-DMD MODEL (SOLID RED LINE). (B) ZOOMED VIEW ON PLOT (A) OVER THE TIME WINDOW OF 5-60 μs. (C) ZOOMED VIEW ON PLOT (A) OVER THE TIME WINDOW OF 60-135 μs. 110

FIG. 4.15 OPT-DMD MODEL WITH THE HIGH-RANK APPROXIMATION (r = 400) TRAINED OVER A LONG-DURATION DATASET (t = 5-98 μs). (A) TIME EVOLUTION OF THE SPATIALLY AVERAGED AXIAL ELECTRON CURRENT DENSITY FROM THE PIC SIMULATION (SOLID BLACK LINE) AND FROM THE OPT-DMD MODEL (SOLID RED LINE). (B) ZOOMED VIEW ON PLOT (A) OVER THE TIME WINDOW OF 5-60 μs. (C) ZOOMED VIEW ON PLOT (A) OVER THE TIME WINDOW OF 60-135 μs. 110

FIG. 4.16 TIME EVOLUTION OF THE LOSS FACTOR (FROM EQ. 4.1) BETWEEN THE PREDICTED AXIAL ELECTRON CURRENT DENSITY FROM THE OPT-DMD MODELS WITH DIFFERENT VALUES OF THE TRUNCATION RANK (r) AND THE GROUND-TRUTH PIC SIMULATION. DASHED BLUE LINES INDICATE THE TRAINING END TIME..... 111

FIG. 4.17 COMPARISON BETWEEN THE 2D SNAPSHOTS OF THE AXIAL ELECTRON CURRENT DENSITY FROM THE GROUND-TRUTH PIC SIMULATION AND THE OPT-DMD MODEL WITH THE RANK OF r = 55 OVER SEVERAL TEST-WINDOW TIME INSTANTS..... 112

FIG. 4.18 COMPARISON BETWEEN THE 2D SNAPSHOTS OF THE AXIAL ELECTRON CURRENT DENSITY FROM THE GROUND-TRUTH PIC SIMULATION AND THE OPT-DMD MODEL WITH THE RANK OF r = 400 OVER SEVERAL TEST-WINDOW TIME INSTANTS 112

FIG. 4.19 TIME EVOLUTIONS OF SEVERAL SPATIALLY AVERAGED (OVER THE ENTIRE DOMAIN) PLASMA PROPERTIES; (A) ELECTRONS' AXIAL DRIFT VELOCITY, (B) ELECTRONS' RADIAL DRIFT VELOCITY, (C) ELECTRONS' AZIMUTHAL DRIFT VELOCITY, (D) RADIAL ELECTRON TEMPERATURE, (E) AZIMUTHAL ELECTRON TEMPERATURE, (F) ELECTRON NUMBER DENSITY, (G) RADIAL ELECTRIC FIELD, (H) AZIMUTHAL ELECTRIC FIELD 114

FIG. 4.20 TIME EVOLUTIONS OF SEVERAL PLASMA PROPERTIES; (**LEFT COLUMN**) ZOOMED-IN VIEWS ON THE TEMPORAL VARIATION PLOTS OF THE SPATIALLY AVERAGED PLASMA PROPERTIES FROM FIG. 4.19. THE ZOOMED-IN PLOTS ARE OVER THE TIME INTERVAL OF 70-110 μs (20 μs BEFORE AND AFTER THE TRAINING END TIME) FOR ALL THE PLASMA PROPERTIES, EXCEPT FOR THE Ex AND THE Ez. FOR THESE TWO PROPERTIES, THE ZOOMED-IN TIMEFRAMES ARE, RESPECTIVELY, 75-105 μs, AND 85-95 μs. (**RIGHT COLUMN**) TIME VARIATIONS OF THE LOCAL VALUES OF THE PLASMA PROPERTIES INCLUDED WITHIN THE MULTI-PROPERTY PLASMA STATE VECTOR 115

FIG. 4.21 TIME EVOLUTIONS OF THE LOSS FACTOR (FROM EQ. 4.1) BETWEEN THE SIMULTANEOUSLY PREDICTED PLASMA PROPERTIES FROM THE OPT-DMD MODEL AND THE GROUND-TRUTH DATA FROM THE PIC SIMULATION 116

FIG. 4.22 COMPARISON BETWEEN THE 2D SNAPSHOTS OF SEVERAL PLASMA PROPERTIES FROM THE GROUND-TRUTH RADIAL-AZIMUTHAL PIC SIMULATION AND THE OPT-DMD MODEL AT THREE RANDOMLY SELECTED TIME INSTANTS WITHIN THE TEST

window; (a) azimuthal electric field, (b) electron number density, (c) radial electron temperature, (d) electrons' radial drift velocity, (e) electrons' axial drift velocity, (f) electrons' azimuthal drift velocity. 116

Fig. 4.23 Comparison between the OPT-DMD's prediction and the ground truth data of the normalized axial electron current density with different levels of added Gaussian noise from the radial-azimuthal simulation case with E = 0.5$E0$ in terms of (a) the time evolution of the spatially averaged signal, and (b) reconstruction of a sample 2D snapshot 117

Fig. 5.1 Schematic of a 1D spatiotemporal stencil of computational nodes corresponding to Phi Method 122

Fig. 5.2 Schematic of the *training phase* of the Phi Method algorithm; matrix X contains the state vectors at time step $k+1$, matrix Θ represents the library of candidate terms, and matrix Φ is the coefficient matrix. 122

Fig. 5.3 Comparison between the learned trajectory of a Lorenz attractor using Phi Method and the ground-truth trajectory obtained from the solution of the system of equations in Eq. 5.9 along (a) $y-x$, (b) $z-x$, and (c) $z-y$ planes. 126

Fig. 5.4 Comparison of the linear and nonlinear ROMs from Phi Method and the ROM from OPT-DMD against the ground-truth simulation for the fluid flow test case; time evolutions of (a) spatially averaged normalized vorticity, (b) local value of the normalized vorticity at the mid-location within the simulation domain, and (c) the loss factor over the entire domain 129

Fig. 5.5 Comparison of the linear and the nonlinear ROMs from Phi Method and the ROM from OPT-DMD against the ground-truth simulation for the fluid flow test case; snapshots of the normalized vorticity field at three different time instances within the test interval. 130

Fig. 5.6 Comparison of the linear and the nonlinear ROMs from Phi Method and the ROM from OPT-DMD against the ground-truth simulation for the fluid flow test case; plots of the absolute difference between the true and the predicted normalized vorticity snapshots at three different time instants during the test interval. 131

Fig. 5.7 Comparison of the linear and nonlinear Phi Method models trained on the transient-state data against the ground-truth simulation for the fluid flow test case; time evolutions of (a) spatially averaged normalized vorticity, (b) local value of the normalized vorticity at the mid-location within the simulation domain, and (c) loss factor over the entire domain 132

Fig. 5.8 Rearranged normalized representations of various coefficient matrices (operators, Φ) from the Phi method's application to the fluid flow test case; (a) visualization of the $\Phi\mathrm{L}$ matrix, (b) visualization of the $\Phi\mathrm{NL}$ matrix, and (c) visualization of the $\Phi\mathrm{NL},\mathrm{ET}$ matrix that represents a larger library of terms compared to the ROM corresponding to $\Phi\mathrm{NL}$. (**left column**) models' training performed on the steady-state data, (**right column**) models' training performed on the transient-state data 133

Fig. 5.9 Comparison of the predictions from the Phi Method (**left column**) and the OPT-DMD (**right column**) ROMs against the ground-truth data for plasma test case; time evolutions of the local values at the mid-location within the simulation domain of (a) normalized electron number density, (b) normalized electrons' azimuthal drift velocity, (c) normalized azimuthal electric field, and (d) normalized electrons' axial drift velocity. Dashed blue lines indicate the end of the training interval. 137

Fig. 5.10 Comparison of the predictions from the Phi Method and the OPT-DMD ROMs against the ground-truth data for the plasma test case; spatiotemporal evolution plots of (a) normalized electron number density, (b)

NORMALIZED ELECTRONS' AZIMUTHAL DRIFT VELOCITY, (C) NORMALIZED AZIMUTHAL ELECTRIC FIELD, AND (D) NORMALIZED ELECTRONS' AXIAL DRIFT VELOCITY..... 138

FIG. 5.11 COMPARISON OF THE ROMS FROM PHI METHOD AND OPT-DMD FOR THE PLASMA TEST CASE; TIME EVOLUTIONS OF THE LOSS FACTOR CALCULATED OVER THE ENTIRE DOMAIN CORRESPONDING TO (A) NORMALIZED ELECTRON NUMBER DENSITY, (B) NORMALIZED ELECTRONS' AZIMUTHAL DRIFT VELOCITY, (C) NORMALIZED AZIMUTHAL ELECTRIC FIELD, AND (D) NORMALIZED ELECTRONS' AXIAL DRIFT VELOCITY..... 140

FIG. 5.12 COMPARISON OF THE PHI METHOD ROMS FOR THE NORMALIZED ELECTRONS' AXIAL CURRENT DENSITY WHEN REGRESSING OVER A SMALL AND A LARGE LIBRARY OF TERMS IN THE PLASMA TEST CASE; (A) &(B) TIME EVOLUTIONS OF THE LOCAL VALUES OF THE Jex AT THE MID-LOCATION WITHIN THE DOMAIN FROM THE GROUND-TRUTH SIMULATION AND FROM THE PHI METHOD MODELS, PLOT (A) WITH A SMALL LIBRARY, PLOT (B) WITH A LARGE LIBRARY OF TERMS. (C) TIME EVOLUTIONS OF THE LOSS FACTOR CALCULATED OVER THE ENTIRE DOMAIN ASSOCIATED WITH EACH ROM..... 141

FIG. 5.13 REARRANGED NORMALIZED REPRESENTATIONS OF THE COEFFICIENT MATRICES (Φ) FROM THE PHI METHOD'S APPLICATION TO THE Jex DATA OF THE PLASMA TEST CASE; (A) $\Phi\mathrm{SL}$ VISUALIZATION (SMALL LIBRARY OF TERMS), (B) $\Phi\mathrm{LL}$ VISUALIZATION (LARGE LIBRARY OF TERMS)..... 142

FIG. 5.14 COMPARISON BETWEEN THE PREDICTIONS OF THE PHI METHOD AND THE OPT-DMD MODELS WITH RESPECT TO THE GROUND-TRUTH DATA IN THE PRESENCE OF A GAUSSIAN WHITE NOISE IN THE FLUID FLOW TEST CASE; TIME EVOLUTIONS OF (A) SPATIALLY AVERAGED NORMALIZED VORTICITY, (B) LOCAL VALUE OF THE VORTICITY AT THE MID-LOCATION WITHIN THE SIMULATION DOMAIN. 143

FIG. 5.15 COMPARISON BETWEEN THE PREDICTIONS OF THE PHI METHOD AND THE OPT-DMD MODELS WITH RESPECT TO THE GROUND-TRUTH DATA IN THE PRESENCE OF A GAUSSIAN WHITE NOISE IN THE FLUID FLOW TEST CASE; SNAPSHOTS OF THE NORMALIZED VORTICITY FIELD AT THREE DIFFERENT TIME INSTANTS DURING THE TEST INTERVAL. 144

FIG. 5.16 COMPARISON BETWEEN THE PHI METHOD AND THE OPT-DMD MODELS IN THE PRESENCE OF A GAUSSIAN WHITE NOISE IN THE FLUID FLOW TEST CASE; PLOTS OF ABSOLUTE DIFFERENCE BETWEEN THE TRUE AND THE PREDICTED NORMALIZED VORTICITY SNAPSHOTS AT THREE DIFFERENT TIME INSTANTS DURING THE TEST INTERVAL. 144

FIG. 5.17 COMPARISON OF THE PREDICTIONS FROM THE PHI METHOD (**LEFT COLUMN**) AND THE OPT-DMD (**RIGHT COLUMN**) ROMS AGAINST THE GROUND-TRUTH DATA IN THE PRESENCE OF A GAUSSIAN WHITE NOISE IN THE PLASMA TEST CASE; TIME EVOLUTION OF THE LOCAL VALUES AT THE MID-LOCATION WITHIN THE SIMULATION DOMAIN OF (A) NORMALIZED ELECTRON NUMBER DENSITY, (B) NORMALIZED ELECTRONS' AZIMUTHAL DRIFT VELOCITY, (C) NORMALIZED AZIMUTHAL ELECTRIC FIELD, AND (D) NORMALIZED ELECTRONS' AXIAL DRIFT VELOCITY..... 145

FIG. 5.18 COMPARISON OF THE PREDICTIONS FROM THE PHI METHOD AND THE OPT-DMD ROMS AGAINST THE GROUND-TRUTH DATA IN THE PRESENCE OF A GAUSSIAN WHITE NOISE IN THE PLASMA TEST CASE; SPATIOTEMPORAL EVOLUTION PLOTS OF (A) NORMALIZED ELECTRON NUMBER DENSITY, (B) NORMALIZED ELECTRONS' AZIMUTHAL DRIFT VELOCITY, (C) NORMALIZED AZIMUTHAL ELECTRIC FIELD, AND (D) NORMALIZED ELECTRONS' AXIAL DRIFT VELOCITY. 146

FIG. 6.1 COMPARISON OF THE PREDICTIONS FROM THE PARAMETRIC PHI METHOD AND THE PARAMETRIC OPT-DMD ROMS AGAINST THE GROUND-TRUTH DATA FOR THE FLUID FLOW TEST CASE ACROSS THE TEST REYNOLDS NUMBERS BELONGING TO DS1; TIME EVOLUTIONS OF THE SPATIALLY AVERAGED NORMALIZED VORTICITY (**LEFT COLUMN**), AND LOCAL VALUES OF THE NORMALIZED VORTICITY AT THE MID-LOCATION WITHIN THE SIMULATION DOMAIN (**MIDDLE COLUMN**). (**RIGHT COLUMN**) TIME EVOLUTIONS OF THE LOSS FACTOR CALCULATED OVER THE ENTIRE DOMAIN. 155

FIG. 6.2 COMPARISON OF THE PREDICTIONS FROM THE PARAMETRIC PHI METHOD AND THE PARAMETRIC OPT-DMD ROMS AGAINST THE GROUND-TRUTH DATA FOR THE FLUID FLOW TEST CASE ACROSS THE TEST REYNOLDS NUMBERS BELONGING TO DS1; 2D SNAPSHOTS OF THE NORMALIZED VORTICITY FIELD AT THREE DIFFERENT SAMPLE TIME INSTANTS FOR THE TEST PARAMETERS OF (A) Re = 150, (B) Re = 250, (C) Re = 350, AND (D) Re = 450. 155

FIG. 6.3 COMPARISON OF THE PREDICTIONS FROM THE PARAMETRIC PHI METHOD AND THE PARAMETRIC OPT-DMD ROMS FOR THE FLUID FLOW TEST CASE; ABSOLUTE-DIFFERENCE PLOTS BETWEEN THE TRUE AND THE PREDICTED NORMALIZED VORTICITY SNAPSHOTS AT THREE TIME INSTANTS FOR THE TEST PARAMETERS OF DS1, NAMELY, (A) Re = 150, (B) Re = 250, (C) Re = 350, AND (D) Re = 450. 156

FIG. 6.4 VISUALIZATION OF THE FIRST FOUR INTERPOLATED DMD MODES ASSOCIATED WITH THE PARAMETRIC OPT-DMD FOR THE FLUID FLOW TEST CASE OVER THE TEST REYNOLDS NUMBERS OF DS1. THE INTERPOLATED MODES ARE COMPARED AGAINST THE CORRESPONDING TRUE DMD MODES. (A) Re = 150, (B) Re = 250, (C) Re = 350, AND (D) Re = 450. 157

FIG. 6.5 COMPARISON OF THE PREDICTIONS FROM THE PARAMETRIC PHI METHOD AND THE PARAMETRIC OPT-DMD ROMS AGAINST THE GROUND-TRUTH DATA FOR THE FLUID FLOW TEST CASE ACROSS THE TEST REYNOLDS NUMBERS BELONGING TO DS2; TIME EVOLUTIONS OF THE SPATIALLY AVERAGED NORMALIZED VORTICITY (**LEFT COLUMN**), AND LOCAL VALUES OF THE NORMALIZED VORTICITY AT THE MID-LOCATION WITHIN THE SIMULATION DOMAIN (**MIDDLE COLUMN**). (**RIGHT COLUMN**) TIME EVOLUTIONS OF THE LOSS FACTOR CALCULATED OVER THE ENTIRE DOMAIN. 158

FIG. 6.6 COMPARISON OF THE PREDICTIONS FROM THE PARAMETRIC PHI METHOD AND THE PARAMETRIC OPT-DMD ROMS AGAINST THE GROUND-TRUTH DATA FOR THE FLUID FLOW TEST CASE ACROSS THE TEST REYNOLDS NUMBERS BELONGING TO DS2; 2D SNAPSHOTS OF THE NORMALIZED VORTICITY FIELD AT THREE DIFFERENT TIME INSTANTS FOR THE TEST PARAMETERS OF (A) $Re = 175$, (B) $Re = 275$ AND (C) $Re = 425$. 158

FIG. 6.7 VISUALIZATION OF THE FIRST FOUR INTERPOLATED DMD MODES ASSOCIATED WITH THE PARAMETRIC OPT-DMD FOR THE FLUID FLOW TEST CASE OVER THE TEST REYNOLDS NUMBERS OF DS2. THE INTERPOLATED MODES ARE COMPARED AGAINST THE CORRESPONDING TRUE DMD MODES. (A) $Re = 175$, (B) $Re = 275$ AND (C) $Re = 425$. 159

FIG. 6.8 COMPARISON OF THE PREDICTIONS FROM THE PARAMETRIC PHI METHOD AND THE PARAMETRIC OPT-DMD ROMS FOR THE FLUID FLOW TEST CASE; PLOTS OF THE TIME-AVERAGED (MEAN) VALUES OF THE LOSS FACTOR VS. THE REYNOLDS NUMBER FROM THE ROMS TRAINED ON (A) DATASET 1 (DS1), AND (B) DATASET 2 (DS2) 160

FIG. 6.9 REARRANGED NORMALIZED REPRESENTATIONS OF THE COEFFICIENTS MATRICES OF THE PARAMETRIC PHI METHOD ROM FOR THE FLUID FLOW TEST CASE; (A) FROM TRAINING ON DS1, AND (B) FROM TRAINING ON DS2. 161

FIG. 6.10 REARRANGED NORMALIZED REPRESENTATIONS OF THE CHARACTERISTIC MATRICES CORRESPONDING TO THE ENSEMBLE PHI METHOD ROM FOR THE FLUID FLOW TEST CASE. (A) ΦENS MATRIX (MEAN OF THE Φ MATRICES OF THE INDIVIDUAL CONSTITUENT MODELS), (B) NORMALIZED (BY THE ABSOLUTE VALUES OF THE ΦENS MATRIX) STANDARD DEVIATIONS OF THE COEFFICIENTS OF THE LIBRARY TERMS ACROSS THE INDIVIDUAL MODELS 162

FIG. 6.11 PREDICTIONS OF THE PARAMETRIC PHI METHOD ROM (PM1) FOR THE PLASMA DISCHARGE TEST CASE COMPARED AGAINST THE GROUND-TRUTH DATA; TIME EVOLUTIONS OF THE LOCAL VALUE OF THE NORMALIZED ELECTRONS' AXIAL FLUX DENSITY AT THE MID-LOCATION WITHIN THE SIMULATION DOMAIN FOR VARIOUS TEST VALUES OF THE AXIAL ELECTRIC FIELD (Ex, **LEFT COLUMN**) AND FOR VARIOUS TEST VALUES OF THE RADIAL MAGNETIC FIELD (By, **RIGHT COLUMN**). 166

FIG. 6.12 PREDICTIONS OF THE PARAMETRIC PHI METHOD ROM (PM1) FOR THE PLASMA DISCHARGE TEST CASE COMPARED AGAINST THE GROUND-TRUTH DATA; SPATIOTEMPORAL MAPS OF THE NORMALIZED ELECTRONS' AXIAL FLUX DENSITY FOR THE TEST VALUES OF THE AXIAL ELECTRIC FIELD (Ex) 167

FIG. 6.13 PREDICTIONS OF THE PARAMETRIC PHI METHOD ROM (PM1) FOR THE PLASMA DISCHARGE TEST CASE COMPARED AGAINST THE GROUND-TRUTH DATA; SPATIOTEMPORAL MAPS OF THE NORMALIZED ELECTRONS' AXIAL FLUX DENSITY FOR THE TEST VALUES OF THE RADIAL MAGNETIC FIELD (By) 168

FIG. 6.14 PREDICTIONS OF THE PM1 (**LEFT COLUMN**) AND THE EM1 (**RIGHT COLUMN**) FOR THE PLASMA DISCHARGE TEST CASE; SPATIALLY AND TEMPORALLY AVERAGED NORMALIZED ELECTRONS' AXIAL FLUX DENSITY VALUES FROM THE ROMS (PREDICTIONS) AND THE SIMULATIONS (GROUND-TRUTH) AGAINST: (A) AND (B) THE VARIOUS ELECTRIC FIELD VALUES, (C) AND (D) THE VARIOUS MAGNETIC FIELD VALUES 169

FIG. 6.15 REARRANGED NORMALIZED REPRESENTATIONS OF THE COEFFICIENTS MATRICES (Φ) CORRESPONDING TO THE PARAMETRIC PHI METHOD ROMS (PM1 AND PM2) AND THE ENSEMBLE PHI METHOD ROMS (EM1 AND EM2) FOR THE PLASMA DISCHARGE TEST CASE 170

FIG. 6.16 NORMALIZED STANDARD DEVIATIONS OF THE COEFFICIENTS OF THE LIBRARY TERMS ACROSS THE INDIVIDUAL MODELS CONSTITUTING THE ENSEMBLE ROMS; (A) STANDARD DEVIATIONS ACROSS THE INDIVIDUAL MODELS OF EM1, (B) STANDARD DEVIATIONS ACROSS THE INDIVIDUAL MODELS OF EM2. 171

FIG. 6.17 COMPARISON OF THE LEARNED COEFFICIENTS OF THE TERMS IN EQ. 6.12 FROM THE PARAMETRIC AND THE ENSEMBLE PHI METHOD AGAINST THE RESPECTIVE ANALYTICAL VALUES OF THE COEFFICIENTS FOR THE PLASMA DISCHARGE TEST CASE. PLOTS (A) AND (B) SHOW THE MAGNITUDES OF THE COEFFICIENTS FOR THE PM1/EM1 AND THE PM2/EM2 ROMS, RESPECTIVELY. PLOTS (C) AND (D) SHOW, RESPECTIVELY, THE NORMALIZED ERROR OF THE OBTAINED COEFFICIENTS RELATIVE TO THE ANALYTICAL VALUES FOR THE PM1/EM1 AND THE PM2/EM2 ROMS. 173

FIG. 7.1 COMPARISON OF THE PHI METHOD MODELS FOR THE ELECTRON NUMBER DENSITY IN THE 1D AZIMUTHAL $\mathrm{E} \times \mathrm{B}$ DISCHARGE PROBLEM (SUBSECTION 5.3.1).. 181

List of Tables

TABLE 3.1 SUMMARY OF THE COMPUTATIONAL AND PHYSICAL PARAMETERS USED FOR THE QUASI-2D SIMULATIONS OF THE RADIAL-AZIMUTHAL TEST-CASE PROBLEM. 71

TABLE 3.2 SUMMARY OF THE VALUES OF THE PLASMA CONDITIONS FOR VARIOUS RADIAL-AZIMUTHAL TEST-CASE SIMULATIONS. 73

TABLE 4.1 SUMMARY OF THE VALUES OF J CORRESPONDING TO VARIOUS AXIAL-AZIMUTHAL TEST-CASE SIMULATIONS; THE BASELINE CASE HAS THE J VALUE EQUAL TO THAT OF THE AXIAL-AZIMUTHAL BENCHMARK. 95

TABLE 4.2 SUMMARY OF THE COMPUTATIONAL AND PHYSICAL PARAMETERS USED FOR THE QUASI-2D SIMULATIONS OF THE AXIAL-AZIMUTHAL TEST-CASE PROBLEM 96

TABLE 4.3 SUMMARY OF THE COMPUTATIONAL AND PHYSICAL PARAMETERS USED FOR THE GROUND-TRUTH QUASI-2D PIC SIMULATION OF THE PENNING DISCHARGE TEST-CASE PROBLEM 97

TABLE 4.4 SUMMARY OF THE VALUES OF THE PHYSICAL PARAMETERS (PLASMA CONDITIONS) USED FOR THE RADIAL-AZIMUTHAL TEST-CASE SIMULATIONS 98

TABLE 5.1 THE VALUES OF THE NON-ZERO ENTRIES OF THE Φ MATRIX FROM PHI METHOD COMPARED AGAINST THE RESPECTIVE TRUE VALUES FROM EQ. 5.11. THE CORRESPONDING ERRORS ARE ALSO PROVIDED 125

List of Publications

Select journal articles

1. **Faraji F**, Reza M, Knoll A, Kutz JN, "Dynamic Mode Decomposition for data-driven analysis and reduced-order modelling of E×B plasmas: I. Extraction of spatiotemporally coherent patterns," *J. Phys. D: Appl. Phys*. **57** 065201 (2024) DOI: 10.1088/1361-6463/ad0910.
2. **Faraji F**, Reza M, Knoll A, Kutz JN, "Dynamic Mode Decomposition for data-driven analysis and reduced-order modelling of E×B plasmas: II. Dynamics forecasting," *J. Phys. D: Appl. Phys.* **57** 065202 (2024) DOI: 10.1088/1361-6463/ad0911.
3. **Faraji F**, Reza M, Knoll A, Kutz JN, "Data-driven local operator finding for reduced-order modelling of plasma systems: I. Concept and verifications," *ArXiv preprint*, arXiv:2403.01523v1 (2024) [in review at the time of submission]
4. **Faraji F**, Reza M, Knoll A, Kutz JN, "Data-driven local operator finding for reduced-order modelling of plasma systems: II. Application to parametric dynamics," *ArXiv preprint*, arXiv:2403.01532v1 (2024) [in review at the time of submission]
5. **Faraji F**, Reza M, Knoll A, "Enhancing one-dimensional particle-in-cell simulations to self-consistently resolve instability-induced electron transport in Hall thrusters," *J. Appl. Phys.* **131**, 193302 (2022) DOI: 10.1063/5.0090853.
6. **Faraji F**, Reza M, Knoll A, "Verification of the generalized reduced-order particle-in-cell scheme in a radial-azimuthal E×B plasma configuration," *AIP Advances* **13**, 025315 (2023) DOI: 10.1063/5.0136889.
7. **Faraji F**, Reza M, Knoll A, "Effects of the neutral dynamics model on the particle-in-cell simulations of a Hall thruster plasma discharge," *J. Appl. Phys.* **133**, 213301 (2023) DOI: 10.1063/5.0143292.
8. Reza M, **Faraji F**, Knoll A, "Concept of the generalized reduced-order particle-in-cell scheme and verification in an axial-azimuthal Hall thruster configuration," *J. Phys. D: Appl. Phys.* **56** 175201 (2023) DOI: 10.1088/1361-6463/acbb15.

Select conference papers

1. **Faraji F**, Reza M, Knoll A, "Digital Twins for a 21st-Century Transformation of the Electric Propulsion Sector: Vision 2030," SP2024_046, *In Procs. of Space Propulsion Conference 2024*, Glasgow, Scotland, 20-23 May 2024.
2. **Faraji F**, Reza M, Knoll A, "Digital twins for Electric Propulsion: concept, game- changing potentials, and building blocks", IEPC-2024-256, *In Procs. of 38th International Electric Propulsion Conference,* Toulouse, France, 23-28 June 2024.
3. **Faraji F**, Reza M, Knoll A, "Data-driven dynamics discovery and discrepancy modeling for advancing plasma science and technology," IEPC-2024-502, *In Procs. of 38th International Electric Propulsion Conference,* Toulouse, France, 23-28 June 2024.
4. **Faraji F**, Reza M, Knoll A, "Machine-learning-enabled plasma modeling and prediction," AIAA 2024-2708, *In Procs. of 2024 SciTech Forum conference*, Orlando, Florida (2024) DOI: 10.2514/6.2024-2708.
5. M. Reza, **Faraji F**, Knoll A, "Latest verifications of the reduced-order particle-in-cell scheme: Penning discharge and axial-radial Hall thruster case," AIAA 2024-2712, *In Procs. of 2024 SciTech Forum conference*, Orlando, Florida, 8-12 January 2024. DOI: 10.2514/6.2024-2712.
6. Reza M, **Faraji F**, Knoll A, "Technical Advancements at Imperial Plasma Propulsion Laboratory on the Building Blocks of Digital Twin Technology for Electric Propulsion," SP2024_061, *In Procs. of Space Propulsion Conference*, Glasgow, Scotland, 20-23 May 2024.
7. Reza M, **Faraji F**, Knoll A, Kutz JN, "Shallow Recurrent Decoder architecture with application to super-resolution enhancement of the reduced-order particle-in-cell scheme," IEPC-2024-503, *In Procs. of 38th International Electric Propulsion Conference,* Toulouse, France, 23-28 June 2024.
8. Reza M, **Faraji F**, Knoll A, Kutz JN, "Progress and future directions on predictive data-driven reduced-order modeling for Electric Propulsion Digital Twins," IEPC-2024-501, *In Procs. of 38th International Electric Propulsion Conference,* Toulouse, France (2024)

Chapter 1 Background and problem statement

Plasmas constitute 99 % of the visible matter in the universe, and span across vast length scales, from the cosmological down to the laboratory size. Often regarded as the "fourth state of the matter", plasmas manifest in many natural phenomena (from the Pillars of Creation in the Eagle Nebula to the auroras and the lightning) and play a fundamental role in a myriad of human-made technologies (such as fusion reactors and the plasma propulsion devices for spacecrafts).

Despite the widespread occurrence and applications of plasmas, and decades of rigorous research since the mid-20th century, many unknowns yet persist regarding the intricate and complex plasma behaviors. A primary reason of this enduring challenge is that the necessary predictive and cost-efficient simulation models and analysis toolset are still lacking to enable comprehensive studies of the multiscale, multidimensional, and often multiphysics phenomena in plasmas. The absence of these numerical capabilities restricts our ability to justify and generalize theoretical predictions and experimental observations, further hindering the advancement in plasma science.

The knowledge gap in plasma physics represents a significant shortcoming, as the unresolved questions have great implications on our ability to understand, predict, and control the plasmas across vast scales. Answering the persisting questions will revolutionize the basic science with respect to how, for instance, the stars are born and die, and how the solar flares' plasma interacts with and affects the Earth's magnetosphere environment. The scientific breakthroughs will also have major applied significance. These will, for instance, enable harnessing the power of nuclear fusion and will underpin the transformation and upscaling of spacecraft plasma propulsion technologies toward realizing mankind's ambitions in exploring the solar system and beyond. The quest for predictive plasma models is thus well warranted and crucial in order to further advance plasma science and engineering.

In this Chapter, we start in section 1.1 by taking a close look at plasmas and reviewing some of their fundamental characteristics and properties. This will elucidate why plasmas are notoriously difficult to understand, predict and control. Section 1.1 also discusses some of the most important and tangible applications of plasmas, including the Hall thruster technology. A Hall thruster is not only an industrially important technology for spacecraft propulsion but

is also an optimal proving ground to study plasma processes of broad scientific and applied significance. Taking Hall thrusters as a reference, section 1.2 elaborates on how predictive models of plasma systems can aid in answering the outstanding scientific questions and can as well facilitate and streamline the design and development of plasma technologies. To clarify the reasons due to which predictive plasma models have remained out of reach, the status quo of numerical tools used today for plasma studies is reviewed in section 1.3. Section 1.4 provides an overview of the reduced-order particle-in-cell (PIC) scheme, a novel high-fidelity kinetic simulation approach that drastically reduces the computational cost of kinetic plasma codes, allowing for cost-efficient generation of extensive high-fidelity datasets. Finally, section 1.5 briefly describes how machine-learning (ML)/data-driven (DD) approaches can open novel pathways toward predictive plasma models.

The above introductory discussions illustrate the motivations behind this research, which are summarized in section 1.6. In this same section, the specific objectives of the work are discussed, and what this thesis has done to meet the defined objectives are enumerated.

1.1. Plasmas: universally ubiquitous, challenging to understand, predict and control

1.1.1. What is a plasma and what are its main characteristics?

In simple terms, plasma is an ionized gas, comprised by negatively charged particles – electrons – and positively charged particles – ions. The term “plasma” was coined in 1928 by Irving Langmuir [1]. Plasma is often referred to as the “fourth state of the matter” alongside solids, liquids, and gases. Plasma can be notionally thought of as a superheated matter, i.e., at such elevated temperatures that the electrons are detached from the atoms, thus forming an ionized gas.

In a more scientific sense, plasma is defined as “a *quasi-neutral* gas of charged particles showing *collective* behavior” [2][3]. It is necessary to further clarify this description because it will cast light on some characteristic features of the plasmas. There are two inherent features of plasmas: (1) quasi-neutrality, and (2) collective behavior. The former implies that, within a plasma at equilibrium, the net global charge density is almost zero despite the fact that the plasma is made up of free electrons and ions. The latter feature means that, in the presence of a charge disturbance in the plasma, all the plasma particles respond in such a way to “mask” the potential arising from the charge disturbance, thus, keeping the plasma globally quasi-neutral.

The distance beyond which any charge disturbance in a plasma is masked by the plasma particles, especially the electrons (because of their lighter mass, and hence, higher mobility), is a characteristic length scale of plasmas, referred to as the “Debye length”. Debye length is usually denoted by the notation λ_D and is defined as

$$\lambda_D = \sqrt{\frac{\epsilon_0 k_B T_e}{e^2 n_e}}. \qquad \text{Eq. 1.1}$$

In Eq. 1.1, ϵ_0 is the permittivity of the free space, k_B is the Boltzmann constant, and e is the unit charge. n_e and T_e are two characteristic properties of plasmas. n_e denotes the electron (plasma) number density, i.e., the number of particles per unit volume, and T_e denotes the electron temperature.

The inverse of the time that it takes a charged particle to traverse a distance equal to the Debye length is called the "plasma frequency (ω_p)", which is a characteristic time scale of plasmas. Accordingly, we can think of electron plasma frequency (ω_{pe}), and the ion plasma frequency (ω_{pi}).

$$\omega_{pe} = \sqrt{\frac{e^2 n_e}{\epsilon_0 m_e}}, \qquad \omega_{pi} = \sqrt{\frac{e^2 n_i}{\epsilon_0 m_i}}. \qquad \text{Eq. 1.2}$$

Referring to Eq. 1.2, it is noted that, for a quasi-neutral plasma, $n_e \approx n_i$, and so the difference between the electron and the ion plasma frequencies is effectively equal to the ratio of their masses (m_e: electron mass, m_i: ion mass). Also note that, in the definition of the ω_{pi}, the ions are assumed to be singly charged.

The presence of a magnetic field can notably alter the trajectories of the plasma particles. If the plasma is immersed in a magnetic field, and assuming that the electrons and the ions do not have a thermal (random) velocity component along the magnetic field lines, the trajectories of the particles' motion will be closed circles around the field lines. In case the particles also exhibit a certain velocity component along the field lines, their trajectories turn into helices. In other words, the center of the circular trajectory (the so-called "guiding center" [4]) moves along the magnetic field line while the particle gyrates around the field line.

The radius of the particles' circular motion around a magnetic field line is another characteristic length scale for the plasmas immersed in magnetic fields. This radius is called the gyroradius or the Larmour radius [2], which is defined as per Eq. 1.3 for the electrons and the ions.

$$r_{L,e} = \frac{m_e v_{th.e\perp}}{eB_0}, \qquad r_{L,i} = \frac{m_i v_{th.i\perp}}{eB_0}. \qquad \text{Eq. 1.3}$$

In Eq. 1.3, B_0 is the intensity of the magnetic field. The $v_{th,e\perp}$ and the $v_{th.i\perp}$ are the thermal velocity components of the electrons and the ions, respectively, in a plane perpendicular to the magnetic field line. v_{th} is often defined as $\sqrt{(k_B T)/m}$ [5], with T and m representing the temperature and the mass of either the electrons or the ions.

The frequency of the particles' gyration around magnetic field lines is a characteristic time scale for the plasmas immersed in magnetic fields. This frequency is called gyrofrequency or cyclotron frequency [2][6] and is given for the electrons and the ions by the relations in Eq. 1.4.

$$\Omega_{ce} = \frac{eB_0}{m_e}, \qquad \Omega_{ci} = \frac{eB_0}{m_i}. \qquad \text{Eq. 1.4}$$

Referring to Eq. 1.3 and Eq. 1.4, it is immediate that $r_{L,e} = v_{th.e\perp}/\Omega_{ce}$, and $r_{L,i} = v_{th.i\perp}/\Omega_{ci}$. It is also evident that the Larmour radius scales directly with the mass of the particles, and as such is much smaller for the electrons than the ions.

A plasma is said to be fully magnetized in case both the $r_{L,e}$ and the $r_{L,i}$ are much smaller than a characteristic length (L) of a plasma system, i.e., $\frac{r_{L,e}}{L}, \frac{r_{L,i}}{L} \ll 1$. If only $\frac{r_{L,e}}{L} \ll 1$, the plasma is said to be partially magnetized.

The real-world plasma systems may have other characteristic time and length scales of importance as well. For instance, the frequency of the collisional interactions between electrons and ions (so-called "Coulomb collisions" [2][7]) or between electrons and neutral atoms can be important in certain plasma applications. In addition, if the plasma deviates from spatial uniformity due to, e.g., non-uniform spatial distribution of an externally applied magnetic field or the plasma being in contact with a surface, the gradient length scales of the plasma properties can become important as well.

1.1.2. Classification of plasmas (types and regimes)

There are many ways to classify plasmas based on various properties and/or phenomena of interest. For example, plasmas can be classified based on the degree of ionization [8] (percentage of charged particles to neutral atoms) or depending on the method of plasma generation [9] (e.g., direct current electron bombardment, electrical breakdown [10], radiofrequency excitation).

A typical, widely used classification, however, is with reference to the temperature of the plasma species. In this regard, plasmas are divided into two general categories of "**high-temperature**" plasmas (HTPs) and "**low-temperature**" plasmas (LTPs). Fusion plasmas belong to the first category, where the electron and the ion temperatures are nearly equal and typically on the order of 10^8 K (equivalently, ≈ 10 keV[1]) or higher [11]. The plasma regime in many industrial plasma applications, including for spacecraft propulsion, falls in the second category, i.e., the LTPs. Low-temperature plasmas have T_e values typically in the range of 1 to 10s of eV [11].

The LTPs can themselves be subdivided into two rather distinct categories. One main difference between the two LTP categories is regarding how the T_e compares against the T_i and the temperature of the neutral particles (T_n). For the so-called "**thermal** (equilibrium)" plasmas [11], $T_e \approx T_i \approx T_n$, with the temperatures typically on the order of 10^4 K (≤ 1 eV) [12]. Atmospheric plasmas used for medical purposes or for the hazard waste treatment are common examples of thermal plasmas.

In case that the $T_e \gg T_i$, the plasma is referred to as "**non-thermal**" (non-equilibrium) [12]. The T_e for the non-thermal plasmas reaches the upper limit of the typical temperature range for LTPs, i.e., on the order of 10 eV. In-space plasma propulsion technologies or the devices used in the manufacturing industry for material processing (e.g., etching or coating) are some applied examples of the non-thermal plasmas.

[1] Electronvolt (eV) is a unit of temperature very often used in plasma physics. Electronvolt is the amount of energy change in a unit charge if it experiences 1 V of electric potential difference. 1 eV is equal to 11,600 K.

Another classification approach, which aids to paint a more complete picture of the ubiquity of plasmas, can be pursued by considering both the electron temperature and the plasma number density. In this regard, plasma types can range from low-density, low-temperature regimes found in interstellar space and auroras to high-density, high-temperature regimes observed in the stars (including the Sun) as well as in the fusion devices. The use of the adjectives “low” and “high” is valid in a relative sense here.

A visual representation of the plasmas classification based on temperature and density is presented in Fig. 1.1. This figure, which is inspired by a similar illustration from Contemporary Physics Education Project [13], highlights the universal presence of plasmas and the vast ranges of temperature and number density properties that they exhibit across different settings and applications.

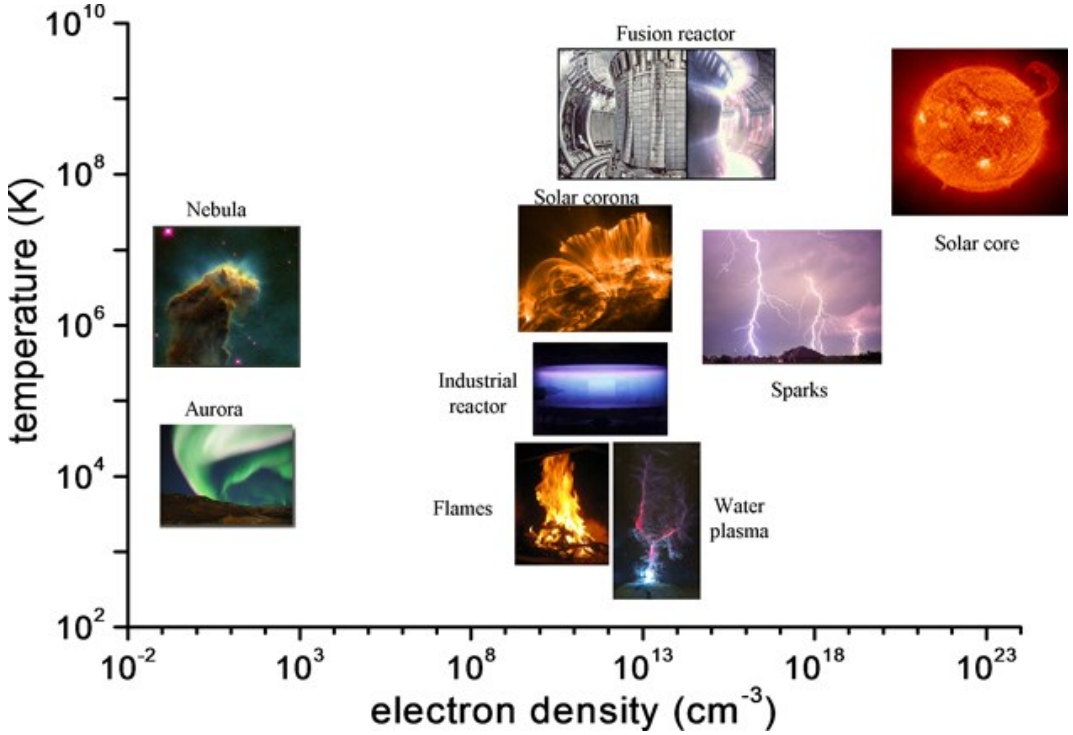


Fig. 1.1 Density and temperature characteristics of typical plasmas; image credit: Laboratoire de Physique des Plasmas (LPP), URL: https://www.lpp.polytechnique.fr/Our-research-in-a-few-words?lang=fr, available on public domain.

1.1.3. Overview of the plasma applications

Arguably, the first and the oldest technological application of plasmas dates back to more than 100 years ago, when plasmas were used in lamps to provide light. Today, plasma applications have expanded to a diverse range of industries. The high-temperature plasmas are at the core of nuclear fusion reactors, and the research on the HTPs in pursuit of the almost limitless source of clean energy is rigorously ongoing.

The LTP applications range from plasma etching of electronic chips and plasma-assisted additive manufacturing to plasma biomedicine and in-space plasma propulsion [14]. LTP technologies in the

manufacturing industry support the supply chain for automotive, aerospace, food packaging, health, and microelectronics industries.

A particular category of LTP technologies consists of the so-called “cross-field” plasma devices. In these devices, an often partially magnetized plasma is immersed in a perpendicular orientation of the electric (E) and the magnetic (B) fields. The term “cross-field” stems from this mutually perpendicular fields’ configuration.

Cross-field (or E×B) plasma technologies have found industrially crucial and important applications [15]. These applications include the magnetrons [16][17], which are devices that are central in the manufacturing industry, especially toward development of (silicon) microchips for computer processors. Another prominent example of the industrial applications of E×B plasmas is the Hall thrusters for spacecraft propulsion [18][19]. Hall thrusters are today the most widely in-use electric propulsion solution for satellites. They are also prime candidates to enable the next generation of near-Earth and interplanetary space missions.

Besides their applied significance, E×B plasmas have received a remarkable scientific attention as well over the past decades [15][20]. This is partly because the cross-field plasma discharges exhibit a complex and rich underlying physics that, in certain respects, resemble the phenomena occurring in the other areas of applied plasma physics, such as fusion energy. Nevertheless, the E×B plasma technologies are more accessible to study and much less costly to test than the fusion reactors. Therefore, investigating the plasma phenomena in E×B devices can serve as a cost-effective approach to derive useful insights for the research to realizing fusion.

In addition, the operation of the E×B technologies is strongly influenced by the underlying plasma processes. Most notably, the efficiency of an E×B device is largely dependent on the effectiveness with which the applied magnetic field can hinder the current of the electron species across the magnetic field lines. However, there are a myriad of physical phenomena that “undesirably” [from an applied perspective] enhance the cross-magnetic-field motion (transport) of the electrons, most of which are yet not fully understood (more details in subsection 1.1.4). This close coupling between the engineering and the physics of operation of E×B technologies gives the scientific research into these devices a critical applied importance. This has been another main driver of academic interest in investigating plasmas in E×B devices.

In order to study many of the yet-unresolved plasma physics questions, and also for the endeavors aimed at realizing reliable and predictive plasma models, the Hall thruster configuration has presented itself as a suitable proving ground over the past years (e.g., Refs. [21]-[27]).

To briefly describe this plasma propulsion technology, Hall thrusters are designed to ionize a neutral gas injected into their discharge channel, typically via DC electron bombardment, and to accelerate the resulting plasma ions in order to generate thrust [18]. The discharge channel is often in annular shape and made of ceramic. The geometry of a Hall thruster is such that the centerline of its discharge channel, which is usually parallel the thruster’s axis, determines the axial coordinate. The direction along the

channel circumference indicates the azimuthal coordinate, and the wall-to-wall direction of the channel is the radial coordinate, which completes a right-handed coordinate system.

The laboratory-scale E×B plasma in a Hall thruster exhibits several important processes and behaviors that are present in other plasma regimes as well, such as in fusion plasmas and in space plasmas. Moreover, a Hall thruster offers a relatively convenient accessibility and affordability for numerical and experimental physics studies over other industrially relevant types of plasma sources while closely matching their plasma configuration. Lastly, the complexity of the physics in a Hall thruster implies that a model (or modelling approach/methodology) capable of predicting the spatiotemporal evolutions of the plasma in this device in an accurate and generalizable manner effectively paves the way to achieve the long-sought-after predictive models across the plasma science domains. All these factors underline why Hall thrusters have become an important test bed for fundamental physics studies as well as for advancement in the computational capabilities.

To have a quantitative sense of the typical properties of the plasma in Hall thrusters, some of the main characteristic length and time scales are provided in the following. The plasma number density ($n_p \equiv n_i \approx n_e$) is $\mathcal{O}(10^{18})\ m^{-3}$, and the electron temperature (T_e) is typically $\mathcal{O}(10)\ eV$ [28]. The peak intensity of the applied magnetic field is $\mathcal{O}(10)\ mT$ [28][29]. The most used propellant today for Hall thrusters is xenon with the ion mass of $2.18 \times 10^{-25}\ kg$. With these values, we can estimate using the equations in subsection 1.1.1 the order of magnitude of the Debye length, Larmour radii, plasma frequencies, and the cyclotron frequencies as: $\lambda_D \approx \mathcal{O}(10)\ \mu m$, $r_{L,e} \approx \mathcal{O}(1)\ mm$, $r_{L,i} \approx \mathcal{O}(1)\ m$, $\omega_{pe} \approx 1 - 10\ GHz$, $\omega_{pi} \approx 1 - 10\ MHz$, $\Omega_{ce} \approx \mathcal{O}(100)\ MHz$, and $\Omega_{ci} \approx \mathcal{O}(1)\ kHz$.

The typical characteristic length of the plasma system in a Hall thruster (L) is on the order of a few centimeters. Noting this, we can immediately deduce that, for the Hall thruster plasma, $\frac{r_{L,e}}{L} \ll 1$, and $\frac{r_{L,i}}{L} \gg 1$. Therefore, the plasma in a Hall thruster is partially magnetized. Due to this partial magnetization, the plasma ions are accelerated in the thruster's electrostatic field with a minimal influence from the applied magnetic field.

It is highlighted that the plasma properties in a Hall thruster feature strong spatial variations (gradients) [18][19][28][29]. This is because of the spatially non-uniform topologies of the applied magnetic field as well as the presence of the channel walls that lead to the formation the so-called "plasma sheaths" [2] – a non-quasi-neutral region formed whenever the plasma is in contact with a surface. As a result, the values of the plasma properties and the characteristic length and time scales vary often notably throughout a Hall thruster domain, but they mostly remain within the ranges of values reported above.

1.1.4. Fundamental knowledge gap in the physics of Hall thrusters' plasma

As shared among the broader family of E×B plasma discharges and technologies, Hall thrusters have a highly complex, multiscale, and multidimensional underlying physics [15][20]. The cross-field configuration of the plasma in Hall thrusters leads to strong anisotropies in the properties of the magnetized electrons with respect to the directions along and perpendicular to the magnetic field lines.

These anisotropies and the strong spatial gradients in the plasma properties result in the excitation of a diverse range of plasma instabilities and turbulence that span over a broad range of spatial and temporal scales [19][30]. The development of instabilities can be an inherent response of the plasma to the conditions imposed on it and a way for the plasma to attempt returning to an equilibrium state [31]. As the instabilities excite, grow, and saturate, they can strongly interact with the plasma species and influence their spatiotemporal dynamics. The instabilities can also interact with each other, exchanging energy, and cause the formation of larger, possibly turbulent, global plasma structures. This is an intriguing phenomena that is referred to as "inverse energy cascade" [32][33].

In a Hall thruster plasma, various types of instabilities and oscillations can exist. The overall dynamics of the plasma is affected by the so-called "breathing mode" oscillations. These are low-frequency longitudinal oscillations with the frequencies in the range of 5 – 25 kHz and wavelengths that can be comparable with the axial extent of the thrusters' channel [34][35]. The breathing oscillations are a characteristic response of the Hall thruster plasma and manifest as a back-and-forth motion of the plasma discharge in unison during one oscillation cycle. Another important axially propagating instability mode is the so-called "Ion Transit Time Instability (ITTI)" [36][37]. As the name suggests, the waves associated with this instability propagate at the speed of the accelerating ion beam. The ITTI frequency is on the order of 100 – 500 kHz.

There are also important and influential instabilities exciting along the azimuthal direction of a Hall thruster. These include the Electron Cyclotron Drift Instability (ECDI) [38][39] – a short wavelength ($\mathcal{O}(1)\ mm$), high-frequency ($\approx$ 5 – 10 MHz) instability mode [40] – and the Modified Two-Stream Instability (MTSI) [41][42] – a relatively longer wavelength, lower frequency instability mode than the ECDI with frequencies in the range of 2 – 5 MHz [20][43]. The ECDI and the MTSI are demonstrated to have notable impacts on the transport of the plasma species and their heating [44]-[51]. A relatively low-frequency (25 – 100 kHz) large-scale azimuthal instability have also been reported to exist in the high-neutral density regions of a Hall thruster domain, which has been termed "(rotating) spoke instability" [52]. This instability has also been observed to induce notable cross-field current of electrons [53][54].

Further details on all the plasma instabilities relevant to a Hall thruster discharge can be found in Refs. [20][34]. In any case, the brief overview in the preceding paragraphs is sufficient to evidence the broad range of time and length scales associated with the dominant plasma instabilities in a Hall thruster. This scales disparity of the plasma phenomena has served to strongly hinder comprehensive studies of the plasma behaviors. We will return to this point in section 1.3.

There are many fundamental questions not yet answered concerning the characteristics and the evolution of the instabilities mentioned above despite extensive prior research efforts [15][20]. These primarily revolve around the three-dimensional effects [15], the interactions and transition dynamics between the axial and azimuthal instability modes [55][56], and the pathways and mechanisms for energy transfer across instabilities spectrum over time that can instigate inverse energy cascade and "self-organization" behaviors [32][57].

Another crucial and outstanding question regarding the underlying physics of Hall thrusters pertains to the problem of “anomalous” cross-magnetic-field transport of electrons. The “anomalous” electron transport in Hall thruster’s physics context refers to the observations, made experimentally [58] and numerically [59], that the rates at which the electrons diffuse across the lines of magnetic field are much larger than those predicted by the classical collision theory [19]. This “anomalous” transport is an example of the yet-not-fully-understood phenomena in Hall thrusters’ plasma that extend in terms of relevance to across the plasma science domains where the plasmas interact with the magnetic fields [60], from the magnetic confinement fusion reactors to the processes that lead to the creation of supernovae.

Even though decades of research has revealed many aspects of the electrons’ dynamics and transport characteristics in Hall thrusters, such as the significant role that the instabilities can play in this phenomenon [15][20], the overall puzzle has still many important missing pieces. This is particularly because of two factors: (1) the spatiotemporal variability of the dominant mechanism(s) behind the electrons’ cross-field transport, and (2) the unanswered questions regarding how the instabilities’ contributions to the electrons’ transport vary during the various phases of their evolution.

1.2. The need for predictive models of plasma systems

Toward answering the long-standing questions in plasma physics, to realize revolutionary plasma applications such as nuclear fusion, and to expedite and streamline the development of advanced plasma technologies across the industries, predictive plasma models serve as indispensable tools.

Before delving into how these models can contribute to progress on the above fronts, it is essential to clearly establish what we mean by a predictive model.

Essentially, a predictive plasma model is one that can self-consistently resolve the spatiotemporal evolution of the plasma across its involved scales. The term “self-consistency” refers to the ability of the model to provide predictions without the need for ad-hoc parameters that require tuning on a-priori information from the system (often experimental).

There are additional factors that, when considered in addition to the predictive capability, will augment the utility and expand the use cases of a predictive plasma model. These factors include cost-efficiency, generalizability, and interpretability.

In this thesis, a model is defined as being cost-efficient when it features a “low” computational cost (e.g., in terms of CPU-hours) to provide reliably accurate solutions. By referring to “low” computational cost, we qualitatively imply a time-to-solution that makes the simulations repeatable and enables carrying out extensive studies within practically meaningful timeframes.

The generalizability refers to a model’s ability to perform well across different scenarios. In our context, this involves the model’s capacity to reliably resolve the parametric variability of the plasma dynamics across the space of relevant state variables. Interpretability refers to the ease of understanding a model’s structure and predictions. By the model’s structure, we mean the relationships

that the model establishes between the involved variables. The interpretability feature of a model is often linked to the generalizability in that a generalizable model can often be interpretable as well.

A general-sense example of interpretable and generalizable models is the Newton's second law of motion ($F = ma$). This model is easily interpretable in the way that it correlates force (F) to mass (m) and acceleration (a). However, the model is also generalizable to a vast range of scenarios, from the falling of an apple to the motion of the planets around the Sun.

Ultimately, the model that we strive to achieve and that can effectively contribute to the scientific and engineering advances in plasma physics will not only need to be predictive, but also cost-efficient, generalizable, and interpretable. Such a model can aid the fundamental studies and the engineering applications as are described in the subsequent paragraphs.

From the scientific research perspective, a predictive model with the characteristics emphasized above will be the long-desired versatile analysis toolset that opens the door to novel approaches and frameworks to study plasmas, with the aim of answering the so-far-unresolved questions. One such approach, which is enabled by predictive plasma models, consist of performing methodical parametric investigations of plasma systems.

It is recalled from the discussions in subsections 1.1.3 and 1.1.4 that the interconnected and multifaceted nature of the underlying physical processes in Hall thrusters' plasma (and in other similar plasma configurations) gives rise to a greatly complex overall picture. Hence, to be able to fully comprehend the complicated global picture of plasmas behavior, an effective pathway entails breaking down the overall complex picture in order to study and understand the effects and contributions that each specific factor influencing the plasma have to the overall behavior and plasma response. Piecing these understandings together, we may solve the mysteries that still surround various plasma phenomena, some remained unanswered from the 1960s such as the problem of "anomalous" electron transport.

With the above in mind, through reasonable simplifications to the overall complex picture of the problem, and by performing predictive parametric simulations in the simplified setups, one will be able to set the foundational knowledge on the effects of the parameters of interest. Building upon the insights derived from the simplified parametric studies, we can methodically progress toward setups that are increasingly more representative of the real world, and ultimately discover the complete picture of the plasma processes in realistic settings, including the dependencies on various factors and parameters.

The parametric studies using fully predictive, cost-efficient models are also essential to complement the experimental research. The simulations provide a more controllable environment of study in which various effects that normally influence the plasma behavior simultaneously can be isolated and investigated more closely. In addition, the simulations enable access to information regarding the underlying plasma processes that might not be readily accessible in an experimental setting due to the spatiotemporal resolution limitations of the diagnostics and/or the physical accessibility constraints related to the device being tested. As a result, performing parametric sweeps of different plasma discharge parameters using predictive models can aid explaining the experimental observations and/or

enriching the results of the test campaigns. Such parametric simulations also serve as a viable method to stress-test the available theoretical descriptions of the plasma to determine the extent to which the theoretically driven correlations between the characteristics of the plasma phenomena and the discharge properties are consistent with the numerical observations. This endeavor may lead to the development of better, more comprehensive theories for plasmas.

On the side of the applied (industrial) applications, predictive plasma models hold immense potential to revamp the design and development processes, yielding groundbreaking advances, and ultimately catalyzing the transition of the plasma industries to more efficient digital engineering practices [61] based on digital tools such as system models/threads/twins [61] (also refer to section 7.2.3).

Predictive models will underpin computer-enabled optimization-based design approaches. These are currently unachievable due to the limitations of the state-of-the-art simulation tools (refer to section 1.3). Thus, plasma industries today heavily rely on empirical design approaches that involve several design, manufacture, and test iterations, resulting in significant time and cost burdens. Nevertheless, the high-dimensional design parameter space typically associated with the plasma technologies makes it impractical to achieve the most efficient solutions through trial-and-error methods alone, emphasizing the necessity for optimization-based practices using predictive, cost-effective modelling tools.

Predictive models can, additionally, inform optimal experimental design methodologies [62], which enable planning of the experiments effectively to extract maximum information about plasmas behavior with minimal resources. These methodologies aid identifying the critical parameters and experimental conditions, leading in turn to more cost-efficient and informative test campaigns. It is evident that realizing optimal experimental design approaches for plasma systems on the back of predictive plasma models equally supports advancement within the fundamental scientific research.

Another area where the predictive plasma models can play a crucial role is complementing the results of the characterization/qualification test campaigns by filling gaps in experimental observations. By simulating plasma behavior over unobserved conditions or scenarios (the so-called “what-if” analyses), predictive models enhance the comprehensiveness and reliability of the experimental data. In the specific case of Hall thrusters, for instance, since they are designed for on-orbit operation, their on-ground testing is performed within vacuum test facilities. However, these facilities struggle to exactly replicate the on-orbit operation environment [63][64]. As a result, the complementary data from predictive Hall thruster models can particularly augment in this case the trustworthiness and extensibility of the on-ground test results with respect to the on-orbit operation.

Finally, predictive plasma models will greatly advance our capabilities in controlling plasmas. By integrating real-time data with the predictive models, in particular, control algorithms can dynamically adjust operating parameters to maintain desired plasma characteristics, enhance process stability, and optimize energy efficiency. The groundbreaking plasma control strategies enable industries to achieve consistent and high-quality plasma processes while minimizing operational costs and possible environmental impacts. Returning to our reference example of a Hall thruster, computer-enabled control enables compatibility of the thruster’s operation with the autonomy requirements of the next-generation

(potentially AI-enhanced) in-space missions and applications, from near-Earth on-orbit servicing to robotic and manned exploration of the Moon, Mars and beyond.

1.3. Status quo of the computational approaches to study plasmas

Through the discussions so far, it was alluded a few times to the fact that the current capabilities to simulate plasmas have major limitations that have hindered the realization of predictive plasma models, which are as well cost-efficient, generalizable, and interpretable.

In this section, an overview of the state-of-the-art numerical tools for the study of plasmas is provided to better clarify the above statement. The discussions are aimed at presenting some of the main limitations and issues associated with the plasma models in use today. Noting this context of the subsequent discussions, and for the sake of conciseness, we focus on the simulation practices being typically used to model partially ionized plasmas in cross-field configurations, including Hall thrusters.

Conventional plasma modeling approaches for the cross-field plasmas can be generally divided into three main categories [65]:

1. **Fluid simulations**: these models are based on solving a system of partial differential conservation equations for the continuity, momentum and energy of the flow of plasma species and, often, the neutral particles. In essence, fluid plasma simulations can be thought of as a generalization of Navier-Stokes equations in fluid dynamics by including the electromagnetic force terms and coupling the equations to the Maxwell laws.

   The fluid equations for plasmas are derived by taking various velocity moments [66][67] of the plasma kinetic equation [8] for each species. By taking these velocity moments, the dimensionality of the problem in the most general case is reduced from 3D3V (three dimensional in configuration (physical) space and three dimensional in velocity (kinetic) space) to 3D. As such, plasma fluid equations can be considered as “reduced-dimension” or “reduced-order” models (ROMs) in the context of plasma modelling. Whereas this order reduction is, in theory, advantageous as it “reduces” the computational requirements, it comes at the expense of losing the ability to resolve important kinetic phenomena that occur within the velocity space.

   The fluid conservation equations are coupled through various source and sink terms that represent various phenomena such as ionization, collisional momentum and energy transfer, etc. Fluid system of equations require closure models so that the effects of higher-order moment terms as well as the nonclassical behaviors can be incorporated in the equations. In the Hall thrusters simulation context, (typically ad-hoc) closure models [19][65] are most often employed to describe the momentum change in the electrons’ flow due to the interactions with the thrusters’ channel walls and with the plasma instabilities. Both types of interactions can in fact play important roles in enhancing electrons’ cross-magnetic-field motion [19].

2. **Kinetic particle-in-cell (PIC) simulations**: these high-fidelity simulations are a way to directly solve the plasma kinetic equation. A PIC simulation is a statistical model whose results satisfy the governing Vlasov-Poisson (Maxwell) system of equations for plasma systems [65][68][69].

The PIC method follows a Lagrangian approach where the motion of several ensembles of real particles, called macroparticles, is followed by integrating the equations of motion for each macroparticle in the presence of electric and magnetic fields [68][69]. The fields themselves are solved on a Eulerian grid [68][69].

The macroparticles can be defined in a more scientifically accurate manner as well. A plasma species macroparticle is a statistical sample of an important property of that plasma species, which is called the "distribution function" [8]. A distribution function is essentially and in the most general sense a seven-dimensional histogram of the particles of a plasma species. It quantifies how many real plasma particles with 3D velocities within a certain range are present at a specific location in the 3D configuration space. The seventh dimension is time, as the distribution functions can evolve over time as well.

Based on the above definition, by kinetically tracing the temporal evolution of the macroparticles of different plasma species, a PIC simulation resolves the evolution of the plasma species' distribution function, which is the solution to the plasma kinetic equation.

As the PIC approach is resolving the variations in the particles' distribution functions, it is capable of capturing kinetic plasma phenomena. Despite this highly desirable feature, the computational cost associated with the PIC simulation approach can be enormous depending on the application, as we will shortly see why in more detail.

3. **Hybrid fluid/PIC simulations**: As a compromise between the computational cost and the fidelity of the simulations, hybrid fluid/PIC codes typically treat the ion and the neutral species kinetically, whereas the fluid description is used for the electrons [65]. The hybrid models are thus capable of resolving the ions' and the neutrals' distribution functions self-consistently. Hence, these models are useful to investigate the kinetic phenomena that is affected and/or driven by the dynamics of the heavy plasma species, e.g., the ion bombardment of the Hall thrusters' channel walls and the consequent erosion. Nevertheless, as the electrons are regarded as a fluid, the limitations of the fluid modeling approach, most importantly the need for transport closure models and the fundamental inability to resolve the electrons' kinetic processes, such as wave-particle interactions, still applies to the hybrid simulations. In this regard, the electrons' cross-field transport and the associated phenomena, e.g., may not be accurately captured, because the results' fidelity is dependent on the accuracy of the model(s) used to incorporate the unresolved behaviors in the hybrid simulation.

With the above overview of the common plasma modelling approaches today, we can now point out the reasons why none of these approaches can currently realize predictive plasma models for the scientific and industrial needs, which were described in section 1.2.

We start by the plasma fluid models, which offer the highest computational cost advantage over the kinetic PIC simulations and the hybrid models. Recalling from the overview of the fluid modelling approach, the fluid simulations can only be self-consistently predictive if the effects of the missing and unresolved phenomena in these simulations can be incorporated in them through closure terms and/or models. However, despite years of effort and multitude of approaches pursued [23][25][26][70][71],

rigorous and generalizable closure models that would not be reliant on a-priori assumptions or tuning on experimental data from the physical system are still unavailable [72]. Even recently developed advanced fluid models with first-principles closure terms are highly simplified and are not applicable to all plasma configurations and/or conditions [15].

The same lack of rigorous and generalizable closure models that hinders the predictiveness of the common fluid simulations of plasmas also impedes the hybrid plasma models from being self-consistently predictive. The most significant challenge in deriving the required closure models is posed by the not-yet-fully-understood problem of "anomalous" electrons' cross-field transport, which plays a major role in determining the behavior and global response of the plasma. The open physics questions surrounding this phenomenon, as were described in subsection 1.1.4, have particularly limited the generalizability and robustness of the first-principles closure terms/models proposed so far for the electrons' transport across the magnetic fields [72][73].

Kinetic PIC simulations have inherently an advantage over the fluid/hybrid models in that, if they hypothetically resolve the full 3D3V distribution functions of the plasma species over time, no closure models are required, and all the kinetic processes will be self-consistently captured. Thus, in theory, a PIC simulation is predictive by construction. However, what impedes a PIC simulation from addressing the scientific and industrial needs for predictive plasma models (section 1.2) is the enormous computational resource demand of this simulation approach [69]. In fact, following conventional PIC modelling methodologies, 3D3V simulations immediately become computationally intractable as one attempts studying full-size plasma technologies/devices. Even lower dimension 2D3V kinetic simulations will be prohibitively expensive for real-scale devices if the use of any unphysical scaling is to be avoided [55][74]; for instance, a 2D PIC simulation of a Hall thruster that captures the plasma dynamics along the axial coordinate for a representative period of 150 μs simulated time without any scaling of the physical or geometrical parameters requires a run time of *about 1 year using 1500 CPUs* [75]. A 3D PIC simulation of this same problem will be at least an order of magnitude more costly.

At this point, one may wonder why a PIC simulation is so computationally demanding, particularly for the practical purpose of modelling real-world technologies. This question is answered for the specific reference example of a Hall thruster. As was described in subsection 1.1.4, the plasma in a Hall thruster evolves globally along the axial direction over a time scale of 0.1 – 1 ms, which corresponds to the typical range of frequencies of the global breathing mode oscillations. As a result, a representative PIC simulation of a real-world Hall thruster shall at least be able to resolve the evolution of the plasma for this time duration. Nevertheless, for the sufficient statistical representativeness of the results, it is typically desired to resolve the plasma evolution over several breathing cycles, therefore, typical simulation times of 1 – 10 ms. It is noted that resolving the axial dynamics of a Hall thruster discharge is important because several critical plasma processes, including electrons' transport, are regulated by this global oscillation [76]. It is also recalled from subsection 1.1.3 that the typical length scales of a Hall thruster are on the order of a few centimeters.

Keeping the above information in mind, we need to also look at the numerical requirements of a PIC simulation. The traditional PIC implementation, often referred to as "explicit, momentum conserving

PIC", has stringent stability and convergence requirements. In more detail, the timestep of the simulation must be smaller than the electron plasma frequency [68][69], which was reported in subsection 1.1.3 to be typically in the range of $1-10\,GHz$ ($1-10\,ns$) for Hall thrusters' plasmas. Another stability criterion is that the simulation macroparticles shall not travel more than a cell length during each timestep [68][69]. As the timestep is limited by the electron plasma frequency, and that, during one electron plasma oscillation, the particles travel a distance equal to the Debye length, the cell size should be typically smaller than this characteristic length. Recalling from subsection 1.1.3, the typical Debye length for Hall thrusters' plasmas is on order of 10 μm.

It now immediately follows that a PIC model developed to simulate the plasma in a real-size Hall thruster over the real-world characteristic evolution time scales can have $\mathcal{O}(10^3)$ computation cells along each involved spatial coordinate and may need to progress for iteration numbers of $\mathcal{O}(10^9)$. Adding to this already huge computational demand, the statistical convergence of a PIC simulation requires that a certain minimum number of macroparticles for each plasma species be present within each computation cell. To provide a quantitative sense for this, Ref. [75] adopted an initial macroparticle-per-cell count of 400 for a 2D PIC simulation. Therefore, that PIC simulation needed to track the spatiotemporal evolution of 4×10^8 microparticles at the beginning of the simulation alone. Exacerbating the computational cost, during a typical Hall thruster simulation, the plasma number density and electron temperature vary temporally and spatially by about an order of magnitude, meaning that the plasma frequency and the Debye length become more stringent over time. Thus, the common practice is to choose the cell size and the timestep of a PIC simulation to be equal to the smallest values expected for the ω_{pe} and λ_D over the simulation time. It is noted that the total macroparticles count also changes proportionally with the change in plasma number density as the simulation progresses.

The above discussions clarify why PIC simulations are enormously costly and, hence, cannot answer the need for predictive plasma models. In part, the enormous computational cost of the traditional PIC simulations strongly limits their applicability to perform parametric investigations (for science studies or engineering design iterations) and to produce statistically representative high-fidelity datasets (as training data, e.g., for ML/DD algorithms).

1.4. Reduced-order particle-in-cell scheme: a cost-effective kinetic tool for high-fidelity data generation

Let us assume for the moment that ML/DD methods and architectures can indeed open viable pathways toward achieving predictive plasma models that are cost-efficient, generalizable, and interpretable. We will elaborate on this premise within section 1.5 as well as in Chapter 2.

In any case, ML/DD methods can only deliver the potential they may hold toward realizing the desired predictive plasma models if sufficiently varied, statistically representative, high-quality databases of plasma systems are available. In this respect, although kinetic PIC simulations offer sufficiently high levels of fidelity, the inadequacy of the traditional PIC schemes for extensive dataset generation due to their significant computational cost represents a major shortcoming. This emphasizes

the prerequisite nature of a simulation tool that enables cost-effective access to high-fidelity, detailed plasma data for the advancement in ML/DD-enabled plasma modelling.

In recent years, several efforts have been aimed at tackling the computational cost issue of the PIC simulations. These efforts include the development of implicit and/or energy-conserving PIC algorithms [77][78], GPU implementation of the PIC method [79], the sparse-grid PIC scheme [80], and the reduced-order PIC scheme [81][82]. The last approach – the reduced-order PIC – has been developed at Imperial Plasma Propulsion Laboratory (IPPL) by Reza [83] as a computationally efficient, high-fidelity kinetic plasma simulation tool. Reza's approach features unique flexibility and extensibility characteristics that set it distinctly apart from the approaches pursued by other authors in Refs. [77]-[80] toward lowering the cost of PIC models.

The innovative reduced-order PIC scheme and the code developments surrounding it allowed effective generation of sufficiently large high-fidelity plasma datasets for the data-driven modelling investigations reported on through Chapter 3 to Chapter 6.

The quasi-2D (Q2D) implementation of the reduced-order PIC scheme forms the basis of a PIC code, which is termed "IPPL-Q2D" [84]. This code has been extensively verified against rigorous benchmark cases available in the literature [81][84][85]. These benchmarks involved 2D configurations representative of a Hall thruster geometry along its axial-azimuthal [86] and radial-azimuthal planes [43]. Furthermore, IPPL-Q2D is successfully benchmarked in a 2D Penning discharge configuration [87], in which, due to the formation of a large spoke-like rotating structure, the principal directions of the involved physical phenomena constantly changes (see subsection 1.1.4 for information on the spoke instability) [85]. The Penning benchmark demonstrated that the formulation underpinning the reduced-order PIC is seamlessly extensible to problems in which the principal directions of the underlying processes may not be aligned and/or evolve with respect to the directions used for the domain decomposition.

The high-fidelity data generated from the PIC simulations of all these benchmark/verification cases using IPPL-Q2D particularly constitute the database used for the assessments of data-driven plasma ROMs in Chapter 3 and Chapter 4. The setup and conditions of the corresponding simulations are elaborated on within these chapters.

1.5. Machine learning for cost-efficient predictive modelling of plasmas: promises and challenges

With increased computational power and abundant generation of data over the last decades, the traditional paradigms of simulation and analysis across sciences have undergone a fundamental transformation. Machine learning (ML) has emerged as a cornerstone of contemporary research, offering novel ways to analyze large and complex datasets, extract insights, and develop predictive models. Machine-learning techniques depart from the conventional reliance on predefined models or theoretical assumptions. Rather, these techniques process data to uncover the underlying patterns, information, relationships, and trends that might otherwise remain obscure. These approaches have

proven particularly powerful where the traditional approaches fall short, such as complex systems with high-dimensional data spaces, or situations where first-principles generalizable equations are challenging/unfeasible to establish.

Plasma modelling and understanding the physics of plasmas are areas that can markedly benefit from machine-learning approaches. In particular, ML-enabled reduced-order models (ROMs) approximate a system's behavior using a reduced-dimensional representation while preserving the essential dynamics, hence, alleviating the common computational burden/intractability of the high-fidelity kinetic plasma simulations. The reduced-order system representations enable faster simulations, real-time control, and optimization of complex systems. The reduced-order representation also helps our understanding of the complex interactions between various processes in the plasma, such as instabilities, by distilling the high dimensional dynamics and providing a more interpretable and lower dimensional description.

Let us focus a bit further on the concept of ROM. A ROM is an approximate model of a system' behavior (dynamics) with a reduced set of degrees of freedom. Reducing the degrees of freedom necessary to describe a system lowers the dimensionality of the model. For example, as was mentioned in section 1.3, plasma fluid equations are derived by taking integrals of the plasma kinetic equation over the velocity space. This in turn reduces the problem's dimensionality from 3D3V in phase space to 3D in configuration space. As such, plasma fluid equations fit the description of ROMs.

From the above example, it is evident that ROMs are not a new concept. Indeed, they have been around for decades, if not centuries. As such, deriving ROMs do not necessarily involve machine learning, and there are classical approaches widely in use today to develop ROMs for various physical systems. The most prominent example perhaps is the POD-Galerkin projection method. The premise of this method is rather straightforward. Proper Orthogonal Decomposition (POD) finds the dominant orthogonal modes underlying the data from a system. These modes constitute a bespoke hierarchical coordinate system, creating a linear subspace to approximate the data. If the system of equations describing the dynamics of the system is known, the equations can be projected (Galerkin projection) onto the n-dimensional coordinate system defined by the n dominant modes of the system. The projected system of equations are in the form of ordinary differential equations (ODEs), resolving the time evolution of the amplitudes of the dominant modes of the physical system under study.

Machine learning allows us to surpass the capabilities of the classical ROM development approaches. Referring to the same example of the POD-Galerkin method, ML approaches may derive a much more compressed (lower dimensional) representation of the system beyond what is feasible with POD. In addition, machine learning enables the "discovery" of the dynamics describing a system's evolution. This is a crucial capability because, for most of the complex physical systems of applied interest today, the governing system of equations is either fully or partially unknown. The latter is in fact true for the plasma fluid equations for which self-consistent and generalizable closure models for the kinetic and nonclassical effects are not yet established.

Machine learning, a subclass of Artificial Intelligence (AI), comprises a broad array of algorithms which aim to learn from data without the need for explicit mathematical models [89]. Machine learning

is closely related to the “data-driven (DD)” framework, where the methods also seek to derive information/models from data. The term “data-driven” can be thought of as a general terminology encompassing both classic statistics/regression methods as well as modern generalized regression and machine-learning approaches.

ML has a rich history that dates back to the 1950s and 1960s. In fact, two significant developments occurred in those years. One was the emergence of the concept of “cybernetics”, as articulated by Wiener in 1965 [90], which aimed to mimic the thinking processes of the human brain. The other was efforts to develop “machines”, such as the perceptron [91], with a focus on automating tasks such as classification and regression. The latter development has particularly laid the groundwork for modern machine learning, with ongoing relevance and impact in contemporary research. Research on ML and the broader field of AI saw a major drop in interest from the early 1970s to the late 1980s, following the criticizing report of Sir James Lighthill in 1974 [92] on the failure of AI to deliver on its grand promises. The interest in AI and ML, especially in neural networks, was reawakened in 1986 after the development of the “back-propagation” algorithm by Rumelhart et al. [93]. This was because the back-propagation algorithm enabled the training of neural networks with multiple layers. The links established by various authors [94][95] around the same time between machine learning and statistical mechanics also contributed to the resurged interest.

Machine learning continued to evolve and advance over the past decades [96], with significant interest and attention gathered in the past few years from the scientific and industrial communities toward incorporating the ML approaches in science discovery and technology development.

Today, ML is in fact extensively applied to data from simulations and experiments for the analysis, modelling, control, and optimization of physical systems. Two main areas in which ML is being rigorously used are “pattern/feature extraction” and “dynamics modelling” [96]. Concerning the latter, ROM development represents only one of the applications. Indeed, ML can be also applied to enhance/accelerate computationally costly, high-fidelity simulations [97] as well as to derive closure models for the fluid system of equations [97]. As the application of ML for physical systems’ modelling is the main theme of this thesis, a more extensive review of the various possibilities on this front will be provided in the next Chapter.

A few remarks are, however, important to be underlined here before moving on. First, despite the great promises that ML offers for science and engineering, there are still several challenges to be addressed so as to realize the full potentials. Some of the most important challenges regard the amounts of data needed, the robustness of the ML algorithms to work with often noisy and sparse data, and the interpretability and generalizability of the ML-enabled models. The data from complex physical systems of real-world applied interest are commonly quite expensive to obtain either numerically or experimentally. As such, the datasets are typically rich only along a single dimension. For example, high-fidelity simulations’ results typically feature high resolution in space but are sparse in time. Contrarily, experimental data can cover long time durations but with low spatial resolution and often high levels of noise. It is, thus, essential to develop and mature ML algorithms that can robustly and reliably work with limited and noisy data. Interpretability and generalizability are also crucial for most

scientific and engineering applications, especially in scenarios where decision making is involved based on the model's outcomes/predictions. Interpretability facilitates understanding the workings of a model, and generalizability can significantly lower the cost of its development by expanding the range of conditions and situations to which the single ML model is applicable. Toward addressing many of these challenges, the incorporation of the known physics of a system in terms of, e.g., the conservations, symmetries, and invariances, provides an important and viable pathway [97]. This is why the theme of "physics-informed" [98] and, more broadly, "physics-consistent" [96] machine learning is being rigorously researched today.

Second, ML applications in plasma physics are still nascent and have remained relatively limited in scope. This is whereas the closely related, adjacent field to plasma physics – fluid mechanics – has seen a major increase in ML-enabled applications for a wide variety of purposes in recent years. This significant progress in ML-related applications within fluid mechanics together with the notable overlaps in tools and practices that exist among these fields provide an excellent opportunity for the cross-transfer of knowledge and capabilities in terms of ML algorithms and use cases to plasma physics. In fact, the mapping of the already demonstrated promising applications of ML for fluid flows modelling to plasmas comprises a main objective of Chapter 2.

Nevertheless, at this point, one may wonder what reason(s) might have led to the creation of the present gap between the two adjacent fields of fluid mechanics and plasma physics in terms of ML applications. Of course, different explanations can be provided for this status quo. However, a fundamental root cause can be traced to the scarcity of large enough volumes of data from plasma systems to support ML research for plasma physics. In this regard, according to the discussions presented in sections 1.1.3 and 1.3, the complexities associated with the plasmas behavior, including the greatly multiscale, multidimensional spatiotemporal characteristics of the involved dominant phenomena, has made the generation of sufficiently varied, statistically representative, and high-quality data a grand challenge in plasma physics. On the one hand, the high-fidelity numerical tools have been largely computationally costly to simulate the real-world geometries of plasma devices over their large relevant parameter spaces of operation. On the other hand, plasma experiments have been highly complicated due to various challenges including accessibility and the cost of operation of the devices, in addition to issues concerning the spatiotemporal resolution and sensitivity of the diagnostics.

As a result, whereas "big data" has become a reality in fluid mechanics over the past decade due to extensive efforts and progress in high-performance computing and experimental measurement capabilities [96][99], similar advances have just begun to manifest and influence the plasma physics research, meaning that plasma physics is still away from the realm of "big data" so that ML can be exploited in the field to its fullest potential. In any case, community-wide efforts and coordination initiated within plasma physics communities in the past years to address the computational cost issue of high-fidelity plasma simulations as well as to improve the employed diagnostics have been changing the circumstances. This implies that ML research for plasma physics, toward which this thesis has made pioneering steps, is today greatly timely and indeed essential for further advancements of the field.

1.6. Problem statement: research motivations and objectives

According to the discussions presented within this Chapter, this thesis is motivated by the fact that, even though the need for predictive plasma models is critically felt today in the academia as well as the industry, such models are not available. Furthermore, conventional plasma simulation approaches are incapable of realizing predictive models that are computationally efficient, robust, interpretable in form and in predictions, and applicable to a broad range of scenarios and plasma conditions. Therefore, as machine-learning techniques have transformed the world in previously unimaginable ways, toward both how we live our lives every day, and how we study and analyze complex systems, it is warranted to examine the promises and potentials of these algorithms in the plasma science domain.

Consequently, the overarching objectives of this PhD research have been to: **(a)** rigorously investigate the pathways that ML/DD algorithms can open to realize predictive plasma models, including how the advances in ML research in adjacent fields to plasma physics, such as fluid dynamics, can be mapped to the field and inform the applications of ML/DD methods in the context of computational plasma physics, **(b)** identify and demonstrate in detail select ML/DD methods and/or methodologies for plasma state forecasting.

On the path to achieving the PhD's overarching objectives, three general tasks were pursued in series, with the second and the third tasks thus defined on the basis of the outcomes from task 1:

Task 1: a detailed, critical literature survey to establish a rigorous knowledge basis of and a well-informed perspective on ML/DD techniques for the analysis and reduced-order modelling of physical systems,

Task 2: assessment and stress-test analysis of the identified global data-driven approach, optimized dynamic mode decomposition, toward dimensionality reduction and state forecasting of plasma systems,

Task 3: development and detailed testing of a data-driven local dynamics discovery approach for reduced-order modelling that surpasses the available techniques in terms of the ease of time forecasting of the dynamics as governed by partial differential equations – in other words, obviating the need for a-posteriori numerical solution of the discovered differential equation as is common among local dynamic discovery algorithms.

Chapter 2 State-of-the-art review on machine learning for physical systems modelling

In this Chapter, the aim is to present a detailed survey of the landscape of machine learning for the analysis and modelling of physical systems. Machine learning provides a vast array of tools and architectures that enable transforming data from simulations and/or experiments into useful and explainable science, hence, augmenting the domain knowledge. ML-enabled/enhanced modelling can revamp scientific computing for real-world complex engineering systems, yielding unique opportunities to examine in detail the operation of the technologies as well as to automate their optimization and control.

In recent years, there has been a significant rise in ML applications across science domains. Particularly relevant advances have occurred within fluid mechanics, with special attention to how the ML techniques can enhance computational modelling of fluid flows. Contrarily, ML applications within plasma physics research have remained relatively limited in extent. Nevertheless, the closely related nature of the two fields of fluid mechanics and plasma physics provides a great opportunity to establish a roadmap of how the advances in ML for fluid flows modelling, in particular, can be cross-transferred to computational plasma physics. This is the overall approach that will be pursued in this Chapter.

Accordingly, section 2.1 starts by discussing some general fundamental aspects of ML. An overview is then presented of various ways by which the ML algorithms have been applied to computational fluid dynamics. This will be done through a review of several insightful efforts on this front. As a disclaimer, the ML algorithms and applications reviewed in this Chapter shall not be construed as exhaustive. Rather, these serve as key highlights of the different directions of research being currently pursued.

Considering that the overall focus of this thesis is on reduced-order plasma modelling, sections 2.2 and 2.3 review the prominent ML approaches that can underpin the main composing elements of ROM development, namely, dimensionality reduction/coordinate transformation on the one hand, and dynamics discovery (system identification) on the other. Finally, section 2.4 provides an overview of the applications that ML has recently found for plasma systems modelling.

2.1. General overview

The discussions begin here by reviewing some fundamentals of machine learning.

ML provides a powerful and versatile information processing framework [96], the primary aim of which is to enable learning from data without the need for explicit programming and/or underlying mathematical model. At the core of ML lies the concept of a “learning machine”. A learning machine is an algorithm that iteratively learns patterns, structures, or relationships from data, allowing it to make predictions, classifications, or decisions based on new, unseen data. Examples of learning machines include neural networks, decision trees, support vector machines, and Bayesian networks. Cherkassky and Mulier [100] in 2007 defined the learning problem itself as “*the process of estimating associations between inputs, outputs, and parameters of a system using a limited number of observations*” [96].

ML algorithms can be categorized into three main types based on the nature of the learning process and the extent to which external supervisory information is available to the learning machine: **(1)** supervised learning, **(2)** unsupervised learning, and **(3)** semi-supervised learning. We will describe in the following each of these learning categories and provide an overview of the most common algorithms belonging to each.

(1) **Supervised Learning**: In supervised learning, corrective information are available to the learning machine. Simply put, the algorithm is trained on a dataset consisting of input-output pairs, where each input is associated with a corresponding output or label. The goal is to learn a mapping or function that predicts the output for new, unseen inputs based on their comprising features. The parameters of the learning machine are determined through the process of minimizing a defined loss function, which is commonly in the form of an L2-norm error between the model’s prediction and the “ground-truth” input data. Common architectures for supervised learning are:

   i. Linear and generalized linear regression: Linear and generalized linear regression are foundational techniques in statistics as well as machine learning, providing interpretable models for understanding the relationship between variables and making predictions based on the observed data.

   Linear regression [101] is a widely used method for predicting a continuous output based on one or more input features. It assumes a linear relationship between the predictors (independent variables) and the target (dependent) variable. This method is particularly effective when the relationship between variables is or can be approximated as linear.

   Generalized linear regression [102] expands upon the linear regression by accommodating, e.g., nonlinear relationships between variables. It encompasses a broader class of models, where the target variable follows a distribution from the exponential family, for instance, Gaussian, binomial, Poisson. It enables modelling relationships between variables that may not be strictly linear. It can also handle various types of target variables, including binary outcomes and continuous variables with non-constant variance. Generalized linear

regression provides a flexible framework for modelling diverse datasets, hence, serving as a versatile tool in data analysis and modelling.

ii. **Support vector machines (SVM)**: SVMs [103] are versatile algorithms typically used for classification and regression tasks. SVM finds the optimal hyperplane that separates different classes or fits the data with a minimal margin of error. In the context of classification, SVMs sit adjacent to another widely used algorithm, the random forests [104]. SVMs have been one of the most successful supervisory data-mining algorithms in the last few decades [101]. The emergence and rapid advances of deep neural networks (see Point iii below), however, implies that when sufficient data is available, SVMs are superseded by the deep neural nets.

iii. **Neural networks**: Neural networks (NNs) are computational architectures that are loosely inspired by the biological networks of neurons in a human brain. They serve as powerful nonlinear function approximators and are commonly used in supervised learning. They consist of interconnected neurons organized in layers, where each neuron processes inputs through an activation function to produce an output. The modular structure of neural networks allows for flexibility in combining neurons into various architectures that are tailored to different problem domains and data types. NNs are closely connected to the universal approximation theorem by Hornik et al. [105], which states that any function can be approximated by a sufficiently large and deep neural network. Deep neural networks (DNNs), which consist of several “hidden” layers of neurons between the input layer and the output layer, have shown remarkable capabilities, largely due to their ability for hierarchical learning. This means that whereas the initial layers of a DNN recover the simple and basic relationships in the data, deeper layers use a combination of the extracted information to derive more abstract relationships. This is highly relevant for physical systems since many such systems also exhibit hierarchical behavior, enabling NNs’ modularity to be effectively exploited for tasks like nonlinear coordinate embedding and dimensionality reduction.

   Feed-forward NNs are one of the most common architectures. Optimization methods, facilitated by back-propagation [93], adjust network weights to minimize the error between predictions and labeled training data. In a neural net, if the activation functions are comprised by convolutional kernels, the network is called a convolutional neural network (CNNs) – a powerful architecture renowned for image and pattern recognition tasks [106].

   Recurrent neural networks (RNNs) are another type of NNs that are pivotal for the processing of sequential information, such as the time-series data. They retain memory of past inputs through recurrent connections, allowing them to capture temporal dependencies inherent in sequential data. However, traditional RNNs face challenges with vanishing or exploding gradients during training. The introduction of long short-term memory (LSTM) algorithms [107] addresses these issues by incorporating memory cells and gating mechanisms, enabling efficient transmission of long-term information. Extensions like multi-dimensional LSTM networks (MD-LSTM) [108] further enhance their capability to handle high-dimensional spatiotemporal data.

In addition to RNNs, alternative architectures like echo-state networks [109] (belonging to the broader category of "reservoir computing") have also emerged as potent tools for predicting dynamical systems.

(2) **Unsupervised Learning**: Here, the learning task is to extract patterns and structures from unlabeled data without explicit supervision in order to discover hidden patterns, groupings, or clusters within the data. The typical problem types encountered in unsupervised learning include clustering and dimensionality reduction. Some common architecture are:

i. **K-means clustering**: This is the most common and widely used algorithm for clustering, i.e., the task of dividing data into different groups based on similarity or distance metrics. K-means clustering partitions the data into k clusters, resulting in the transformation of the data space into several Voronoi cells [96].

ii. **Proper orthogonal decomposition (POD)**: POD, also referred to as linear principal component analysis (PCA), is dimensionality-reduction technique that projects high-dimensional data onto a lower-dimensional subspace while preserving the maximum variance in the data. POD/PCA algorithm has been playing a central role in reduced-order modelling for decades. This is because POD enables deriving the dominant modes underlying data from a system. We will discuss this algorithm in more details in section 2.2 and in Chapter 3, where we will present its application to data from a plasma system.

To link POD or the linear PCA to machine learning, we can imagine the algorithm as a shallow neural net with linear activation functions.

iii. **Autoencoders**: These are NN architectures designed to learn efficient, compressed representations of the input data by first encoding it into a lower-dimensional space (latent space) and then decoding it back to the original input. Autoencoders are used for tasks such as feature extraction and denoising. The autoencoder architecture is a nonlinear generalization of the POD method, achieved by extending the number of layers and incorporating nonlinear activation functions. A closely related architecture is variational autoencoder [110], which can be considered as generative ML because it produces new outputs from the reduced-dimension latent space representation of the input data.

The power of autoencoders to learn nonlinear manifold coordinates to represent the data can enable achieving improved compression in the latent space [97]. As such, autoencoders can play a crucial role in ROM development and, hence, we will have a closer look at these architectures as well in section 2.2.

(3) **Semi-supervised Learning**: The algorithms belonging to semi-supervised learning learn under partial supervision, meaning that either the amount of labeled data is limited, or the corrective information are received directly from the learning environment. Generative adversarial networks (GANs) [111] and Reinforcement Learning (RL) algorithms [112] are the main semi-supervised learning algorithms.

i. **Generative adversarial networks (GANs)**: GANs are learning algorithms designed to produce generative models capable of generating data that closely resembles the distribution of the training data. The essence of GANs lies in a dynamic interplay between

two neural networks: the generative network and the discriminative network. In this setup, the generative network generates candidate data samples, while the discriminative network evaluates these samples to distinguish between real and synthetic data. The training process resembles a zero-sum game [111], where the generative network aims to produce data examples that deceive the discriminative network into misclassifying them as real data. Conversely, the discriminative network tries to accurately differentiate between real and generated data. Through this adversarial learning process inspired by game theory, the networks iteratively refine their parameters to improve their performance.

The adversarial training paradigm inherent to GANs allows the networks to learn from each other without the need for explicit supervision, leading to a so-called "self-supervised" learning process. While this training approach adds to the interest and appeal of the GANs, there remains a challenge in ensuring convergence to an equilibrium point in the adversarial game. Although larger datasets can enhance the training process, convergence is not guaranteed [96]. This highlights the need for further research on GANs.

ii. **Reinforcement Learning (RL)**: RL is a mathematical framework [112] designed for solving complex problems through goal-directed interactions between an agent and its environment. Unlike supervised learning, where the agent learns from labelled information, RL operates in a semi-supervised manner, where the agent learns from its own experiences, guided by infrequent and partial rewards. The primary objective of RL is not merely to uncover patterns in actions or in the environment but to maximize long-term rewards. In RL, two fundamental components are the agent's policy and the value function. Agent's policy defines the mapping between states of the system and optimal actions. The value function assesses the utility of reaching a particular state for maximizing long-term rewards.

RL poses significant computational challenges due to the large number of episodes required to effectively account for the agent-environment interaction. Additionally, a core challenge in RL is the long-term credit assignment problem, especially when rewards are sparse or delayed over time [96]. This challenge involves inferring causal relations between individual decisions and rewards from a long sequence of states and actions. Efforts to address these issues include augmenting sparsely rewarded objectives with densely rewarded subgoals [113], aiming to enhance the learning process and facilitate effective decision-making in complex environments.

Following the introductory discussions above on the ML fundamentals and the main categories of architecture, we next review the primary contexts for the application of ML to physical systems and, in turn, look into the applications particularly related to systems' modelling.

2.1.1. Primary areas of ML applications to physical systems

When applying ML algorithms to physical systems, we may intend to solve several types of problem. One problem type can be the discovery of new physics and/or new physical correlations/relationships, for which governing equations may not exist or that may be difficult to develop using first-principles.

Model discovery, where the aim is to derive models describing the dynamics of a system under study, is one of the ML applications that falls, at least partly, within this category.

Another problem type is resolving cost-efficiently physics that are expensive to capture. ROM and surrogate model developments are prominent examples of ML applications in this category of problems. ML can also be used to complement an existing system of equations for a physical system by finding relations and/or models for the physics that are not captured, hence, reconciling the simulations and the real-world behaviors. This application (problem type) is often referred to as “discrepancy modelling” [114], where we may, e.g., aim to find “closure” terms for the system of equations or intend to augment the system of equations by enabling it to capture multiphysics interactions.

Selection of the problem type that we intend to solve using ML consequently affects the data that needs to be curated as well as the choice of the ML architecture from the general categories of architecture reviewed above. Two aspects closely linked to ML architecture selection are the definition of a loss function (for instance, an L2-norm error between the predictions and the ground-truth for supervised learning) and the choice of an optimization algorithm. The optimization acts to minimize the defined the loss function during the architecture’s training by adjusting the ML model parameters.

It is emphasized that, in the recently emerged framework of “physics-informed” ML [98], the primary objective is to embed prior knowledge of physics into the learning machine so as to achieve higher levels of generalizability, robustness, and interpretability. As were pointed out in section 1.5, conservations, symmetries, and invariances constitute various ways by which physics can be incorporated into ML. Another relevant consideration for physics embedding is the concept of model “parsimony”, which is inspired by the observation that all generalizable and interpretable physics laws we currently have (Newton’s second law or Maxwell’s equations, e.g.) feature as few terms as necessary to describe the relevant phenomena. The incorporation of the physics into ML can be manifested within the loss function definition and/or within the optimization algorithm. The former embedding acts to “promote” physics because the physics-related loss term(s) would be competing against other loss factors, for instance, the L2-norm error. Whereas, the latter physics embedding approach serves to “enforce” physics in the learning process of the machine. Nevertheless, embedding physics within the optimization procedure is usually effort-intensive.

When considering physical knowledge embedding within ML as part of the architecture’s loss function, a well-known and prevalent family of ML algorithms are physics-informed neural networks (PINNs) [96][115][116]. Pioneered by Raissi et al. [115], PINNs offer an elegant approach to solving partial differential equations (PDEs) by incorporating the satisfaction of the underlying physics of the problem directly into the loss function of a neural network. This method leverages deep learning principles to tackle complex PDEs, utilizing automatic differentiation within the back-propagation algorithm to compute partial derivatives and formulate the equations to be enforced through the loss function. PINNs have been used for a wide variety of applications, including turbulence modeling [117], developing ROMs [118], and accelerating traditional solvers [119]. PINNs have shown notable capability in handling noisy data [120], enabling robust predictions in scenarios where data uncertainty is present. One notable challenge with PINNs is their underlying assumption that the system of

equations describing a system's evolution is fully known so that satisfying the equations can be included within the loss function of the architecture [121]. Despite this, PINNs can be effective for discrepancy (closure) modelling in case the problem can be formulated as an inverse problem or system identification, which PINNs can handle [121].

2.1.2. Select examples of ML applications to computational modelling

When it comes to the integration of ML into computational modelling, we can distinguish three promising pathways, along each of which interesting and insightful efforts have been carried out over the past years: **(1)** accelerating high-fidelity, computationally heavy simulations, **(2)** closure model development (more broadly, discrepancy modelling), and **(3)** reduced-order modelling. We briefly discuss each of these pathways here and review some of the most relevant prior research from fluid mechanics, aiming to paint an overall picture of the opportunities for cross-disciplinary transfer of ideas and advances into computational plasma physics.

2.1.2.1. Accelerating high-fidelity simulations

Starting with the acceleration of high-fidelity simulations, several efforts in fluid dynamics have focused on improving the cost-efficiency of direct numerical simulations (DNS). DNS is the highest fidelity modelling approach in computational fluid dynamics (CFD), which aims to solve the Navier-Stokes system of equations for the fluid flow across all relevant spatiotemporal scales, from the smallest turbulence scale to the largest and, hence, requires very fine-resolution grids. The computational cost of a DNS simulation roughly scales with the third power of the Reynolds number [122] and, thus, the method is computationally burdensome even for moderate values of Reynolds number. Accordingly, researchers have aimed to accelerate these simulations using ML along two main paths: **(i)** improving the discretization schemes, and **(ii)** developing correlations to link fine- and coarse-grid simulations.

Before looking at some examples of prior work along each path, it is important to underline one point. For plasmas, it is often difficult for a fluid system of equations to properly describe the behaviors and the dynamics. In many applications, plasmas are in thermodynamic nonequilibrium, exhibit strong anisotropies, and their global response is majorly affected by the underlying instabilities and/or turbulence. As a result, with some exceptions such as the case of ideal magnetohydrodynamics (MHD) [123], a system of equations for plasmas that may (partly) resemble Navier-Stokes will not be able to generally describe a wide range of plasma conditions. In fact, plasma fluid systems of equations for most practical applications and scenarios lack closure models to represent the missing kinetic effects (moment closures) and the nonclassical effects, such as the influence of instabilities (transport closures).

In any case, methods to accelerate DNS using ML may still be applied in plasma physics modelling from different perspectives. In particular, MHD simulations, where applicable, and other high-fidelity grid-based models, such as direct-kinetic simulations that aim to solve the Vlasov-Maxwell equations in an Eulerian manner [124], can benefit from ML-enhanced discretization schemes as well as ML-enabled correlations to map the results of a low-resolution simulation to a high-resolution one.

With the above remarks in mind, some notable examples of accelerating high-fidelity simulations using ML approaches in fluid dynamics are now reviewed. In this regard, toward improving the discretization schemes, the method of “stencil learning” was pursued by authors such as Bar-Sinai et al. [125] and Zhuang et al. [126], who used deep learning to estimate the spatial derivatives such that the learned numerical discretization retains high accuracy even with low-resolution computation grids. Stevens and Colonius [127][128] and Jeon et al. [129] worked to enhance the accuracy of the classic finite-difference and finite-volume schemes using ML and deep learning. In particular, Stevens and Colonius proposed the FiniteNet [128] – an architecture based on Convolutional and LSTM neural networks – to improve solving time-dependent PDEs.

Toward developing correlations to link fine- and coarse-grid simulations, a prominent work has been carried out by Kochkov et al. [130], who pursued the ideas of super-resolution from image processing to derive a deep-learning-based correction that enables resolving detailed features of a turbulent flow evolution with a coarse-grid simulation (Fig. 2.1). Their method enabled capturing the results of a high-resolution reference simulation with mesh resolutions that were 8-10 times coarser along each dimension [130]. A similar work has been performed by Fukami et al. [131], who employed CNNs to reconstruct high-resolution turbulent flow fields from highly under-resolved field data. In addition, there have been other efforts to improve the performance of PDE solvers on coarse grids, e.g., by Li et al. [132].

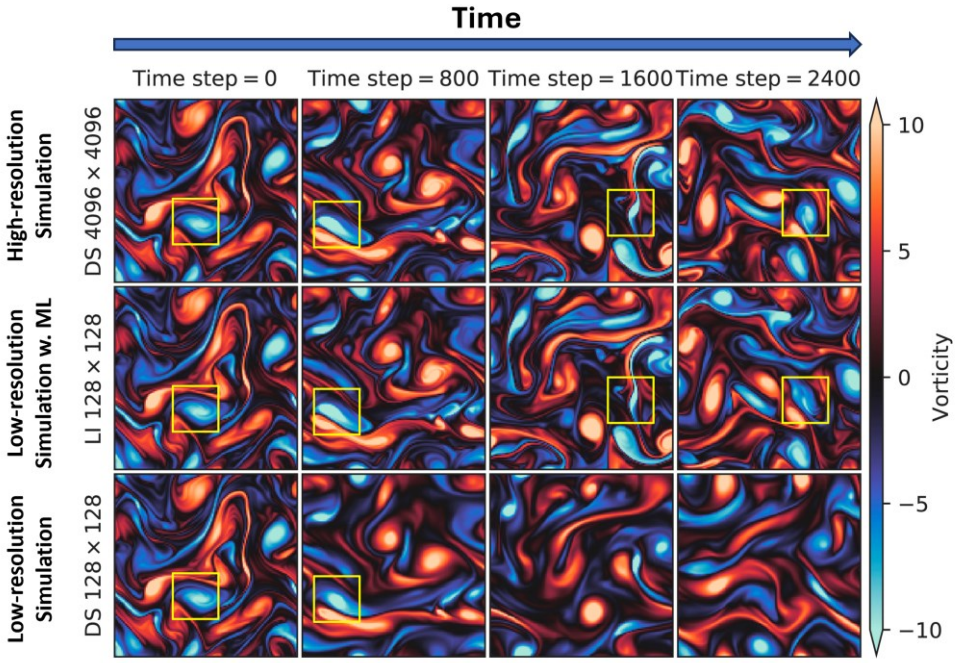


Fig. 2.1 Sample figure from the work of Kochkov et al. [130] (super-resolution acceleration of numerical simulations). The plot shows the time evolution of the vorticity field from simulations with a fine (**top row**) and a coarse grid (**bottom row**) as well as a coarse-grid simulation enhanced with ML (**middle row**). The vorticity field evolution is shown at four different time instants, and some key vortical structures in the flow are specified by yellow square boxes. Reproduced from Ref. [130], available under CC-BY NC-ND license.

It is noteworthy at this point that, although traditional kinetic PIC simulations for plasma physics do not provide any evident pathway to use the super-resolution ideas – due to the stringent requirements

on their spatiotemporal resolutions – the reduced-order PIC [81][82] (section 1.4) can greatly benefit from super-resolution ML enhancements in order to further augment the cost-efficiency of the approach. In fact, as the computation grid associated with the reduced-order PIC comprises “regions” that can amount to a coarse mesh, super-resolution ideas, such as those reviewed above from fluid dynamics, may be used to account for the details of the physics that might be lost due to the order-reduction treatment. Further discussions on the prospects of this approach are provided in Ref. [83].

Before moving on, it is pointed out that ML has been employed as well to solve Poisson-like equations, which are encountered in fluid dynamics, and are evidently of high relevance in plasma physics. For example, Tang et al. [133] have assessed developing deep-learning-based Poisson solvers, and Ajuria et al. [134] have applied deep learning to solve the Poisson-type corrector step in the incompressible flow simulations.

2.1.2.2. Discrepancy modelling and closure model development

Discrepancy modelling involves addressing errors and inaccuracies in traditional (first-principles) physics-based models used for simulations and predictions. These errors can, in part, arise from missing physics, which, in turn, can significantly affect the predictions of complex physical systems using the models. ML techniques have, thus, been developed to learn models for these discrepancies. For example, Levine and Stuart [135] recently proposed leveraging recurrent neural networks (RNNs) to model hybrid problems in non-Markovian settings, demonstrating the potential of ML in capturing missing physics and, hence, improving dynamical models.

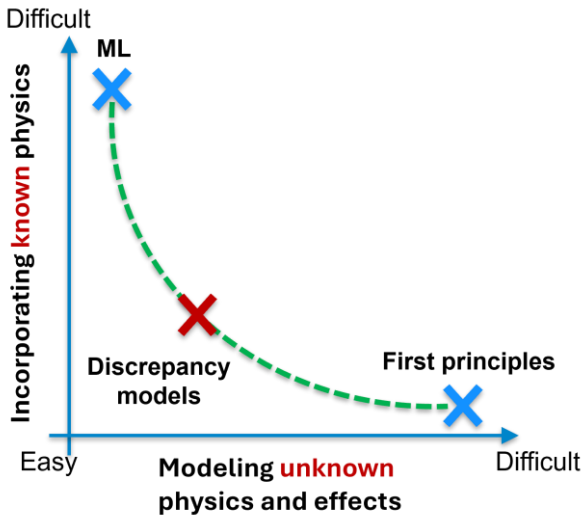


Fig. 2.2 Schematic diagram of discrepancy modeling as an alternative path to inform ML-based dynamics discovery with known physics so as to facilitate learning unknown physics and effects that may be difficult to model via first-principles methods.

By augmenting known first-principles models with discrepancy models learned from data, it is possible to improve the accuracy and reliability of dynamical models (refer to Fig. 2.2 and the description in the figure’s caption). Historically, early models of systems were often coarse approximations of the underlying physics, leading to discrepancies between the model and reality. While computational advancements have enabled more detailed simulations (models), model-measurement mismatch has

remained a challenge across various fields. This challenge has been especially pressing for plasma systems due to the complexities of the physics in realistic settings, on the one hand, and the computational burden and/or incompleteness of the plasma simulation tools and models, on the other.

In modern applications, discrepancy modelling has become increasingly relevant, especially for use cases such as digital twins, where precision is crucial. As we will return to in section 7.2.3, predictive and scalable digital twin technologies will rely on computational models. However, if derived from first-principle physics solely, these underpinning models will at least in plasma physics fail to accurately match reality. This is where discrepancy modelling becomes crucial because it enables improving the models in order to ensure that they align increasingly better with real-world observations.

ML techniques, such as neural networks, have emerged as powerful tools for learning the discrepancy models directly from data, providing a data-driven framework to address model inaccuracies and enhance predictive capabilities. In this respect, whereas Levine and Stuart [135] derived discrepancy models using NNs in a dynamical representation, e.g., in the form of differential equations, Ebers et al. [114] considered a broader class of ML/data-driven techniques and also explored learning a discrepancy model in the state space. Their work demonstrated the versatility and potential of different DD approaches in capturing and modelling discrepancies in physical systems [114].

Developing closure models for the system of governing equations can be considered as a subcategory of discrepancy modeling. Data-driven closure finding represents significant opportunities for plasma systems. This is particularly because the closure problem is an active subject of research in computational plasma physics, and that several approaches pursued so far [23][25][26][70][71][136] have not yet succeeded in achieving generalizable and self-consistent closure models for the plasma fluid equations. This is whereas plasma fluid descriptions are of interest due to their higher computationally affordability compared to kinetic simulation approaches, such as PIC.

In the most general terms, the closure problem in plasma physics refers to two aspects: first, models that can represent higher-order moments of the particles' distribution function (capturing deviations from an isotropic, Maxwellian distribution function [67][137]) in terms of the lower-order moments (density, mean velocity, and pressure tensor); this is referred to as "moment closure". The second aspect of the closure problem is to develop models that resolve the nonclassical effects of plasma phenomena, such as instabilities and turbulence, on the momentum and energy transport of plasma particles; this aspect of closure can be called "transport closure".

In fluid mechanics, when the Navier-Stokes equations are valid (continuum fluids in thermal equilibrium), the closure problem is often synonymous to finding a turbulence model that obviates the need for highly fine-resolution DNS simulations. In this respect, turbulence models for the Reynolds-averaged Navier-Stokes (RANS) equations try to correlate the so-called "Reynolds stress tensors" to the mean flow properties. Turbulence closure models for the Large Eddy Simulations (LES) aim to resolve the effects of the flow turbulence that cannot be captured because they fall below the grid resolution. In any case, moment closure problem can also arise in fluid mechanics, for instance, in case of the rarified and/or non-equilibrium gases, for which the Navier-Stokes equations can become less valid [138].

Accordingly, despite certain differences in what constitutes a closure problem in fluids vs plasmas modelling, the ML applications toward closure finding in fluid dynamics can still be relevant and beneficial in plasma physics domain. In this regard, two specific examples of prior work in fluid mechanics are reviewed. A more detailed, excellent review of ML for closure modelling of fluids is provided by Duraisamy et al. in Ref. [139].

The first example showcases how the incorporation of partial knowledge of the physics improves the accuracy and performance of the ML-derived closure models. Ling et al. [140] developed a deep neural network architecture that featured a multiplicative layer with an invariant tensor basis in conjunction with the last hidden layer of the network before a merged output layer [140]. The inclusion of this multiplicative layer enforced the closure model learned for the Reynolds stress tensor to respect Galilean invariance, meaning the model was valid in any inertial frame of reference. By embedding this physics into the network, the authors showed an improved performance of their RANS closure model, better than any traditional model, even for the cases that are challenging to predict with the RANS models [140].

As for the second example, Novati et al. [141] followed an agent-based approach (Reinforcement Learning) to estimate and determine more accurately the free parameter (coefficient) of a subgrid scale (SGS) model in a coarse-grid LES simulation. Several RL agents were deployed at various fixed grid locations within the simulation domain. These agents computed the values of the SGS model's coefficient at their grid locations over time and learned under the influence of a policy and a reward. The policy depended on the state of the agents that incorporated both local and global variables [141]. The reward, which was also dependent on the local and global variables, was received based on the accuracy of the simulation [141]. The RL-based approach pursued by Novati et al. [141] exhibited good generalization characteristics across different grid sizes and flow conditions. The outcomes from this work open interesting prospects for closure finding and subgrid modelling, as appropriate, in plasma simulations' context.

2.1.2.3. Reduced-order modelling

Reduced-order modelling is a powerful technique, which is used to simplify the descriptions of complex systems by representing their essential dynamics in a lower-dimensional space, typically by focusing on dominant coherent structures underlying the system's evolution. The ROMs are intended to provide the best trade-off between accuracy and computational cost (refer to Fig. 2.3 and the descriptions in the caption). As these models are faster alternatives to high-fidelity, more expensive simulations, they can enable rapid predictions for tasks such as optimization and control. Traditionally, the great computational efficiency of ROMs has come at the cost of the models having limited range of applicability. However, enhancing the generalizability of ROMs with ML is currently an active area of research, toward which embedding prior and/or partial knowledge of the physics is believed to play a critical role [97]. Interpretability (through the principle of parsimony, in particular), reduced data demands, and physics consistency comprise additional targets to achieve with ML-enabled "physics-informed/inspired" ROMs.

Generally speaking, ROM development consists of two main steps: first, finding a suitable *reduced* coordinate system (set of dominant modes) and, second, discovering a dynamical system model (a set of ordinary differential equations (ODEs) for instance) in the identified reduced coordinates. For example, from high-fidelity, high-dimensional simulation data, a dimensionality-reduction technique, such as POD, can extract the dominant modes (structures) underlying the data. Then, a dynamics model can be constructed that describes how the amplitudes of these dominant modes evolve over time. Such a model is reduced order because instead of needing to resolve a large number of degrees of freedom, it now just resolves the time evolution of the amplitudes of a few spatial coherent structures.

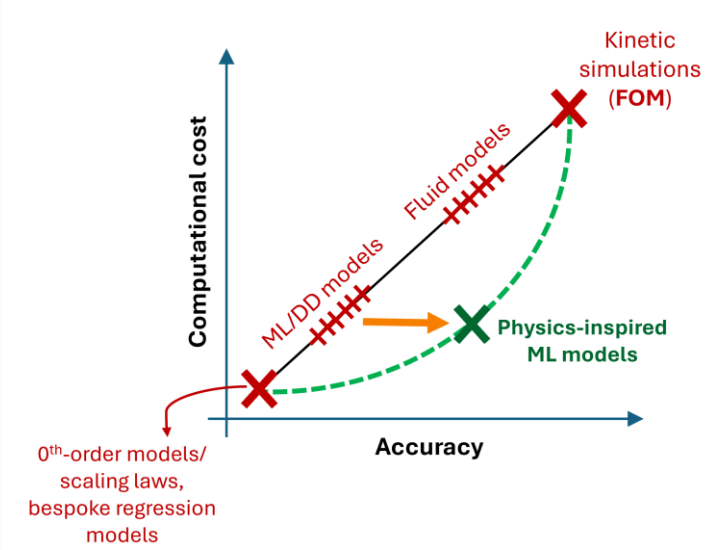


Fig. 2.3 Graph of the computational cost vs accuracy for various possible plasma physics models. The kinetic simulations represent full order models (FOMs), whereas fluid models and ML/DD models are reduced-order models (ROMs). Incorporation of "physics-inspired" constraints [in particular, low-dimensionality and sparsity (parsimony, subsection 2.3.1)] into learning can promote accuracy (and generalizability) while maintaining computational cost at the level of ML/DD models purely based on data.

In the case of plasmas, the view toward reduced-order modelling shall be rather more nuanced compared to the descriptions above. Plasmas generally evolve within a 6-dimensional phase space and over time. As a result, if one succeeds in establishing a closed 3D system of partial differential equations for a given plasma system, such system of PDEs can still be interpreted as a ROM. As a result, we can argue that deriving a dynamical system model directly from the high-fidelity data in the high-dimensional space in order to construct a set of governing PDEs amounts to reduced-order modelling as well for plasmas. Nonetheless, a 3D system of PDEs would still correspond to a significant computational cost (Fig. 2.3), preventing the model's applicability to design optimization and control tasks, e.g. Therefore, deriving predictive ODE-like ROMs for plasmas is warranted and in fact necessary for most practical applications.

It is emphasized as well that there are classical approaches for reduced-order modelling, too. As was also mentioned in section 1.5, one widely employed approach in ROM is the POD-Galerkin method. POD learns a low-dimensional coordinate system from data, enabling the representation of the dominant structures. A dynamical system is obtained within the POD-identified reduced subspace via

performing a Galerkin projection of the governing equations (the Navier-Stokes equations for fluids e.g.), onto these modes. While the POD step is data-driven, Galerkin projection requires a functional numerical implementation of the governing equations [97]. Additionally, it often necessitates custom modifications to the numerical solver, making it rather intrusive [97]. The advancements in ML and data-driven techniques allow us to expand the capabilities of the classical ROM development methodologies. Indeed, in the two subsequent sections, we will see how ML can aid each of the two main constituting elements of reduced-order modelling, namely, dimensionality reduction (coordinate transformation) and dynamics discovery (system identification).

2.2. ML for dimensionality reduction and coordinate transformation

Enhancing the coordinate system used to represent the reduced dynamics serves as a major opportunity for integrating ML into ROM development. In fact, POD [142] and the underpinning concept of Singular Value Decomposition (SVD) provide an orthogonal set of modes, comprising a linear subspace for approximating data. However, many physical systems evolve on nonlinear manifolds. ML and deep learning, thus, present a potent method for generalizing dimensionality reduction from learning linear subspaces as with POD/PCA/SVD to learning coordinates on curved manifolds.

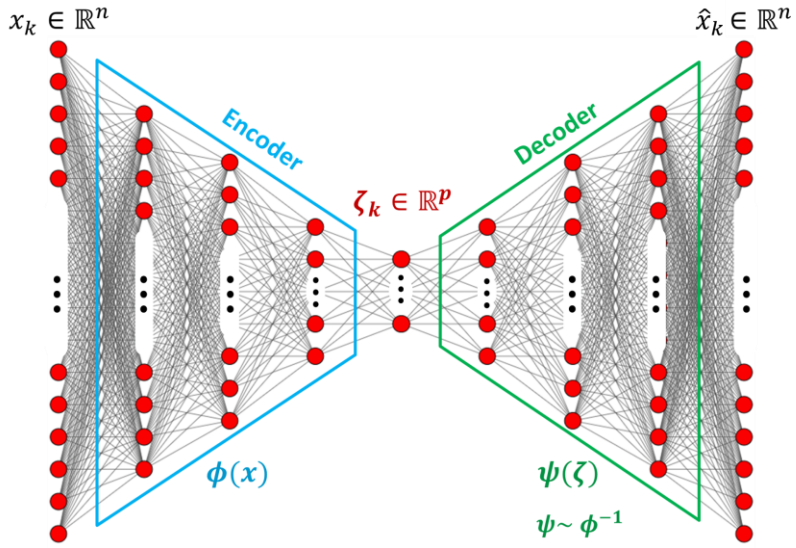


Fig. 2.4 Schematic representation of an autoencoder network. The encoder ($\phi(x)$) reduces the dimensionality of the data ($x_k \in \mathbb{R}^n$), finding the latent space variables (ζ_k) belonging to a reduced subspace $\mathbb{R}^p$, with typically $p \ll n$. The decoder ($\psi(\zeta)$) maps the latent variables of the autoencoder back to the high-dimensional $\mathbb{R}^n$ space. The network is trained by minimizing the loss between the input data to the encoder (x_k) and the output data of the decoder ($\hat{x}_k$).

In particular, autoencoder networks [97] open viable pathways to learn nonlinear data embeddings. Fig. 2.4 shows a schematic representation of an autoencoder with multiple hidden layer (7 in this case). The hidden layers of an autoencoder can include fully connected layers with nonlinear activation functions, convolutional layers, etc.

As is shown in Fig. 2.4, autoencoders feature an input and output layer that matches in size the high-dimensional system state and a bottleneck hidden layer that reduces the state to a low-dimensional

space, consisting of one or more latent variables (ζ). The encoder part (ϕ) of the network maps the high-dimensional state (x_k, k being a time instant for time-series data, e.g.) to the latent space. Autoencoders belong to the unsupervised learning category. As a result, even though it is the latent space representation at the bottleneck of the autoencoder that we are interested in, the decoder part that reconstructs an estimate of the high-dimensional state ($\hat{x}_k$) from the latent space is crucial for the unsupervised training of the network. A common loss function for the autoencoder architectures is the L2-norm error between the decoder's state reconstruction and the original input state.

A shallow autoencoder with single-layer encoder and decoder as well as linear identity activation functions is closely related to SVD/POD. However, the modular nature of the autoencoders, similar to any neural net, enables expanding the number of layers as well as employing nonlinear activation functions. Hence, the network can learn nonlinear manifold coordinates, potentially improving compression in the latent space.

The use of autoencoders have gathered significant attention from across sciences, including in fluid mechanics [143][144]. Lee and Carlberg [145] demonstrated the effectiveness of deep convolutional autoencoders in enhancing classical ROM techniques based on linear subspaces [146]. Additionally, promising data-driven methods for constructing ROMs on spectral submanifolds (SSMs) have been proposed [147].

Clustering and classification are not directly related to dimensionality reduction but are used in innovative ways to aid reduced-order modelling. Classification algorithms, such as the K-nearest neighbors (KNN), can enable detecting and categorizing various dynamical regimes and/or operating conditions of a system. As a result, a potential application may be building a library of ROMs from which a specific relevant ROM compatible with the dynamics regime of the system as represented by the data and distinguished by the classifier would be chosen and used for prediction and/or control.

With regards to clustering, the k-means algorithm has proven effective in various fluid dynamics applications in particular. K-means were employed by Kaiser et al. to develop a data-driven discretization of the high-dimensional phase space for the mixing layer of a fluid flow [148]. Their approach yielded a low-dimensional representation (a small number of clusters). This facilitated the creation of tractable Markov transition models for the flow evolution from one state to another over time. Amsallem et al. [149] used k-means clustering to partition the phase space into distinct regions. Within these regions, local reduced-order bases were constructed, leading to enhanced stability and robustness against parameter variations.

The above utilizations of k-means clustering highlight the potential applicability areas of clustering in general for reduced-order modelling and analysis.

2.3. Data-driven discovery of the dynamics

There are many ML architectures and data-driven algorithms that can be used to learn from data how the state of a certain system evolves over time, i.e., to construct a model for the system's dynamics. Many, if not all, of these architectures and methods can be combined with dimensionality-reduction

techniques, such as POD and autoencoders, to discover the dynamics in a suitably identified reduced coordinate system (subspace of underlying modes/structures), hence, yielding computationally efficient reduced-order models.

In this section, a high-level overview of various categories and examples of linear and nonlinear dynamics discovery algorithms is first provided, followed by a closer look at some of the promising and interesting methods and architectures in subsection 2.3.1.

Development of linear-dynamics models or linear representations of the nonlinear dynamics are of particular appeal in science and engineering because linear systems offer analytical tractability and there are a wide range of established techniques for their evaluation and analysis. Two well-known data-driven approaches that aim to construct linear time-dynamics models/representations for physical systems are the dynamic mode decomposition (DMD) [150][151] and the more general, closely related framework of Koopman analysis [152].

Various neural network architectures enable effectively capturing nonlinear dynamics. Some notable examples include the LSTM [153] and echo-state networks [109] along with GANs [111]. Among these, the applications of GANs have been limited but promising enough to warrant further exploration [154]-[156]. Alternative regression techniques exist as well to derive nonlinear dynamical system models. Cluster Reduced-Order Modeling (CROM) [148] has been demonstrated as a straightforward yet robust unsupervised learning method. The CROM approach decomposes time-series data into a few number of representative clusters and then yields models for the transition-state probability between these clusters. Another approach is the operator-inference method [157], which resembles Galerkin projection and aims to learn neighboring operators from data. Sparse Identification of Nonlinear Dynamics (SINDy) [158] learns a parsimonious model by fitting the observed dynamics from data to the fewest terms in a library of candidate functions.

2.3.1. Closer review of several dynamics discovery methods

A. Dynamic mode decomposition: DMD is a data-driven technique that was introduced by Schmid [150] in fluid dynamics to decompose a flow field into series of coherent spatiotemporal modes with exponential time dynamics. Due to its simplicity of implementation and interpretability, the DMD method has gained remarkable popularity and has been used in various applications, particularly in the field of fluid dynamics [152][159]-[163].

Several variants of the DMD method have emerged over the years as well, including multi-resolution DMD (mrDMD) [164], high-order DMD (HODMD) [165], and parametric DMD [166].

The recent developments of the optimized DMD (OPT-DMD) [167] and the bagging optimized DMD (BOP-DMD) [168] algorithms have significantly changed the potential of DMD to perform critical modeling tasks due to the robustness of these new algorithms to noise. Importantly, OPT-DMD can be used to constrain best-fit linear models to be stable by construction.

Physics-informed DMD (Pi-DMD) [169] aims to constrain the DMD's linear regression of the data by imposing known physics of the problem such as the existing symmetries. The constrained regression underpinning Pi-DMD limit the search space of the optimization to the subspaces (or

the manifolds) that respect the symmetry of interest. This has been shown in Ref. [169] to yield significant performance improvements over the basic, unconstrained DMD implementations.

The DMD method presents major promises for plasma physics applications, in particular with regards to the extraction of spatiotemporal coherent structures (instability modes, e.g.) as well as for reduced-order modelling. As a result, in Chapter 3 and Chapter 4, we focus on demonstrating these promises in several test cases that represent different plasma configurations and involved dynamics.

B. **Koopman analysis/transformation**: Many dynamical systems of applied interest exhibit nonlinear and/or transient dynamics. The Koopman analysis framework, based on the Koopman theory [170], aims to find a coordinate transformation (embedding) in which the nonlinear dynamics can be linearly represented.

The Koopman operator, a central concept in this framework, involves defining an infinite-dimensional operator that acts on observables (functions) of the state vector from a finite-dimensional nonlinear dynamical system. By applying this operator to suitable observables, the dynamics can be described linearly in the observables space. The Koopman operator lifts the dynamics from a finite-dimensional state space to an infinite-dimensional function space, where the evolution of the original system becomes linear. Despite being an abstraction, the Koopman theory suggests that such a mapping exists, although identifying the appropriate observables to achieve this linear representation is nontrivial and a subject of ongoing research. In this regard, several researchers, e.g. in Refs. [171]-[174], have attempted using deep-learning architectures and autoencoders, like the one illustrated schematically in Fig. 2.5, to identify nonlinear observables and/or coordinate embeddings that may realize Koopman linear representation of nonlinear dynamics.

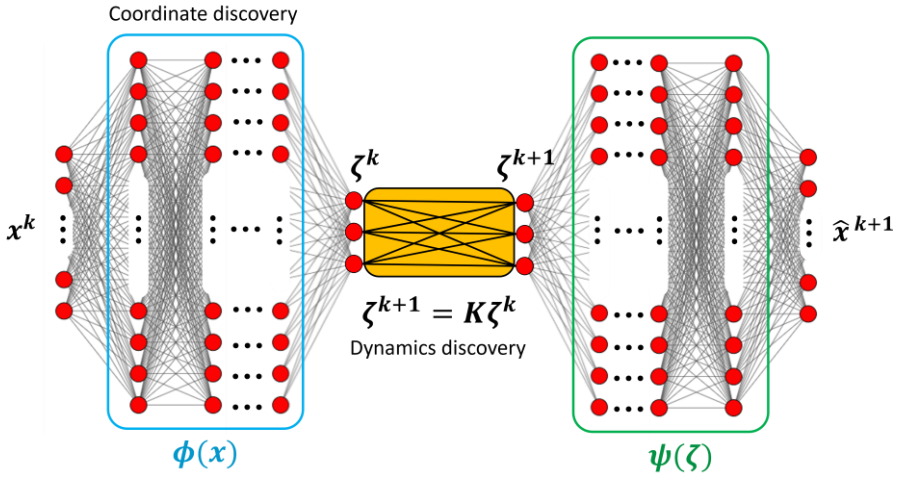


Fig. 2.5 A schematic representation of deep Koopman networks for the simultaneous discovery of the linear embedding of nonlinear dynamics (Koopman transformation) and the associated time-stepping operator K.

The DMD method is closely related to the Koopman analysis framework, and efforts have been made to extend DMD with nonlinear observables [175][176] in order to find an approximation for the Koopman operator.

The Koopman analysis and the related DMD implementations serve as powerful tools for ROM development of nonlinear systems and, thus, these ideas are further described in subsection 3.2.1.1 along with presenting some results for plasma systems in Chapter 4 and Chapter 5.

C. **Parsimonious (sparse) nonlinear models**: ML/DD techniques are increasingly utilized to develop parsimonious nonlinear models that strike a balance between predictive accuracy and model complexity. This is meant to mitigate overfitting while ensuring interpretability and generalizability of the derived model.

One approach to parsimonious ROM development is "symbolic regression" based on genetic programming [177]. Genetic programming within symbolic regression works by generating random populations of mathematical expressions, which are then evolved over successive generations through processes such as mutation, recombination, and selection to optimize a specified fitness criterion ("survival of the fittest"). Schmidt and Lipson in 2009 demonstrated genetic programming for the discovery of conservation laws and governing equations [178].

Another approach involves sparse regression, with one of most prominent examples being the SINDy algorithm. SINDy has been employed to identify various dynamical systems [158] as well as partial differential equations [179][180]. Particularly relevant to the paradigm of physics-informed/constrained ML, Loiseau and Brunton in 2018 identified sparse ROMs for various flow systems while enforcing energy conservation as a constraint [181].

Both symbolic regression and sparse model identification methods like SINDy utilize Pareto analysis to identify models that offer the optimal trade-off between the predictive accuracy and model complexity as quantified by the number of included terms. Remarkably, in scenarios where the underlying physics is known, these approaches often uncover the correct governing equations, leading to superior generalizability compared to other prominent machine learning algorithms [96].

Nevertheless, contrary to the computationally expensive symbolic regression, SINDy offers a more cost-effective alternative via sparse regression on a predefined library of candidate terms, which can potentially describe a system's dynamics [158]. Thus, some more words are spent in the following to further describe SINDy.

The SINDy algorithm uses time series data to extract dynamical system models best describing the data. As was pointed out above, this is done by performing a sparse regression on a library (Θ) of candidate linear and non-linear terms such that the resulting dynamical system model best approximates the time derivatives ($\dot{X}$) of a system's state variables (X). The sparse regression finds the coefficient values of the minimum number of "active" terms in matrix Θ. The terms in matrix Θ can include arbitrary functional forms of the state variables, such as polynomials, spatial derivatives, and/or any combination of these. The derivatives are computed from the data following standard discretization approaches like finite differencing or equivalent.

The derived dynamical systems from SINDy can take the form of PDEs as well as ODEs. In the latter case, SINDy is coupled to an autoencoder (see Fig. 2.6). The autoencoder finds the latent space of the data, and SINDy then learns an ODE in that space which best describes the time evolution of identified dominant underlying modes [coefficient values (a_is) in Fig. 2.6] [182].

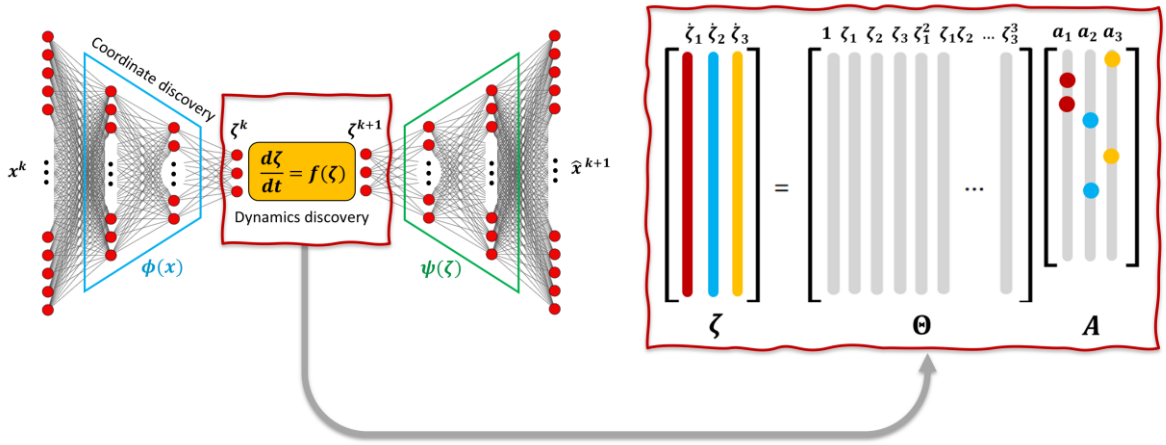


Fig. 2.6 Schematic visualization of the SINDy algorithm as the dynamics discovery method in the latent space of an autoencoder.

Multiple variants of SINDy have been developed since the algorithm's introduction in 2016 [183]-[188], and the approach is used across science and engineering applications (e.g., in Refs. [189]-[191]). Recently, SINDy has been combined with Bayesian autoencoders in order to augment the simultaneous discovery of the latent space and the dynamics with uncertainty quantification [192].

D. **Shallow Recurrent Decoder (SHRED)**: SHRED is an ML framework introduced in 2023 [193][194] that optimally combines the power of the LSTM networks for dynamics discovery (time-series learning) with the reconstructive advantages of the decoders.

The aim of SHRED is to establish a mapping between the time history of sparse, minimal sensor measurements (at three randomly placed spatial points) and the high-dimensional spatiotemporal data [194]. To this end, the LTSM learns the time dynamics from the sensors' trajectory information (time-history measurements), whereas the decoder learns the spatial mapping between the latent output space of the LSTM and the high-dimensional state space of the data. In practice, the latent outputs of the LSTM are mapped to a compressive representation of the high-dimensional data as encapsulated within the rank-r truncated matrix of right singular values (V) from singular value decomposition (SVD) of the data ($X = U\Sigma V^*$), where the superscript * denotes conjugate transpose (refer to section 3.1 for more information). This mapping to compressive (truncated) representation improves the computational efficiency and overall performance of the SHRED architecture.

The extracted latent space representation from the LSTM network is agnostic to sensor placement, which is a great advantage of SHRED over other state-of-the-art methods with similar purposes [194]. Moreover, the fully connected nonlinear decoder architecture is by construction noise-robust, hence, drastically improving the quality of reconstruction over linear methods, such as POD.

Fig. 2.7 shows a schematic diagram of the SHRED architecture, illustrating how it can reconstruct full spatial maps of various plasma properties from the time history of local probe measurements of a single property – in this case the electron number density (n_e). Local density

probes collect measurements at three spatial points, which are then processed through an LSTM network. The latent outputs of the LSTM are passed to decoder networks, with each decoder mapping the LSTM's latent variables to the V matrix of the SVD of the data corresponding to a specific plasma property. The reconstructed V matrix for each property is projected to the high-dimensional space using the precomputed SVD modes of that property.

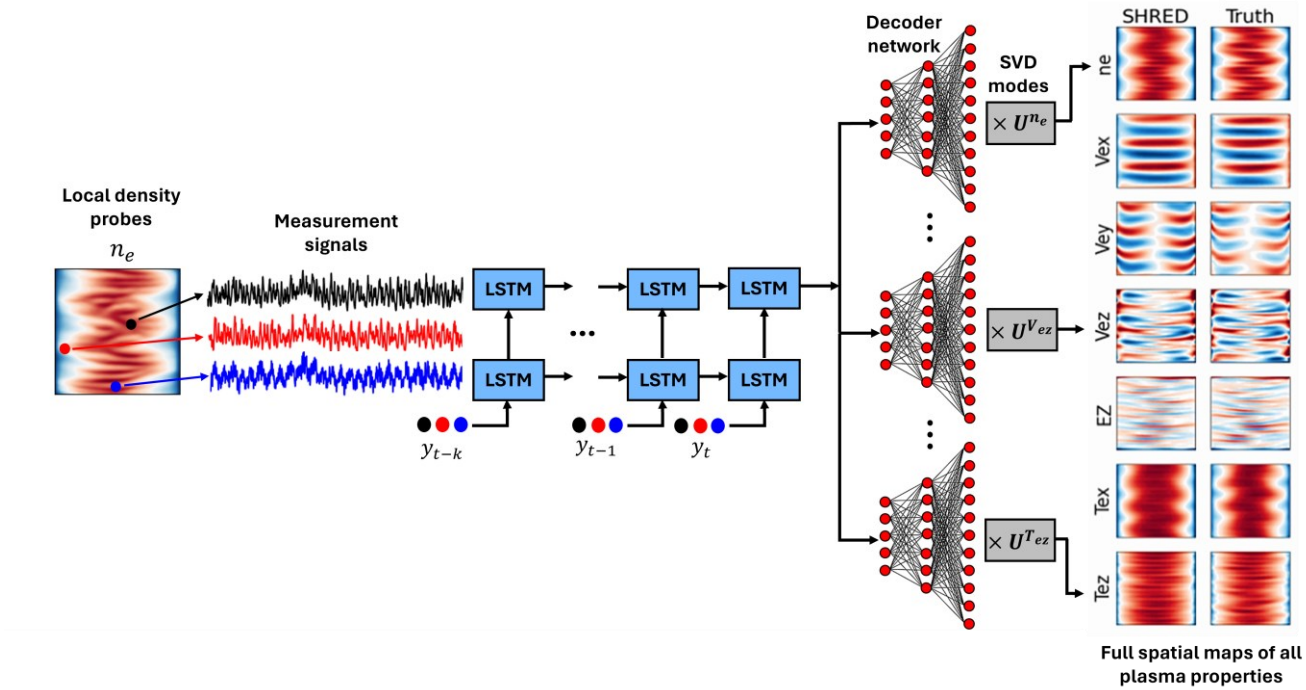


Fig. 2.7 Schematic visualization of the SHRED architecture as applied to data from the simulation of a 2D plasma discharge representative of a radial-azimuthal cross-section of a Hall thruster [48][195]. Reconstruction of full spatial maps of all plasma properties from the time history of local probe measurements of a single property.

The SHRED approach can be thought of as the nonlinear ML generalization of the classical method of separation of variables for solving linear PDEs [195]. In this respect, the incorporation of the time-history measurements from the sensors at specific spatial locations into the learning process (system identification problem) serves to make the solution unique, in much the same way that the general form of the solution to a linear PDE from separation of variables is anchored by imposing the initial conditions (local values at *all* spatial locations at *one* time instant).

SHRED has been successfully applied for dynamic discovery in a wide range of problems [193][194]. In particular, the authors in Ref. [193] have demonstrated that SHRED can use extremely limited sensor trajectories to provide an "on-the-fly" compressive representation of the dynamics as well as to provide predictions of systems' full state based on the sensor measurements.

2.4. Prior applications of ML/DD techniques in plasma modelling

The interest in ML/DD applications for plasma modelling and simulation has surged over the past years. This rise in community attention has been accompanied by the emergence of a range of research efforts. These can be mainly categorized into enhancing/accelerating plasma simulations, on one side, and system identification (dynamics discovery) and reduced-order modelling, on the other side.

In this last section of Chapter 2, we aim to provide a demonstrative overview of some of the main prior efforts on the above two fronts. The overview presented here is not meant to be exhaustive. Rather, it aims at painting a picture of the landscape of applications that ML/DD techniques have already found in computational plasma physics. Detailed reviews have been recently published in the literature on data-driven plasma science [196] as well as on ML promises for modelling and analysis of low-temperature plasmas [197]. Additionally, in Ref. [198], the authors have provided an in-depth description of the ML fundamentals for plasma applications, particularly for LTPs, and have discussed some specific use cases of ML to study LTP systems. All these publications, and the references therein, can be referred to for a more extensive reading.

We start by looking at several examples of prior research where the authors have investigated ML enhancement and/or acceleration of high-fidelity PIC simulations. In this context, machine learning and data-driven techniques have often been employed to substitute a specific, computationally intensive module of PIC codes, e.g., the Poisson solver [199]. Aguilar and Markidis [200] employed deep learning (in the form of a fully connected neural network and a CNN architecture) to enhance a PIC simulation by learning a mapping between the electric field and the phase-space information of the particles. They assessed their DL-PIC algorithm against the two-stream instability test case, observing that the code provides good predictions in this case [200]. However, they reported that their DL-PIC does not conserve momentum and energy, requiring explicit enforcing of these in the DL field solver as the next step [200]. Kube et al. [201] proposed the use of neural networks to accelerate implicit, energy-conserving PIC algorithms (e.g., see Ref. [77]) by reducing the number of GMRES (generalized minimal residual) iterations required to advance the PIC solution in time. Nicolini et al. [202] and later Nayak et al. [203] aimed at accelerating the field-solve step of the PIC simulations by substituting the high-dimensional field solver with reduced-order models for the electromagnetic fields. Nicolini et al. used POD for this purpose [202], whereas Nayak et al. employed the DMD method [203]. Despite promising results, the generalizability of the derived ROMs for electromagnetic fields in extrapolation, whether in time or in parameter space of a plasma system, was highlighted as a challenge that needs to be addressed [202]. In Ref. [204], Nayak et al. used the DMD method with a sliding window to identify the system state transition from transient to equilibrium, allowing the identification of the limit cycle, and hence predicting the electric field evolution and the resulting plasma particles dynamics. The authors discussed the implications of this approach for accelerating PIC simulations [204]. Badiali et al. [205] proposed the application of ML-based models to speed up the multiphysics modules of a PIC simulation, such as the collisions operator.

In a separate context to the above efforts but still with the aim of accelerating plasma simulations, Zhong et al. used PINNs in Ref. [206] to solve stationary and time-dependent PDEs governing thermal plasmas, particularly 1D arc discharges.

We now shift our attention to some examples of ML/DD applications for reduced-order modelling, dynamics discovery, and closure model development for plasmas. Dynamic mode decomposition has been used by several researchers toward reduced-order plasma modelling. Sasaki et al. [207] applied DMD to the simulation data of the Kelvin-Helmholtz instability [208] in order to extract the underlying dominant plasma structures and develop models for the time evolution of these structures (modes). The time dynamics of the DMD modes was calculated using an approach based on "convolution-type correlation integrals" [207]. The authors observed that the derived time dynamics reflects the essential physical behaviors much better that what can be achieved with Fourier analysis [207]. Taylor et al. [209] and Kaptanoglu et al. [210] applied the DMD method to the experimental and simulation data from the HIT-SI fusion experiment. Both studies provided interpretable insights into the dominant patterns underlying the data. Taylor et al. recovered the three dominant modes of the HIT-SI plasma using DMD [209]. The application of DMD by Kaptanoglu et al. [210] enabled uncovering several coherent magnetic modes from the data, which subsequently allowed a previously unrecognized three-dimensional structure in the simulations to be discovered. Maddaloni et al. [211] used DMD to analyze the dynamics of the global discharge oscillations in a Hall thruster plasma using the data from a hybrid fluid/PIC simulation. Kohne et al. [212] employed a data-driven clustering method – the self-organizing maps – to analyze the data from the full kinetic simulations of the plasmoid instability [213], observing that the identified clusters reflect the existing knowledge concerning the characteristics of this instability mode. Even though clustering is not a ROM development method per se, the authors in Ref. [212] pointed out the potential applicability of their method to represent the transition between various models (including ROMs) describing the plasma.

Dynamics discovery and system identification for plasmas has seen a number of interesting efforts in the past few years as well. Kaptanoglu et al. investigated the first-principles and data-driven approaches toward reduced-order modeling of a class of MHD plasmas, comparing the performance of the classical POD-Galerkin method against POD-SINDy and evaluating the effects of imposing physics constraints on the ROMs [214]. Recently, Kaptanoglu et al. have examined in Ref. [215] the applications of sparse regression methods, such as SINDy, to various tasks in plasma physics, including system identification, compressed sensing, and design optimizations. The SINDy algorithm has been used for plasma dynamics discovery by several other authors as well. The weak (integral) form of the SINDy algorithm was employed by Alvez and Fiuza in Ref. [216] to recover from PIC simulation data the known differential equations such as the Vlasov and the MHD equations. The authors demonstrated the importance of using the weak-form SINDy so as to robustify the model discovery to the discrete-simulation-particle noise [216]. In Ref. [217], Abramovic et al. applied SINDy to data from the simulations of drift-wave turbulence [218] based on the Hasegawa–Wakatani and modified Hasegawa–Wakatani models. They reported the efficacy of SINDy as a sample sparse regression method to derive the governing nonlinear PDEs accurately describing the evolution of the system [217].

Connecting dynamics discovery and closure model development, Donaghy and Germaschewski [219] employed SINDy to derive from kinetic PIC simulation data of a magnetic reconnection configuration – the "*collisionless Harris sheet reconnection problem*" – a ten-moment fluid system of equations for the plasma with a closure model for the tenth moment term, aiming to improve the "*Hammett–Perkins closure"*. The salient finding was that even though the closure term from SINDy was improved over the Hammett–Perkins closure for the nonlinear phase of the system evolution, it was not adequate for the linear phase [219], hence, warranting further work along the direction of closure discovery with SINDy.

Further to the application of ML for closure model discovery, van de Plassche and collaborators [220] developed a fast feed-forward neural network, QLKNN, to find ML closure (surrogate) models for the heat and particle transport fluxes from a tokamak's core from a database comprising 300 million flux calculations of "*the quasilinear gyrokinetic transport model QuaLiKiz*." Integrating these ML models into a tokamak modelling framework and a control-oriented tokamak transport solver, they reported orders of magnitude speed-up while maintaining a good accuracy [220]. Heinonen and Diamond in Ref. [221] applied a deep neural network to data from a direct numerical simulation of the 2D Hasegawa-Wakatani plasma system in order to derive mean-flow closure terms for the turbulent transport fluxes, showing promising results. In the context of Hall thruster plasmas, Jorns followed symbolic regression to find a functional relationship for the "anomalous" electron collision frequency in terms of the local plasma properties [26]. The derived data-driven symbolic function showed a predictive capability as well as a quantitative improvement over five first-principles models of the anomalous collision frequency [26]. The limitations of the approach, partly in terms of the extensibility of the derived functional forms to parameter spaces notably outside the training range, were examined and discussed [26].

According to the above overview, the integration of ML/DD techniques into plasma modeling has demonstrated notable potential in various respects. These advances highlight the transformative impact of ML/DD and warrant further progress and consolidated efforts, particularly on reduced-order plasma modelling, as this thesis aims to accomplish within the subsequent chapters.

Chapter 3 Proper orthogonal decomposition and dynamic mode decomposition for the plasma data analysis and pattern extraction

Informed by the detailed review of the landscape of scientific ML in Chapter 2, this first technical Chapter of the thesis starts by focusing on dynamic mode decomposition and its underpinning method, proper orthogonal decomposition/singular value decomposition, in the context of data analysis and mode identification for plasma systems. We assess how POD/SVD enables understanding the low-dimensional representability of data. Moreover, we intend to evaluate the DMD's application for pattern (mode) extraction from high-dimensional simulation data.

The discussions begin by providing a review of the theory behind POD/SVD in section 3.1. Section 3.2 presents the DMD concept and introduces several DMD variants that will be tested and assessed in this thesis. Section 3.3 provides the details of the 2D plasma problem that are adopted to demonstrate the POD/SVD and two of the DMD methods introduced in section 3.2, namely, the basic DMD, and the Optimized DMD (OPT-DMD). Section 3.4 discusses the results of the application of POD/SVD to the data from high-fidelity simulations of our test case. The performances of basic DMD and OPT-DMD toward reproducing the original dataset by extracting the underlying dominant coherent structures and patterns in the data are compared in section 3.5. Informed by the outcomes from section 3.5, the application of OPT-DMD to the data are presented in section 3.6, where the results in terms of the identification and extraction of dominant spatiotemporal modes underlying the data are discussed.

3.1. The theory of the proper orthogonal/singular value decomposition

Proper orthogonal decomposition, also referred to as Principal Component Analysis (PCA) in statistics, is a technique used for dimensionality reduction, data compression, and noise reduction. It aims to capture the dominant patterns or modes present in a dataset. In POD, we start with a set of data often represented as snapshots or measurements taken over time or space. The goal is to find a reduced set of basis functions, also called modes, that can accurately approximate the original data. These modes are determined based on the statistical properties and the correlations of the data. The POD modes are also orthogonal (perpendicular) to each other, which helps in capturing effectively the

most important variations in the data. POD is closely related to the singular value decomposition (SVD) [101], which is a matrix factorization technique that enables computing the basis modes (see also section 2.2).

SVD decomposes a data matrix $X \in C^{n\times m}$, the columns of which can be measurements in space or time, into three matrices, $U \in C^{n\times m}$, $\Sigma \in R^{m\times m}$, and $V^* \in C^{m\times m}$, thus,

$$X = U\Sigma V^*. \qquad \text{Eq. 3.1}$$

In Eq. 3.1, n denotes the dimensionality of the data, and m represents the number of measurements or snapshots. In addition, the superscript * indicates conjugate transpose of matrix V. The columns of matrix U are the left singular vectors, which represent the SVD (or POD) modes. The SVD/POD modes are orthonormal and form a basis for the column space of X. These modes capture the correlation between the columns of matrix X (different snapshots), and they are arranged in a hierarchical order from left to right based on the extent of the correlation they represent. Matrix V contains the right singular vectors. Similar to U, the vectors in V are orthonormal as well and provide a basis for the row space of X. The columns of V capture the correlation between the rows of X and, hence, in a time series data, they represent the temporal pattern. Σ is a diagonal matrix with non-negative entries σ_i , $i = 1, 2, \dots, m$ along its diagonal. The σ_i entries are the singular values associated with the SVD/POD modes. The singular values are arranged in a descending order and represent the “strength” or “importance” of the corresponding singular vectors in the matrices U and V.

In practice, it is not necessary to retain all the SVD modes (i.e., full U, V, and Σ matrices) to be able to represent the data. In many cases, the data matrix can be well approximated by a lower-rank matrix because the dominant patterns and significant information in the data are captured by a subset of the SVD modes. The dominant patterns are captured by the larger singular values and their corresponding singular vectors in SVD. Therefore, we can truncate the SVD by considering only the most significant singular values and their corresponding columns in the matrices U and V. This reduces the dimensionality of the data while retaining its important features. Besides, real-world data often contains noise or insignificant information represented by smaller singular values. By truncating these smaller singular values and their associated modes, we can effectively remove the noise and irrelevant information, focusing on the more significant patterns.

The low-rank truncation and the determination of how many ranks to retain are performed by looking at the distribution of the singular values (σ_i). The ratio given by Eq. 3.2 determines the proportion of the total variance in the data that is captured by the first k singular values and their corresponding modes.

$$K = \frac{\sum_{i=1}^{k} \sigma_i}{\sum_{i=1}^{m} \sigma_i}. \qquad \text{Eq. 3.2}$$

Typically, we aim to retain around 90-95% of the total variance to capture the most important features. The rank-r approximation of the data ($\tilde{X}_r$) is obtained by keeping the first r columns of U (u_i) and V (v_i^*) and the first r singular values (σ_i) as per Eq. 3.3.

$$\tilde{X}_r = \sum_{i=1}^{r} u_i \sigma_i v_i^* = \tilde{U}\tilde{\Sigma}\tilde{V}^*. \qquad \text{Eq. 3.3}$$

In Eq. 3.3, $\tilde{U} \in C^{n\times r}$, $\tilde{\Sigma} \in R^{r\times r}$, and $\tilde{V} \in C^{m\times r}$, with $r \leq m$.

3.2. Dynamic mode decomposition: basic formulation, optimized DMD, and parametric DMD

DMD is a technique to decompose the dynamics of a system into a combination of several spatiotemporal modes with arbitrary spatial bases and linear time dynamics. Therefore, unlike Fourier transform, which assumes purely harmonic basis functions, DMD modes are specific to each system and are driven directly from its data. The time dynamics of the DMD modes are constrained to be linear, which means that each spatial mode is associated with a complex frequency representing an oscillatory behavior with possibly a growth or damping rate.

DMD can be thought of as an alternative dimensionality reduction technique to the SVD/POD. Indeed, the SVD/POD modes are orthogonal and, therefore, provide an optimum set of bases for representing the data [101][151]. However, the SVD/POD modes do not contain any temporal evolution information. In contrast, DMD provides dynamical information about the system at the expense of the fact that the modes are no longer orthogonal because they are now constrained by the linearity of the time dynamics.

3.2.1. The basic DMD algorithm

The central idea of DMD is to find a best-fit linear-dynamics approximation of the spatiotemporal data from either a linear or nonlinear system. The basic algorithm of DMD is outlined in the following.

The collected time-series data representing the time evolution of the system (which can come from either an experiment or a high-fidelity simulation) are rearranged into data matrices X and $X' \in R^{n\times m-1}$

$$X = \begin{bmatrix} | & | & & | & & | \\ \mathcal{S}_1 & \mathcal{S}_2 & \dots & \mathcal{S}_k & \cdots & \mathcal{S}_{m-1} \\ | & | & & | & & | \end{bmatrix}, \qquad X' = \begin{bmatrix} | & | & & | & & | \\ \mathcal{S}_2 & \mathcal{S}_3 & \dots & \mathcal{S}_{k+1} & \cdots & \mathcal{S}_m \\ | & | & & | & & | \end{bmatrix}.$$

Each column of X ($\mathcal{S}_k$) represents an n-dimensional state vector at time step t_k, where $k = 1, 2, \dots, m-1$, and the corresponding column in X' is the system's state vector at the subsequent time step t_{k+1}. In the context of the discussions in this Chapter, the $\mathcal{S}_k$ is a snapshot of a plasma flow field at each time step of the simulation rearranged into a column vector. The DMD algorithm finds the optimal linear transformation/operator (A) that approximates simultaneous advancement of all data one time-step forward such that

$$X' \approx AX, \qquad \text{Eq. 3.4}$$

with A defined as,

$$\mathrm{A} = \arg\min_{A} \|\mathcal{S}_{k+1} - A\mathcal{S}_k\|_2 . \qquad \text{Eq. 3.5}$$

The solution of the above minimization problem is

$$A = X'X^{\dagger} = X'\,V\Sigma^{-1}U^{*}, \qquad \text{Eq. 3.6}$$

where $X^{\dagger}$ is the Moore-Penrose pseudo-inverse of X obtained using the SVD of matrix X.

Typically, the data snapshots are high dimensional such that $n \gg m$. This means that matrix $A \in R^{n\times n}$ will be prohibitively large to compute and store. As such, DMD relies on SVD/POD to project matrix A onto a reduced-dimensional subspace defined by r POD modes so as to find a reduced rank approximation of the full matrix A, which is denoted by $\tilde{A} \in R^{r\times r}$. Therefore, a practical step toward calculating DMD is to take the SVD of the data matrix X and project matrix A onto the rank-r truncated SVD modes, $\widetilde{U}$, to obtain $\tilde{A}$

$$\tilde{A} = \widetilde{U}^{*}A\widetilde{U} = \widetilde{U}^{*}X'\tilde{V}\tilde{\Sigma}^{-1}. \qquad \text{Eq. 3.7}$$

Having obtained $\tilde{A}$, we compute the eigendecomposition of this matrix to obtain the DMD spatial modes and their time dynamics

$$\tilde{A}V = \Lambda V. \qquad \text{Eq. 3.8}$$

In Eq. 3.8, Λ is a diagonal matrix containing the eigenvalues ($\lambda_i,\ i = 1, 2, \dots, r$) of $\tilde{A}$. These eigenvalues coincide with the r leading eigenvalues of the full matrix A. The columns of matrix V are the eigenvectors of $\tilde{A}$, which can be expanded to reconstruct the eigenvectors of A using the relation in Eq. 3.9. The eigenvectors of A are represented by the columns of matrix Ψ ($\psi_i,\ i = 1, 2, \dots, r$).

$$\Psi = X'\tilde{V}\tilde{\Sigma}^{-1}V. \qquad \text{Eq. 3.9}$$

Each eigenvalue-eigenvector pair (λ_i and ψ_i) represents a distinct spatiotemporal DMD mode, where the ψ_i characterizes the spatial behavior and the λ_i describes temporal evolution of the mode. The reconstructed time series of the state vectors from the DMD modes is obtained through their linear combination

$$\mathcal{S}_{k+1} \approx \sum_{i=1}^{r} \psi_i b_i \lambda_i^{k} = \Psi \mathrm{B}\Lambda^{k}, \qquad k = 1, 2, \dots \qquad \text{Eq. 3.10}$$

where, matrix B is a diagonal matrix containing the initial amplitudes of the DMD modes (b_i corresponding to the i-th DMD mode). The modes' initial amplitudes are obtained by projecting the data at $t = 0$ ($\mathcal{S}_1$) onto the DMD modes

$$B = \Psi^{\dagger}\mathcal{S}_1. \qquad \text{Eq. 3.11}$$

Eq. 3.10 represents the discrete form of the dynamics describing the evolution of the system over discrete time intervals Δt corresponding to the time step at which the measurements are taken. This equation can also be written in the continuous dynamics form as in Eq. 3.12.

$$\mathcal{S}(t) \approx \sum_{i=1}^{r} \psi_i b_i \exp(\omega_i t) = \Psi \mathrm{B} \exp(\Omega t). \qquad \text{Eq. 3.12}$$

In Eq. 3.12, Ω is a diagonal matrix that contains the continuous-time eigenvalues ($\omega_i\ ,\ \ i = 1, 2, \ldots, r$) on its main diagonal. ω_i and λ_i are related through the relationship below

$$\omega_i = \ln(\lambda_i)\ /\Delta t. \qquad \text{Eq. 3.13}$$

Eq. 3.12 and Eq. 3.13 can be used to forecast the system's behavior at future times.

3.2.1.1. DMD application to multi-property state vectors with linear and nonlinear variables

Through the discussions so far, we have considered applying DMD to the spatiotemporal data (signals) corresponding to individual state variables of a system. However, DMD can also work with *multi-property* state vectors. Assembling a multi-property state vector comprised by several physical properties can amount to representing the full state of the system. This will then allow us to develop a linear ROM using DMD that best describes the time evolution of all the properties (state variables) simultaneously. The enhancement of the state vector in this manner enables capturing the interdependency between the physical properties and is, thus, expected to improve the predictive performance of the derived ROMs.

To develop a DMD model based on a multi-property state vector, the only necessary change compared to what was presented in subsection 3.2.1 is to construct the matrices of the time-series data snapshots X and X' as per Eq. 3.14.

$$X = \begin{bmatrix} | & | & & | & & | \\ \boldsymbol{\mathcal{X}}_1 & \boldsymbol{\mathcal{X}}_2 & \ldots & \boldsymbol{\mathcal{X}}_k & \cdots & \boldsymbol{\mathcal{X}}_{m-1} \\ | & | & & | & & | \end{bmatrix}, \qquad X' = \begin{bmatrix} | & | & & | & & | \\ \boldsymbol{\mathcal{X}}_2 & \boldsymbol{\mathcal{X}}_3 & \ldots & \boldsymbol{\mathcal{X}}_{k+1} & \cdots & \boldsymbol{\mathcal{X}}_m \\ | & | & & | & & | \end{bmatrix}. \qquad \text{Eq. 3.14}$$

Referring to Eq. 3.15, each column $\boldsymbol{\mathcal{X}}_k$ in X represents a column-wise stacking of the rearranged snapshots of several plasma properties at the time t_k, $\mathcal{S}_{p,k}$, with p denoting the index of a single specific property, and $p = 1, 2, \ldots, P$. $\mathcal{S}_{p,k}$ is a snapshot of the p-th property at the time t_k ($k = 1, 2, \ldots, m-1$) rearranged into a n-dimensional column vector. Also, m indicates the number of timesteps over which the data is collected from the system under study.

$$\boldsymbol{\mathcal{X}}_k = \left[\mathcal{S}_{1,k}^T,\ \mathcal{S}_{2,k}^T, \ldots, \mathcal{S}_{P,k}^T\right]^T. \qquad \text{Eq. 3.15}$$

The corresponding column of $\boldsymbol{\mathcal{X}}_k$ in X', i.e., $\boldsymbol{\mathcal{X}}_{k+1}$, is the system's state vector at a subsequent time step t_{k+1}.

Besides the above, another point is worth highlighting at this stage. A limitation of the basic DMD algorithm is its underlying assumption of the linearity of the time dynamics (Chapter 10 of Ref. [151]). Nonetheless, nonlinearities can be introduced into a DMD model by expanding the state vector's space so that it includes nonlinear observables (functions) of the state variables as well. To this end, the state vectors, $\boldsymbol{\mathcal{X}}_k$s, in the data matrices X and X' can be augmented to include the state variables as well as their nonlinear observables, $g_i,\ i = 1, 2, \ldots, N$ (Eq. 3.16).

$$\boldsymbol{\mathcal{X}}_k = \left[\mathcal{S}_{1,k}^T,\ \mathcal{S}_{2,k}^T, \ldots, \mathcal{S}_{P,k}^T,\ g_1(\boldsymbol{\mathcal{S}}_k^T),\ g_2(\boldsymbol{\mathcal{S}}_k^T), \ldots,\ g_N(\boldsymbol{\mathcal{S}}_k^T)\right]^T, \qquad \text{Eq. 3.16}$$

where,

$$\boldsymbol{\mathcal{S}}_k = \left[\mathcal{S}_{1,k}^T,\ \mathcal{S}_{2,k}^T, \dots, \mathcal{S}_{P,k}^T\right]. \qquad \text{Eq. 3.17}$$

The above approach is inspired by the Koopman theory, which refers to representing a *finite-dimensional* nonlinear dynamical system using an *infinite-dimensional* linear transformation known as the Koopman operator [170]. The Koopman operator acts on an infinite-dimensional state space that includes all the possible nonlinear observables of the physical state variables of the system.

The concepts of an infinite-dimensional operator and state space are, however, an abstraction. In practice, we need to find a *finite-dimensional approximation* of the Koopman operator by choosing a finite observable subspace. Choosing an appropriate subset of the observables, which enables to best approximate the full Koopman transformation, may require expert knowledge. In the absence of an a-priori knowledge, techniques such as Kernel-based methods and, in particular, Support Vector Machines (SVM) [222]-[225], as well as deep neural networks (refer to section 2.1) can be used to find the optimal subset of observables (see also subsection 2.3.1).

3.2.2. Optimized DMD method based on variable projection

In the previous subsection, it was pointed out that one limitation of the basic DMD algorithm is its underlying assumption of the linearity of the time dynamics. Another, perhaps more major, limitation of the basic DMD algorithm, which is also referred to as “Exact” DMD, is its susceptibility to bias errors from noisy data. This can lead to poor model fits and unstable forecasting capabilities. The sensitivity of Exact (basic) DMD to noise arises from the fact that DMD considers the relationship between the consecutive snapshot pairs in the dataset. Therefore, the presence of noise can misinform the pairwise relation between the snapshots, which can in turn disrupt an accurate identification of the underlying patterns and modes in the data. Moreover, the identified modes can involve notable growth or damping rates resulting in the model predictions to either vanish to zero or go to infinity over longer prediction time windows. In this regard, noting that most real-world data contain certain noise levels, the standard Exact DMD algorithm leads to unstable and inaccurate models, which are incapable of reproducing data over practically desirable long durations.

The optimized DMD algorithm based on variable projection [167] represents a remarkable advancement over the Exact DMD method. The OPT-DMD particularly circumvents the limitation of Exact DMD with respect to noise-induced bias, making the DMD method more stable, robust, and predictive. OPT-DMD mitigates the sensitivity to noise by accounting for the relationship among all snapshots collectively rather than relying exclusively on pairwise connections. This algorithm allows us to minimize the real part of the eigenvalues, thus, constraining the modes to have purely oscillatory dynamics and as a result stabilizing the model which will be in turn capable of excellent reconstruction of the original (ground-truth) dataset. Additionally, OPT-DMD can function with non-equispaced snapshots in time, which is another notable improvement compared to Exact DMD.

OPT-DMD relies on variable projection technique to efficiently solve the following nonlinear least-square optimization problem to find Ω and $\Psi_{\mathrm{B}} = \Psi \mathrm{B}$ [167]

$$\min_{\Omega, \Psi_B} \|X - \Psi_B \exp(\Omega t)\|_2 . \quad \text{Eq. 3.18}$$

The full mathematical details of the OPT-DMD method and the approach to solve the optimization problem given by Eq. 3.18 are available in Ref. [167].

To apply the OPT-DMD algorithm to the data from the test cases assessed in this Chapter, as well as in Chapter 4 to Chapter 6, the open-source MATLAB code of Ref. [167] was adopted and used. This code is available at the URL indicated in Ref. [226].

3.2.3. Parametric dynamic mode decomposition

In a "parametric" system, the dynamics is not only dependent on the state variables of the system but is also influenced by some additional parameters. The inclusion of these additional parameters is necessary to obtain a comprehensive and adaptable representation of complex systems that could be applicable to a broad range of scenarios. In this section, a parametric extension to the DMD algorithm is presented. The results from the "parametric DMD" method will be presented and discussed in Chapter 6.

The parametric DMD implementation is intended to enable the development of ROMs for dynamical systems that exhibit parametric dependencies, i.e., systems described by the following equation

$$\frac{d\boldsymbol{x}(t)}{dt} = \boldsymbol{\mathcal{F}}(\boldsymbol{x}(t); \boldsymbol{\mu}). \quad \text{Eq. 3.19}$$

In Eq. 3.19, $\boldsymbol{x}$ is the state vector, and $\boldsymbol{\mu}$ is the "parameter vector" containing the individual parameters (μ_j with $j = 1, 2, \dots, n_p$) that the system's dynamics is dependent on, hence,

$$\boldsymbol{\mu} = \left[\mu_1, \mu_2, .., \mu_{n_p}\right]^T. \quad \text{Eq. 3.20}$$

Multiple variants of parametric DMD are proposed in the literature [166][227], which rely on interpolation to achieve "parameter realization". The proposed algorithms are different from each other in the way that they interpolate between the parameters. In the following, the specific parametric DMD implementation that is used for the studies in this thesis is described.

We start by gathering the data snapshots, $X^{\boldsymbol{\mu}_s}$, that correspond to different sets of values of the parameter vector ($\boldsymbol{\mu}$) for the parametric dynamical system described by Eq. 3.19. These different sets of values are denoted as $\boldsymbol{\mu}_s \in \mathrm{M}$ ($s = 1, 2, \dots, S$), where

$$\mathrm{M} = \{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \dots, \boldsymbol{\mu}_S\} \quad \text{Eq. 3.21}$$

The arrangement of the spatiotemporal data in the matrix $X^{\boldsymbol{\mu}_s}$ is identical to that described in subsection 3.2.1 for the basic (Exact) DMD method. Thus, $X^{\boldsymbol{\mu}_s}$ comprises the time series of column-wise rearranged snapshots, $X^{\boldsymbol{\mu}_s} \in R^{n \times m-1}$, where n represents the dimension of each data snapshot, and m specifies the total number of snapshots.

We then apply the OPT-DMD algorithm (subsection 3.2.2) to the individual datasets, $X^{\boldsymbol{\mu}_s}$s, to determine the eigenvalues ($\Lambda^{\boldsymbol{\mu}_s}$) and the eigenfunctions ($\Psi^{\boldsymbol{\mu}_s}$) of the DMD's operator $A^{\boldsymbol{\mu}_s}$ for each parameter realization such that

$$X'^{\boldsymbol{\mu}_s} \approx A^{\boldsymbol{\mu}_s} X^{\boldsymbol{\mu}_s}, \qquad \text{Eq. 3.22}$$

where, $X'^{\boldsymbol{\mu}_s}$ represent the data matrix $X^{\boldsymbol{\mu}_s}$ shifted by one time step forward. The amplitudes of the modes corresponding to each parameter value set ($B^{\boldsymbol{\mu}_s}$) are also directly obtained from the OPT-DMD algorithm (subsection 3.2.2).

The matrices $\boldsymbol{\Lambda}$, $\boldsymbol{\Psi}$, and $\mathbf{B}$ contain the eigenvalues, the spatial modes, and the modes' amplitude, respectively, corresponding to each parameter realization

$$\boldsymbol{\Lambda} = \begin{bmatrix} \Lambda^{\boldsymbol{\mu}_1} \\ \Lambda^{\boldsymbol{\mu}_2} \\ \vdots \\ \Lambda^{\boldsymbol{\mu}_s} \end{bmatrix}, \qquad \boldsymbol{\Psi} = \begin{bmatrix} \Psi^{\boldsymbol{\mu}_1} \\ \Psi^{\boldsymbol{\mu}_2} \\ \vdots \\ \Psi^{\boldsymbol{\mu}_s} \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} B^{\boldsymbol{\mu}_1} \\ B^{\boldsymbol{\mu}_2} \\ \vdots \\ B^{\boldsymbol{\mu}_s} \end{bmatrix}. \qquad \text{Eq. 3.23}$$

For a new parameter vector $\boldsymbol{\mu}_q \notin \mathrm{M}$, the eigenvalues ($\Lambda^{\boldsymbol{\mu}_q}$), the spatial modes ($\Psi^{\boldsymbol{\mu}_q}$), and the corresponding amplitudes ($B^{\boldsymbol{\mu}_q}$) are obtained by evaluating the Lagrangian interpolation between the respective quantities of the neighboring parameter vectors to $\boldsymbol{\mu}_q$. Interpolation can be performed using linear, cubic, or any higher order Lagrange interpolant polynomials. Following the interpolation, the time series data for the "query" parameter vector $\boldsymbol{\mu}_q$ can be reconstructed using the relation in Eq. 3.24

$$x^{\boldsymbol{\mu}_q}|_{k+1} \approx \Psi^{\boldsymbol{\mu}_q}\, \mathrm{B}^{\boldsymbol{\mu}_q}\, \Lambda^{\boldsymbol{\mu}_q}|_k\,, \qquad k = 1, 2, \dots. \qquad \text{Eq. 3.24}$$

It is emphasized that the interpolation would result in a reasonable estimation of the dynamics only if the variation of the spatiotemporal modes is smooth across the parameter space. Strong nonlinear dependencies of the DMD modes to the systems' parameters can cause significant discrepancy between the interpolated and the "true" spatial and temporal characteristics of the modes for the query parameter-vector $\boldsymbol{\mu}_{\mathbf{q}}$.

In the parametric DMD implementation described above, OPT-DMD is used as the core DMD algorithm that is applied to the datasets of the individual parameters. Besides this, the parametric DMD scheme of this thesis has minor differences with respect to the "reduced eigen-pair interpolation (rEPI)" approach that was followed by Huhn et al. in Ref. [227] for the interpolation among the spatial modes. In this regard, another noteworthy difference is that, whereas we directly interpolate the DMD modes ($\Psi^{\boldsymbol{\mu}}$) for a new parameter realization, in Ref. [227], the eigenvectors ($V^{\boldsymbol{\mu}}$) of the DMD's operator and the spatial POD modes ($\tilde{U}^{\boldsymbol{\mu}}$) were interpolated separately. Subsequently, the DMD modes were computed as $\Psi^{\boldsymbol{\mu}_q} = \tilde{U}^{\boldsymbol{\mu}_q}\, V^{\boldsymbol{\mu}_q}$ [227].

Another parametric DMD algorithm has been presented in Ref. [227], which is called "reduced Koopman operator interpolation (rKOI)". This algorithm involves interpolating the reduced DMD operator $\tilde{A}^{\boldsymbol{\mu}}$ directly and then calculating the $\Lambda^{\boldsymbol{\mu}_q}$ and $\Psi^{\boldsymbol{\mu}_q}$ from the eigendecomposition of $\tilde{A}^{\boldsymbol{\mu}_q}$.

Finally, a rather different parametrization technique is suggested by Ref. [228]. In this approach, all data matrices for the different parameter realizations ($X^{\boldsymbol{\mu}_s}$) are stacked column-wise in an augmented

data matrix $\boldsymbol{X} \in R^{(n_p \times n) \times m-1}$. The DMD is then applied on the augmented data matrix $\boldsymbol{X}$ [228]. A major limitation of this approach is that, while the spatial modes and their amplitudes are specific to each parameter realization, the eigenvalues (time dynamics) are shared and assumed constant across the entire parameter space.

3.3. Description of the adopted test case: a cross-field plasma discharge in a 2D radial-azimuthal geometry

The setup of the "ground-truth" reduced-order quasi-2D simulations carried out using IPPL-Q2D [81][229][230] represents a 2D cross-field plasma discharge configuration that resembles a radial-azimuthal cross-section of a Hall thruster. The computational and physical parameters of the simulations summarized in Table 3.1 closely follow those of the benchmark problem by Villafana et al. in Ref. [43].

Parameter	Value [unit]
Computational parameters	
Domain length ($L_x = L_z$)	1.28×10^{-2} [m]
Virtual axial length (L_y)	1×10^{-2} [m]
Cell size ($\Delta x = \Delta z$)	50 [μm]
Number of cells in each direction ($N_i = N_k$)	256
Time step (ts)	1.5×10^{-11} [s]
Total simulated duration (t_{sim})	30 [μs]
Initial number of macroparticles per cell (N_{ppc})	100
Physical parameters	
Initial plasma density ($n_{i,0}$)	5×10^{16} [m^{-3}]
Initial electron temperature ($T_{e,0}$)	10 [eV]
Initial ion temperature ($T_{i,0}$)	0.5 [eV]
Electric potential at the walls (ϕ_w)	0 [V]

Table 3.1 Summary of the computational and physical parameters used for the quasi-2D simulations of the radial-azimuthal test-case problem.

The simulations of the radial-azimuthal test case were carried out in a domain that corresponds to a 2D $x - z$ Cartesian plane. The adopted domain is illustrated in Fig. 3.1. As it is shown in this figure, the x-coordinate is along the radial direction, and the z-coordinate represents the azimuthal direction. The y-axis is along the axial direction. The domain is 0.0128-m long along both the radial and the azimuthal coordinates.

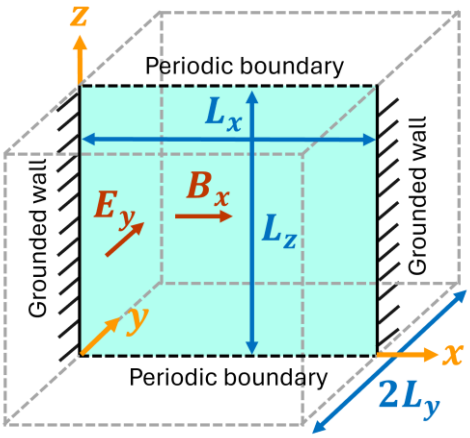


Fig. 3.1 Schematics of the coordinate system, the computational plane, and the applied constant electric and magnetic fields corresponding to the radial-azimuthal test-case simulations.

An overview of how the simulations were set up follows: at the beginning of the simulations, the electrons and ions are sampled from a Maxwellian distribution at 10 eV and 0.5 eV, respectively, and are loaded uniformly throughout the domain as pairs. To limit the growth of the energy of the azimuthal waves and the particles in the simulations [23], a virtual axial extent with the length of L_y = 0.01 m is considered [43]. The particles crossing a domain's boundary along the axial direction are resampled from their initial distribution and are re-injected onto the simulation plane, maintaining their azimuthal and radial positions. In all simulation cases, a zero-volt Dirichlet boundary condition is used for the electric potential at the walls. A periodic boundary condition is considered in the potential solver for the nodes at the two azimuthal ends of the domain. Moreover, particles that exit the domain along the azimuth from one side are re-introduced into the domain from the opposite side.

The domain decomposition associated with the reduced-order PIC scheme [81] was carried out using 50 vertical regions (M) and 50 horizontal regions (N). This specific domain decomposition translates into quasi-2D simulations with a five-fold reduction in the computational cost compared to corresponding conventional full-2D PIC simulations [84].

The radial-azimuthal simulations feature a temporally invariant cosine-shaped ionization source along the radial direction with the peak value of S. This ionization source serves to compensate for the flux of plasma particles lost to the walls [43][48]. The simulations also include an applied constant axial electric field (E_y) and a constant radial magnetic field (B_x) [43][48].

The various test cases in this Chapter correspond to simulations of the radial-azimuthal discharge problem with different values of the parameter S, which is equivalent to various plasma number densities at the system's quasi-steady state. The sets of plasma conditions, i.e., the values of the parameters S, E_y, and B_x, associated with these test cases are summarized in Table 3.2.

For each test case, the total number of plasma properties' snapshots is 2000, corresponding to 30 μs of simulation time. Each snapshot additionally represents an average over $1.5 \times 10^{-2}\ \mu s$ (1000 simulation timesteps). The first 5.25 μs (350 snapshots) of the simulations, which captures the initial

transient of the system, is excluded from the dataset. This leaves us with a 1650-snapshot training/testing dataset for each test case.

Test case	S value	E_y value	B_x value
1 (Baseline)	$S_0 = 8.9 \times 10^{22}\ m^{-3}s^{-1}$	$E_0 = 10{,}000\ Vm^{-1}$	$B_0 = 0.02\ T$
2	$1/32\ S_0$	E_0	B_0
3	$1/16\ S_0$	E_0	B_0
4	$1/8\ S_0$	E_0	B_0
5	$1/4\ S_0$	E_0	B_0
6	$1/2\ S_0$	E_0	B_0
7	$3S_0$	E_0	B_0
8	$6S_0$	E_0	B_0

Table 3.2 Summary of the values of the plasma conditions for various radial-azimuthal test-case simulations; the baseline case has the S value equal to that of the radial-azimuthal benchmark reported in Ref. [43].

3.4. Application of the POD/SVD to understand the low-dimensional representability of the data

In this section, the results of the application of SVD to the data from the radial-azimuthal test case are presented. The primary aim is to demonstrate an example of how the results of SVD (equivalently, POD) allow us to gain insights into the structure and dimensionality of our data and to investigate whether the data can be represented with low-enough number of ranks. We discuss the influence of the truncation rank (r) and the training data ratio on SVD. We also evaluate the effects of noise on the low-rank representation of the data using SVD.

For the analyses in this section, we have focused on the baseline radial-azimuthal test case (see Table 3.2) and, unless stated otherwise, have taken the axial electron current density (J_{ey}) data as a representative plasma property. The selection of the J_{ey} data was because the effects of the main instabilities and the dynamical processes ongoing in the radial-azimuthal plasma setup are well reflected in the J_{ey} signal.

Fig. 3.2(a) and (b) present, respectively, the normalized distribution of the singular values ($\sigma_r / \sum_{i=1}^{m} \sigma_i$) and their normalized cumulative sum ($\sum_{i=1}^{r} \sigma_i \,/ \sum_{i=1}^{m} \sigma_i$) in SVD of J_{ey} vs the number of rank. From these plots, we can identify the significance of each mode and how much of the variance within the data is captured by keeping the first r ranks. The maximum number of ranks is equal to the number snapshots (m), which is equal to 1650 for the radial-azimuthal simulations' data.

Fig. 3.2 illustrates that the entire ranks (or the SVD modes) are not necessary to faithfully reconstruct the original signal. In fact, the "knee" point in the profiles of the singular values indicates that a major part of the useful information within the data are represented by the first 400-500 modes.

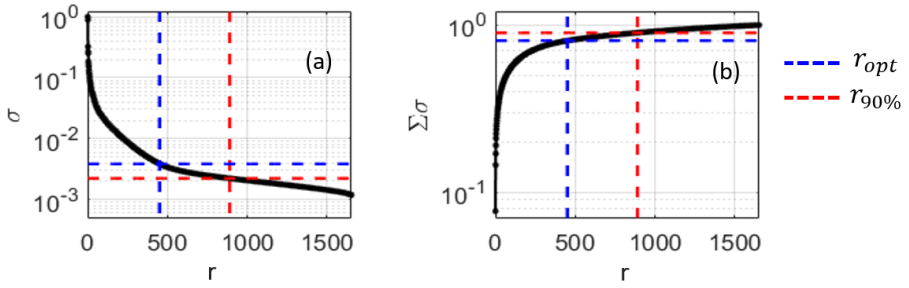


Fig. 3.2 The distribution of (a) normalized singular values and (b) normalized cumulative sum of the first r singular values in SVD of axial electron current density (J_{ey}) data.

To quantitatively determine the minimum SVD rank necessary for a high fidelity reconstruction of the original data, there are two possible approaches: **(1)** we can consider the minimum r value ($r_{90\%}$) as the fewest number of modes that carry at least 90% of the total variance in the data, i.e., $\frac{\sum_{i=1}^{r}\sigma_i}{\sum_{i=1}^{m}\sigma_i} \geq 0.9$. **(2)** Gavish and Donoho (Ref. [231]) introduced the "Optimal Hard Threshold" of $4/\sqrt{3}$ for singular values of the SVD of a data matrix. This threshold serves to determine the noise floor of the data and corresponds to a number of ranks (r_{opt}) beyond which the modes represent the noise. The number of necessary ranks determined by either of these criteria are shown as dashed lines in Fig. 3.2. The Optimal Hard Threshold (dashed blue line) suggests r = 450 for the low-rank approximation of the data matrix. This is much smaller than $r_{90\%}$ (dashed red line), which gives a required rank of about 900. Regardless, it will become clear later in this section that, even with a much lower number of ranks than those given by either criterion, we can still reconstruct the original data with sufficiently high fidelity.

Before that, it is of interest to assess the effectiveness of the SVD bases in representing unseen ("test") data, i.e., data which has not been included in the "training" dataset for which the SVD is calculated. In the context of SVD, testing implies whether the projection of the unseen data onto the SVD modes calculated using the training dataset can provide an accurate low-rank representation of those unseen data.

For this purpose, we divide the entire data into a training and a test set with three different proportions of R_t = 0.2, 0.5, and 0.8. R_t denotes the ratio of the length of the training set to the total dataset. The SVD is calculated on the training dataset, and the reconstruction of the training data is obtained from the rank-r truncated SVD as given by Eq. 3.3. The average reconstruction loss ($\mathcal{L}_{avg}$) with respect to the original data for different number of ranks is then calculated using Eq. 3.25. In Eq. 3.25, $\|\mathcal{S}_i\|_F$ denotes the Frobenius norm of the i-th data snapshot ($\mathcal{S}_i$), N_i and N_j are the number of grid points along the horizontal and vertical directions, respectively, and s_{ij} represents the value of the J_{ey} at each node ij.

$$\mathcal{L}_{avg} = \frac{1}{m}\sum_{i=1}^{m}\frac{\left\|\mathcal{S}_i^{true} - \mathcal{S}_i^{SVD}\right\|_F}{\left\|\mathcal{S}_i^{true}\right\|_F}, \quad \|\mathcal{S}\|_F \equiv \sqrt{\sum_{i=1}^{N_i}\sum_{j=1}^{N_j}|s_{ij}|^2}, \qquad \text{Eq. 3.25}$$

$$\mathcal{S}_{k,r} = U_r U_r^T \mathcal{S}_k. \qquad \text{Eq. 3.26}$$

The test data snapshots ($\mathcal{S}_k$) are next projected on the SVD modes (U_r) according to Eq. 3.26, which yields the rank-r reconstruction of the test data. Similar to the training set, the average loss of reconstruction of the test data with various SVD modes is computed using Eq. 3.25.

Fig. 3.3 presents the results in terms of the variation against the number of ranks of the losses in the reconstructed training (plot (a)) and test (plot (b)) datasets. It is evident from Fig. 3.3 that the loss in data reconstruction is smaller when using larger dataset to calculate SVD. This is because, with larger R_t values, more statistics of the signal are made available to the algorithm, which results in the identification of more representative and accurate modes to describe the data.

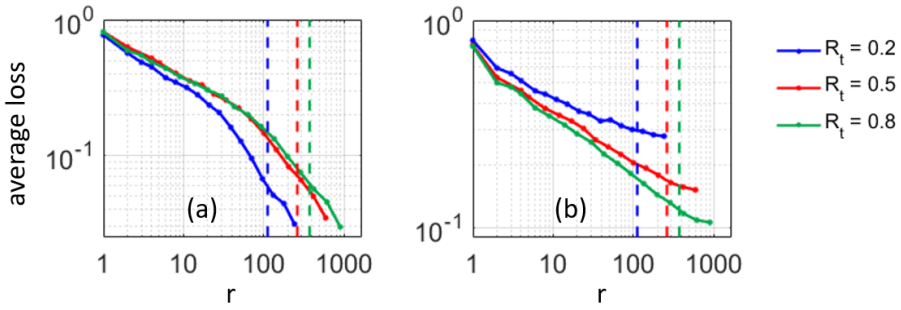


Fig. 3.3 Variation vs rank number (r) of the average loss factor (Eq. 3.25) between the ground-truth data (axial electron current density, J_{ey}) from radial-azimuthal baseline test case and the reconstructed data from SVD. Various curves correspond to different ratios of the training to test data (R_t). Plot (a) represents the variation in the loss on the training data, whereas plot (b) shows the loss factor variation over the test data. The vertical dashed lines indicate the r_{opt} for each R_t case based on the criterion by Gavish and Donoho [231].

To further demonstrate the efficacy of SVD in approximating the data in a reduced-order coordinate, we present in Fig. 3.4 a comparison between the time series of the spatially averaged values of the reconstructed J_{ey} with various number of ranks and R_t ratios and the original signals from the PIC simulations. The time evolutions of the instantaneous loss between the individual original and reconstructed snapshots are superimposed. The instantaneous loss is calculated as $\frac{\left\|\mathcal{S}_i^{true}-\mathcal{S}_i^{SVD}\right\|_F}{\left\|\mathcal{S}_i^{true}\right\|_F}$.

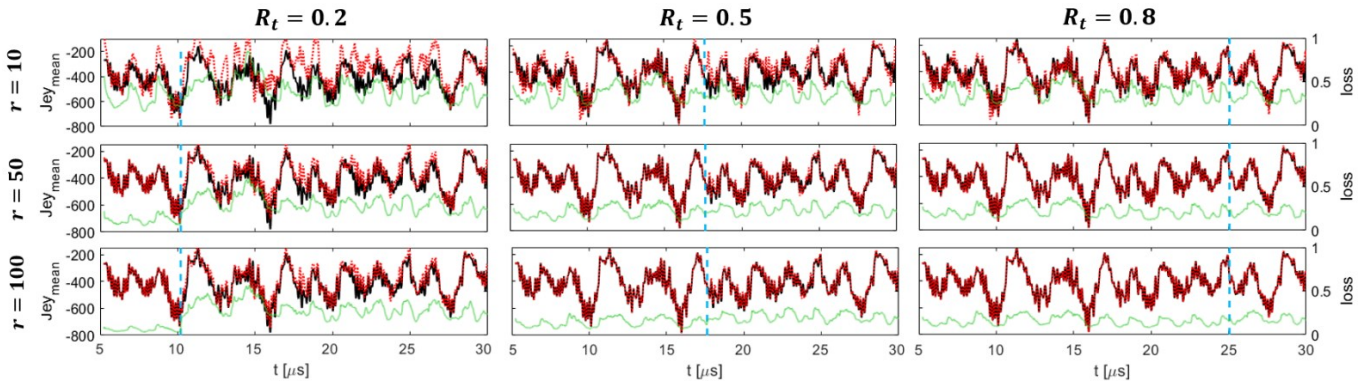


Fig. 3.4 Time evolution of the spatially averaged axial electron current density (J_{ey}) from the quasi-2D PIC simulation of the radial-azimuthal test case with the baseline conditions (solid black lines) and the reconstructed signal from the SVD (dotted red lines) using different number of ranks and training-to-test data ratios. Superimposed on each plot is the time evolution of the instantaneous loss factor for the corresponding (r, R_t) value pair. Dashed blue lines indicate the training end time.

It is highly interesting that, when a large enough dataset is used to calculate SVD ($R_t = 0.5$ and 0.8), even 10 ranks can capture the main information content so that the reconstructed J_{ey} signal can follow closely the original signal. With 50 ranks, the reconstruction is almost in perfect agreement with the original signal for R_t values of 0.5 and 0.8. Beyond 50 ranks (as observed for instance with 100 ranks), any further enhancement in accuracy becomes negligible. The reconstruction of the test data in the case of R_t = 0.2 is rather poor and includes noise. This is because the statistical information of 20% of the signal has not been sufficient for the SVD modes to be representative of the entire signal.

In Fig. 3.5, the reconstructions of a sample "true" 2D snapshot with different ranks and R_t ratios are provided. The reference 2D snapshot is randomly selected from the test window (within the time interval of 25-30 μs). The plots in Fig. 3.5 convey similar information to those inferred from Fig. 3.3 but in a more visual sense. The reconstructed snapshots with 50 ranks capture all features of the original snapshot (labeled as True). We further observe that the reconstruction quality in the case of $R_t = 0.2$ is poorer than the other two cases with R_t values of 0.5 and 0.8. In addition, it is noticed that the true snapshot has some noise superimposed on the main features. However, when we truncate SVD at lower ranks than the maximum rank of the data matrix, we remove the modes which represent the noise embedded within the data. Consequently, this process leads to the removal of noise from the data, which is another important application of SVD. This denoising effect is clearly observed from the reconstructed snapshots in the cases of R_t = 0.5 and 0.8 across various ranks.

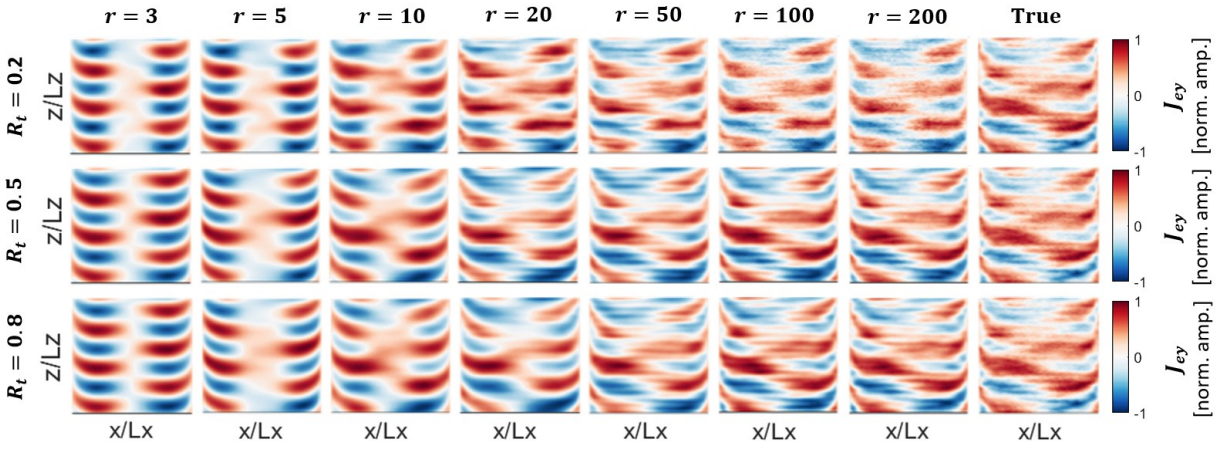


Fig. 3.5 Comparison of the reconstructed 2D snapshots of the axial electron current density (J_{ey}) from SVD using different number of ranks and R_t ratios against the corresponding ground-truth snapshot (last column) from the quasi-2D PIC simulation of the radial-azimuthal baseline test case.

For the case with R_t = 0.5, Fig. 3.6 additionally presents the sum of contributions of the remaining SVD modes which are not used for the reconstruction of the data at each truncation rank ($\hat{X} = \sum_{i=r}^{N_{modes}} \sigma_i u_i v_i^*$, where N_{modes} represents the total number of SVD modes). We can see that after removing the leading modes that carry the essential information, the modes in the tail of SVD spectrum (e.g., after rank 100) represent noise with no meaningful information content. The impact of the artificially added noise on the performance of SVD is assessed in the subsection 3.4.1.

It is recalled from section 2.2 that there is a connection between SVD and the autoencoder architecture. In fact, the autoencoder is a nonlinear generalization of the SVD method, achieved by

extending the number of layers and incorporating nonlinear activation functions. Although autoencoders and similar machine learning methods are not the focus of this thesis in terms of results, for the sake of completeness, we demonstrate the above stated connection between SVD and the autoencoder architecture in Appendix A.1. Moreover, we will show in the Appendix that learning nonlinear manifold coordinates to represent the data using autoencoders leads to much improved faithful compressions of the data in the latent space compared to linear decomposition methods like SVD.

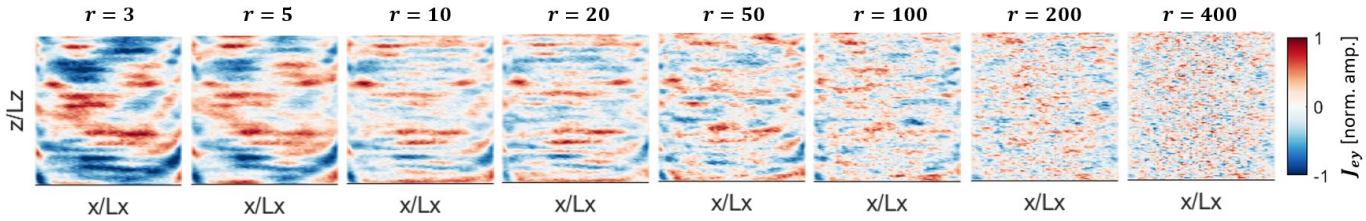


Fig. 3.6 Comparison of contribution of the remaining SVD modes of the axial electron current density (J_{ey}) with $R_t = 0.5$ (corresponding to middle row of Fig. 3.5), which are not used for reconstruction of the data at each truncation rank ($\hat{X} = \sum_{i=r}^{N_{modes}} \sigma_i u_i v_i^*$) in the radial-azimuthal baseline test case. The scale of the figures is normalized.

To conclude this section, we have shown in Fig. 3.7 a set of SVD spatial modes calculated for the axial electron current density (J_{ey}) as well as for the azimuthal electric field (E_z) data from the baseline radial-azimuthal test case. The modes shown are among the first 50 SVD modes, which represent distinct patterns within the data. The illustrated modes capture the most prominent structures in the data, which resemble the Electron Cyclotron Drift Instability (ECDI) and the Modified Two Stream Instability (MTSI) [refer to subsection 1.1.4 for some descriptions and relevant references on the ECDI and the MTSI modes]. In particular, among the plotted modes of the azimuthal electric field (E_z) (plot (b)), the first mode with purely azimuthal oscillations corresponds to the ECDI, the third and fourth modes are associated with the radial-azimuthal structures of the MTSI, and the remaining modes capture the transition states between the two instabilities. In any case, it is once again emphasized that SVD modes are static and do not inherently capture the evolution of the recovered patterns over time. Whereas the DMD modes are dynamic, which implies that DMD is a more suitable approach to extract the spatiotemporal coherent modes as it will be demonstrated in the subsequent sections.

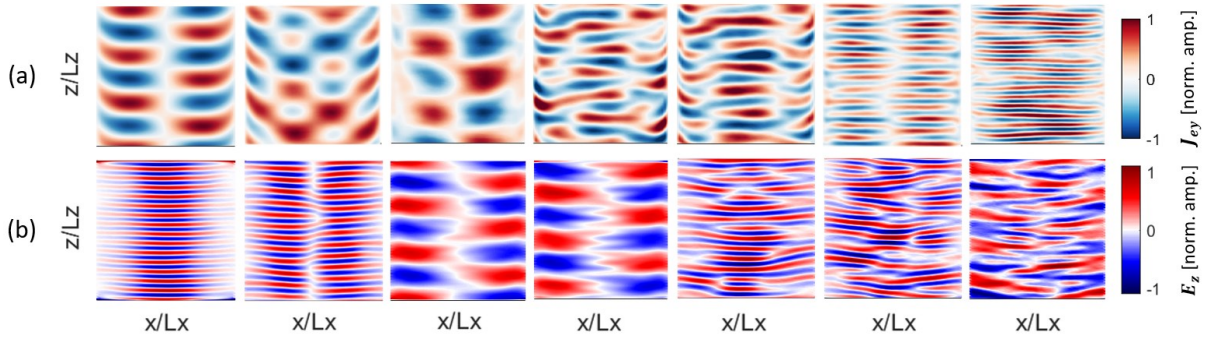


Fig. 3.7 Visualization of seven distinct SVD modes of the ground-truth data from the radial-azimuthal baseline simulation when using r = 100 and R_t = 0.5 for the SVD. (a) Modes corresponding to the axial electron current density (J_{ey}) data, (b) modes corresponding to the azimuthal electric field (E_z) data.

3.4.1. Sensitivity of SVD to added Gaussian noise

To investigate the impact of noise on the low-rank representation of the data using SVD, we introduced Gaussian noise with various standard deviations (σ), ranging from 0.01 to 0.2, in the normalized axial electron current density signal (J_{ey}) from the radial-azimuthal baseline test case. The singular value spectrum for the J_{ey} with various noise levels and the reconstructions of a sample snapshot with different number of truncation ranks are presented in Fig. 3.8 and Fig. 3.9, respectively.

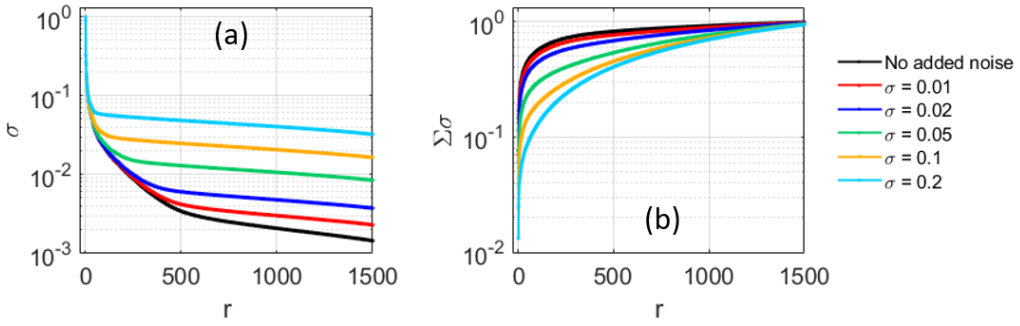


Fig. 3.8 The distribution of (a) normalized singular values and (b) normalized cumulative sum of the first r singular values in the SVD of the normalized axial electron current density (J_{ey}) data with added Gaussian noise of different standard deviations.

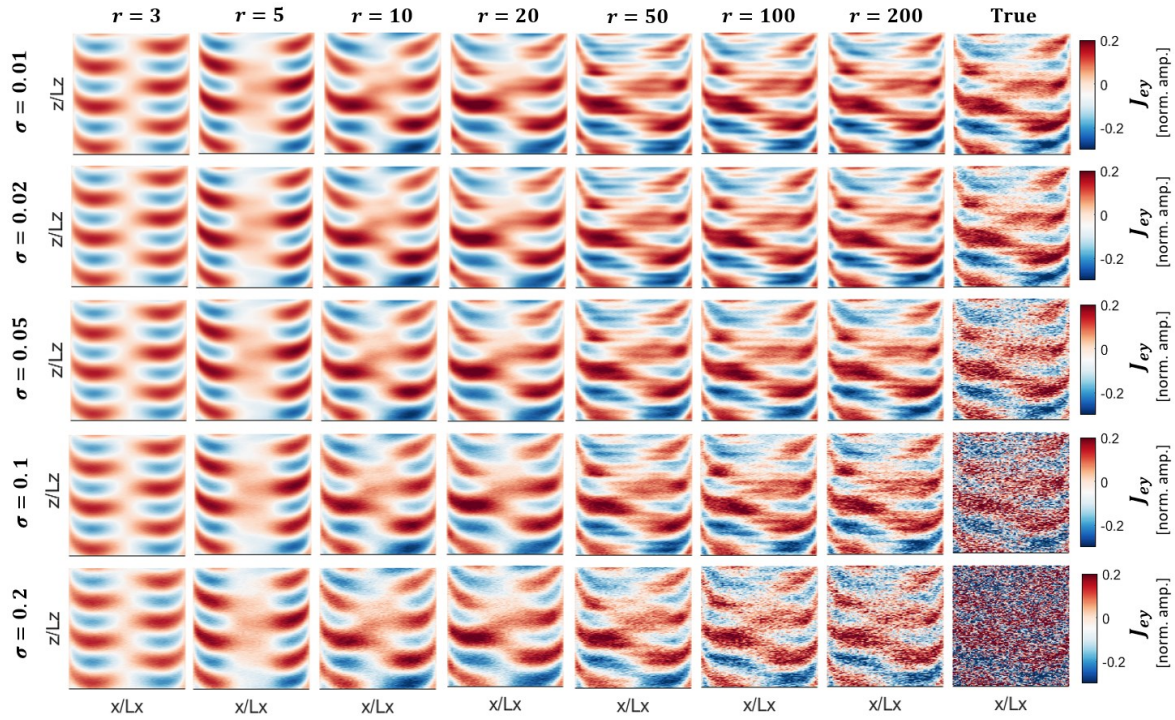


Fig. 3.9 Comparison of the reconstructed 2D snapshots of the normalized axial electron current density (J_{ey}) from the SVD using different number of ranks against the ground-truth snapshot (last column) corresponding to J_{ey} signal from the quasi-2D PIC simulation with different levels of added Gaussian noise.

Fig. 3.8 shows that, as we add more noise to the data, the magnitude of the singular values in the tail of the singular values distribution become increasingly larger, and the knee point, which separates the essential information content of the data from the noise, shifts towards smaller ranks. However, we

observe that, even in the presence of modest levels of noise, the leading singular values, which capture the most important feature in the data, remain nearly unaffected by the noise. As a result, SVD is still able to identify the main patterns and to reconstruct the original signal while denoising it as is observed from Fig. 3.9.

At very high noise levels ($\sigma = 0.2$), the few first ranks are still intact from the effect of noise and, therefore, only a very low rank reconstruction of the signal ($r < 20$) can separate the noise from the data. However, the higher ranks are corrupted by the strong noise, and this is seen to have led to the loss of some important information carried by the corresponding modes.

3.5. Basic DMD and OPT-DMD for data reconstruction

In this section, the aim is to evaluate and compare the ability and performance of the DMD models obtained from the Exact (basic) and OPT-DMD methods toward reproducing the original dataset by extracting the underlying dominant coherent structures and patterns in the data.

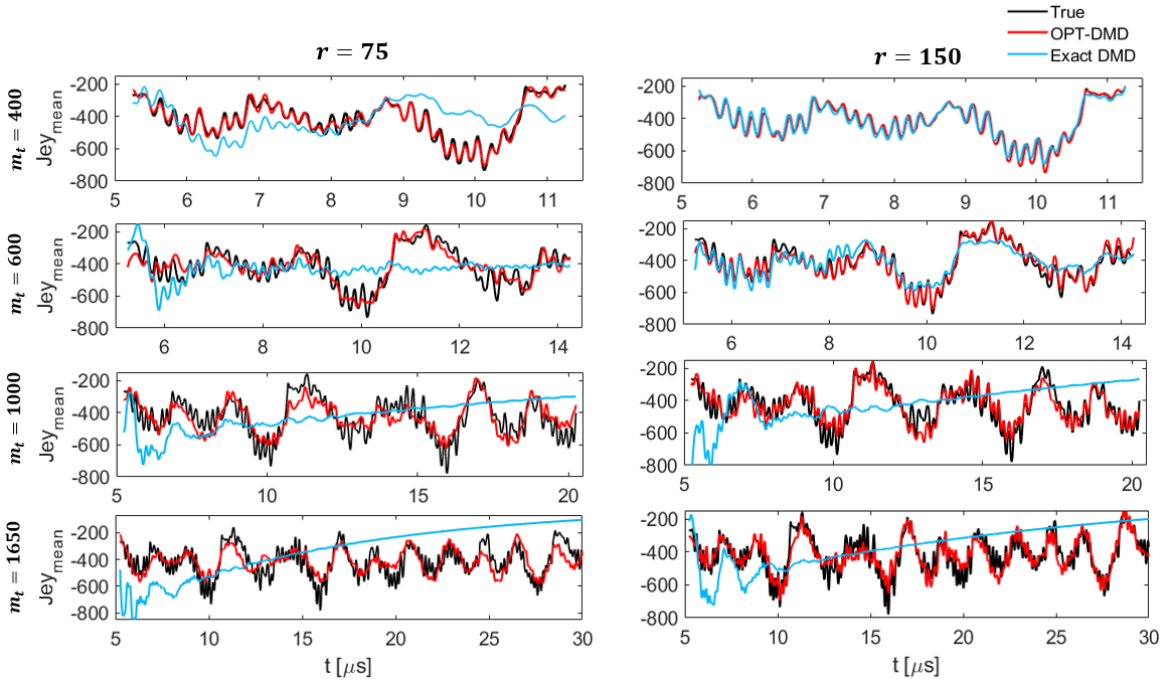


Fig. 3.10 Time evolution of the spatially averaged J_{ey} for the radial-azimuthal test case with the baseline conditions. The rows from top to bottom correspond to different lengths of the training signal (or, equivalently, number of snapshots (m_t)). The left and right columns, respectively, correspond to two different DMD ranks of 75 and 150. Ground-truth traces from the PIC simulation are illustrated as solid black lines, solid red lines show the OPT-DMD predicted traces, and the solid blue lines are the predicted traces from the Exact DMD.

To this end, we computed Exact DMD and OPT-DMD of the J_{ey} dataset from the baseline radial-azimuthal test case with various number of snapshots (m_t), corresponding to different lengths of the averaged J_{ey} signal. The respective DMD models were then used to reproduce the data using 75 and 150 DMD modes (or, alternatively, the truncation rank, r). The OPT-DMD models are obtained by constraining the eigenvalues of the modes such that their real parts are minimized (as close to zero as possible) through the optimization.

The time evolutions of the spatially averaged reconstructed and original data are shown in Fig. 3.10. The plots show that, only for short time duration ($m_t = 400$) and with relatively large number of modes (r = 150), the Exact DMD model yields similar reconstruction capability as the OPT-DMD model. However, over longer durations (larger m_t values), the Exact DMD model immediately approaches a steady value corresponding to the constant mode representing the mean of the data and fails to follow the trend of the ground-truth. Whereas, the OPT-DMD model with either 75 or 150 modes successfully reproduces the overall behavior in the signal across all the tested time durations.

The inability of the reconstructed signal from the Exact DMD models to persist over extended times is because the modes' eigenvalues exhibit relatively large negative real values. This results in rapid damping of the oscillatory modes. It is recalled from Eq. 3.12 that the time dynamics in the DMD method is represented as $exp(\omega_i t)$, hence, it is the real frequency component in this context that denotes growth or damping.

The constraining of the modes' eigenvalues in OPT-DMD to lie on the imaginary axis stabilizes the associated model. As a result, the OPT-DMD modes are sustained over arbitrarily long durations without vanishing to zero or growing indefinitely over time. To better illustrate this, Fig. 3.11 compares the imaginary vs real frequency of the modes obtained with the Exact DMD and the OPT-DMD methods applied to the J_{ey} data of the radial-azimuthal baseline test case.

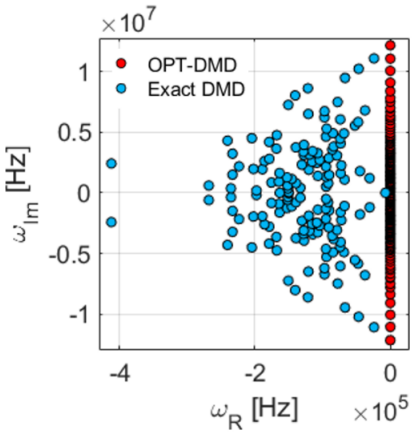


Fig. 3.11 Distribution of the eigenvalues over the complex frequency plane from the Exact DMD and the OPT-DMD algorithms. The eigenvalues correspond to the DMD of the J_{ey} data for the radial-azimuthal baseline test case with the rank of 150 and the training snapshots number of 1650 (Second-column, last-row plot in Fig. 3.10).

The presented results through Fig. 3.10 and Fig. 3.11 highlight that OPT-DMD is notably more reliable than Exact DMD for data reconstruction and pattern extraction. Thus, we will base the remainder of this section and Chapter, as well as the rest of the thesis, on various investigations using OPT-DMD.

3.5.1. Convergence of OPT-DMD vs truncation rank

The DMD convergence vs the number of modes (or the truncation rank of the underlying SVD) is an important consideration when approximating complex dynamical systems. The number of modes chosen for the DMD determines the accuracy and fidelity of the resulting approximation. Therefore, in

this subsection, we evaluate the convergence characteristics of the OPT-DMD model as the number of modes increases.

Here, we are not concerned with the predictive capability of the OPT-DMD model beyond the training dataset as this aspect will be addressed in Chapter 4, section 4.2. Rather, our current focus is on evaluating the loss associated with the *reconstruction* of the dataset with various number of modes. To this end, we present in Fig. 3.12 a comparison between the time evolutions of the spatially averaged J_{ey} data from the radial-azimuthal baseline test case reconstructed with different number of DMD modes and the respective ground-truth data. In addition, samples of a 2D snapshot reconstructed with various number of DMD modes corresponding to those shown in Fig. 3.12 are depicted in Fig. 3.13.

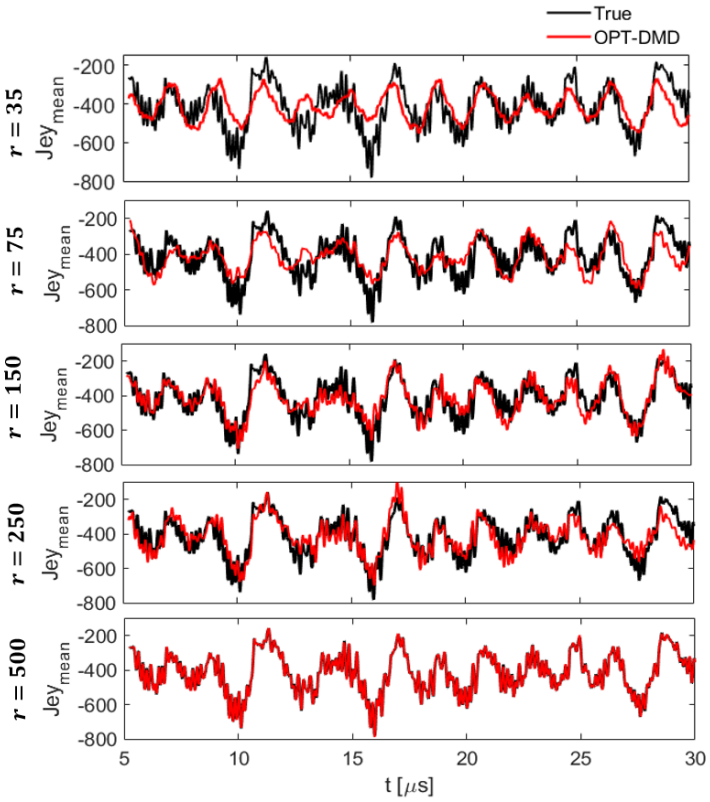


Fig. 3.12 Time evolution of the spatially averaged axial electron current density (J_{ey}) for the radial-azimuthal baseline test case from the PIC simulation (solid black line) and the OPT-DMD model (solid red line) with various number of DMD ranks.

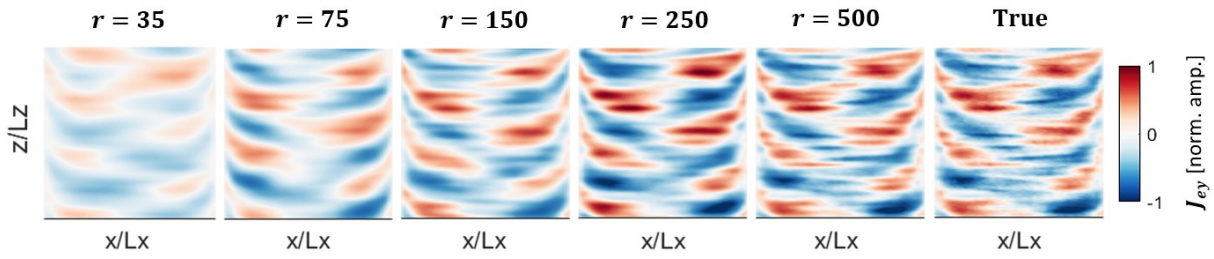


Fig. 3.13 Visualization of a sample reconstructed snapshot of electron axial current density (J_{ey}) from the radial-azimuthal baseline test case using the OPT-DMD model with different number of ranks. The subplot labeled as “True” represents the same snapshot from the PIC simulation.

From Fig. 3.12 and Fig. 3.13, we can observe that, as the number of modes increases, the DMD algorithm captures finer details present in the data. This allows a more accurate representation of the system's behavior. Selecting too few modes leads to underfitting, resulting in an oversimplified representation of the system and the loss of critical information. For instance, in the cases with r = 35 and 75, the lower energy modes such as the short wavelength azimuthal oscillations superimposed on the main radial-azimuthal patterns are not adequately captured.

It must be noted, however, that too many number of modes can also lead to overfitting, where the model fits the noise in the data rather than the true underlying dynamics. This might cause spurious oscillations or divergence of the models' forecast beyond the training data. Thus, finding the optimal number of modes involves a trade-off between capturing the complexity of the system and avoiding overfitting. The cross-validation by assessing the ability of the DMD model to generalize well to the unseen data is an indication of a proper selection of mode number. This is particularly relevant when considering the application of DMD model for forecasting, which will be extensively explored in Chapter 4.

To summarize the conclusions of the above discussions, Fig. 3.14 represents the convergence of the OPT-DMD's average loss (Eq. 3.25) vs the number of modes used to reconstruct the data. The loss factor continuously decreases with increasing r value, and the rate of decrease in the reconstruction loss also increases for higher number of DMD ranks, especially from about r = 150 forward.

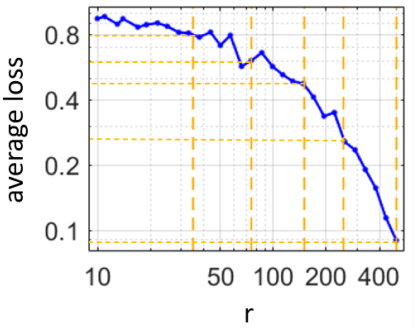


Fig. 3.14 Variation vs rank number (r) of the average loss factor (Eq. 3.25) between the ground-truth J_{ey} data from the PIC simulation of the radial-azimuthal baseline test case and the reconstructed J_{ey} data from the OPT-DMD model.

3.6. Optimized DMD for data-driven identification of dominant plasma instability modes

It was emphasized through the discussions so far that the DMD method identifies the dominant coherent structures (patterns, or modes) in the data and associates each derived mode with a linear dynamics, i.e., a complex frequency with the imaginary part representing the frequency of the mode's temporal oscillations and the (potential) real part representing growth or damping of that mode. The power of the OPT-DMD method was highlighted to partially lie in the fact that the real frequency component of the modes can be minimized and/or eliminated through the optimization process of the algorithm, thus, enabling the OPT-DMD-derived modes to remain fully stable.

In this section, we present a novel and practically significant use case of the pattern extraction using OPT-DMD. Through comparing the discrete frequency spectra of the DMD modes against the continuous temporal fast Fourier transform (FFT) spectra of the time evolution of the plasma properties across various radial-azimuthal test cases, we demonstrate that OPT-DMD allows us to derive information on the spatiotemporal characteristics of the involved plasma phenomena, such as fluctuations and instabilities, in a more accessible and comprehensive manner beyond what is feasible with conventional FFT analyses.

Before delving into the results, we would underline that the generic spatial vector bases that the DMD method employs to reconstruct and represent a dataset allows this method to circumvent certain limitations of the spatial FFT analysis, which is carried out to determine the wavenumber characteristics of the plasma oscillations. Most notably, DMD is not limited by the FFT's strict requirement for data periodicity in space [56][232].

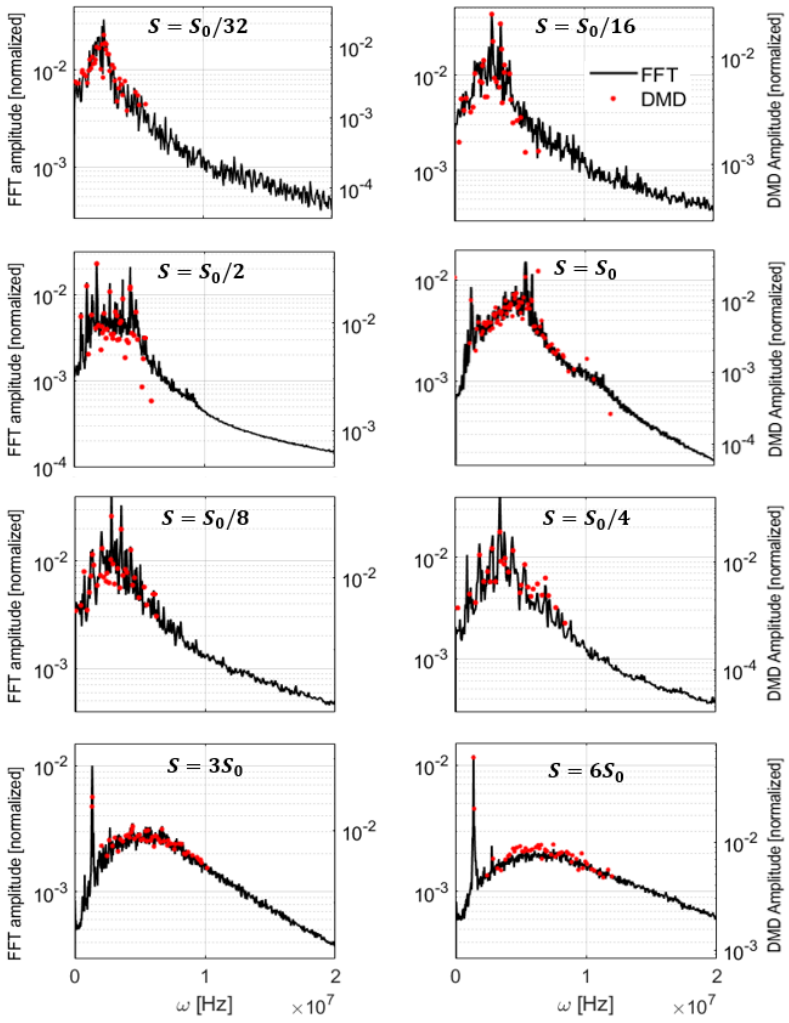


Fig. 3.15 Comparison of the temporal FFT spectra of the azimuthal electric field (E_z) for different radial-azimuthal test cases against the distribution of the DMD modes' frequency. The solid black lines represent the FFT of the ground-truth data from the PIC simulations, whereas the red dots correspond to frequency spectra of the E_z DMD modes. Depending on the plasma density across various test cases, the number of DMD ranks used vary between 80 to 150.

Moving to the results, in Fig. 3.15, the frequency spectra of the modes obtained by applying OPT-DMD to the E_z data for various radial-azimuthal test cases are superimposed on the temporal FFT spectra of the same data. It is noticed that, for all the cases, the frequency spectra of OPT-DMD and FFT are highly consistent. In particular, there are distinct frequency peaks associated with the dominant modes from OPT-DMD that coincide with the FFT frequency peaks. This superimposition of the frequency spectra allows us to pinpoint the OPT-DMD modes that correspond to dominant and distinct instability/fluctuation modes in the plasma discharge. Following this identification, the frequency information, and the spatial structure of the OPT-DMD modes enable a comprehensive, simultaneous, and quantitative spatial-temporal characterization of the instabilities.

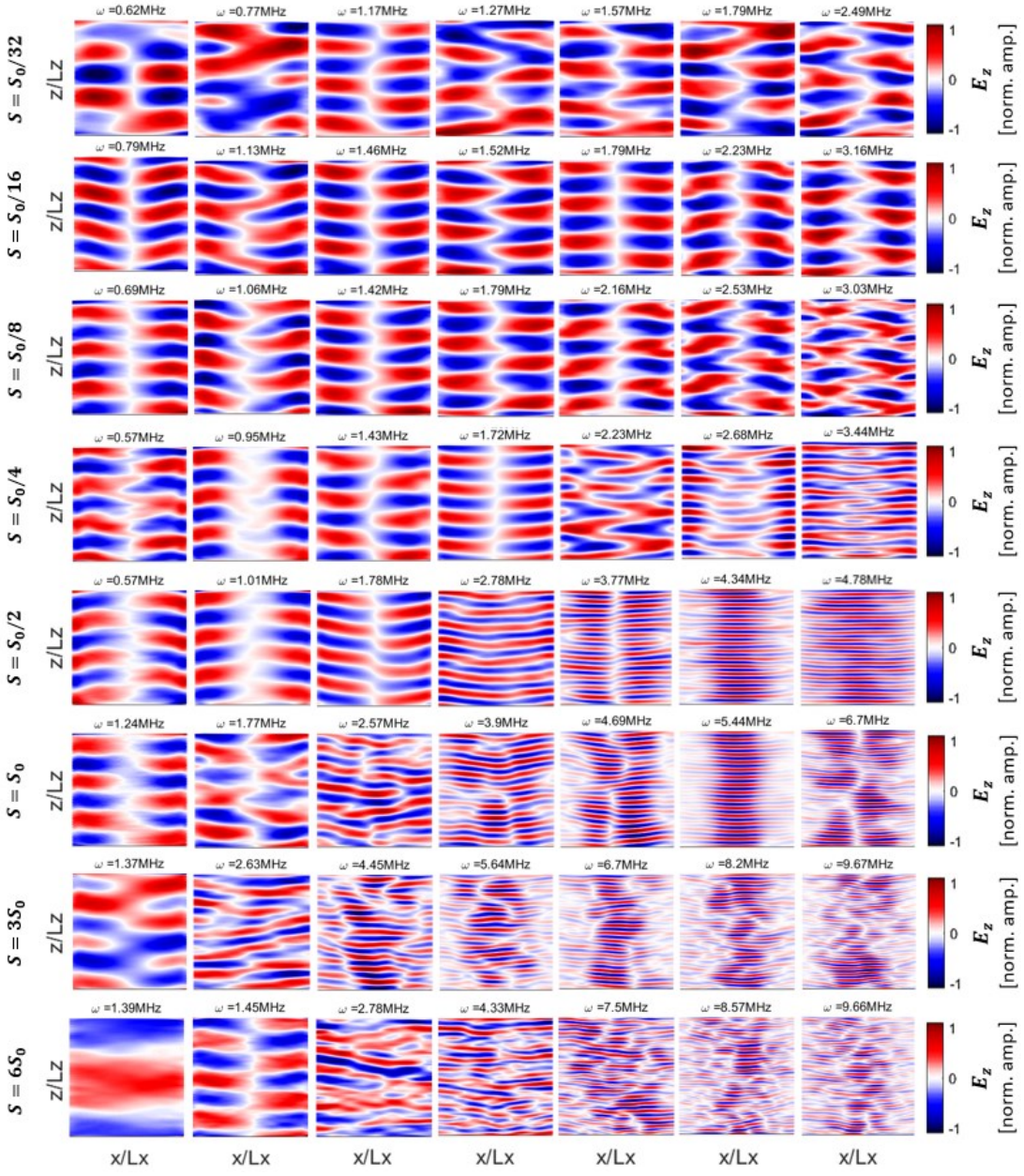


Fig. 3.16 Visualization of the dominant DMD modes of the azimuthal electric field data for several radial-azimuthal test cases. The illustrated DMD modes are those whose frequencies coincide with the peaks in the temporal FFT plot of each test case.

This is better illustrated in Fig. 3.16, in which, for each radial-azimuthal test case, we have shown the spatial structure of the dominant OPT-DMD modes whose frequencies coincide with the peaks in the FFT plot of each test case. From these 2D mode structures, the wavelength of each specific mode can be readily determined. Plots similar to those shown in Fig. 3.16 can have a remarkable applied significance in that they allow informing experiments by, for instance, providing information on the spatial spacing needed between electrostatic Langmuir probes along each physical direction of the system to experimentally capture a specific fluctuating pattern.

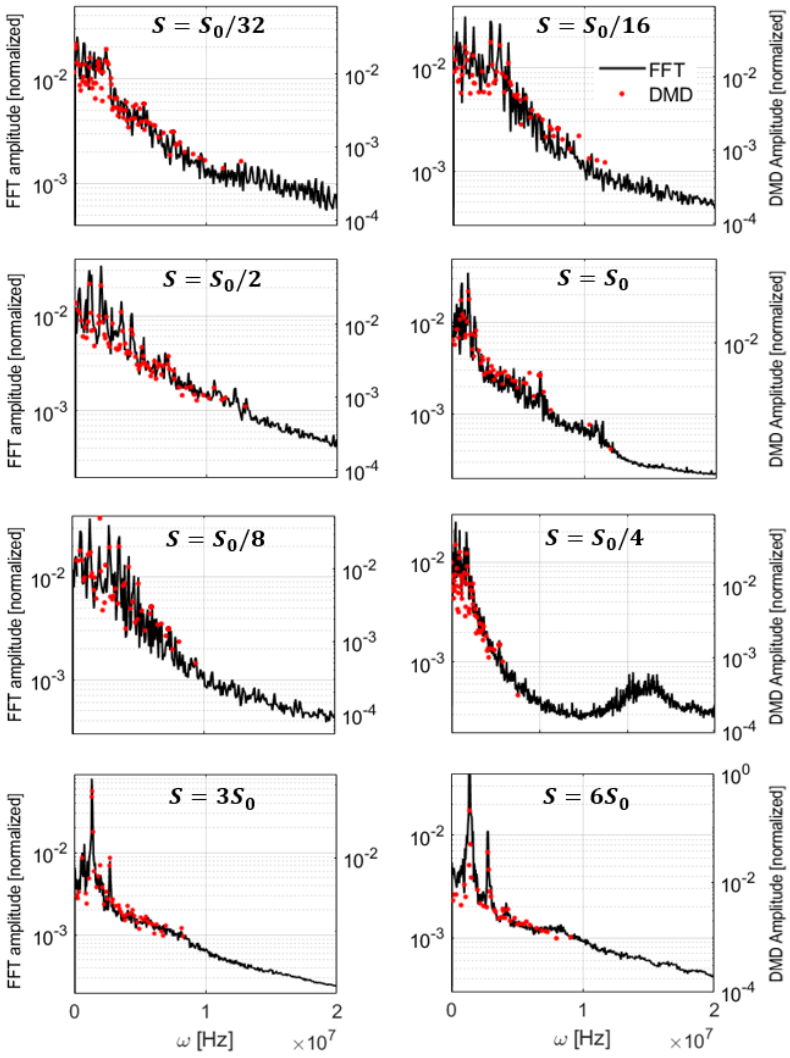


Fig. 3.17 Comparison of the temporal FFT spectra of the axial electron current density (J_{ey}) for different radial-azimuthal test cases against the distribution of the DMD modes' frequency. The solid black lines represent the FFT of the ground-truth data from the PIC simulations, whereas the red dots correspond to frequency spectra of the J_{ey} DMD modes. Depending on the plasma density across various test cases, the number of DMD ranks used vary between 80 to 150.

The information presented by the plots in Fig. 3.15 and Fig. 3.16 fully support the findings of Ref. [48], in which the influence of the plasma number density (the value of S) on the characteristics of the ECDI and MTSI modes had been assessed. In that work, the spectral analyses relied solely on the spatial and temporal FFT of the PIC simulations' data. Fig. 3.15 and Fig. 3.16, however, provide the

same insights in a more visually accessible and easier-to-comprehend manner. In fact, aligned with the outcomes of the study in Ref. [48], it is evident from the plots in Fig. 3.16 that, at low plasma densities (S values from $S_0/32$ to $S_0/8$), the MTSI-like patterns are the only present modes, whereas at high plasma densities (S values of $3S_0$ and $6S_0$), a long-wavelength mode appears that coexists with a spectrum of shorter-wavelength wave modes.

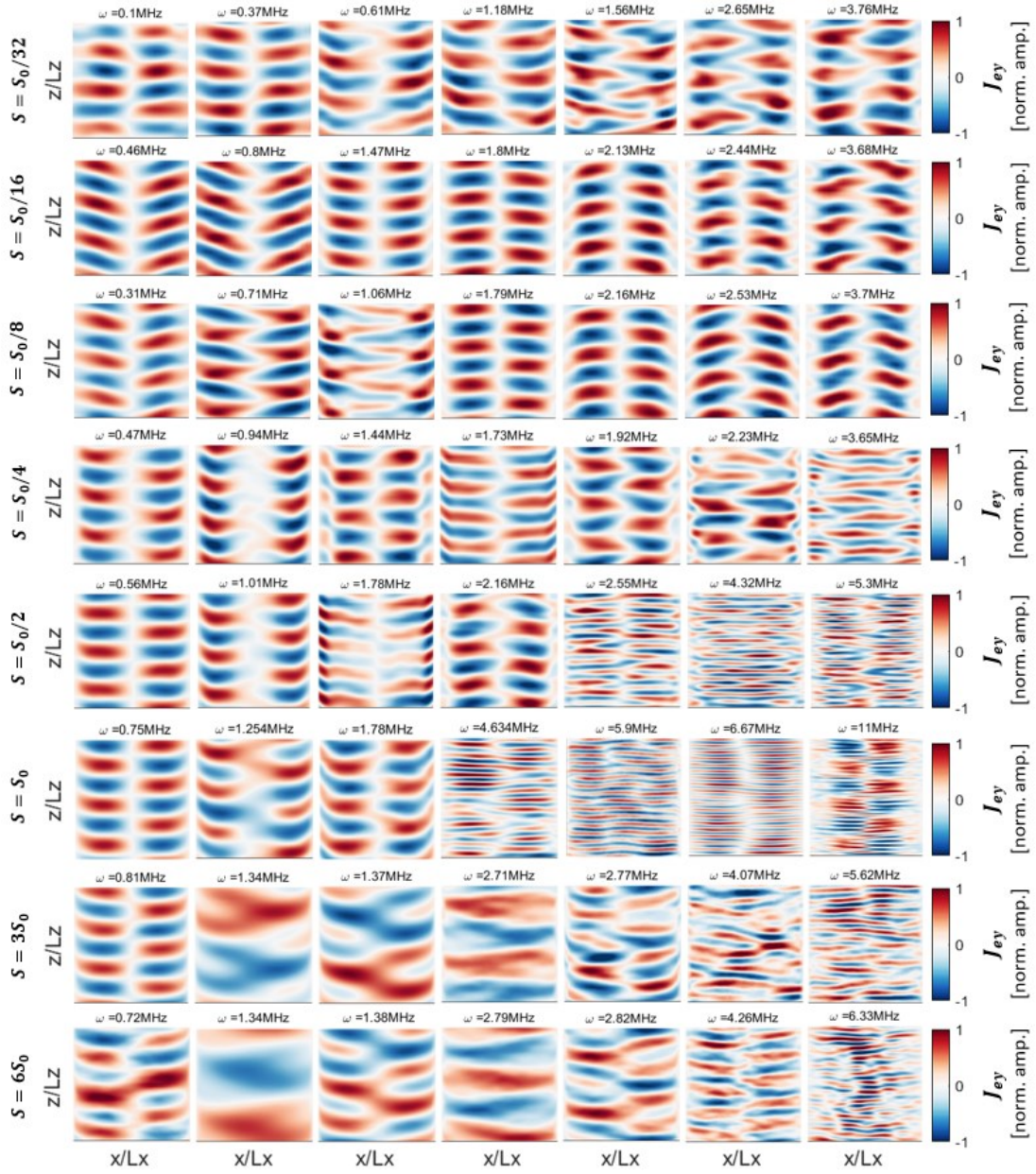


Fig. 3.18 Visualization of the dominant DMD modes of the axial electron current density data for several radial-azimuthal test cases. The illustrated DMD modes are those whose frequencies coincide with the peaks in the temporal FFT plot of each test case.

Another important point to highlight is regarding the plots of Fig. 3.15. We notice that, occasionally, a single frequency peak in the FFT corresponds to several DMD modes. These are manifestations of the DMD “rank inflation” due to the travelling nature of the waves in our plasma system. The influence of the rank inflation on the derived spatiotemporal characteristics of the instability modes using OPT-DMD will be assessed in subsection 3.6.2.

The plots in Fig. 3.17 and Fig. 3.18 convey similar information as those pointed out concerning Fig. 3.15 and Fig. 3.16, but from the application of OPT-DMD to the J_{ey} data of various radial-azimuthal test cases. From Fig. 3.18, in particular, the dominance of the MTSI-like structures in cases with the S values from $S_0/32$ to $S_0/8$, and the appearance of long-wavelength modes for cases with the S values of $3S_0$ and $6S_0$ can again be clearly discerned, as was also reported in Ref. [48].

It is mentioned that, toward fully understanding the underlying processes in a system, applying OPT-DMD to different plasma property data can be advantageous since certain physical information and patterns may be best represented within the evolution data of a specific plasma property.

Furthermore, it is emphasized that although, in the demonstrated test cases so far, the FFT and OPT-DMD may appear as equally well suited for spatiotemporal characterization of observed oscillatory phenomena, in cases where the underlying physics is less known and/or the FFT might become crowded with numerous peaks, OPT-DMD finds its main advantage. To clarify this statement, we present an example where the DMD method can provide insights into the coexisting modes that cannot be easily derived using the FFT analysis. This example focuses on a radial-azimuthal test case from Chapter 4, for which the setup is similar to the one described in section 3.3 but the value of the applied axial electric field is $E_y = 5E_0 = 50{,}000\ Vm^{-1}$ (test case number 4 in Table 4.4).

A comparison between the temporal FFT and the DMD spectra of the azimuthal electric field for this test case in addition to the corresponding spatial FFT spectrum are shown in Fig. 3.19.

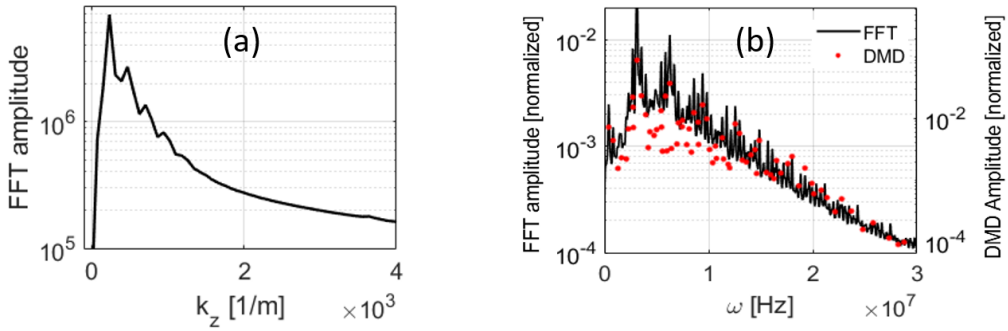


Fig. 3.19 (a) Spatial FFT spectrum, and (b) the temporal FFT spectrum of the azimuthal electric field (E_z) for the radial-azimuthal case with E_x = 50,000 Vm^{-1}. In plot (b), the distribution of the frequency of the DMD modes is compared against the temporal FFT spectrum.

We notice that, while the spatial FFT displays four distinct peaks representing multiple harmonics of the ECDI, the temporal FFT exhibits numerous peaks. The crowdedness of the temporal FFT spectrum makes it challenging to infer the frequencies associated with the peaks observed in the spatial FFT plot. In contrast, OPT-DMD (Fig. 3.20) offers simultaneous access to both the spatial and the temporal information for each mode, providing a clear understanding of the association between the frequency and the spatial structures, including specific wavenumbers.

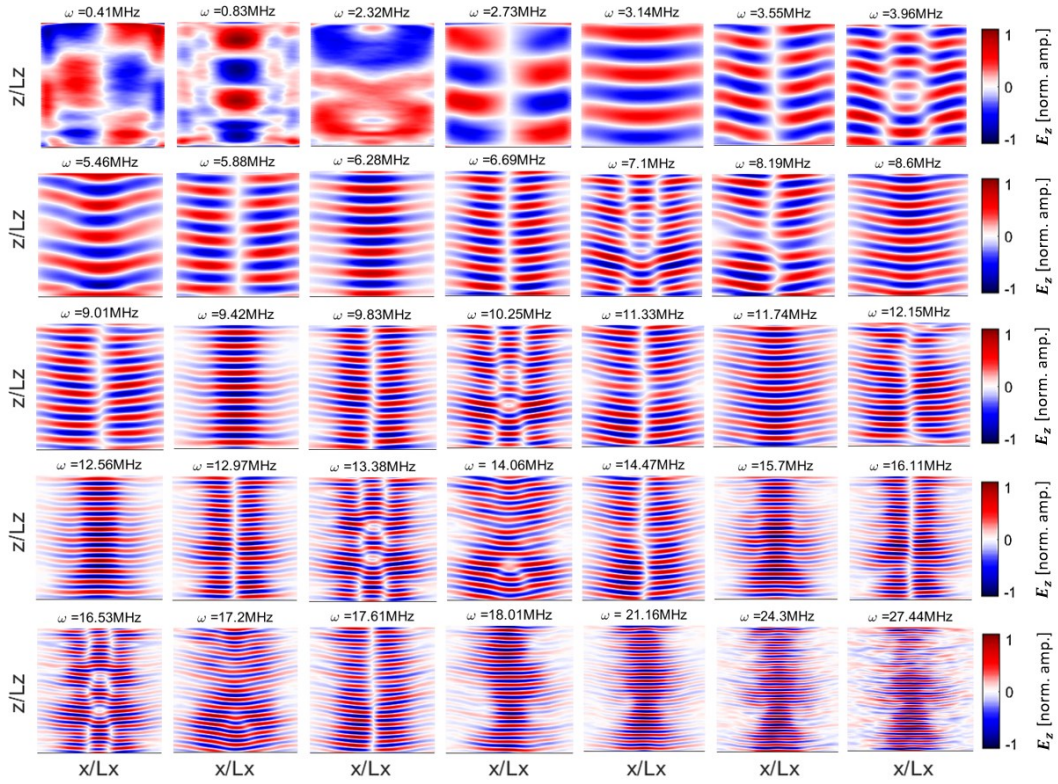


Fig. 3.20 Visualization of the dominant DMD modes of the azimuthal electric field data for a radial-azimuthal test case with $E_y = 50{,}000\ Vm^{-1}$.

The presented example is indeed a simple one with a rather known physics, which we used to demonstrate a situation where DMD can offer a significant advantage over the FFT analysis. This advantage can be further appreciated in more complex cases where the underlying physics and, consequently, the principal instabilities within the system, are less understood. In such situations where the dispersion relations of the observed instabilities might not be available or well established, it becomes increasingly challenging to establish a connection between the spatial and the temporal spectra of the instabilities through classical FFT analyses.

3.6.1. Analysis of the sensitivity of the DMD-identified spatial modes and frequencies to the number of ranks

In this subsection, the sensitivity of the OPT-DMD-identified modes and their spatial structures and frequencies to the truncation rank is assessed. We performed OPT-DMD on the J_{ey} data of the baseline radial-azimuthal test case with different number of ranks and compared the characteristics of the most dominant DMD modes that correspond to four distinct peaks in the temporal FFT spectrum of the data.

Fig. 3.21 illustrates the spatial patterns of the modes, their corresponding frequencies, and the discrete DMD spectra with different number of ranks (r). It is first observed that, for each value of r, the DMD spectrum agrees with the FFT plot, and that the high-energy modes coinciding with the FFT peaks are represented. Second, the spatial structures of the modes are similar when DMD is calculated with different number of ranks. Third and last, the frequencies associated with each mode show a variation of around 1-6 % across all the cases.

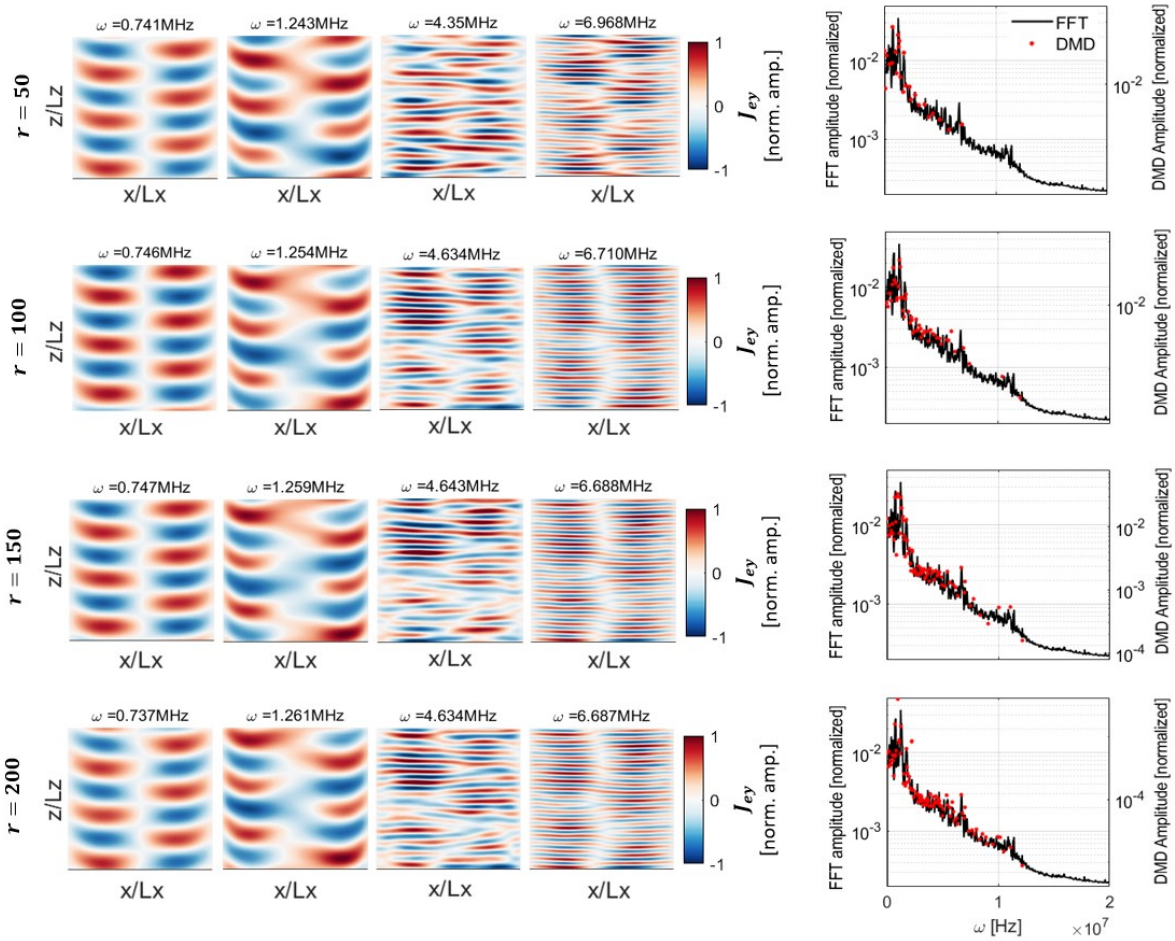


Fig. 3.21 Effect of the number of ranks on the DMD frequency spectra and on the shape of the sample dominant frequency modes. The DMD spectra are compared against the temporal FFT plots. The radial-azimuthal test case corresponds to the baseline conditions. The FFT and the DMD are applied to the J_{ey} data.

Based on these observations, it is concluded that the spatiotemporal characteristics of the modes exhibit a very low sensitivity to the number of ranks for the DMD model. This implies that the most significant features are captured even when the number of retained modes varies. The consistency of the modes' characteristics regardless of the chosen rank underscores the robustness of the OPT-DMD method for the task of dominant patterns extraction and modes identification.

3.6.2. Analysis of the effect of rank inflation on the DMD-identified spatial modes

A well-known fundamental limitation of the SVD/POD-based approaches, such as DMD, is their ineffective treatment of translational/rotational invariances in the data. The reason is that SVD relies on the correlation between the rows and the columns of the data matrix, and any misalignment in the data can lead to an overestimation of the rank of the data due to the extra variations induced by translation and/or rotation. This has an implication for the effectiveness of the dimensionality reduction we hope to achieve with the SVD-based methods for data that features fundamentally low-rank objects but shifted through space. In such cases, SVD struggles to accommodate the shifts or translational invariances in the patterns, and instead represent the object with an increased number of ranks, leading to an artificial rank inflation. A relevant example of this in the plasma applications is the existence of travelling waves.

Looking at the DMD spectra in Fig. 3.15 and Fig. 3.17, we notice that, around the FFT peaks, the DMD spectra often contain multiple modes. These instances correspond to dominant travelling waves, to represent which DMD requires several modes.

Here, for the baseline radial-azimuthal test case, we demonstrate the rank inflation by showing in Fig. 3.22 the spatial patterns of a group of modes that represent the first four dominant DMD modes of the J_{ey} data calculated using 200 ranks. Each row in Fig. 3.22 illustrates a set of modes with close frequencies to a certain peak in the FFT spectrum (Fig. 3.21, last row), hence, representing a single wave.

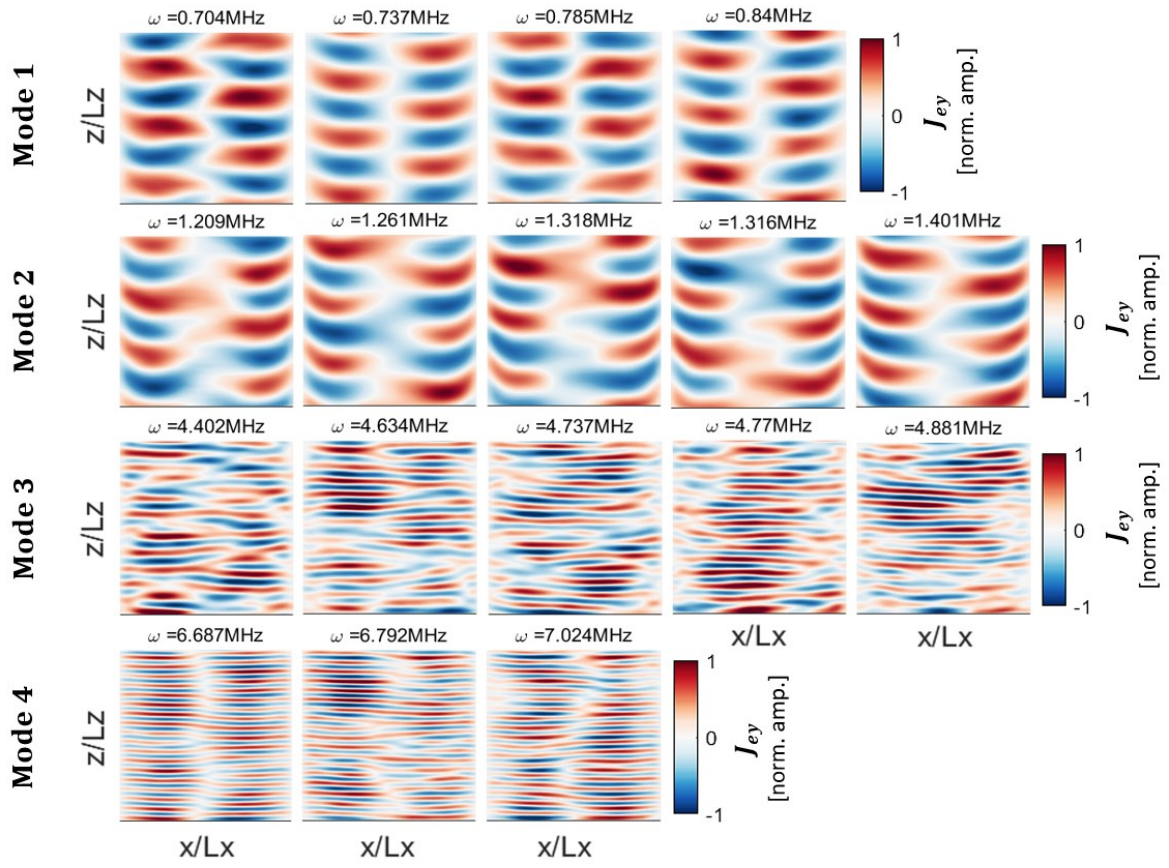


Fig. 3.22 Demonstration of the effect of DMD rank inflation, and the similarity of the shape of the DMD modes at close frequencies. The distinct J_{ey} DMD mode numbers 1 to 4 correspond to those on the bottom row of Fig. 3.21.

It is evident that each group of modes share similar spatial characteristics except that they are shifted along the azimuthal direction. Therefore, although the rank inflation due to presence of travelling waves artificially increases the dimensionality of the system's dynamics as captured by the OPT-DMD method, it does not raise any concern regarding modes identification since selecting any mode from a "mode cluster" equivalently provides similar information about the spatial structure of the wave.

3.7. Chapter conclusions

In this Chapter, a practical introduction was provided to the concepts and the mathematical foundations of SVD/POD and DMD. We underlined the relationship between these data-driven methods

and presented their applications toward dimensionality reduction of high-dimensional plasma simulations' data as well as for pattern extraction and modes identification.

Regarding SVD/POD, we demonstrated that the method enables understanding the nature of the data in terms of low-rank (low-dimensional) representability and can be used for data compression and de-noising. It was also shown how the SVD's capability to represent and reconstruct unseen (test) data based on the reduced coordinates it has learned from the ground-truth data varies with the training-to-test ratio.

Concerning DMD, three principal variants of the method were introduced, namely, Exact (basic) DMD, OPT-DMD, and parametric DMD with the OPT-DMD algorithm at its core. We additionally discussed the possibility of applying DMD to multi-property state vectors of plasma systems that may include linear state variables as well as their nonlinear observables (functions).

We focused on the Exact DMD and the OPT-DMD variants in this Chapter applied to single-property state vectors. The performance of Exact DMD and OPT-DMD were compared toward low-rank dynamics representation of the ground-truth data. We observed that OPT-DMD is much more reliable to carry out this task. This was highlighted to root in the fact that OPT-DMD enables constraining the eigenvalues (frequencies) associated with the modes to near the imaginary axis. This constraining of the eigenvalues allows the OPT-DMD-derived modes to remain stable over arbitrary time durations and not tend to zero or diverge to infinity as is often the case with the Exact DMD method when applied to noisy data of real-world systems. These results emphasized the noise robustness of OPT-DMD, which makes this variant well suited for applications to the plasma systems' data.

For OPT-DMD, we evaluated the convergence characteristics of the algorithm against the number of ranks. It was pointed out that, even though increasing the truncation rank improves consistently the similarity between the OPT-DMD-reconstructed and the ground-truth data, a very high rank number can lead to overfitting the training data by forcing the algorithm to learn the noise as well. This may in turn affect the forecasting accuracy of an OPT-DMD model, an aspect that will be discussed in detail in Chapter 4. However, too few number of ranks leads to the loss of important information in the data. It is for these reasons that the selection of the optimal number of DMD ranks comes as a trade-off between the accuracy of reconstructing training data and predicting the test data.

We presented a novel application of OPT-DMD to complement and/or augment the spectral analyses conventionally carried out using FFT technique. We demonstrated the consistency between the frequency spectra of the modes from the OPT-DMD and the FFT of plasma data (azimuthal electric field and axial electron current density) across various radial-azimuthal test cases. This illustrated how OPT-DMD enables a more visually accessible, simultaneous characterization of the spatial-temporal properties of the plasma instabilities and fluctuations. The important benefits of the use of OPT-DMD for spectral analyses of data in terms of informing the experimental characterizations of plasma oscillations were underlined.

Finally, we analyzed the sensitivity of the spatial structure and the frequency of the identified OPT-DMD modes to the number of DMD ranks. It was shown that the modes' characteristics from OPT-DMD

are highly robust with respect to the truncation rank. We also assessed the influence of rank inflation on the identification of physically meaningful dominant coherent structures in the plasma data using OPT-DMD. We showed that different DMD modes with frequencies around any specific peak in the temporal FFT plot of the data feature very similar spatial distributions. This underlined that the rank inflation does not affect the reliability of OPT-DMD as a powerful tool to infer the spatiotemporal characteristics of the instabilities.

Chapter 4 Optimized dynamic mode decomposition for the plasma state forecasting

In the previous Chapter, Optimized DMD (OPT-DMD) was demonstrated to enable a robust identification of the dominant spatiotemporally coherent modes underlying the data. This was done across various test cases that corresponded to different plasma conditions of a cross-field radial-azimuthal simulation configuration. It was emphasized that OPT-DMD significantly improves the analysis of complex plasma processes, revealing information that cannot be derived using conventionally employed spectral analyses such as FFT. Because OPT-DMD can be constrained by construction to produce stable ROMs, we extend the application of the method in this Chapter and investigate the capabilities of the linear ROMs from OPT-DMD toward forecasting in time of the plasma dynamics in several plasma configurations. The demonstration test cases correspond to cross-field plasmas in 2D Hall-thruster-representative axial-azimuthal and radial-azimuthal geometries as well as in a 2D Penning configuration. The predictive capacity of the OPT-DMD-derived ROMs is assessed primarily in terms of short-term dynamics forecast or, in other words, for large ratios of training-to-test data. However, the utility of the ROM for long-term dynamics forecasting is also presented for an example case in the radial-azimuthal configuration. For this same example case, we further assess the predictive performance of the OPT-DMD ROM when used for long-term, simultaneous forecasting of a multi-property state of the plasma system.

This Chapter comprises four sections. Section 4.1 provides an overview of the setups and conditions of the high-fidelity simulations performed to generate the training and the test data for OPT-DMD evaluations. Section 4.2 discusses the performance of the OPT-DMD ROMs toward forecasting the plasma state across the various adopted demonstration cases. Section 4.3 assesses the sensitivity of the OPT-DMD ROMs to noise and, finally, section 4.4 summarizes the main results and findings.

4.1. Description of the adopted demonstration cases

The setups of the test-case simulations in this Chapter follow, in general, those of the recent benchmark problems defined within the E×B plasma community for code-to-code verifications of PIC plasma models. These benchmark problems represent cross-field plasma discharges in various 2D

configurations, namely, a configuration that resembles an axial-azimuthal cross-section of a Hall thruster [86], a configuration representing a radial-azimuthal cross-section of a Hall thruster [43], and a Penning-trap-representative configuration [87]. As was mentioned in section 1.4, the IPPL-Q2D code has been verified in detail against all these benchmark cases [81][82][84][85]. The axial-azimuthal and the radial-azimuthal benchmark setups were additionally used to perform extended parametric studies using IPPL-Q2D on the influences that various physical factors have on the physics of instabilities and electron transport in the plasma configurations of these benchmark problems [48]-[51].

All these verification efforts and studies allowed us to generate an expanded set of high-fidelity data that spans over a broad range of plasma conditions. The data to be used in the present Chapter for the demonstrations of the predictive performance of OPT-DMD ROMs correspond to a subset of our extensive dataset. The plasma conditions associated with the selected data subset will be clarified in the following subsections, where we will also provide some details on the setups for the different test-case simulations.

4.1.1. Axial-azimuthal Hall-thruster-representative configuration

This test-case problem captures certain dynamics of a Hall thruster plasma along an axial-azimuthal cross-sectional plane of the thruster. These include the propagation of azimuthal instabilities and the spatial transition of small-wavelength instability waves to longer-wavelength modes as we cross the plane of maximum radial magnetic field intensity (vertical dashed red line in Fig. 4.1). The computational domain is a Cartesian $x - z$ plane as depicted in Fig. 4.1(a), with the dimensions of 0.025 × 0.0128 m. The x-axis is along the axial coordinate and the z-axis is along the azimuth. The y-coordinate represents the unresolved radial direction.

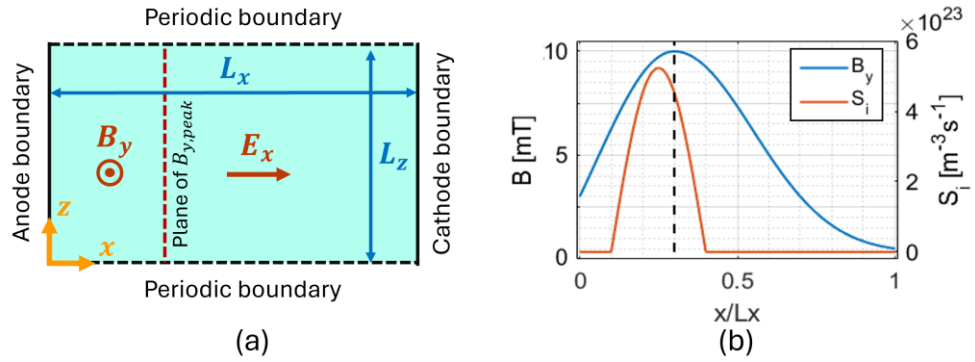


Fig. 4.1 (a) Schematics of the coordinate system, the computational domain, and the applied electric and magnetic fields corresponding to the axial-azimuthal test-case simulations. (b) Axial profiles of the imposed magnetic field (B_y) and the ionization source (S_i).

The quasi-2D axial-azimuthal "ground-truth" simulations feature an imposed temporally invariant ionization source (S_i) with a cosine-shaped profile along the x-direction and a peak value corresponding to the total axial ion current density (J), which is set as a simulation parameter [86]. The applied radial magnetic field (B_y) follows a Gaussian axial distribution with the peak value of 20 mT. The profiles of the imposed ionization source and the magnetic field are presented in Fig. 4.1(b).

The various axial-azimuthal test cases differ from each other based on the selected value of J. Table 4.1 provides the values of this parameter for each of the test cases.

Test case	J value
1	$1/8\ J_0 = 50\ Am^{-2}$
2	$1/4\ J_0 = 100\ Am^{-2}$
3	$1/2\ J_0 = 200\ Am^{-2}$
4 (Baseline)	$J_0 = 400\ Am^{-2}$

Table 4.1 Summary of the values of J corresponding to various axial-azimuthal test-case simulations; the baseline case has the J value equal to that of the axial-azimuthal benchmark [86].

In the following paragraphs, we provide an overview of how the test-case simulations were set up.

To initiate the simulations, the electron and the ion particles are loaded uniformly throughout the simulation domain with a density of $5 \times 10^{16}\ m^{-3}$. The particle species are sampled from a Maxwellian distribution function at 10 eV for the electrons and 0.5 eV for the ions.

Dirichlet boundary conditions are implemented for the plasma potential. The boundary conditions in the axial direction are 200 V at x = 0 (domain's anode side) and 0 V at x = 0.025 m (domain's cathode side). In the azimuthal direction, a periodic boundary condition with a zero-volt reference potential has been applied to both ends.

With regards to the particles' boundary conditions, both the ions and the electrons reaching either the anode or the cathode side are removed from the simulation. However, to maintain the discharge, electrons are constantly injected into the domain from the cathode side. The number of electrons to inject is obtained using the so-called "quasi-neutrality" approach, more details on which can be found in Refs. [75][83]. The cathode electrons are sampled from a Maxwellian distribution at 10 eV. The azimuthal direction is periodic, and the particles leaving the domain at one end are injected back from the other end.

Table 4.2 summarizes the values of the main computational and physical parameters used in our axial-azimuthal test-case simulations.

The ground-truth data for the different axial-azimuthal test cases (Table 4.1) are obtained from PIC simulations using with IPPL-Q2D [81][82] with a domain decomposition of M = 40 and N = 20. Compared to equivalent full-2D simulations, the quasi-2D ones have a computational gain by a factor of about 6 [81].

4.1.2. Penning discharge configuration

The setup and conditions of the ground-truth simulation performed for the Penning discharge test case are identical to the corresponding benchmark problem [87]. The benchmark is based on recent numerical research into the physics of plasma within the Penning trap configuration, e.g., in Ref. [52], in order to investigate the formation and the characteristics of large coherent structures called the spoke modes [see also subsection 1.1.4]. In the Penning discharge case, after an initial transient, a large coherent spoke-like structure forms within the simulation domain that exhibits a counter-clockwise

rotation. The discharge evolution, hence, represents a quasi-periodic behavior at the system's steady state.

Parameter	Value [unit]
Computational parameters	
Axial domain length (L_x)	2.5×10^{-2} [m]
Azimuthal domain length (L_z)	1.28×10^{-2} [m]
Cell size ($\Delta x = \Delta z$)	50 [μm]
Number of cells along the axial direction (N_i)	500
Number of cells along the azimuthal direction (N_k)	256
Time step (ts)	5×10^{-12} [s]
Total simulated duration (t_{sim})	30 [μs]
Initial number of macroparticles per cells for axial grid (N_{ppc_x})	150
Initial number of macroparticles per cells for azimuthal grid (N_{ppc_z})	375
Physical parameters	
Initial plasma density ($n_{i,0}$)	5×10^{16} [m^{-3}]
Initial electron temperature ($T_{e,0}$)	10 [eV]
Initial ion temperature ($T_{i,0}$)	0.5 [eV]
Discharge Voltage (V_d)	200 [V]

Table 4.2 Summary of the computational and physical parameters used for the quasi-2D simulations of the axial-azimuthal test-case problem

The simulation domain of the Penning discharge test case corresponds to a 2D Cartesian plane in the $x - y$ coordinate, with x denoting the axial direction, and y representing the radial direction. Fig. 4.2 shows a schematic of the simulation setup.

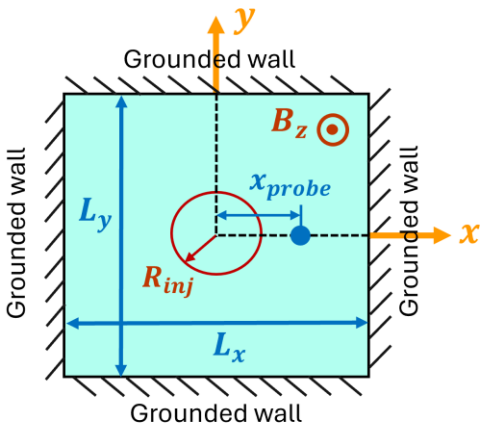


Fig. 4.2 Schematic of the setup of the Penning discharge simulations; adopted from Ref. [87] and re-illustrated.

The length of the simulation domain is the same along the x and the y directions and is equal to 0.05 m, i.e., L_x = L_y = 0.05 m. The total simulation time is 500 μs. A uniform, constant magnetic field is externally applied. A virtual ion density probe is placed within the simulation domain at the coordinates of x = 0.0125 m and y = 0.

As a summary of how the simulation was set up, the simulation domain is initially empty. A cylindrical particle injection source with a radius of 5 mm is envisaged in the center of the domain that injects helium ions and electrons at fixed rates throughout the simulation time [85][87]. The injected electrons and ions are sampled from Maxwellian distribution functions at the temperatures of 15 eV for the electrons and 0.025 eV for the ions. All four boundaries of the domain are electrically grounded and are perfectly absorbing. Thus, a zero-volt potential boundary condition is applied to all sides of the domain, and the particles crossing any of the four domain boundaries are removed from the simulation.

Table 4.3 presents a summary of the physical and the numerical conditions used for the Penning test-case simulation.

Parameter	Value [unit]
Computational parameters	
Domain length $(L_x = L_y)$	5×10^{-2} [m]
Number of cells in each direction $(N_i = N_j)$	256
Time step (ts)	4×10^{-11} [s]
Total simulated duration (t_{sim})	500 [μs]
Physical parameters	
Intensity of the applied magnetic field (B_z)	10 [mT]
Radius of the injection source (R_{inj})	5 [mm]
Electron injection current (I_e)	-20×10^{-3} $[Am^{-1}]$
Electron injection temperature $(T_{e,0})$	15 [eV]
Ion injection current (I_i)	8×10^{-3} $[Am^{-1}]$
Ion injection temperature $(T_{i,0})$	0.025 [eV]
Electric potential at the boundaries (ϕ_w)	0 [V]

Table 4.3 Summary of the computational and physical parameters used for the ground-truth quasi-2D PIC simulation of the Penning discharge test-case problem

Quasi-2D simulation of the Penning test-case problem was carried out using IPPL-Q2D [81][85] with a domain decomposition of M = N = 45 along the axial and the radial directions. This approximation order faithfully captures the essential underlying physics of the problem [85] and is, thus, suitable from the fidelity perspective for our purposes of demonstrating OPT-DMD in subsection 4.2.2.

4.1.3. Radial-azimuthal cross-field discharge configuration

The setup of the radial-azimuthal test-case simulations was presented in detail in section 3.3. For the radial-azimuthal simulations, the coordinate system is defined such that the x-axis represents the radial direction, the z-axis is along the azimuth, and the y-axis shows the unresolved axial dimension.

As was pointed out in section 3.3, the test-case simulations feature a temporally constant ionization source with a cosine distribution along the x-direction and the peak value of S. The ionization source is imposed to compensate for the flux of particles lost to the walls. Furthermore, for all the test cases, a constant axial electric field (E_y) and a constant radial magnetic field (B_x) are applied.

The various radial-azimuthal test cases in this Chapter differ from each other based on the set of values chosen for the ionization source peak, the axial electric field, and the radial magnetic field. Table 4.4 reports the values of these plasma conditions for each adopted test case.

Test case	S value	E_y value	B_x value
1 (Baseline)	$S_0 = 8.9 \times 10^{22}\ m^{-3}s^{-1}$	$E_0 = 10{,}000\ Vm^{-1}$	$B_0 = 0.02\ T$
2	S_0	$0.5E_0$	B_0
3	S_0	$3E_0$	B_0
4	S_0	$5E_0$	B_0
5	S_0	E_0	$1.5B_0$
6	$1/16\ S_0$	E_0	B_0
7	$1/8\ S_0$	E_0	B_0
8	$3S_0$	E_0	B_0
9	$6S_0$	E_0	B_0

Table 4.4 Summary of the values of the physical parameters (plasma conditions) used for the radial-azimuthal test-case simulations; the values of the plasma conditions for the "Baseline" case are equal to those of the radial-azimuthal benchmark [43].

All test-case simulations are run for 30 μs. Case 1 (the baseline) is also run for a longer duration of 135 μs. This long-run was to generate the data for the investigations in subsections 4.2.3.1 and 4.2.3.2. The set of values of the physical parameters S, E_y, and B_x for the test case 1 correspond to the conditions of the radial-azimuthal benchmark [43].

The plasma in the adopted radial-azimuthal configuration exhibits a complex behavior that is primarily influenced by two instability modes, the ECDI and the MTSI (see subsection 1.1.4). Following the nonlinear development of these instabilities during the transient phase of the discharge evolution, a quasi-steady state is reached, which is characterized by the nonlinear interactions between the ECDI and the MTSI. These nonlinear interactions lead to the periodic mitigation and growth of these instabilities. This dynamics is reflected in the time evolution of the macroscopic plasma properties, particularly the ion number density and the radial electron temperature, in terms of a cyclic rise and drop [43][48][84].

The complex nature of the system's dynamics and the dominant role of the underlying instability modes in determining the global system response make the radial-azimuthal problem an interesting one to assess the predictive performance of reduced-order models from OPT-DMD.

4.2. Performance of OPT-DMD for plasma dynamics forecasting

We assess the utility of OPT-DMD to derive reliable ROMs for plasma systems that enable forecasting the time evolution of the dominant spatiotemporally coherent structures in the data.

The emphasis here is on gauging the extent of predictiveness of the linear models from the OPT-DMD algorithm across a range of plasma test cases with varying degrees of nonlinearity in the underlying processes and interactions.

The performance of the OPT-DMD ROMs is assessed through comparing the ground-truth and the predicted temporal evolutions of the spatially averaged plasma properties in each test case. Additionally, the 2D snapshots of plasma properties from the OPT-DMD model for each test case are juxtaposed against the snapshots from the corresponding ground-truth simulation at sample time instants within the test window. This provides a visual means of comparing the extent of similarity between the OPT-DMD models' prediction and the ground-truth data.

The difference between each ground-truth snapshot from the PIC simulation and the predicted one from the OPT-DMD model is also quantified in terms of a loss factor ($\mathcal{L}$), which is defined as per Eq. 4.1. This equation represents the instantaneous equivalent of Eq. 3.25, which was used to calculate the average reconstruction loss ($\mathcal{L}_{avg}$).

$$\mathcal{L} = \frac{\|\mathcal{S}^{true} - \mathcal{S}^{DMD}\|_F}{\|\mathcal{S}^{true}\|_F}, \qquad \text{Eq. 4.1}$$

In Eq. 4.1, $\|\mathcal{S}\|_F$ denotes the Frobenius norm of the data matrix that represents a 2D plasma property snapshot from either the ground-truth simulation ($\|\mathcal{S}^{true}\|_F$) or the OPT-DMD model ($\|\mathcal{S}^{DMD}\|_F$). The Frobenius norm is calculated using the relation in Eq. 3.25. Loss factor definition like that provided in Eq. 4.1 aligns with the standard approach widely accepted and used in the machine learning community (see, e.g., Refs. [193] and [227]).

It is noted with regard to the loss factor, however, that reduced-order models, such as those from DMD, operate in lower-dimensional spaces by focusing on dominant modes and disregarding minor details, including noise. This dimensionality reduction inherently causes some information loss, making the loss factor sensitive to accumulated discrepancies across dimensions and often resulting in relatively large values, even when the model successfully captures essential patterns. Since ROMs are purposefully designed to omit less significant data, including noise, a higher loss factor does not necessarily indicate poor predictive quality; rather highlights the trade-off between quantitative precision and capturing dominant dynamics.

Consequently, throughout this work, the loss factor serves primarily as a comparative indicator rather than an absolute measure of accuracy. This has been in particular used to ensure loss remains bounded

over time in the ROMs' predictions in this Chapter, as well as to compare the performance of the OPT-DMD and the Phi Method models in Chapters 5 and 6. The importance of visual inspection, such as comparing snapshots, is highlighted to effectively assess the predictions' quality.

4.2.1. Axial-azimuthal test cases

To present the performance of the OPT-DMD models in the axial-azimuthal test cases, we first look at the time evolution plots of the spatially averaged azimuthal electric field (E_z) shown in Fig. 4.3 for various values of the total ion current density (J). We compare in this figure the ground-truth and the OPT-DMD-predicted E_z signals for the different test cases. A DMD truncation rank of 55 is used to develop ROMs from the E_z data. We have also superimposed on the plots of Fig. 4.3 the time variations of the loss factor between the ground-truth and the reconstructed data in each test case.

It is evident from the time evolution plots in Fig. 4.3 that the E_z signal becomes increasingly more complex with the increasing total ion current density. As is shown from previous works in the literature [24][75][82][229], the higher complexity of the E_z signal at higher J values roots in the stronger presence of underlying nonlinear and transient processes at these values of J, such as the development of the ECDI and its transition to the ion acoustic waves.

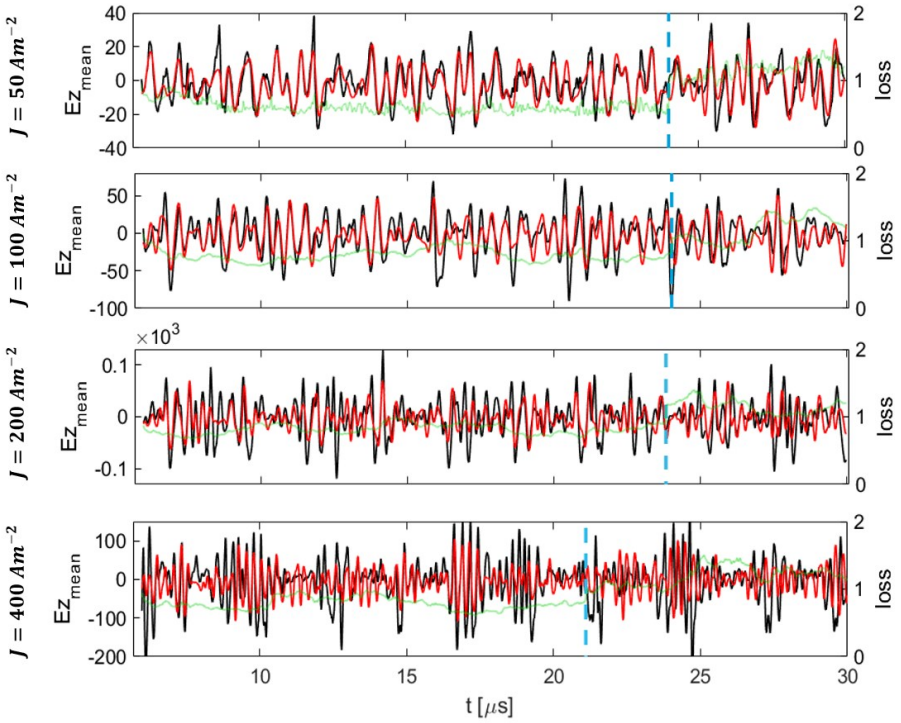


Fig. 4.3 Time evolution of the spatially averaged azimuthal electric field (E_z) for the axial-azimuthal test cases representing four different values of the total ion current density (J). Ground-truth traces from the quasi-2D simulations are illustrated as solid black lines and the OPT-DMD-predicted traces as solid red lines. The green traces represent the time variation of the loss factor (Eq. 4.1). Dashed blue lines indicate the training end time.

Nevertheless, the linear models from OPT-DMD are observed to provide reasonable predictions of the ground-truth data evolution, especially at lower ion current densities of 50 and 100 Am^{-2}, where the OPT-DMD-predicted traces follow the ground-truth very closely. The loss factor also varies quite gradually in time from the training to the test window, and it remains stable and bounded throughout the training and testing.

For the axial-azimuthal test case 1 (J = 50 Am^{-2}, Table 4.1), for which the OPT-DMD model is noticed from Fig. 4.3 to present a great predictive capability, we have shown in Fig. 4.4 the effect of the length of the training time window. This is done by comparing the time evolution and the 2D snapshots of the azimuthal electric field from the ground-truth PIC simulation and the OPT-DMD model.

Referring to the time evolution plots (Fig. 4.4(a) and (b)), we observe that, with a larger training set, the OPT-DMD model can more closely predict the ground-truth signal. This is further evident by referring to the comparison of the snapshots at four random test time instants. It is indeed clear that the predicted 2D snapshots from the ROM trained on a larger dataset are quite more similar to the ground-truth snapshots, particularly in terms of the details of the spatial distribution of the azimuthal electric field.

In any case, the model trained on a smaller dataset also captures the main spatial patterns in the data, and its predicted E_z signal largely follows the overall evolution trend of the ground-truth data as well.

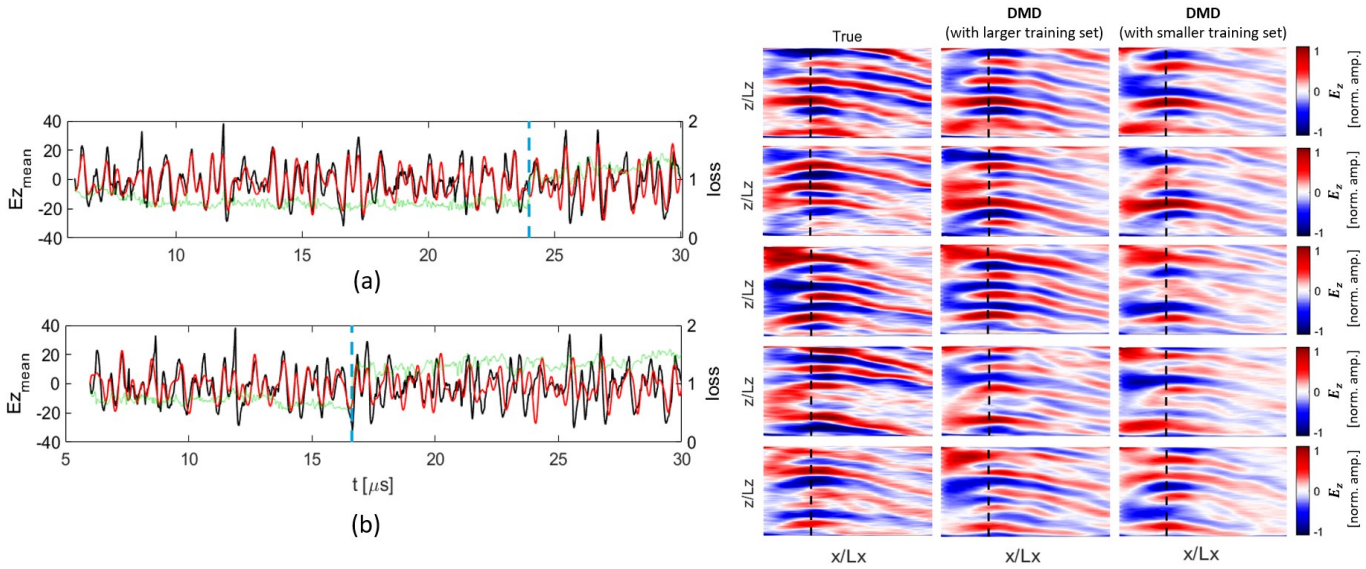


Fig. 4.4 (**Left column**) time evolution of the spatially averaged azimuthal electric field (E_z) for the axial-azimuthal test case with J = 50 Am^{-2} and two different lengths of the training window: (a) larger training set of 6 – 24 μs, (b) smaller training set of 6 – 17 μs. Ground-truth traces from the quasi-2D simulation are illustrated as solid black lines and the OPT-DMD predicted traces as solid red lines. The green traces represent the time variations of the loss factor (Eq. 4.1). Dashed blue lines indicate the training end time. (**Right column**) Comparison between the 2D snapshots of E_z from the ground-truth PIC simulation and the OPT-DMD model obtained with the larger and smaller training windows.

Concerning the above observations, we would emphasize that for OPT-DMD to correctly learn the dynamics, the training window needs to be long enough to capture the important features within the data with sufficient statistical representativeness. In this regard, for both of the cases compared in Fig.

4.4, the training window is sufficiently long to enable the model to capture the important modes. However, in the case of a longer training window, the learning can be more accurate simply due to more statistics available. Furthermore, the non-stationary nature of the signal can also influence the impact of the training data size on the accuracy of the model. The larger the training dataset is, a better fit can be found to represent data that is not fully periodic.

4.2.2. Penning discharge test case

For the Penning discharge test case, we applied OPT-DMD to the spatiotemporal data of two plasma properties of interest, namely, the ion number density (n_i) and the radial component of the electric field (E_y). OPT-DMD was applied to time-series data of these two properties from 200 μs forward to ensure that the system has arrived at its quasi-steady state [85][87]. We used a DMD truncation rank of 100 to derive the ROMs for the n_i and the E_y properties.

Fig. 4.5 and Fig. 4.6 compare the predicted dynamics of the n_i and the E_y from the OPT-DMD ROMs against the ground-truth dynamics of these properties from the quasi-2D PIC simulation. Superimposed on the plots of Fig. 4.5 are the time evolutions of the instantaneous loss between the reconstructed/predicted snapshots of each property and the corresponding ground-truth snapshots.

Before discussing the observations from Fig. 4.5 and Fig. 4.6, we point out that, even though the truncation rank is the same between the OPT-DMD ROMs for the n_i and the E_y properties, the length of the training window is different. This is because OPT-DMD was found to require more statistics of the n_i data to provide its best forecast compared to when the method was applied to the E_y data. This point roots in the inherent difference that may exist between the different plasma properties in terms of the complexity of the associated spatiotemporal data.

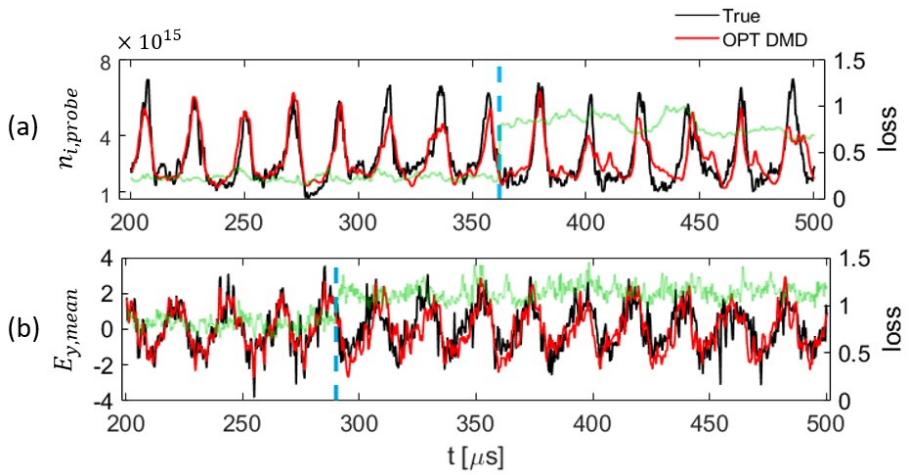


Fig. 4.5 Time evolution of the plasma properties from the Penning discharge test case: (a) ion number density at the location of the density probe ($n_{i,probe}$), (b) spatially averaged radial electric field ($E_{y,mean}$). Black traces represent the ground-truth signals from the quasi-2D PIC simulation, whereas the red traces are the reconstructed/predicted signals from the OPT-DMD models with the truncation rank (r) of 100. The green traces represent the time variations of the loss factor (Eq. 4.1). The vertical dashed blue lines indicate the end of the training windows.

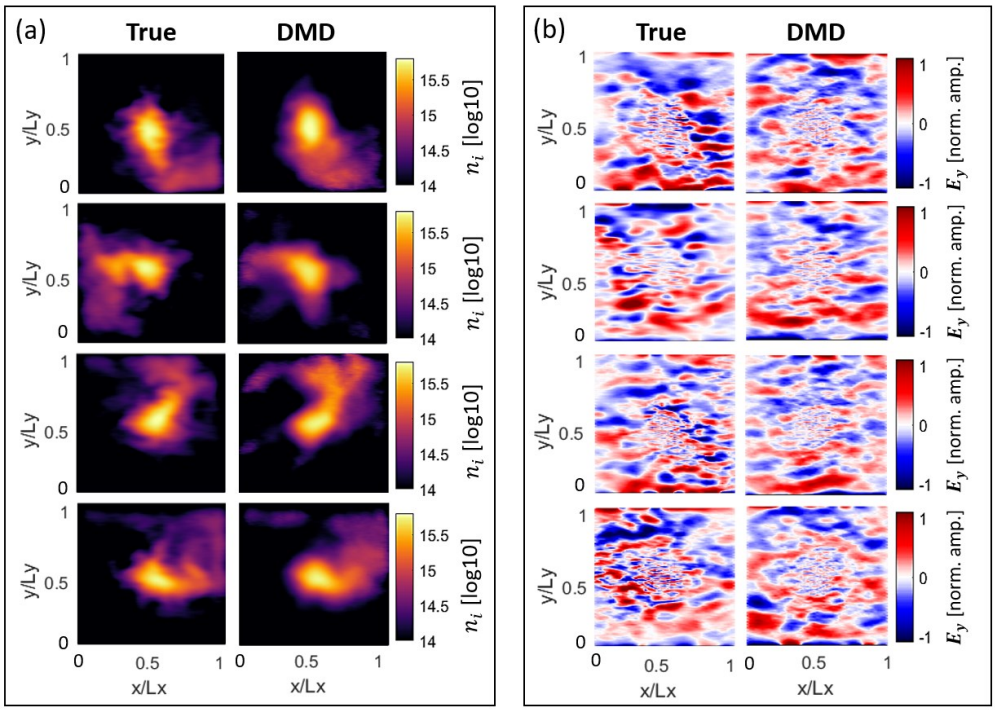


Fig. 4.6 Comparison between the 2D snapshots of plasma properties from the ground-truth quasi-2D simulation of the Penning problem and the OPT-DMD models at four time instants within the shared test window for both plasma properties (360 to 500 $\boldsymbol{\mu}$s); (a) ion number density (n_i), (b) radial electric field (E_y). The selected time instants represent one full rotation cycle of the spoke structure.

Now, in Fig. 4.5(a), we have compared the time variations of the predicted and the ground-truth n_i signal at the location of the virtual density probe. As the ion density measurements using this virtual probe were important outputs of the Penning discharge benchmark [87], and considering that the high-number-density region in the Penning problem has a rotating nature, the most appropriate measure to verify the accuracy of the n_i predictions is by comparing the OPT-DMD-predicted and the PIC-simulated n_i evolutions at the probe location. With this in mind, we observe that the predicted n_i signal within the test window is quite consistent with the ground-truth, exhibiting a similar oscillation frequency and overall comparable amplitudes.

Moreover, in Fig. 4.6(a), we have juxtaposed the predicted and the “true” (ground-truth) 2D snapshots of the ion number density at four time instants within the test window that correspond to one rotation-cycle of the spoke structure. The OPT-DMD-predicted snapshots are seen to closely resemble the ones from the PIC simulation, capturing the main spatial features.

Similar statements to those mentioned above regarding the ion number density results can also be said with regard to the predicted and the ground-truth E_y data. Indeed, the true time evolution of the spatially averaged radial electric field from the quasi-2D PIC simulation is observed from Fig. 4.5(b) to be quite closely captured by the OPT-DMD model. The predicted and the ground-truth 2D snapshots of the E_y in Fig. 4.6(b) also show a reasonable degree of resemblance to each other in terms of both the spatial structure and the magnitudes across the four sample time instants.

4.2.3. Radial-azimuthal test cases

We begin by presenting a number of radial-azimuthal test cases in which the OPT-DMD models demonstrated remarkably good predictiveness. In this regard, Fig. 4.7 shows the time evolutions of the spatially averaged plasma properties from the ground-truth PIC simulations (black lines) and the OPT-DMD models (red lines) for the test cases with the axial electric field intensity (E_y) of $0.5E_0$ and $5E_0$.

In each of these two test cases, and for all the plasma properties shown, we used the data in the time window of about 5-22 μs for the training of the OPT-DMD model, which was subsequently tested over the timeframe of about 22-30 μs. It is recalled from subsection 4.1.3 that the total duration of the radial-azimuthal test-case simulations was, in general, 30 μs. Across the two cases and all the plasma properties, we used a truncation rank (r) of 50 to 60 in OPT-DMD. The data from the first 5 μs of the discharge evolution was discarded for the training as it represents the initial transient of the system.

We observe from Fig. 4.7 that, in terms of the time evolution of the plasma properties, the traces corresponding to the OPT-DMD models closely follow the ground-truth traces within both the training and testing time windows.

The time evolutions of the loss factor ($\mathcal{L}$, Eq. 4.1), illustrated as green lines in the plots of Fig. 4.7, show an expected sudden variation when moving from the training window to the test, but they remain stable and bounded for both the training and the test datasets. The stability and boundedness of the loss of OPT-DMD's forecasts were also observed from the axial-azimuthal and the Penning discharge test cases.

We highlight that, in the section 4.3, we present an analysis of the sensitivity of the OPT-DMD ROM to the noise in data for the $0.5E_0$ case as an example.

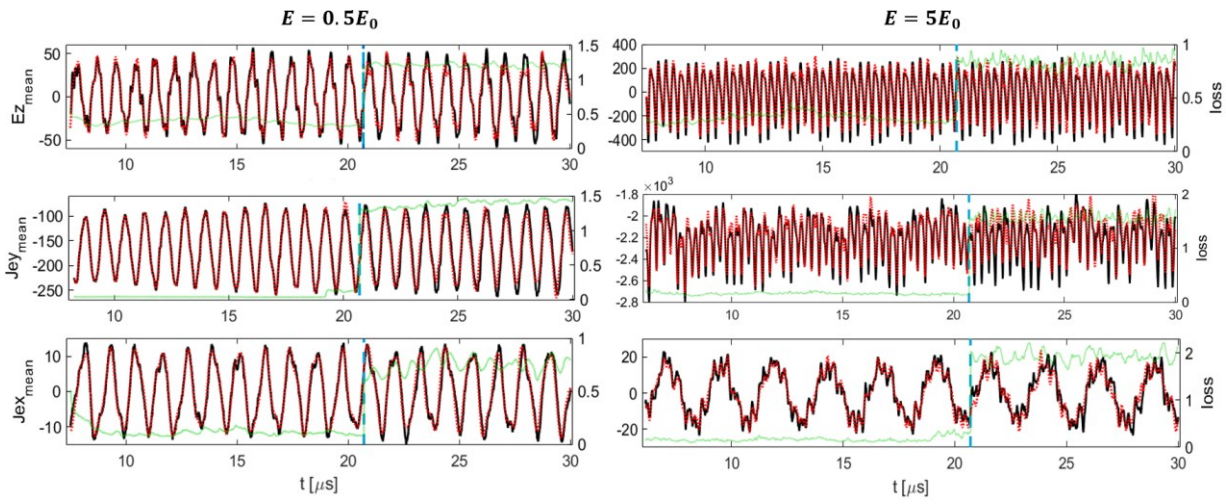


Fig. 4.7 Time evolution of the spatially averaged plasma properties for two test cases with the axial electric field values of $E_y = 0.5E_0$ (**left column**), and $E_y = 5E_0$ (**right column**). First-row plots represent the azimuthal electric field (E_z) signal, second-row plots show the axial electron current density (J_{ey}), and third-row plots correspond to the radial electron current density (J_{ex}). Ground-truth traces from the PIC simulations are shown with solid black lines, OPT-DMD-predicted traces with dotted red lines, and the green traces represent the time variations of the loss factor (Eq. 4.1). Dashed blue lines indicate the training end time.

In Fig. 4.8, we have compared the 2D snapshots of the azimuthal electric field (E_z) and the axial electron current density (J_{ey}) from the ground-truth simulations and the OPT-DMD models for the test cases of $0.5E_0$ and $5E_0$ at three randomly selected time instants within the test window. The predicted snapshots of either plasma property are seen to be closely representative of the ground-truth ones in both test cases and at all three time instants.

Another radial-azimuthal test case, in which the derived ROM from OPT-DMD exhibited great predictive capability, is the case with B_x = $1.5B_0$. Comparisons of the OPT-DMD model predictions against the ground-truth for this test case in terms of the time evolution of the averaged plasma properties and their 2D snapshots are shown in Fig. 4.9. Similar to the test cases with E_y = $0.5E_0$ and $5E_0$, we used the time window of about 5-22 μs for OPT-DMD training and an r value of 50-60 across the different plasma properties shown.

It is observed from Fig. 4.9 that the OPT-DMD predictions within the test window are highly consistent with the ground-truth data. This is true considering either the time evolutions of the plasma properties or the 2D snapshots at sample time instants. We note that, in this test case, the variation in the loss from the training set to the test is more gradual and does not feature the step change that was observed for the $0.5E_0$ and $5E_0$ test cases from Fig. 4.7. In any case, the loss factor variation over time is again seen to be bounded.

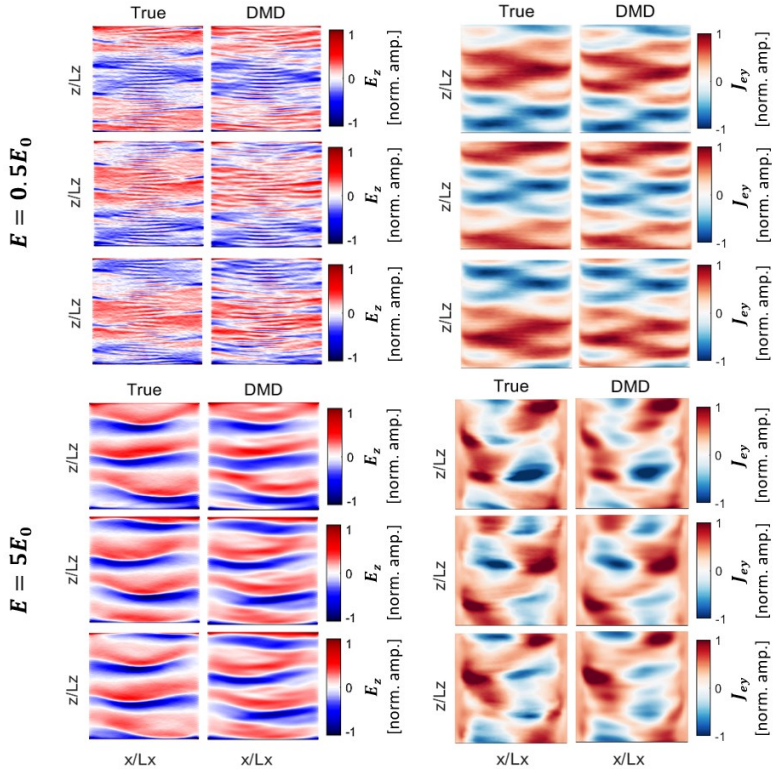


Fig. 4.8 Comparison of the 2D snapshots of plasma properties from the ground-truth radial-azimuthal simulations and the OPT-DMD models for two values of the axial electric field at three randomly selected time instants within the test window ($\sim 22 - 30\ \mu s$). (**Left column**) azimuthal electric field (E_z), (**right column**) axial electron current density (J_{ey}). The first three rows correspond to $E_y = 0.5E_0$, whereas the second three rows are for the simulation with $E_y = 5E_0$.

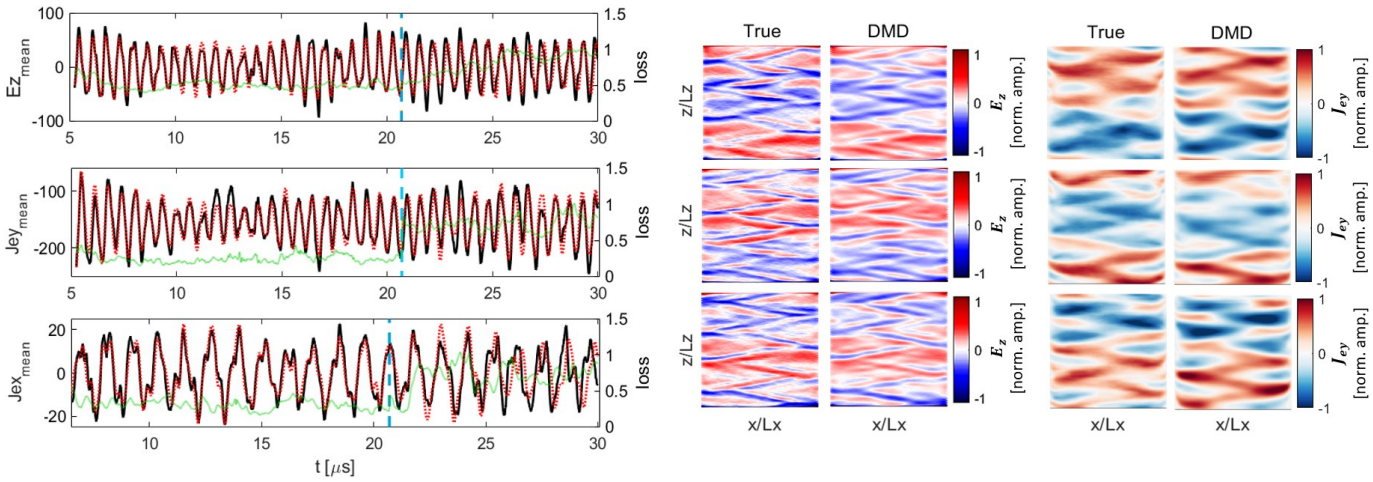


Fig. 4.9 (**Left column**) time evolution of the spatially averaged plasma properties, E_z, J_{ey}, and J_{ex}, for the radial-azimuthal test case with the radial magnetic field intensity of $B_x = 1.5B_0$. Ground-truth traces from the quasi-2D simulation are shown with solid black lines, OPT-DMD-predicted traces with dotted red lines, and the green traces represent the time variation of the loss factor (Eq. 4.1). Dashed blue lines indicate the training end time. (**Middle column**) Comparison of the 2D snapshots of the E_z from the ground-truth simulation and the OPT-DMD model at three randomly selected time instants within the test window ($\sim 22 - 30\ \mu s$). (**Right column**) Comparison of the 2D snapshots of the J_{ey} from the ground-truth simulation and the OPT-DMD model at three randomly selected time instants within the test window.

For the test cases presented so far, a common feature among the time evolution plots of the spatially averaged plasma properties (Fig. 4.7 and Fig. 4.9(left column)) is that the signals are rather periodic. This implies that, under the simulations' setup conditions corresponding to these test cases, the system exhibits a relatively quasi-steady behavior after its initial transient.

We will demonstrate next in several test cases that the predictive performance of the linear OPT-DMD models is to some extent reduced for the cases where the signal evolution becomes less regular, i.e., when the plasma system is characterized by strong underlying transient processes and highly nonlinear interactions.

In this respect, Fig. 4.10 compares the time evolution signals of the spatially averaged axial electron current density ($J_{ey,mean}$) from the PIC simulations and the OPT-DMD models for six different sets of plasma conditions that correspond to the test cases 1, 3, and 6 to 9 in Table 4.4. Noting the specificities of the discharge evolution in each test case, which are evident from the different temporal variation characteristics of the $J_{ey.mean}$ across the test cases, the end time of the training window varies between about 17 to 22 μs for the various cases, and the r value used has been between 100-150.

We observe from Fig. 4.10 that, even though the OPT-DMD predictions may not be at the same level of accuracy as those shown before for the test cases 2, 4, and 5 (Table 4.4), they still provide a reasonably close approximation of the ground-truth data. Given the simplicity and minimal computational cost of the models obtained from the OPT-DMD algorithm, on the one hand, and the complexity of the involved physics, on the other, the models' accuracy is deemed acceptable. In addition, we note that the loss factor once again remains bounded for both the training and the testing.

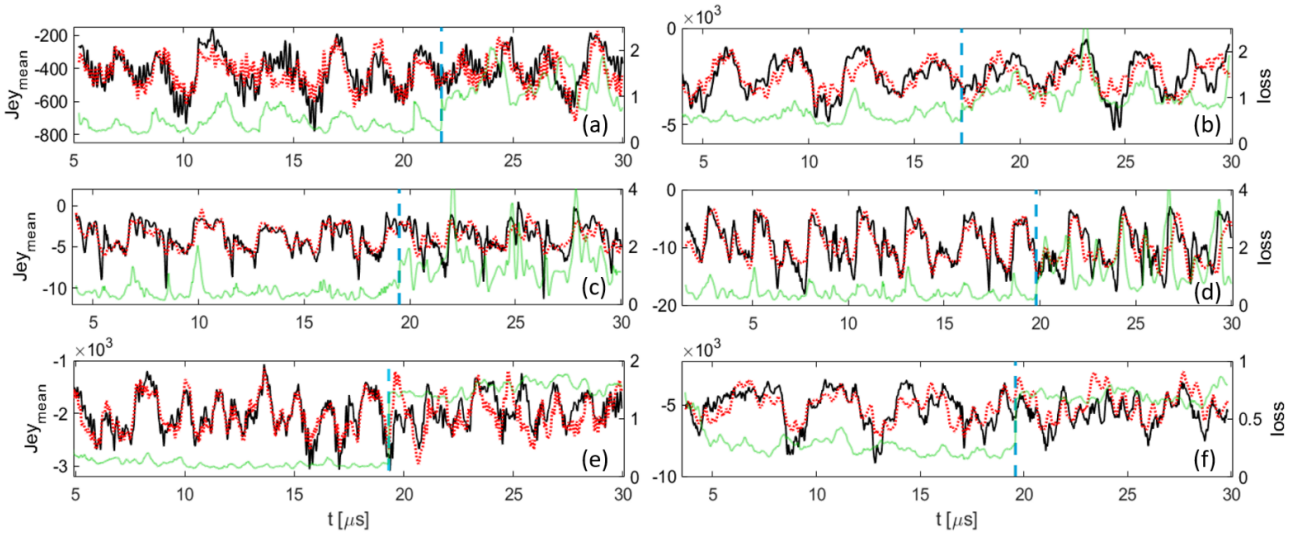


Fig. 4.10 Spatially averaged axial electron current density ($J_{ey,mean}$) signals for the radial-azimuthal test cases with various setup parameters (plasma conditions): (a) baseline conditions with axial electric field (E_y) being E_0, and the peak amplitude of the ionization source (S) being S_0. (b) $E_y = 3E_0,\ S = S_0$, (c) $E_y = E_0,\ S = \frac{S_0}{16}$, (d) $E_y = E_0,\ S = \frac{S_0}{8}$, (e) $E_y = E_0,\ S = 3S_0$, and (f) $E_y = E_0,\ S = 6S_0$. Ground-truth traces from the PIC simulations are shown with solid black lines, OPT-DMD-predicted traces with dotted red lines, and the green traces represent the time variations of the loss factor (Eq. 4.1). Dashed blue lines indicate the training end time.

We highlight that OPT-DMD can be sensitive to the time at which we truncate the data as the end of the training interval. The extent of this sensitivity varies from case to case and is related to the periodicity characteristics of the data. For the cases shown in Fig. 4.7 and Fig. 4.9, which exhibit quasi-periodic behaviors with small enough frequencies to enable capturing enough cycles during the training interval, we found this sensitivity to be minimal. As a result, for these cases, we used nearly identical training durations.

Nonetheless, the cases presented in Fig. 4.10 exhibit different periodic behaviors from each other in terms of either the time scales of the periodicity or the level of similarity between various cycles in the signal. In this respect, we found that, although a training window of about 5-22 μs is sufficient in these cases to train OPT-DMD, the corresponding models are more sensitive to the alterations in the training interval compared to the previous cases of Fig. 4.7 and Fig. 4.9. As a result, minor adjustments to the training window of 5-22 μs were made to obtain the most optimal OPT-DMD model performance.

Despite the differences observed from Fig. 4.10 between the time evolution signals of the J_{ey} from the OPT-DMD models and the ground-truth simulations, the predicted 2D snapshots in Fig. 4.11 and Fig. 4.12, presented for four randomly selected time instants within the test window, show a high degree of resemblance to the corresponding ground-truth data for all the test cases 1, 3, and 6 to 9 (Table 4.4). Indeed, we observe that the main features in the ground-truth data are captured by the predictions of the OPT-DMD models, and the magnitudes are quite similar as well.

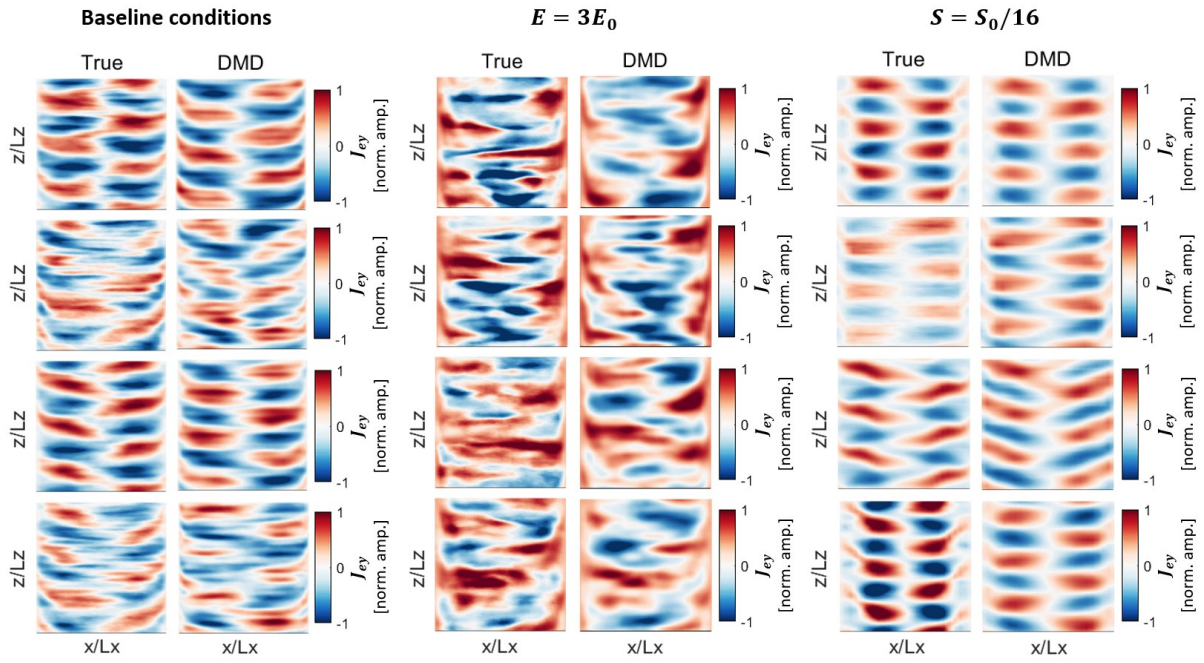


Fig. 4.11 Comparison between the 2D snapshots of the axial electron current density (J_{ey}) from the ground-truth PIC simulations and the OPT-DMD models for three sets of setup parameters (plasma conditions); (**left column**) baseline conditions, $E_y = E_0$, $S = S_0$, (**middle column**) $E_y = 3E_0$, $S = S_0$, (**right column**) $E_y = E_0$, $S = \frac{S_0}{16}$. For each set of plasma conditions, the snapshots along the rows correspond to various instances of the system evolution within the associated test window (from the dashed blue line in Fig. 4.10 until t = 30 μs).

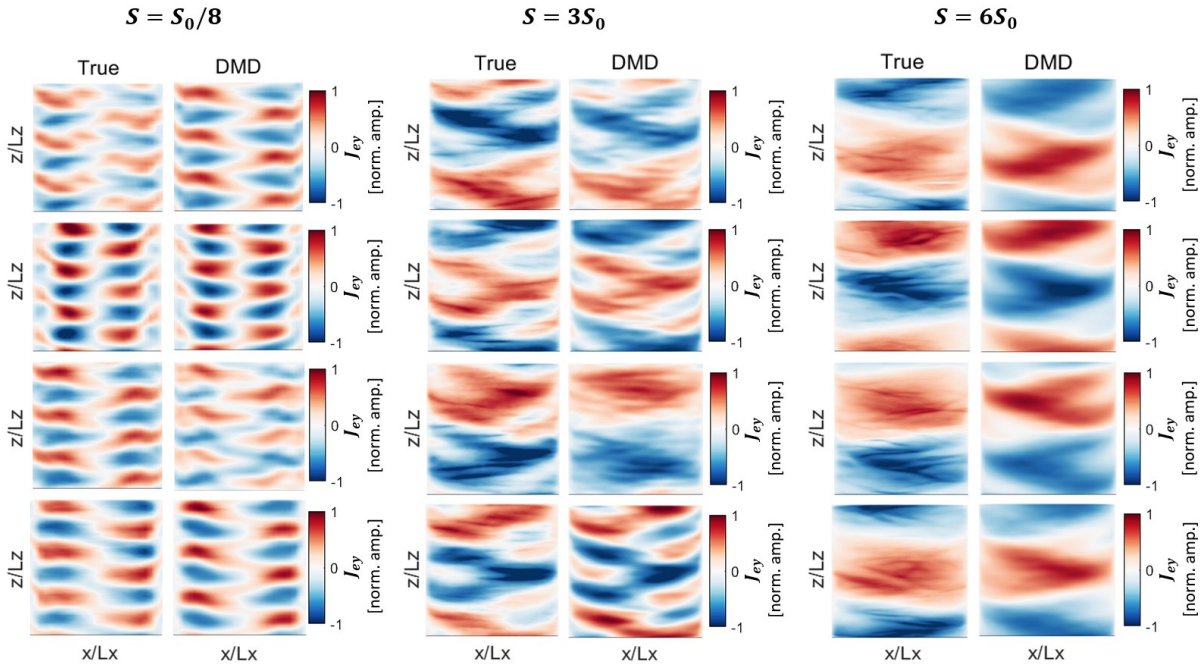


Fig. 4.12 Comparison between the 2D snapshots of the axial electron current density (J_{ey}) from the ground-truth PIC simulations and the OPT-DMD models for three sets of setup parameters; (**left column**) $E_y = E_0$, $S = \frac{S_0}{8}$, (**middle column**) $E_y = E_0$, $S = 3S_0$, (**right column**) $E_y = E_0$, $S = 6S_0$. For each set of setup parameters, the snapshots along the rows correspond to various instances of the system evolution within the associated test window (from the dashed blue line in Fig. 4.10 until t = 30 μs).

4.2.3.1. An example of long-term dynamics prediction

At this point, we would like to assess the stability and the predictive capability of an OPT-DMD model for long-term predictions of the dynamics. This investigation is done for test case 1 from Table 4.4.

We have used the data from the extended quasi-2D PIC simulation of test case 1 (with 135 μs duration) and have evaluated the performance of three OPT-DMD models obtained with different values of the truncation rank (or, equivalently, the number of DMD modes), namely, r = 55, r = 150, and r = 400. For the models with r = 55 and r = 150, the training time window is from about 5 to 22 μs. For the model with r = 400, we needed to extend the end time of the training window to about 98 μs. This was necessary because the relatively large truncation rank of 400 requires sufficient number of training data snapshots as well so that the corresponding model will still be learning the physical patterns in the data (rather than the noise) and will thus not overfit the training data.

Fig. 4.13 shows the evolution of the spatially averaged J_{ey} from the ground-truth PIC simulation and the OPT-DMD model with r = 55 over the timeframe of 5-135 μs. It is observed that, even with a training-to-test ratio of about 0.15, the predicted signal from the OPT-DMD model remains fully stable over time, and it follows the general trends of the ground-truth throughout the test window. The stability of the predicted signal from OPT-DMD is indeed remarkable in itself because, as was discussed in detail in section 3.5, obtaining a stable and, hence, reliable DMD model is quite often challenging with the original version of the DMD method (basic/Exact DMD).

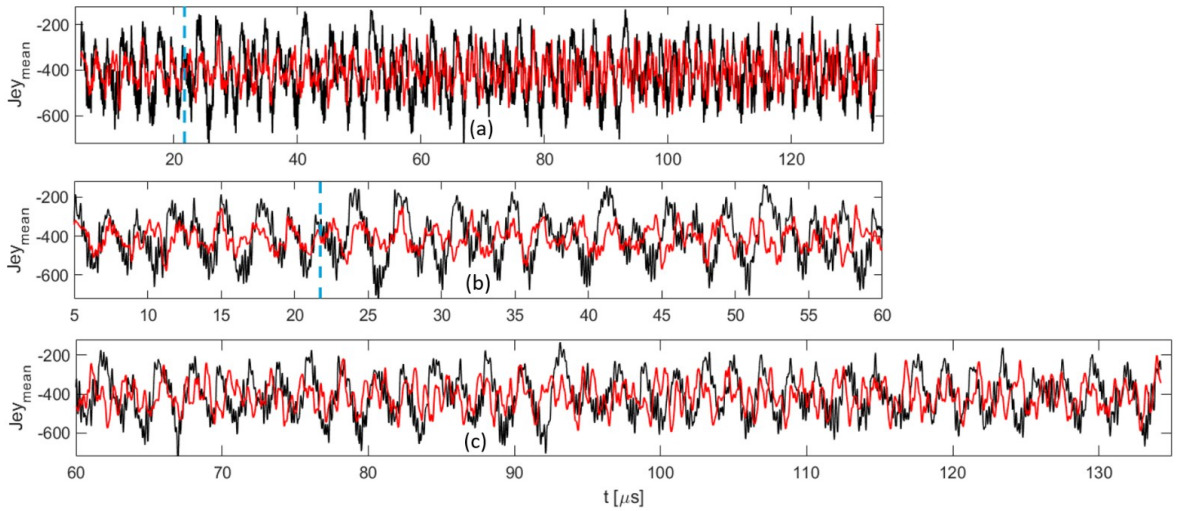


Fig. 4.13 OPT-DMD model with the low-rank approximation (r = 55) trained over a short-duration dataset (t = 5-22 μs). (a) Time evolutions of the spatially averaged axial electron current density (J_{ey}) from the PIC simulation (solid black line) and from the OPT-DMD model (solid red line). (b) zoomed view on plot (a) over the time window of 5-60 μs. (c) zoomed view on plot (a) over the time window of 60-135 μs. The setup parameters correspond to those of the radial-azimuthal baseline test case. Dashed blue line indicates the training end time.

When increasing the number of DMD modes from 55 to 150, the time evolution plots of the averaged J_{ey} in Fig. 4.14 illustrate that, whereas the model now better fits the training, its prediction within the test window follows less closely the ground-truth compared to the model with r = 55. The predicted signal

still has the same mean value as that of the ground-truth and remains stable over time, but it is overall out of sync with the signal from the PIC simulation.

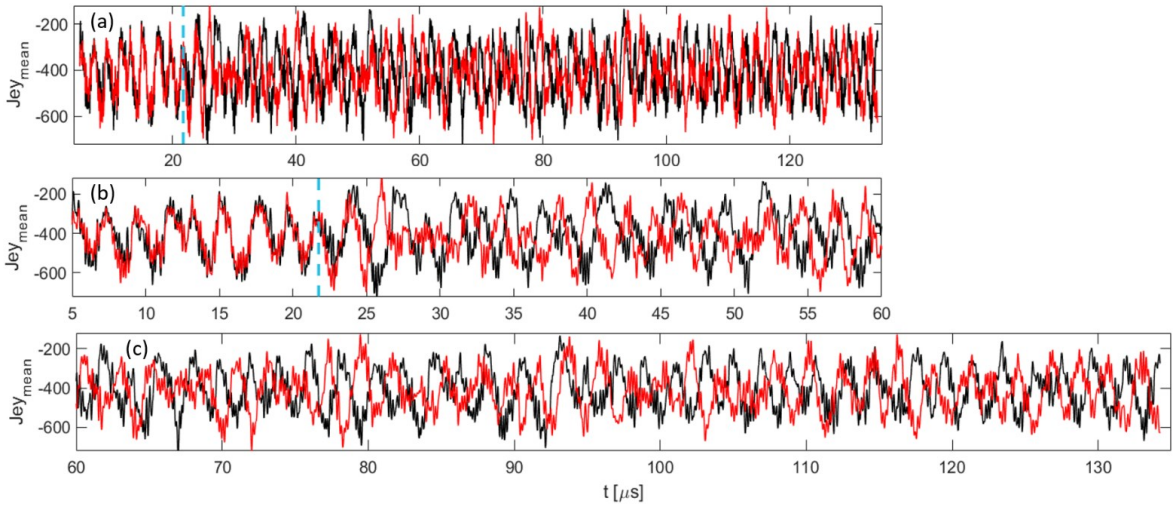


Fig. 4.14 OPT-DMD model with the moderate-rank approximation (r = 150) trained over a short-duration dataset (t = 5-22 μs). (a) Time evolution of the spatially averaged axial electron current density (J_{ey}) from the PIC simulation (solid black line) and from the OPT-DMD model (solid red line). (b) zoomed view on plot (a) over the time window of 5-60 μs. (c) zoomed view on plot (a) over the time window of 60-135 μs. The setup parameters correspond to those of the radial-azimuthal baseline test case. Dashed blue line indicates the training end time.

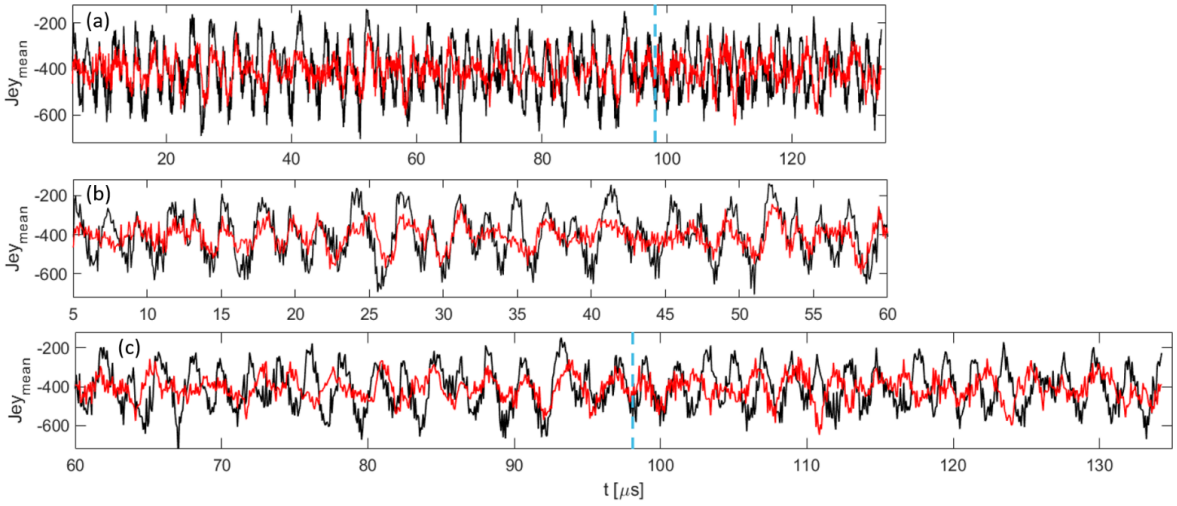


Fig. 4.15 OPT-DMD model with the high-rank approximation (r = 400) trained over a long-duration dataset (t = 5-98 μs). (a) Time evolution of the spatially averaged axial electron current density (J_{ey}) from the PIC simulation (solid black line) and from the OPT-DMD model (solid red line). (b) zoomed view on plot (a) over the time window of 5-60 μs. (c) zoomed view on plot (a) over the time window of 60-135 μs. The setup parameters correspond to those of the radial-azimuthal baseline test case. Dashed blue line indicates the training end time.

For the case of r = 400, the averaged J_{ey} signal from the OPT-DMD model is noticed from Fig. 4.15 to capture the main evolution trend of the training data, and this behavior is also exhibited by the

predicted signal into the test window. However, we can again observe that the prediction falls frequently out-of-sync with the ground-truth signal.

The time evolutions of the loss factor in J_{ey} between the ground-truth data and the predictions of the OPT-DMD models with different r values are shown in Fig. 4.16. It is again observed that, for all the OPT-DMD models, the loss remains bounded, which is particularly interesting for the models with r = 55 and 150, which were tested over a notably extended timeframe.

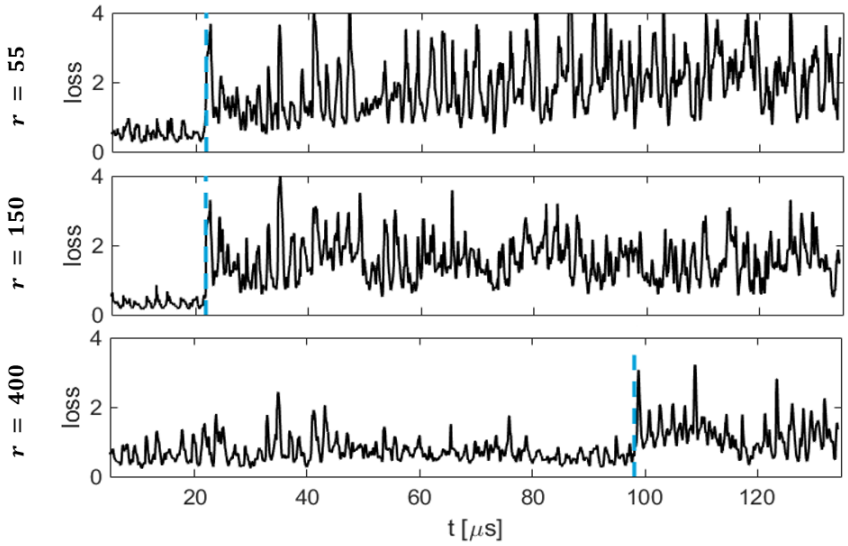


Fig. 4.16 Time evolution of the loss factor (from Eq. 4.1) between the predicted axial electron current density (J_{ey}) from the OPT-DMD models with different values of the truncation rank (r) and the ground-truth PIC simulation. Dashed blue lines indicate the training end time.

We would point out that comparing the values of loss across the different models might be, in any case, rather misleading. In fact, it is recalled that increasing the DMD's truncation rank enables the model to capture more of the patterns underlying the training data. This reduces the loss factor, which is calculated by comparing the ground-truth and the reconstructed snapshots point-by-point. However, as is seen from the time evolution plots of the J_{ey} in Fig. 4.14 and Fig. 4.15, a higher rank model does not necessarily perform better in terms of the time variation of the spatially averaged data over the testing window. This can be in part explained by the fact that DMD assumes the underlying modes to remain invariant over time. Thus, the higher the number of DMD modes retained during training, i.e., for larger r values, the more probable it is in real nonlinear systems, that feature transient behaviors, that some of the learned modes would not exist within the test set. This can consequently impact the ability of the model to properly trace the evolution trends in the ground-truth beyond the training window.

To conclude this subsection, we refer to Fig. 4.17 and Fig. 4.18 in which, for the OPT-DMD models with r = 55 and 400, respectively, we have compared the predicted 2D snapshots of the J_{ey} against the ground-truth at several time instants within the test interval.

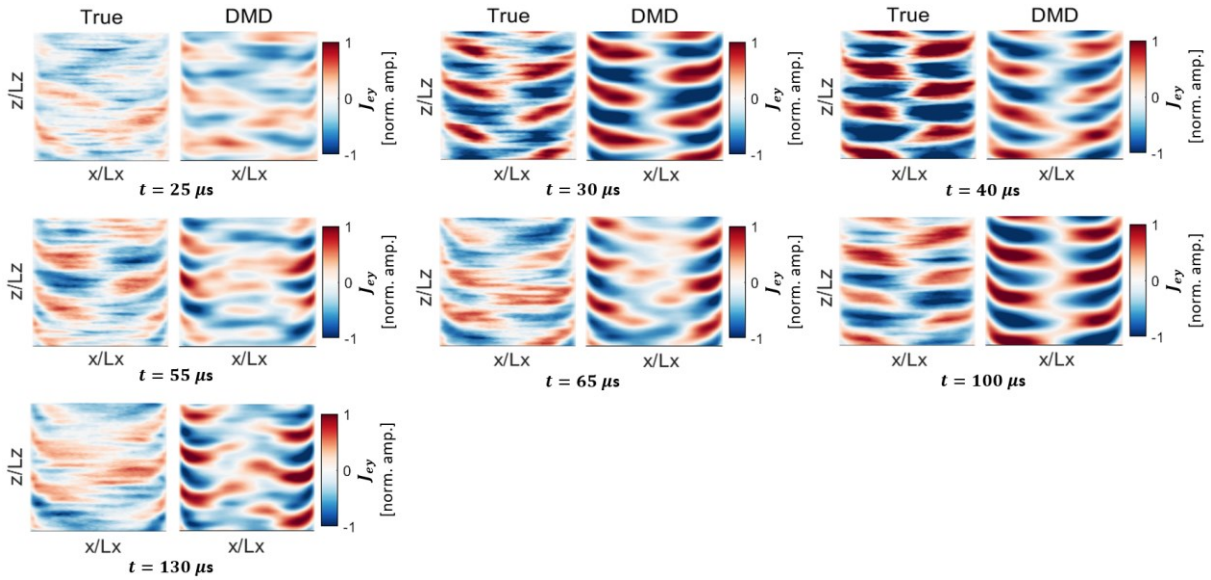


Fig. 4.17 Comparison between the 2D snapshots of the axial electron current density (J_{ey}) from the ground-truth PIC simulation and the OPT-DMD model with the rank of r = 55 over several test-window time instants. The setup parameters correspond to those of the radial-azimuthal baseline test case.

It is noted that generally, for both cases, the predicted snapshots feature the main patterns of the ground-truth, and the magnitudes are also overall comparable. However, the OPT-DMD model with r = 400 has expectedly captured finer details of the spatial distribution. This has, in turn, resulted in an increased degree of resemblance between the DMD-predicted and the ground-truth data for this case.

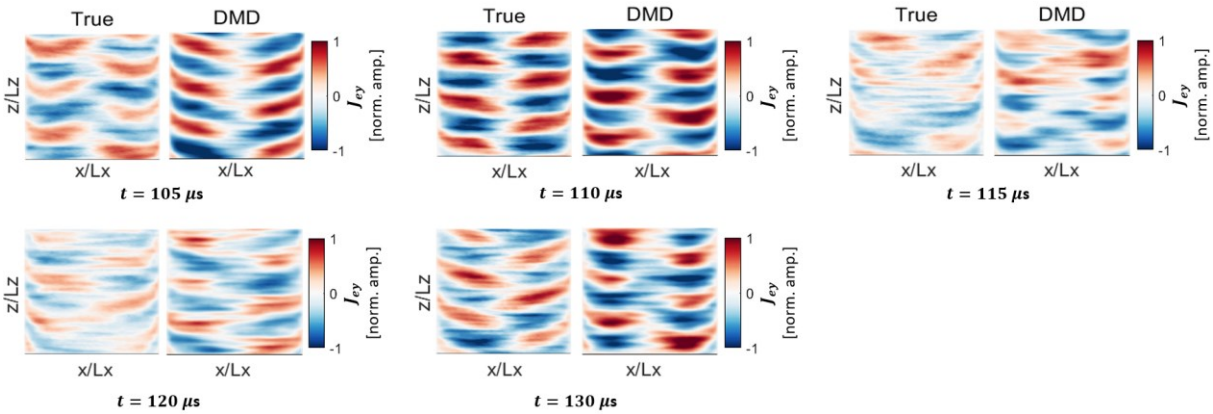


Fig. 4.18 Comparison between the 2D snapshots of the axial electron current density (J_{ey}) from the ground-truth PIC simulation and the OPT-DMD model with the rank of r = 400 over several test-window time instants. The setup parameters correspond to those of the baseline radial-azimuthal test case.

4.2.3.2. An example of simultaneous forecasting of a multi-property plasma state

In subsection 3.2.1.1, we pointed out that DMD can be also applied to augmented data matrices that comprise the spatiotemporal data (time-series snapshots) of several plasma (physical) properties. We mentioned that this augmentation of the system's state vector to which DMD is applied can enhance the predictive performance of the DMD model by enabling it to capture the interdependency between various physical properties. In this subsection, we aim to assess whether applying OPT-DMD to a multi-

property state vector of the radial-azimuthal plasma system as an example can improve the predictive performance of the resulting ROM for this problem.

To the above end, we have once again used the data from the long-duration PIC simulation of the radial-azimuthal test case 1 (the baseline case, Table 4.4). As per the approach described in subsection 3.2.1.1, the data matrix to which OPT-DMD is applied now comprises the time-series snapshots of several plasma properties, namely, the electrons' axial drift velocity ($V_{de,y}$), the electrons' radial drift velocity ($V_{de,x}$), the electrons' azimuthal drift velocity ($V_{de,z}$), the radial electron temperature (T_{ex}), the azimuthal electron temperature (T_{ez}), the electron number density (n_e), the radial electric field (E_x), and the azimuthal electric field (E_z). A DMD truncation rank of 150 is used, and the OPT-DMD model is trained over the time window of 30-90 μs.

Fig. 4.19 compares the time evolutions of the spatially averaged values of the above plasma properties from the ground-truth PIC simulation against the signals obtained from the OPT-DMD ROM. The OPT-DMD model has simultaneously learned the time dynamics of all the properties and has, in turn, provided a simultaneous prediction of their time variations over a rather extended time window of 45 μs (90-135 μs).

It is remarkable to observe that the predicted signals from the OPT-DMD model remain fully stable over time and are quite closely following the ground-truth signals across all the properties. Since the agreement between the predicted and the ground-truth signals seems to be overall very good, we have looked more closely in Fig. 4.20 at how well the OPT-DMD-predicted signals compare against the ground-truth. Fig. 4.20 provides zoomed-in views on the plots of Fig. 4.19 over a time interval around the end of the training window, i.e., t = 90 μs. Moreover, an additional comparison is shown in Fig. 4.20 between the ground-truth data and the OPT-DMD ROM predictions in terms of the time variations of the local values of the plasma properties at the mid-point within the radial-azimuthal domain.

The zoomed-in plots on the left column of Fig. 4.20 show that the OPT-DMD-predicted signals faithfully follow the ground-truth traces for all the plasma properties. In particular, the predicted time dynamics (oscillations' frequency) remains consistent with the ground-truth from the training into the testing interval. This is a marked improvement in predictions compared to the results shown through Fig. 4.13 to Fig. 4.15, where the predicted signals were observed to fall out-of-sync with the ground-truth after a rather short time into the test window, especially for the moderate-to-high truncation ranks of DMD.

The time evolution plots of the local values of the plasma properties (Fig. 4.20(right-column)) also show an overall good agreement between the predictions and the ground-truth, with the main oscillation patterns of the ground-truth signals being captured by the OPT-DMD-predicted signals. It is emphasized that the ability of a data-driven model to provide rather accurate predictions of the local behaviors of complex systems is most often a major challenge. However, the OPT-DMD model trained on a multi-property state vector is seen here to overcome this challenge in an effective manner.

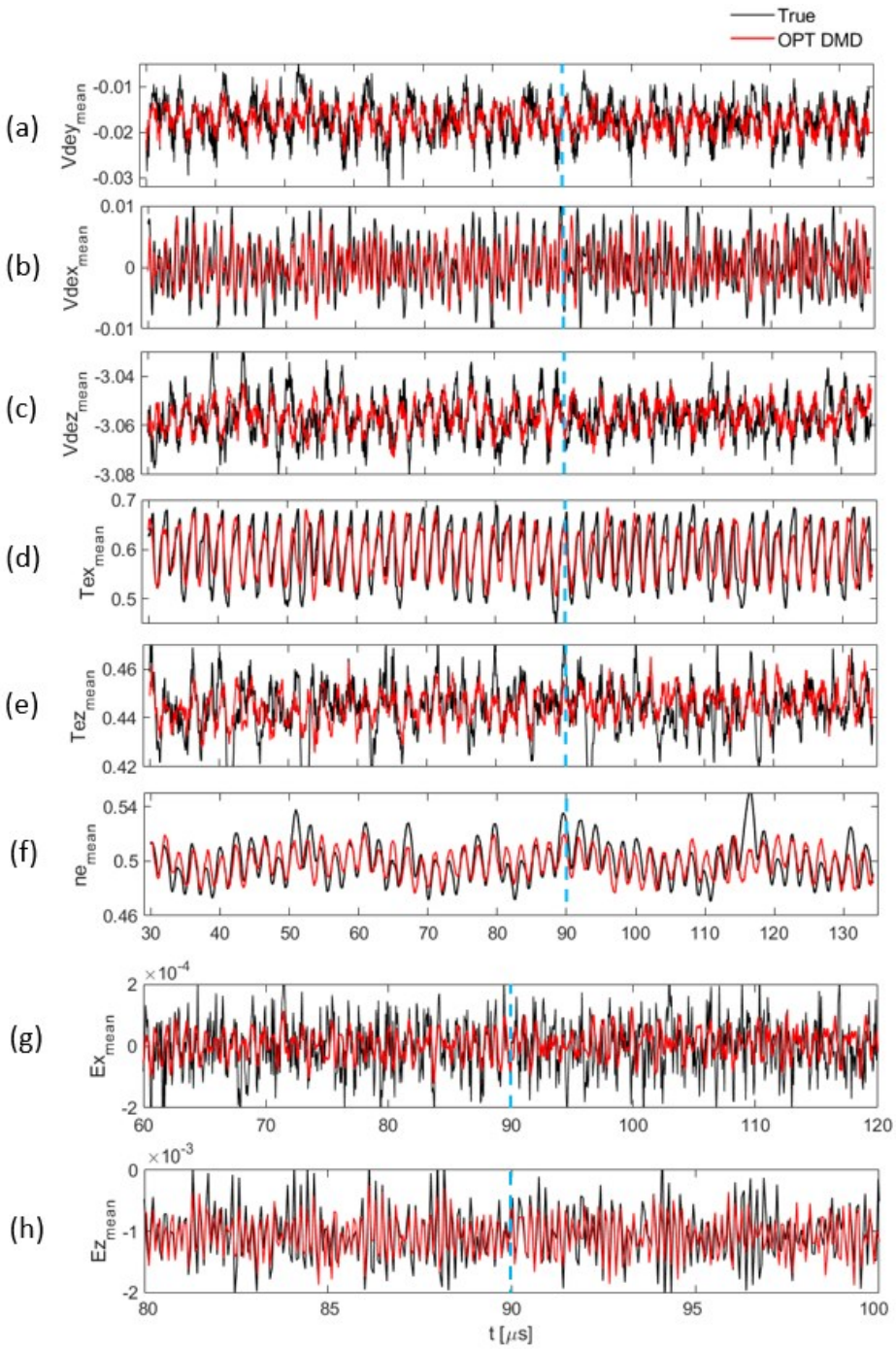


Fig. 4.19 Time evolutions of several spatially averaged (over the entire domain) plasma properties; (a) electrons' axial drift velocity ($V_{de,y}$), (b) electrons' radial drift velocity ($V_{de,x}$), (c) electrons' azimuthal drift velocity ($V_{de,z}$), (d) radial electron temperature (T_{ex}), (e) azimuthal electron temperature (T_{ez}), (f) electron number density (n_e), (g) radial electric field (E_x), (h) azimuthal electric field (E_z). Black traces represent the ground-truth signals from the radial-azimuthal PIC simulation, whereas the red traces are the reconstructed/predicted signals from the OPT-DMD model with the truncation rank (r) of 150. The vertical dashed blue line indicates the end of the training window, which corresponds to t = 90 μs. Plots (g) and (h) are zoomed-in views on the E_x and E_z signals within the time intervals of 60-120 μs and 80-100 μs, respectively, to aid better distinguishing the ground-truth and the predicted evolutions.

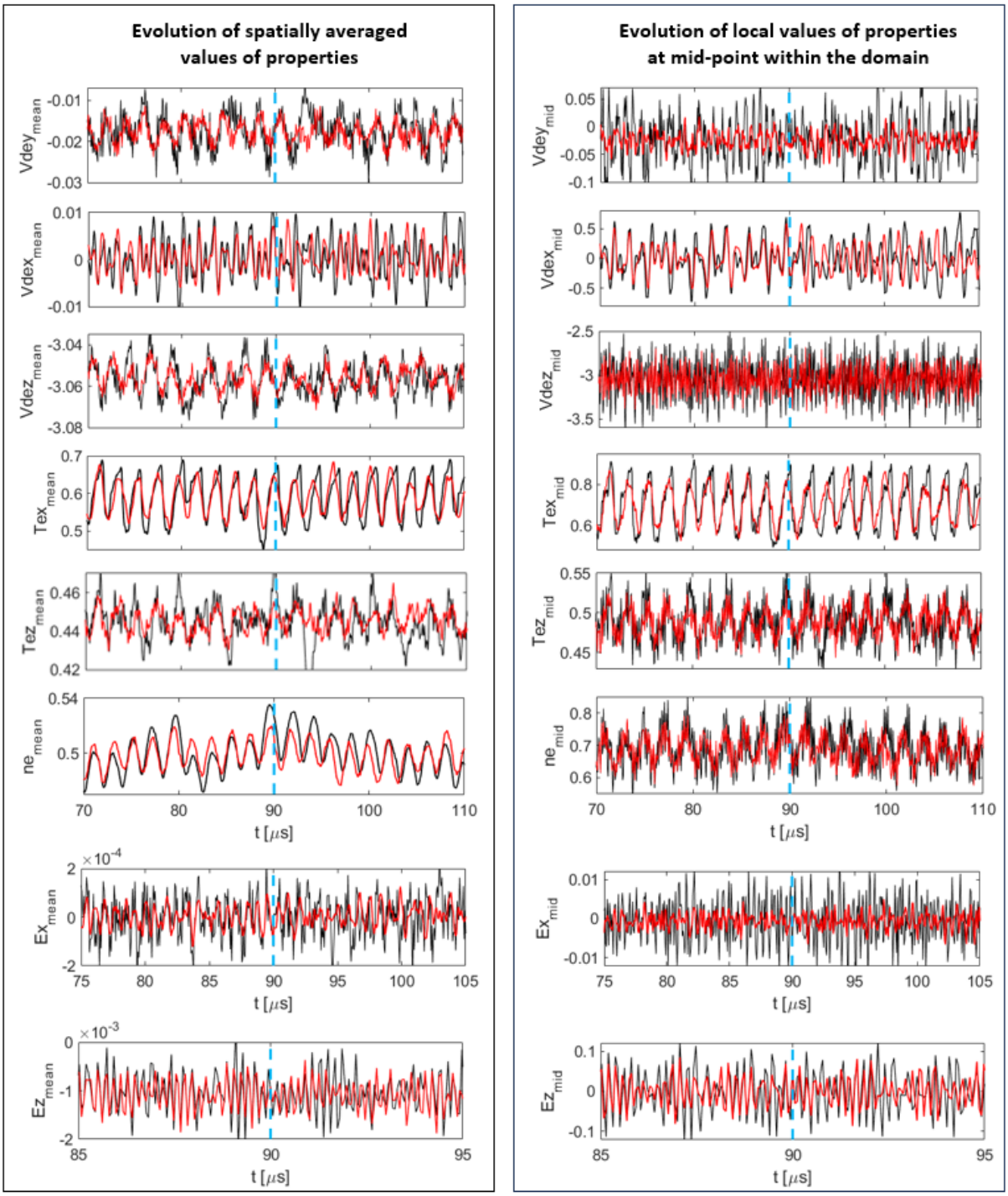


Fig. 4.20 Time evolutions of several plasma properties; (**left column**) zoomed-in views on the temporal variation plots of the spatially averaged plasma properties from Fig. 4.19. The zoomed-in plots are over the time interval of 70-110 μs (20 μs before and after the training end time) for all the plasma properties, except for the E_x and the E_z. For these two properties, the zoomed-in timeframes are, respectively, 75-105 μs, and 85-95 μs. This is done to aid better distinguishing the ground-truth and the predicted evolutions. (**right column**) time variations of the local values of the plasma properties included within the multi-property plasma state vector. The signals' duration for all the properties are consistent with those of the plots on the left column.

To provide a quantitative comparison between the OPT-DMD predictions and the ground-truth data from the PIC simulation, Fig. 4.21 shows the time variations of the instantaneous loss factor calculated

as per Eq. 4.1 between the predicted and the “true” snapshots corresponding to the signals that were compared in Fig. 4.19. The most notable observation is again the stability and the boundedness of the loss over the entire duration of the signals. Also, for all the plasma properties, the loss factor shows a very gradual increase from the training into the test window.

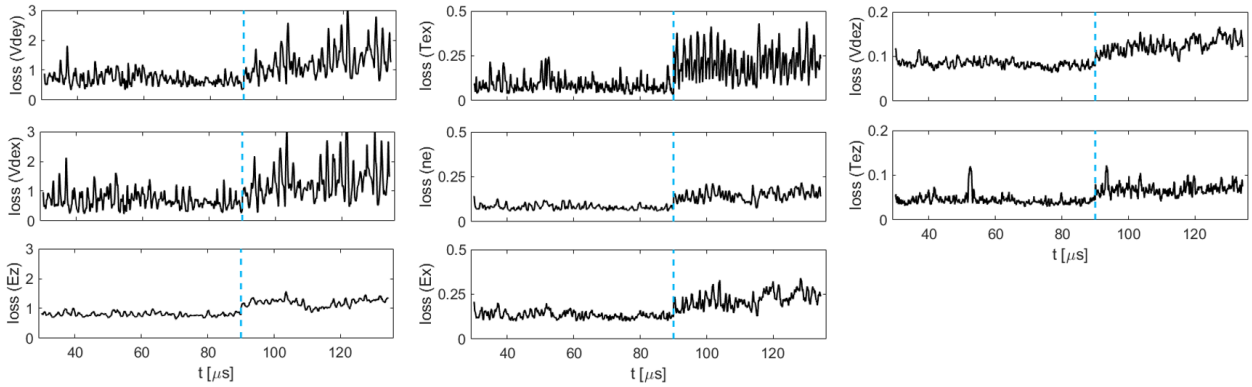


Fig. 4.21 Time evolutions of the loss factor (from Eq. 4.1) between the simultaneously predicted plasma properties from the OPT-DMD model and the ground-truth data from the PIC simulation. The range of the y-axis values is the same for the plots under the same columns. Dashed blue lines indicate the training end time.

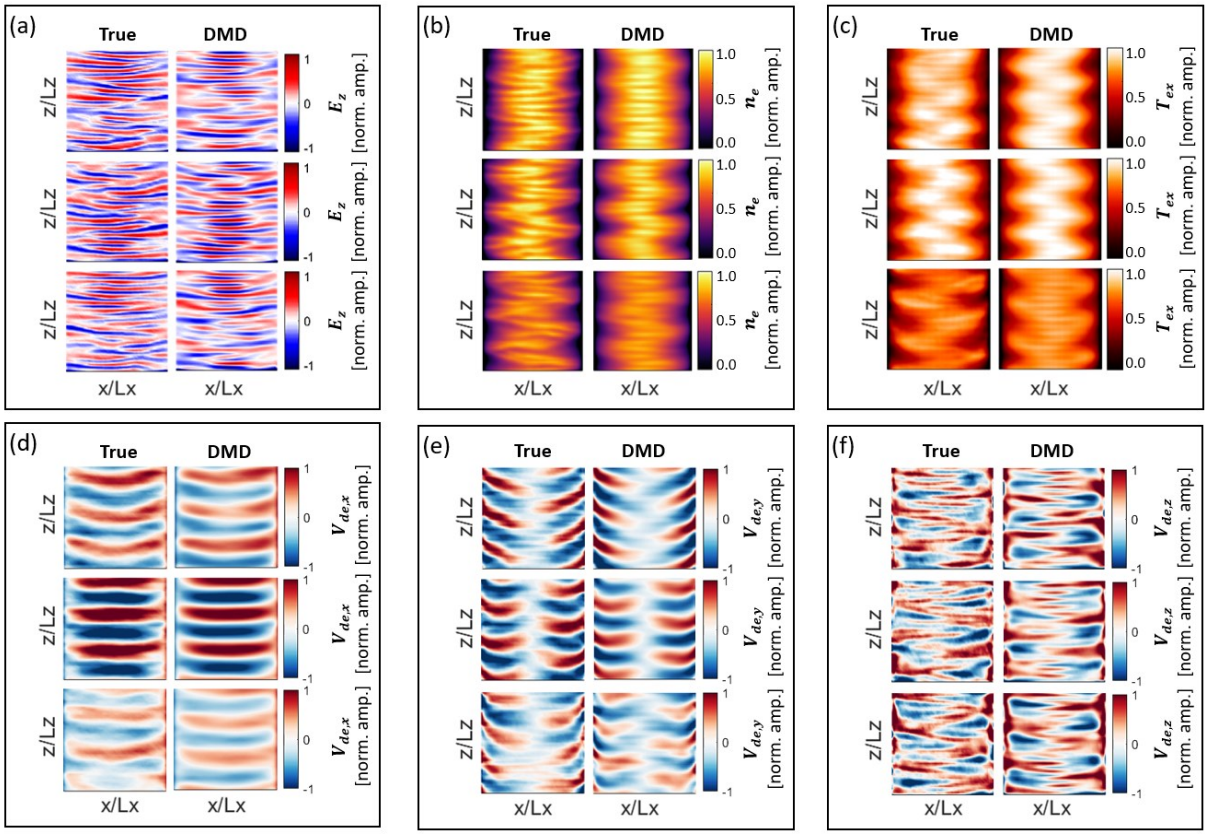


Fig. 4.22 Comparison between the 2D snapshots of several plasma properties from the ground-truth radial-azimuthal PIC simulation and the OPT-DMD model at three randomly selected time instants within the test window (90-130 μs); (a) azimuthal electric field (E_z), (b) electron number density (n_e), (c) radial electron temperature (T_{ex}), (d) electrons’ radial drift velocity ($V_{de,x}$), (e) electrons’ axial drift velocity ($V_{de,y}$), (f) electrons’ azimuthal drift velocity ($V_{de,z}$).

To additionally have a visual qualitative assessment of the performance of the OPT-DMD ROM for simultaneous forecasting of a multi-property plasma state vector, we refer to Fig. 4.22. This figure compares the predicted 2D snapshots of several plasma properties from the OPT-DMD ROM against the ground-truth snapshots at three randomly selected time instants within the test window (beyond the 90-μs mark in Fig. 4.19). Across all the time instants and plasma properties, we observe a remarkable degree of similarity between the OPT-DMD-predicted and the ground-truth snapshots.

4.3. Sensitivity of the OPT-DMD performance to added Gaussian noise

In this section, we evaluate the sensitivity of OPT-DMD to noise in the data. This has been assessed by adding artificial Gaussian noise with a wide range of standard deviations (σ = 0.01 to 0.5) to the normalized axial electron current density (J_{ey}) data from the radial-azimuthal test case with E_y = 0.5E_0 (test case 2 in Table 4.4). This particular test case is selected just as an example. We would nonetheless remind from subsection 4.1.3 that the OPT-DMD model demonstrated an excellent predictive performance for this test case in the absence of any artificially added noise.

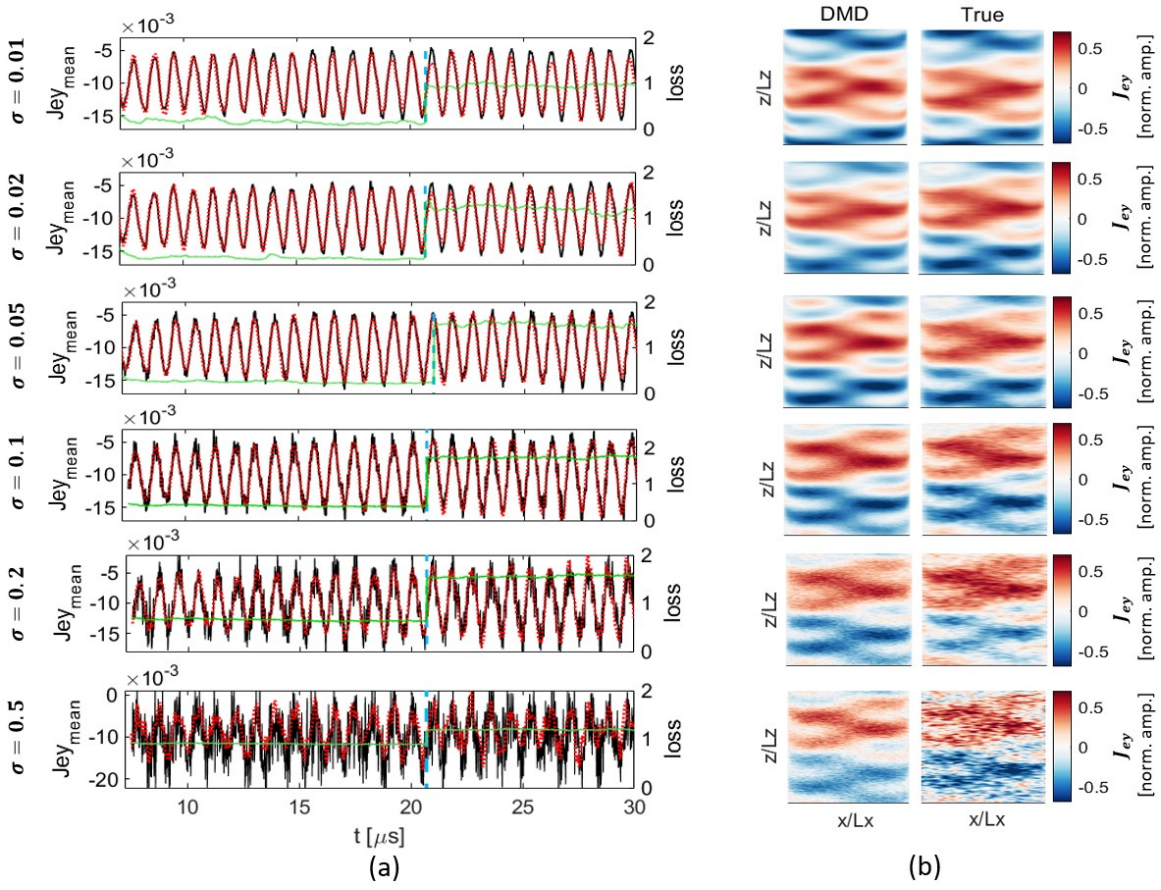


Fig. 4.23 Comparison between the OPT-DMD's prediction and the ground truth data of the normalized axial electron current density (J_{ey}) with different levels of added Gaussian noise from the radial-azimuthal simulation case with E_y = 0.5E_0 in terms of (a) the time evolution of the spatially averaged signal, and (b) reconstruction of a sample 2D snapshot. In the plots of column (a), the ground-truth traces from the PIC simulation are shown with solid black lines and the OPT-DMD-predicted traces with dotted red lines. The green traces represent the time variations of the loss factor (Eq. 4.1). Dashed blue line indicates the training end time.

The results of the robustness analysis on OPT-DMD are illustrated in Fig. 4.23. In this figure, the time variations of the spatially averaged J_{ey} data with various degrees of noise as predicted by the OPT-DMD model are compared against the corresponding ground-truth. Moreover, the reconstructions of a sample 2D snapshot of the J_{ey} from the OPT-DMD model in the presence of various noise levels are compared against the respective "true" 2D snapshot.

Considering the various comparisons shown in Fig. 4.23, OPT-DMD is observed to exhibit great robustness to noise. It is indeed able to successfully identify the dominant dynamics and to produce a stable model with good predictive performance even for highly noisy data.

4.4. Chapter conclusions

This Chapter focused on assessing the applicability of OPT-DMD to provide reliable and predictive reduced-order models of plasma systems. We evaluated in various test cases the performance of the linear data-driven ROMs from OPT-DMD to predict the temporal evolution of the plasma state and the spatial distribution of the properties.

The adopted test cases corresponded to 2D configurations representative of an axial-azimuthal and a radial-azimuthal cross-section of a Hall thruster as well as a 2D Penning discharge configuration. The axial-azimuthal and the radial-azimuthal test cases each involved several sets of plasma conditions (setup parameters), whereas the Penning test case involved only one set of setup parameters.

We overall observed that the OPT-DMD ROMs have a promising predictive potential across all the test cases, with a remarkable ability to predict the ground-truth data in cases where the plasma system exhibits relatively quasi-periodic behavior. We additionally noted that the OPT-DMD models provide stable predictions over time, and that the loss factor between the ground-truth and the reconstructed/predicted data remains bounded throughout the training and the test windows for all the cases tested.

For the majority of the test cases, we examined the short-time predictive capacity of the OPT-DMD models. This was because short-duration dynamics forecasting is the most prudent use of a data-driven algorithm such as dynamic mode decomposition that provides linear dynamics models under the assumption that the underlying patterns (modes) in the training data remain invariant over time. This assumption can in fact be a stringent one for real-world plasma systems, such as Hall thrusters and the broader family of E×B discharges, for which nonlinear, highly transient processes and interactions often take place.

Nevertheless, for a radial-azimuthal test case, we investigated the behavior of the OPT-DMD model for long-term prediction of the dynamics. We noticed that models developed using three different DMD truncation ranks all present great stability and can acceptably reproduce the time evolution trends of the ground-truth data as well as the spatial structures in the 2D distributions of the plasma properties.

A remarkable improvement in the predictive performance of the OPT-DMD ROM was observed toward time-extended dynamics forecast, in particular, when OPT-DMD was applied to a multi-property state vector of the radial-azimuthal plasma system as an example. We highlighted that the enhanced

predictive capability roots in the fact that the data matrix in the case of a multi-property plasma state is augmented by the time-series information of several plasma properties, and this enables the OPT-DMD model to capture the interdependency between various physical properties and not be solely reliant on the spatiotemporal correlations in the data of a single property.

To elaborate a bit further, the capability of OPT-DMD to capture interdependencies among plasma observables when multiple observables are incorporated into the data matrix enables the method to analyze their joint dynamics. By including multiple observables, OPT-DMD extracts dynamic modes that reflect the relationships among various observables. Each observable in the data matrix indeed resembles a state variable, and by expanding this set of state variables, OPT-DMD captures the interdependencies in the time evolution of each variable. In this way, each observable's dynamics can reflect the influence of others, effectively embedding these interdependencies within its learned dynamics.

This is conceptually linked to the Koopman operator framework [170], where the high-dimensional space of observables can enable a linear representation of nonlinear interactions. In the context of OPT-DMD, including multiple observables in the data matrix enables the method to approximate this Koopman space, where the interactions among observables are captured as part of the identified dynamic modes. Thus, OPT-DMD, when applied to a multi-observable data matrix, can approximate the dynamics by taking into account the couplings between various observables.

Lastly, we demonstrated in this Chapter that OPT-DMD is greatly robust to noise in the data. The method was proved to be able to reliably identify the dominant modes underlying the data and provide stable ROMs even when the training data featured high levels of added Gaussian noise (large standard deviations).

Chapter 5 Data-driven local operator finding for reduced-order modelling of plasmas

In Chapter 3 and Chapter 4, we observed how the robust OPT-DMD algorithm enables the identification of dominant "global" modes (patterns) underlying the data of a plasma system and yields linear time-dynamics models that best describe the time evolution of the global modes. In this Chapter, the focus is turned to a data-driven method for "local" dynamics discovery. We introduce a novel algorithm that is developed through this thesis work and is termed the "Phi Method". Phi Method enables learning the discretized form of differential equations describing a dynamical system and, thus, allows straightforward time forecasting of the system's state. The performance and broad applicability of Phi Method toward deriving reliable and robust ROMs are demonstrated for three test cases – the Lorenz attractor problem, the flow-past-a-cylinder problem, and a 1D Hall-thruster-representative plasma problem. In the fluid and the plasma test cases, the performance of the ROMs from Phi Method is compared for reference against the performance of OPT-DMD-derived ROMs.

The concept of Phi Method and its underlying formulation are described in section 5.1. The method is then tested in section 5.2 on the canonical problems of the Lorenz attractor and the fluid flow past a cylinder. Section 5.3 assesses Phi Method's performance when applied to the data of the plasma test case, and section 5.4 evaluates the sensitivity of Phi Method to noise in the data in comparison against OPT-DMD. The noise robustness analyses of section 5.4 are carried out using both the fluid and the plasma test cases. Section 5.5 summarizes the main observations and findings from this Chapter.

5.1. Phi Method: a novel approach for the discovery of discretized PDEs

Phi Method is a novel data-driven dynamics discovery technique, which finds the discretized form of the PDEs governing a system's dynamics. In other words, the algorithm is designed to discover the involved dynamics and the optimum discretization stencil simultaneously. This, hence, yields an optimum local operator that acts as a time stepper, advancing the system's state vector through time.

The data-driven finding of the optimum discretization stencil that is embedded within Phi Method is inspired from the general "stencil-learning" research in fluid dynamics [125][126], where ML/DD approaches have been pursued to relax the stringent stability requirements on solving the differential

equations with nonlinear terms (representing turbulence, e.g.) that make the numerical solutions computationally burdensome (refer also to subsection 2.1.2.1).

The underlying formulation of Phi Method is presented in the following. For simplicity, and without losing generality, we describe the formulation for the case of a system with one-dimensional (1D) spatial variations.

Consider the relation in Eq. 5.1, which describes the dynamics (time evolution) of a generic system

$$\frac{d\boldsymbol{x}(t)}{dt} = \boldsymbol{\mathcal{F}}\big(\boldsymbol{x}(t)\big), \qquad \text{Eq. 5.1}$$

with $\boldsymbol{x}$ denoting the system's state vector. $\boldsymbol{x}$ can represent all the physical properties of interest from the system. We aim to discover the discretized form (in time and space) of Eq. 5.1, which can be written as follows

$$\boldsymbol{x}_i^{k+1} = \boldsymbol{F}(\boldsymbol{y}_i^k)\Phi. \qquad \text{Eq. 5.2}$$

In Eq. 5.2, Φ represents the linear discrete-form counterpart of the continuous-time operator $\boldsymbol{\mathcal{F}}$ (Eq. 5.1), which we would like to find, hence, the name of the algorithm, "Phi Method". The relationship in Eq. 5.2 is applicable to all the nodes, i, inside a computational domain ($i = 2, 3, \dots, N_g - 1$, with N_g being the total number of computational grid nodes), and all the time steps $k = 1, 2, \dots, m - 1$.

Referring to Eq. 5.3, $\boldsymbol{x}_i^k$ is the system's state vector comprised by P variables ($x_{p,i}^k$, $p = 1,2, \dots, P$) on the i-th node and at the time step k. $\boldsymbol{y}_i^k$ represents the set of state vectors on node i and on its neighboring nodes at the time step k.

$$\boldsymbol{x}_i^k = \begin{bmatrix} x_{1,i}^k \\ x_{2,i}^k \\ \vdots \\ x_{P,i}^k \end{bmatrix}^T, \qquad \boldsymbol{y}_i^k = [\boldsymbol{x}_{i-1}^k \quad \boldsymbol{x}_i^k \quad \boldsymbol{x}_{i+1}^k]. \qquad \text{Eq. 5.3}$$

Here, in the definition of $\boldsymbol{y}$, we have included only the immediate adjacent nodes to node i along a single dimension, i.e., nodes $i - 1$ and $i + 1$. However, the definition can be readily generalized to contain arbitrary number of surrounding nodes along all the dimensions possibly involved in the system's dynamics.

The inclusion of farther neighboring nodes (e.g., nodes $i - 2$, $i + 2$, etc) allows capturing higher-order spatial derivatives (if present) as well as to obtain higher accuracy discretization of the lower-order derivatives. The schematic of the spatiotemporal computational stencil corresponding to Phi Method in a 1D configuration is illustrated in Fig. 5.1.

In Eq. 5.2, $\boldsymbol{F}$ represents a library of candidate terms that includes the linear and nonlinear "observable" functions of $\boldsymbol{y}$, and is defined as below

$$\boldsymbol{F}(\boldsymbol{y_i}) = \begin{bmatrix} \boldsymbol{f}_1(\boldsymbol{y_i}) \\ \boldsymbol{f}_2(\boldsymbol{y_i}) \\ \vdots \\ \boldsymbol{f}_L(\boldsymbol{y_i}) \end{bmatrix}^T, \qquad \boldsymbol{f}_l(\boldsymbol{y_i}) = \begin{bmatrix} f_l(\boldsymbol{x}_{i-1}) \\ f_l(\boldsymbol{x}_i) \\ f_l(\boldsymbol{x}_{i+1}) \end{bmatrix}^T;\ l = 1, 2, \dots L, \qquad \text{Eq. 5.4}$$

where L denotes the total number of distinct observable functions ($\boldsymbol{f}$) in the library.

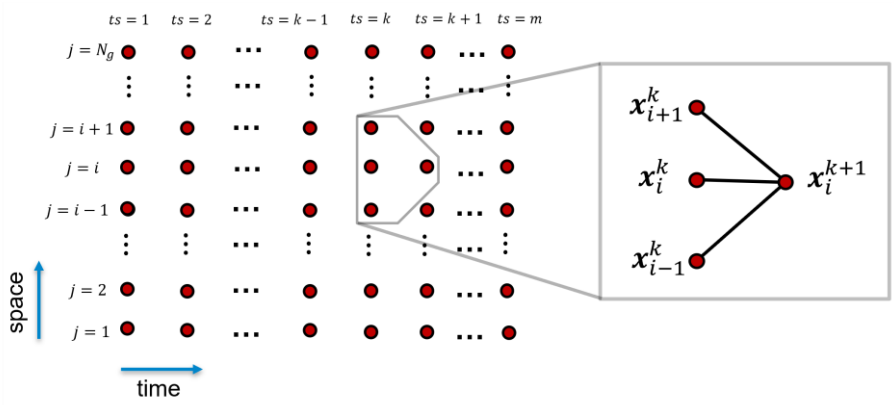


Fig. 5.1 Schematic of a 1D spatiotemporal stencil of computational nodes corresponding to Phi Method. j denotes the nodes' index, and t_s denotes the timestep.

To determine Φ from a system's data, i.e., in the training phase, we construct the data matrices X and Y (Eq. 5.5 and Eq. 5.6), which, respectively, contain the state vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ evaluated on different nodes and over several time steps

$$X = \left[\boldsymbol{x}_2^2, \boldsymbol{x}_3^2, \dots, \boldsymbol{x}_{N_g-1}^2, \boldsymbol{x}_2^3, \boldsymbol{x}_3^3, \dots, \boldsymbol{x}_{N_g-1}^3, \dots\dots\dots\dots\dots, \boldsymbol{x}_2^m, \boldsymbol{x}_3^m, \dots, \boldsymbol{x}_{N_g-1}^m\right]^T, \qquad \text{Eq. 5.5}$$

$$Y = \left[\boldsymbol{y}_2^1, \boldsymbol{y}_3^1, \dots, \boldsymbol{y}_{N_g-1}^1, \boldsymbol{y}_2^2, \boldsymbol{y}_3^2, \dots, \boldsymbol{y}_{N_g-1}^2, \dots\dots\dots\dots\dots, \boldsymbol{y}_2^{m-1}, \boldsymbol{y}_3^{m-1}, \dots, \boldsymbol{y}_{N_g-1}^{m-1}\right]^T. \qquad \text{Eq. 5.6}$$

The above data matrices establish the following system of equations

$$X = \Theta\Phi, \quad \Theta = \boldsymbol{F}(Y) \qquad \text{Eq. 5.7}$$

The coefficient matrix Φ is obtained by performing a least-squares regression on both sides of the system of equations given by Eq. 5.7, thus,

$$\Phi = \Theta^{\dagger} X \qquad \text{Eq. 5.8}$$

where $\Theta^{\dagger}$ denotes the Moore–Penrose pseudo-inverse of matrix Θ. The schematic representation of the training phase of Phi Method, as was just described, is illustrated in Fig. 5.2.

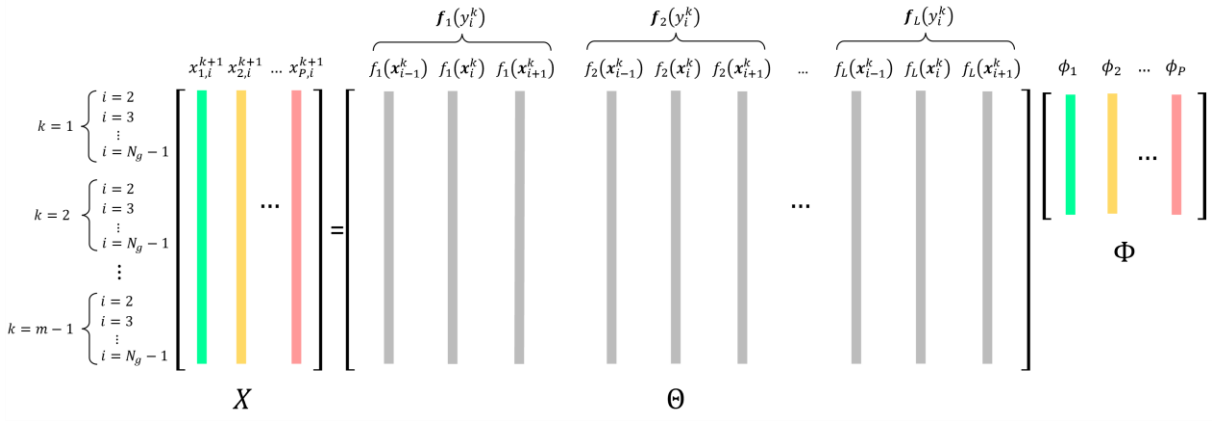


Fig. 5.2 Schematic of the *training phase* of the Phi Method algorithm; matrix X contains the state vectors at time step $k+1$, matrix Θ represents the library of candidate terms, and matrix Φ is the coefficient matrix.

Once Φ is obtained, Eq. 5.2 is used as a time stepper that advances the entire state vector forward in time from a specified initial condition ($\boldsymbol{x}_i^1,\ i = 1, 2, \dots, N_g$) and given the boundary conditions, $\boldsymbol{x}_{i_b}^k$, where i_b denotes the index of the nodes on the boundaries, e.g. $i_b = 1,\ N_g$ in a 1D domain.

We highlight that, in its current implementation, Phi Method does not incorporate a sparsity promoting regularization as is included in some other dynamics discovery algorithms like SINDy (refer to subsection 2.3.1). However, a sparse regression can be used in a similar manner to that in SINDy, for example. In this way, the Phi Method algorithm would select as few terms as required from the library matrix Θ by zeroing out the coefficients of the unnecessary library terms. Sparse regression can further prevent overfitting the data, and the resulting "parsimonious" model will be more generalizable and interpretable. This can be particularly important for the cases where an a-priori knowledge of the functional forms of the nonlinear observables of the system does not exist. In such cases, the library of the observables must be large enough so as to ensure accounting for the involved nonlinearities in the system. As a result, it is necessary to find a *sparse* set of terms that can best represent the dynamics out of the constructed large library.

According to the descriptions provided so far, the methodology behind Phi Method sits in between the SINDy and the DMD algorithms (sections 2.3 and 3.2, respectively).

Similar to SINDy, Phi Method discovers a nonlinear system of differential equations by regressing on a library of terms, which are evaluated using the local values of the system's state vector data. However, Phi Method features an advancement over SINDy in that, in Phi Method, we simultaneously find the system of equations and the optimal discretization stencil. As a result, unlike SINDy which derives the continuous form of the governing differential equations [158], Phi Method provides the discretized version of the system of equations. This implies that, in Phi Method, we no longer need to pre-evaluate the spatial differential terms in the library (on a finite difference stencil, for instance) as is done in SINDy [158]. Instead, we evaluate the candidate terms on each grid node and on its neighboring nodes separately and store these values as *individual* terms in the library (Θ).

A similar approach is followed to treat the temporal differentiations. In SINDy, the time derivatives of the state variables appear on the left-hand side of the equations for which the right-hand-side terms are to be found [158]. Phi Method, however, directly relates the state variables at a subsequent time step, $t_s = k + 1$ (which represent the left-hand-side terms of the equations), to the values of the state variables at the present time step, $t_s = k$.

From the above perspective of learning the discrete time-dynamics, Phi Method shares some similarities with DMD, particularly in the case that the DMD's state vector (X) would represent a broader set of observables that also include nonlinear terms (see subsection 3.2.1.1). Like DMD, Phi Method finds the best-fit linear operator, which maps the state vector of the present time step to the next. The key distinction between the methods lies in the fact that the Phi Method's operator (Φ) is a local operator that links the state vector at a certain spatial location to that of the immediate surrounding, i.e., the values of the state vector on the neighboring nodes. Whereas, the DMD's operator (A) is, by definition,

global and, hence, it establishes a connection between the value of the state vector on each node to the corresponding state vector values on the entire computational domain (an entire snapshot).

5.2. Assessment of Phi Method in canonical test problems

In this section, we evaluate the predictive performance and the characteristics of the ROMs from Phi Method in two test cases, namely, the Lorenz attractor problem, and the problem of a fluid flow past a 2D cylinder.

In the canonical Lorenz attractor test case, our aim is to show the Phi Method's capability to recover dynamics from trajectory data that are governed by a system of nonlinear ordinary differential equations (ODEs). The fluid flow problem is another canonical test case that has been used as a benchmark in the literature for the verifications of different ML/DD algorithms (e.g., in Refs. [158][166][179][181]). The dynamics in the fluid flow problem is governed by a system of PDEs. For this problem, the goal is to assess the performance of a Phi-Method-derived ROM toward forecasting the fluid flow field in comparison against an OPT-DMD-derived ROM.

For each test case, we first provide an overview of the problem description and the corresponding setup that we had used to perform the ground-truth simulations. Next, we present and discuss the predictions and the behavior of the ROMs from Phi Method. For the fluid flow test case, the robustness of the Phi-Method-derived ROM to noise in the data is assessed in section 5.4.

5.2.1. Lorenz attractor test case

5.2.1.1. Description of the problem setup

The Lorenz attractor is a well-known dynamical system introduced by the meteorologist Edward Lorenz in 1963 as a simplified model to study atmospheric convection [233]. The Lorenz system exhibits chaotic behavior, which is characterized by strong sensitivity to the initial conditions and the formation of a butterfly-shaped attractor. The Lorenz attractor describes the evolution of a system in three-dimensional space using a set of three coupled nonlinear ODEs (Eq. 5.9).

$$\begin{aligned} \frac{dx}{dt} &= \sigma y - \sigma x, \\ \frac{dy}{dt} &= \rho x - xz - y, \\ \frac{dz}{dt} &= xy - \beta z. \end{aligned} \qquad \text{Eq. 5.9}$$

In Eq. 5.9, x, y, and z are the state variables representing the system's evolution over time (t). σ, ρ, and β are positive constants that characterize the behavior of the system.

Due to its simple and well-represented dynamics description, the Lorenz attractor serves as a common benchmark case for the data-driven techniques. This is why, before proceeding to higher dimensional test cases, we begin by the Lorentz attractor problem and test the Phi Method's ability in recovering the dynamics from the simulation data of this problem.

To generate the data, we have numerically integrated the Lorenz system of ODEs with the time step (dt) of 1 ms, and with the following values of the constant parameters: $\sigma = 10$, $\rho = 28$, and $\beta = 8/3$.

The Phi Method's library of candidate terms (Θ) for this test case includes the linear state variables and their quadratic functions. Hence, Eq. 5.2 in this example becomes

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}_{k+1}^{T} = [x \;\; y \;\; z \;\; x^2 \;\; y^2 \;\; z^2 \;\; xy \;\; yz \;\; xz\,]_k \Phi. \qquad \text{Eq. 5.10}$$

It is noted that, since the state of the system for the Lorenz test case does not have a spatial distribution, i.e., there are no spatial derivatives and, hence, the dynamics is represented by a system of ODEs, $\boldsymbol{x}$ and $\boldsymbol{y}$ in Eq. 5.2 are equivalent.

To train the Phi Method model, the left-hand-side vector and the library matrix in Eq. 5.10 were evaluated using the first half of the data points collected from our numerical solution of the Lorenz problem. Following a regression on the resulting "ground-truth" training dataset, the coefficient matrix $\Phi \in \mathrm{R}^{9\times3}$ is determined.

5.2.1.2. Results

Table 5.1 shows the values of the entries of the obtained Φ matrix. These entries correspond to the "non-zero" coefficients in Eq. 5.10 after the training of Phi Method. The Φ matrix entries not shown in Table 5.1 all had values below 10^{-5}, which effectively amount to zero because they are about two orders of magnitude smaller than the values of the smallest coefficients shown.

	Phi Method	**True**	**Error [%]**
$\Phi_{1,1}$	9.9018×10^{-1}	9.9×10^{-1}	0.018
$\Phi_{2,1}$	9.9455×10^{-3}	1×10^{-2}	0.545
$\Phi_{1,2}$	2.7955×10^{-2}	2.8×10^{-2}	0.161
$\Phi_{2,2}$	9.9910×10^{-1}	9.99×10^{-1}	0.010
$\Phi_{9,2}$	-9.9604×10^{-4}	-1×10^{-3}	0.396
$\Phi_{3,3}$	9.9735×10^{-1}	9.973×10^{-1}	0.005
$\Phi_{7,3}$	9.9318×10^{-4}	1×10^{-3}	0.682

Table 5.1 The values of the non-zero entries of the Φ matrix from Phi Method compared against the respective true values from Eq. 5.11. The corresponding errors are also provided. The error represents the difference between the learned and the true values of each coefficient normalized with respect to the true value.

In order to compare the learned coefficients' values against the analytical values, we discretized the time derivatives in Eq. 5.9 using first-order finite difference scheme. This leads to the discretized system of equations in Eq. 5.11.

$$x^k = (1 - \sigma dt)\, x^{k-1} + \sigma dt y^{k-1},$$
$$y^k = \rho dt\, x^{k-1} + (1 - dt) y^{k-1} - dt\, (xz)^{k-1}, \quad \text{Eq. 5.11}$$
$$z^k = dt(xy)^{k-1} + (1 - \beta dt) z^{k-1}.$$

Substituting the values of dt and the system's parameters (σ, ρ, and β), the analytical values of the coefficients are obtained and are presented in Table 5.1 as well. By comparing the Phi-Method-derived coefficients against their true values, we verify that Phi Method has effectively learned the dynamics, as evidenced by the fact that the errors in all the coefficients are below 1%.

In Fig. 5.3, the Phi Method's prediction of the trajectory of the Lorentz attractor is compared against the true trajectory as obtained from the numerical integration of Eq. 5.9. This comparison further illustrates the consistency between the predicted dynamics by the Phi Method ROM and the ground-truth dynamics.

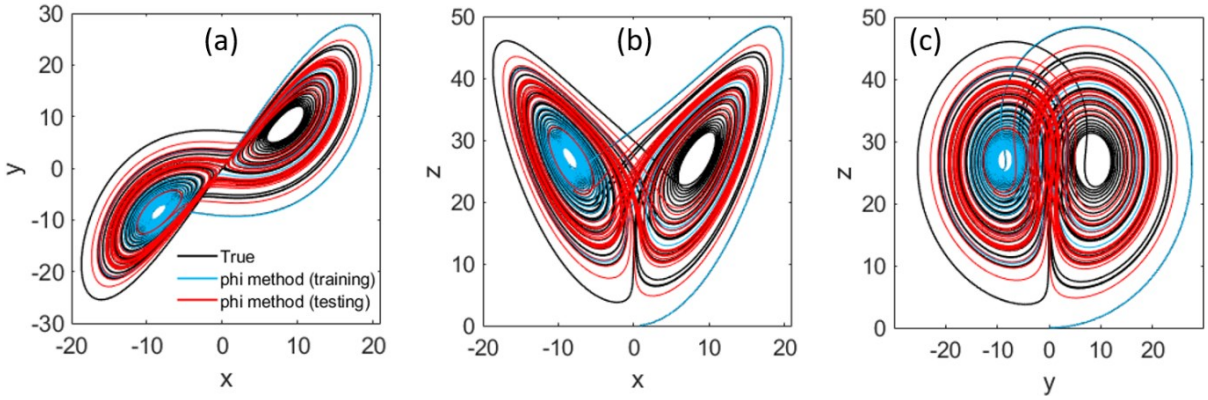


Fig. 5.3 Comparison between the learned trajectory of a Lorenz attractor using Phi Method and the ground-truth trajectory obtained from the solution of the system of equations in Eq. 5.9 along (a) $y - x$, (b) $z - x$, and (c) $z - y$ planes.

5.2.2. Fluid flow past a cylinder test case

5.2.2.1. Description of the problem setup

We now extend the demonstration of Phi Method to the discovery of spatiotemporal dependencies in a dynamical system that is governed by a PDE. Accordingly, the present test case involves a 2D unsteady fluid flow around a cylinder in the vortex-shedding regime. This example represents a classic fluid dynamics phenomenon, which is characterized by the formation of alternating vortices in the wake of a cylindrical object placed in front of a fluid stream. In this case, as the fluid flows around the cylinder, vortices are periodically shed from either side of the cylinder. This leads to the formation of a distinctive oscillatory pattern in the vorticity field (known as von Kármán vortex street), which is comprised by vortices of opposite signs.

Under the assumption of an incompressible and barotropic flow, the spatiotemporal evolution of the vorticity field (Ω) is given by

$$\frac{\partial \Omega}{\partial t} + (\boldsymbol{V}.\boldsymbol{\nabla})\Omega = \frac{1}{Re}\nabla^2\Omega, \quad \text{Eq. 5.12}$$

where, Re represents the Reynolds number of the free stream, and $\boldsymbol{V}$ is the local flow velocity vector. Re is a dimensionless quantity defined at UL/ν, with U denoting the characteristic velocity, L the characteristic length, and ν the kinematic viscosity. Reynolds number is usually used to determine the regime of the flow – laminar, turbulent or transitional.

For the ground-truth data generation, the open-access "ViscousFlow" module [234] was used. ViscousFlow is a software package written in Julia language and enables simulating viscous incompressible flows around arbitrary body shapes. The 2D Cartesian domain of the simulation comprises a rectangular box in the $x-y$ plane with the dimensions $L_x = 6\ cm$ and $L_y = 4\ cm$. x and y represent the involved simulation directions along and perpendicular to the free stream velocity, respectively. A cylindrical object with a diameter of 0.01 m is placed into the domain with its center being 0.01 m from the left boundary. The left boundary serves as the flow inlet, where the Reynolds number of the freestream is 300. A time step of 1 ms is adopted, and the number of nodes along the x and y coordinates of the simulation domain are N_i = 300 and N_j = 200, respectively. The total simulated duration is 100 s. We output the flow properties from the simulation every 0.1 s, which resulted in a total of 1000 snapshots. The boundary conditions include a no-slip condition on the cylinder surface and the far-field conditions at the outer boundaries.

We used the collected data from the "ground-truth" simulation to train and test Phi Method as well as OPT-DMD for the spatiotemporal evolution of the vorticity field. We discarded the initial 45 s of the simulation, which represented the transient of the system. This left us with the next 55 s duration of the simulation for the training and testing of data-driven ROMs. This 55-second duration is shown as the time interval of 0 - 55 s in the results plots of subsection 5.2.2.2. The first 25 s within the 55-second time window was used for the training of the models, whereas the rest was reserved to validate the forecasts obtained from the developed ROMs.

To apply OPT-DMD to the data snapshots of the vorticity field, the number of retained DMD ranks (section 3.2) in the model was 14. The OPT-DMD ROM serves as a comparison reference for the Phi-Method-derived ROM.

For the application of Phi Method in this test case, we considered two "scenarios" in terms of the candidate library terms:

A. The **first scenario** involves a "linear" model, where the terms in the library are the system's state vector only. Thus, the candidate library terms include the components of the flow velocity field along the x and the y directions, denoted by $V_x(x,y)$ and $V_y(x,y)$, respectively, as well as the vorticity field $\Omega(x,y)$. In the case of the linear model, $\Phi_{\mathrm{L}} \in R^{15\times 1}$, and Eq. 5.2 becomes

$$\Omega_{i,j}\big|_{k+1} = \begin{bmatrix} \begin{bmatrix} \Omega_{i,j-1} \\ \Omega_{i-1,j} \\ \Omega_{i,j} \\ \Omega_{i+1,j} \\ \Omega_{i,j+1} \end{bmatrix}^T & \begin{bmatrix} V_{x_{i,j-1}} \\ V_{x_{i-1,j}} \\ V_{x_{i,j}} \\ V_{x_{i+1,j}} \\ V_{x_{i,j+1}} \end{bmatrix}^T & \begin{bmatrix} V_{y_{i,j-1}} \\ V_{y_{i-1,j}} \\ V_{y_{i,j}} \\ V_{y_{i+1,j}} \\ V_{y_{i,j+1}} \end{bmatrix}^T \end{bmatrix}_k \Phi_{\mathrm{L}}. \qquad \text{Eq. 5.13}$$

In Eq. 5.13, each of the inner vectors of the matrix on the right-hand side characterizes a state variable on the node (i, j) and on its neighboring nodes. These vectors were represented by $\boldsymbol{y}_i$ in Eq. 5.2. Note that, for a linear model, $\boldsymbol{f}_l(\boldsymbol{y}_i) = \boldsymbol{y}_i$, and $l = 1, 2, \dots, P$, where P is the total number of state variables.

B. The **second scenario** involves a "nonlinear" model, which, in addition to the linear terms – $V_x(x, y)$, $V_y(x, y)$, and $\Omega(x, y)$ – includes some quadratic nonlinear functions of the state vector as well, namely, $\Omega \mathrm{V_x}(x, y)$ and $\Omega \mathrm{V_y}(x, y)$. Choosing these specific nonlinear terms have been in accordance with the terms that appear in the dynamics PDE in Eq. 5.12. The impact of considering a broader set of terms in the library that includes all the quadratic functions of the state variables will be discussed in the following subsection. Eq. 5.2 for the nonlinear Phi Method model takes the following form

$$\Omega_{i,j}\big|_{k+1} = \begin{bmatrix} \begin{bmatrix} \Omega_{i,j-1} \\ \Omega_{i-1,j} \\ \Omega_{i,j} \\ \Omega_{i+1,j} \\ \Omega_{i,j+1} \end{bmatrix}^T & \begin{bmatrix} V_{x\,i,j-1} \\ V_{x\,i-1,j} \\ V_{x\,i,j} \\ V_{x\,i+1,j} \\ V_{x\,i,j+1} \end{bmatrix}^T & \begin{bmatrix} V_{y\,i,j-1} \\ V_{y\,i-1,j} \\ V_{y\,i,j} \\ V_{y\,i+1,j} \\ V_{y\,i,j+1} \end{bmatrix}^T & \begin{bmatrix} \Omega V_{x\,i,j-1} \\ \Omega V_{x\,i-1,j} \\ \Omega V_{x\,i,j} \\ \Omega V_{x\,i+1,j} \\ \Omega V_{x\,i,j+1} \end{bmatrix}^T & \begin{bmatrix} \Omega V_{y\,i,j-1} \\ \Omega V_{y\,i-1,j} \\ \Omega V_{y\,i,j} \\ \Omega V_{y\,i+1,j} \\ \Omega V_{y\,i,j+1} \end{bmatrix}^T \end{bmatrix}_k \Phi_{\mathrm{NL}}, \qquad \text{Eq. 5.14}$$

where $\Phi_{\mathrm{NL}} \in R^{25\times 1}$.

In order to avoid possible numerical issues due to the discrepancy in the order of magnitudes of the different state variables (flow properties), each state variable was normalized with respect to its maximum value over the entire spatiotemporal data. The Φ operators for the "linear" and the "nonlinear" models, i.e., Φ_{L} and Φ_{NL}, respectively, were obtained by performing regression on the data matrices in Eq. 5.13 and Eq. 5.14. The data matrices were constructed by evaluating the left-hand sides of the Eq. 5.13 and Eq. 5.14 along with their respective library terms over a sufficiently large number of data points during the training interval.

5.2.2.2. Results

The obtained linear and nonlinear ROMs from Phi Method (as represented by the operators Φ_{L} and Φ_{NL}, respectively) were used to reconstruct the vorticity field data within the training window, starting from a given initial condition, and to forecast the vorticity field beyond the training dataset. The results in terms of the time evolution of the spatially averaged vorticity (Ω_{mean}) and the local value of the vorticity field (Ω_{mid}) at the center of the domain (x = 0.03 m and y = 0.02 m) are presented in Fig. 5.4. The corresponding predicted traces from the OPT-DMD ROM are superimposed on the plots in Fig. 5.4 for comparison.

The instantaneous loss ($\mathcal{L}$) of the predictions from the Phi Method ROMs and the OPT-DMD ROM with respect to the original ground-truth data are calculated according to Eq. 4.1, shown below again for convenience. The time evolution of the loss factor corresponding to each DD model is also plotted in Fig. 5.4.

$$\mathcal{L} = \frac{\|\mathcal{S}^{true} - \mathcal{S}^{ROM}\|_F}{\|\mathcal{S}^{true}\|_F}, \qquad \text{Eq. 4.1 (repeated)}$$

It is recalled that, in Eq. 4.1, $\|\mathcal{S}\|_F$ denotes the Frobenius norm (Eq. 3.25) of the data matrix that represents a 2D snapshot from either the ground-truth simulation ($\|\mathcal{S}^{true}\|_F$) or the data-driven ROM ($\|\mathcal{S}^{ROM}\|_F$).

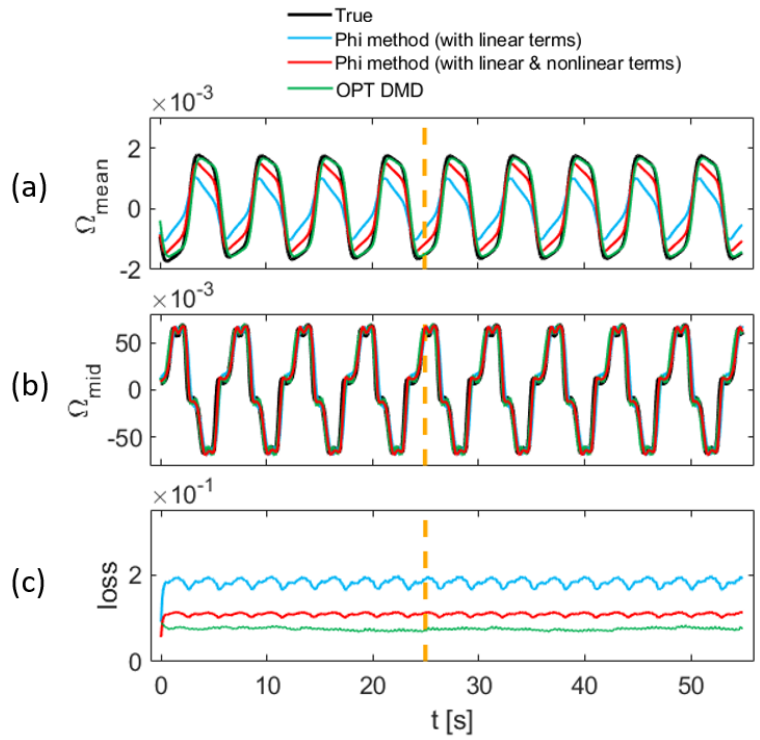


Fig. 5.4 Comparison of the linear and nonlinear ROMs from Phi Method and the ROM from OPT-DMD against the ground-truth simulation for the fluid flow test case; time evolutions of (a) spatially averaged normalized vorticity (Ω_{mean}), (b) local value of the normalized vorticity at the mid-location within the simulation domain (Ω_{mid}), and (c) the loss factor over the entire domain. Dashed yellow lines indicate the end of the training interval.

From Fig. 5.4, we see that even the Phi Method's linear model provides a notably good approximation of the original data, which is particularly evident when looking at the time evolution plot of the local value of the vorticity. This point is further evidenced by referring to Fig. 5.5, in which we have compared the predicted 2D snapshots of the vorticity field from the Phi Method linear ROM against the true snapshots.

In terms of the spatially averaged vorticity (Ω_{mean}), however, Fig. 5.4(a) shows that the inclusion of the nonlinear terms leads to improvement in the accuracy of the Phi Method model. The time evolution of the loss factor in Fig. 5.4(c) shows quantitatively the degree of improvement achieved by the Phi Method nonlinear ROM. In particular, the loss factor for the nonlinear model is almost half of that for the linear model.

Another interesting observation from Fig. 5.4 is that the Phi Method linear model exhibits the largest loss when the absolute value of the average vorticity reaches the maximum in each cycle. Contrarily, the linear model's loss is minimum when the average vorticity is close to zero. A similar variation in the

loss factor is also noticed for the nonlinear ROM, albeit with a smaller amplitude. The loss associated with the OPT-DMD model, nevertheless, remains nearly constant throughout the entire duration of the signal (from the training into the test window). The mean loss of the Phi Method nonlinear model is about 1.4 times that of the OPT-DMD model. Finally, for all the models, the level of loss is seen to remain consistent between the training and the test intervals.

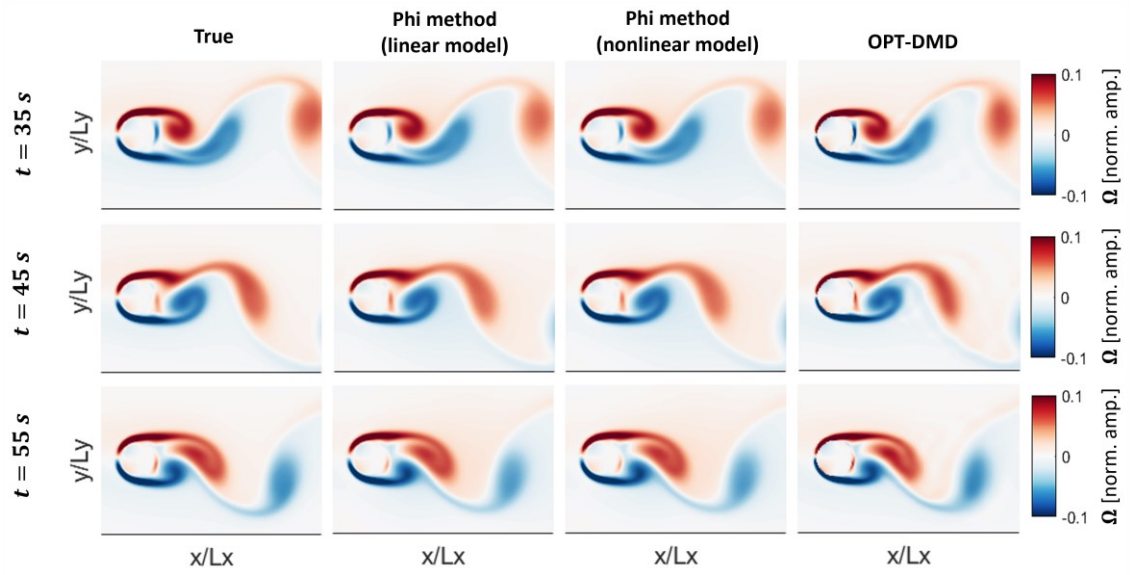


Fig. 5.5 Comparison of the linear and the nonlinear ROMs from Phi Method and the ROM from OPT-DMD against the ground-truth simulation for the fluid flow test case; snapshots of the normalized vorticity field at three different time instances within the test interval.

In Fig. 5.5, we have shown the predicted 2D snapshots of the normalized vorticity field from each derived ROM at three sample time instants beyond the training duration. The “true” 2D snapshots from the ground-truth simulation are also shown for reference. The overall patterns in the snapshots from all the models are almost indistinguishable from their true counterparts. To illustrate, however, the subtle discrepancies not readily apparent in the snapshots of Fig. 5.5, Fig. 5.6 presents the absolute-difference plots between the true snapshots and the corresponding predicted ones ($S_k^{true} - S_k^{\mathrm{ROM}}$) at the three sample time instants. From this figure, it is evident that the error of the Phi Method linear ROM is the highest among the models presented, whereas the errors associated with the Phi Method nonlinear model and the OPT-DMD ROM are within a comparable range.

Also, we notice from Fig. 5.6 that the patterns in the difference plots corresponding to the Phi Method linear and nonlinear ROMs are different from the difference-plot patterns for the OPT-DMD model. For OPT-DMD, the non-zero error is confined to the downstream wake region, whereas for Phi Method, the error extends across the entire flow field. This distinction arises from the inherent differences between the Phi Method and the OPT-DMD models. OPT-DMD yields a “global” model, and its error represents the sum of the discarded DMD modes (i.e., beyond rank 14, where we truncated the SVD/POD expansion). In this regard, although the upstream and the lateral regions of the flow are fully represented by the leading DMD modes that we retained, the wake exhibits higher dimensionality which requires a larger number of modes for a complete representation. The Phi Method models, on the contrary, are “local” operators that best fit the dynamics across the entire flow field because a single PDE is expected

to govern the dynamics everywhere in the flow. Hence, for the case of Phi Method, the error arises from a dynamical term not having been entirely captured by the model, thereby resulting in prediction errors throughout the entire domain.

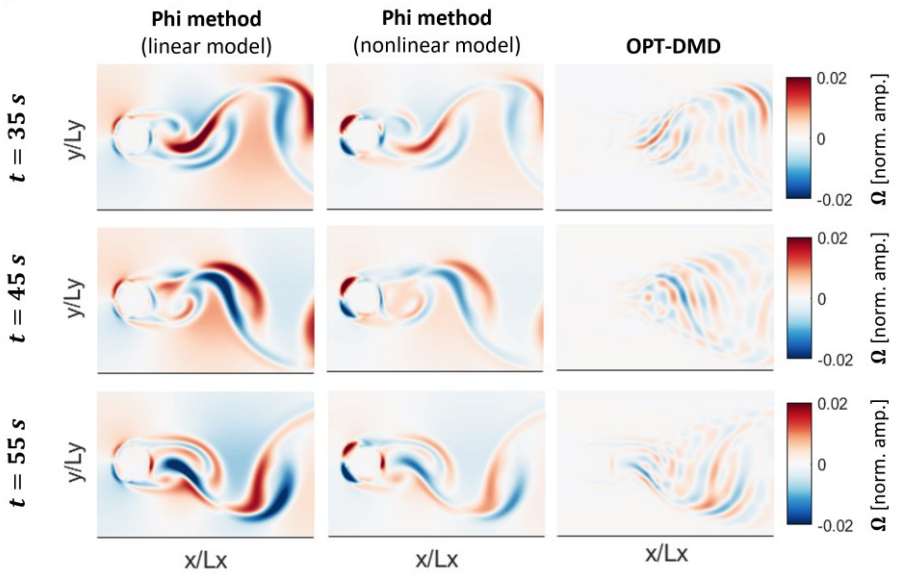


Fig. 5.6 Comparison of the linear and the nonlinear ROMs from Phi Method and the ROM from OPT-DMD against the ground-truth simulation for the fluid flow test case; plots of the absolute difference between the true and the predicted normalized vorticity snapshots at three different time instants during the test interval.

Moving on with the discussions, we have so far used the data from the quasi-steady oscillatory state of the system for the training of the data-driven models, excluding the initial transient behavior from the dataset. However, there might be scenarios in which the dynamics does not feature a steady state, or the system takes a long time to reach a quasi-steady state. In these scenarios, our only option would be to train a data-driven model using the transient data of the system. Accordingly, to evaluate Phi Method's capability to learn the dynamics from transient data, we trained both the linear and the nonlinear models on the initial 0-25 s duration of the simulation time during which the flow exhibits a highly transient behavior. This 25-second time window falls within the initial 45-second interval (see subsection 5.2.2.1) that we had discarded before for the results discussed through Fig. 5.4 to Fig. 5.6.

The outcomes of the above exercise are provided in Fig. 5.7 in terms of the time evolutions of the predicted vorticity field signals from both the linear and the nonlinear Phi Method models as well as the time evolution of the associated loss factors. It is observed that the models' predictions of the mean and the local vorticity values are reasonably consistent with the original (true) data. Furthermore, the mean loss of the models when trained on the transient-state data is only about 1.7 times that of the corresponding models when trained on the steady-state data. This verifies that Phi Method is indeed able to identify the optimal Φ operator from the transient data as well.

It is highlighted that, since the characterization of the transient phenomena is a well-known limitation of the DMD-based (POD-based) approaches (refer to section 3.6 for more details), OPT-DMD could only be reliably trained on the quasi-steady data and, hence, the predictions of the OPT-DMD model are not shown in the plots of Fig. 5.7.

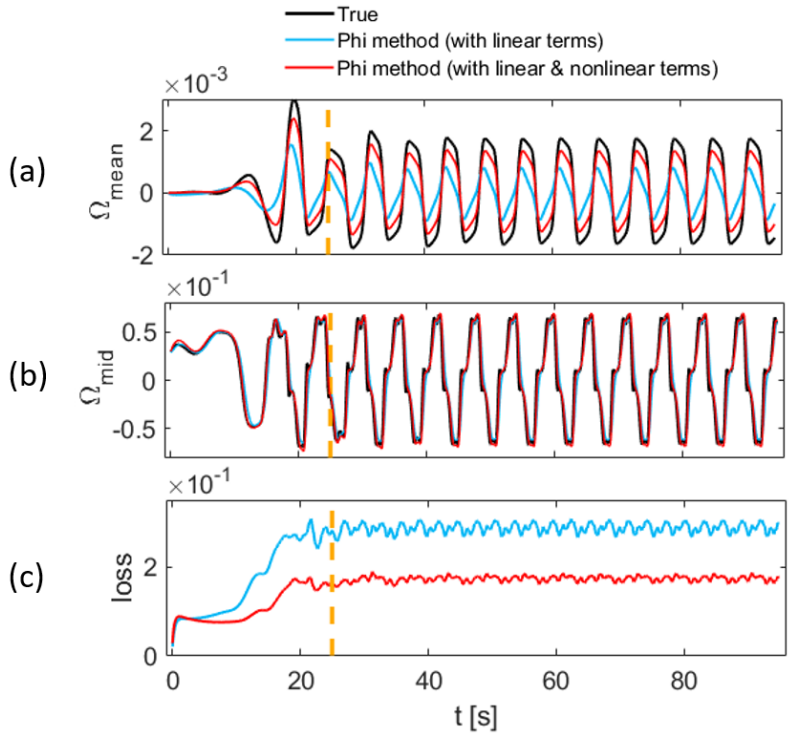


Fig. 5.7 Comparison of the linear and nonlinear Phi Method models trained on the transient-state data against the ground-truth simulation for the fluid flow test case; time evolutions of (a) spatially averaged normalized vorticity, (b) local value of the normalized vorticity at the mid-location within the simulation domain, and (c) loss factor over the entire domain. Dashed yellow lines indicate the end of the training interval.

As the final discussion in this section, we have provided in Fig. 5.8 the visualizations of the Φ_{L} and the Φ_{NL} matrices (operators), corresponding respectively to the linear and the nonlinear Phi Method ROMs. We have visualized the Φ_{L} and the Φ_{NL} matrices from the Phi Method training on both the steady-state (45 – 65 s) and the transient-state (0 – 25 s) data. It is noted that, although the Φs are column matrices, we have illustrated a rectangular rearrangement of them in Fig. 5.8.

Apart from the Φ_{L} and the Φ_{NL} matrices that were discussed up to this point, a Φ matrix corresponding to another nonlinear model is also visualized. This new nonlinear model incorporates extra terms, namely, $V_x^2(x,y)$ and $V_y^2(x,y)$, in addition to the terms that were present in the so-far-analyzed nonlinear model (Eq. 5.14). The Φ matrix associated with this new model ($\Phi_{\mathrm{NL,ET}}$) is also visualized from the model's training on the steady-state and the transient-state data.

The matrices Φ_{L}, Φ_{NL}, and $\Phi_{\mathrm{NL,ET}}$ correspond to the Phi Method ROMs with increasing levels of complexity as is determined by the number of candidate terms in the respective libraries of the models.

Several points can be highlighted from the comparisons among the various Φ matrices in Fig. 5.8: first, at each complexity level (each row), the main characteristics of the Φ operators are shared between the models trained on the steady or the transient data. This includes the structure of the operator, which represents the dominant terms identified and the discretization stencil.

Second, across the complexity levels (column-wise comparison), the coefficients corresponding to each specific term are very similar. This provides confidence that the addition of extra terms in the

library, which may not be necessary to describe the dynamics, will not significantly impact the coefficients found for the essential and dominant terms. In practice, to achieve a balance between model's fidelity and complexity, we can plot the Pareto front of the loss factor vs the number of involved dynamical terms and identify the optimum model.

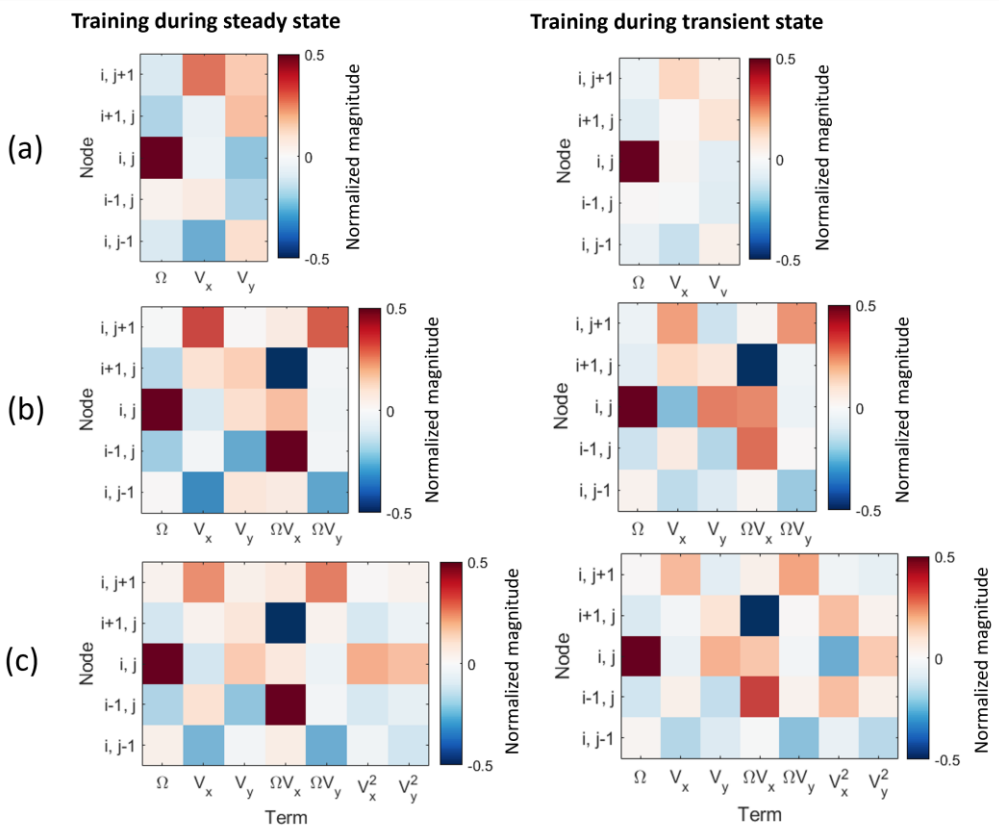


Fig. 5.8 Rearranged normalized representations of various coefficient matrices (operators, Φ) from the Phi method's application to the fluid flow test case; (a) visualization of the Φ_{L} matrix, (b) visualization of the Φ_{NL} matrix, and (c) visualization of the $\Phi_{\mathrm{NL,ET}}$ matrix that represents a larger library of terms compared to the ROM corresponding to Φ_{NL}. (**left column**) models' training performed on the steady-state data, (**right column**) models' training performed on the transient-state data. i and j represent the node's indices along the x and the y directions, respectively.

Third, it is emphasized that, by looking at a Φ operator (matrix), we can infer information about the discretized dynamics in terms of the relative importance of the candidate terms and their optimal discretization stencil. To be able to compare the obtained data-driven coefficients with their analytical values, we rewrite Eq. 5.12 in the following form

$$\frac{\partial \Omega}{\partial t} = -\frac{\partial \Omega \mathrm{V_x}}{\partial x} - \frac{\partial \Omega \mathrm{V_y}}{\partial y} + \Omega \frac{\partial \mathrm{V_x}}{\partial x} + \Omega \frac{\partial \mathrm{V_y}}{\partial y} + \frac{1}{Re}\left(\frac{\partial^2 \Omega}{\partial x^2} + \frac{\partial^2 \Omega}{\partial y^2}\right). \qquad \text{Eq. 5.15}$$

According to Eq. 5.15, the terms that appear in the dynamics include Ω, ΩV_x, and ΩV_y, which are also the terms with the dominant coefficients as can be seen from Fig. 5.8.

Now, we discretize Eq. 5.15 using first-order Euler scheme in time and second-order finite difference in space, and obtain

$$\Omega_{i,j}^{k+1} = \left(1 - \frac{2\Delta t}{Re}\left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2}\right)\right)\Omega_{i,j}^{k} + \frac{\Delta t}{Re}\left(\frac{1}{\Delta x^2}\left(\Omega_{i+1,j}^{k} + \Omega_{i-1,j}^{k}\right) + \frac{1}{\Delta y^2}\left(\Omega_{i,j+1}^{k} + \Omega_{i,j-1}^{k}\right)\right)$$
$$- \frac{\Delta t}{2\Delta x}\left((\Omega V_x)_{i+1,j}^{k} - (\Omega V_x)_{i-1,j}^{k}\right) - \frac{\Delta t}{2\Delta y}\left((\Omega V_y)_{i,j+1}^{k} - (\Omega V_y)_{i,j-1}^{k}\right) \qquad \text{Eq. 5.16}$$
$$+ \frac{\Delta t}{2\Delta x}\Omega_{i,j}^{k}\left(V_{x}{}_{i+1,j}^{k} - V_{x}{}_{i-1,j}^{k}\right) + \frac{\Delta t}{2\Delta y}\Omega_{i,j}^{k}\left(V_{y}{}_{i,j+1}^{k} - V_{y}{}_{i,j-1}^{k}\right).$$

Referring to the relation in Eq. 5.16, we point out that, in the current implementation of Phi Method, all constituent variables of a library term are evaluated on the same node. Accordingly, the terms such as $\Omega\frac{\partial \mathrm{V_x}}{\partial x}$ and $\Omega\frac{\partial \mathrm{V_y}}{\partial y}$, which appear as "cross-node" terms (i.e., for example, $\Omega_{i,j}^{k}V_{x}{}_{i+1,j}^{k}$, $\Omega_{i,j}^{k}V_{x}{}_{i-1,j}^{k}$, $\Omega_{i,j}^{k}\,V_{y}{}_{i,j+1}^{k}$, and $\Omega_{i,j}^{k}V_{y}{}_{i,j-1}^{k}$) in the discretized relation of Eq. 5.16 are not represented in a Phi Method model. As a result of this, a direct comparison between the learned coefficients and their analytical counterparts is not straightforward. In fact, for the fluid flow test case, Phi Method is effectively finding an *alternative* model that can optimally represent the data. Despite this point, a symmetrical pattern (with respect to node (i,j)) can be overall observed in the Φ matrices depicted in Fig. 5.8. This symmetrical pattern is reminiscent of the finite difference stencil, which has indeed been the underlying discretization stencil that was used for the simulation that generated the training and the test data.

5.3. Assessment of Phi Method for the prediction of a plasma discharge dynamics

The goal in this section is to demonstrate the utility of the ROMs from Phi Method for plasma systems and toward simultaneously forecasting the plasma state variables. This is done by applying Phi Method to the data from a plasma problem that represents the 1D azimuthal dynamics of a Hall thruster discharge. Similar to the fluid flow test case of the previous section, the aim here is also to verify the Phi Method's ability to recover PDEs that describe the time evolution of the system.

An OPT-DMD model is also presented for the plasma discharge test case. This is primarily to have a comparative point of reference for the performance of the Phi Method model. In addition, applying OPT-DMD to this test case allows us to further illustrate the effectiveness of the method for ROM development in plasma applications, adding to the demonstration cases that were presented for OPT-DMD in Chapter 3 and Chapter 4.

Following the same order of discussions as that pursued in section 5.2, we will first present the problem description and the corresponding setup that we had used to perform the ground-truth simulations. We then discuss the predictions and the characteristics of the ROMs from Phi Method in comparison against the OPT-DMD models. The robustness of the Phi-Method-derived ROMs to noise in the plasma data is assessed in section 5.4, alongside the same analysis in the fluid flow test case.

5.3.1. Description of the test-case problem setup

The setup of our plasma discharge test case is inspired by the well-defined setup that Lafleur et al. used in Ref. [23] for their physics studies. The test case corresponds to the problem of fluctuations in

plasma properties along the azimuthal coordinate of an $\mathrm{E} \times \mathrm{B}$ plasma discharge. The azimuthal direction (z) is perpendicular to the directions of the externally applied electric and magnetic field vectors, hence, along the $\mathrm{E} \times \mathrm{B}$ direction. In this test case, the electric field is along the axial (x) direction, and the magnetic field is along the radial (y) direction. The physics of the problem involves the development, growth, and saturation of the ECDI waves (see subsection 1.1.4), which exhibit nonlinear behaviors during the growth and saturation phases [23].

For the data generation, a 1D electrostatic PIC simulation using IPPL-1D code [229][230] was performed. The simulation solves the Poisson's equation along the azimuthal direction using a direct tridiagonal matrix solve algorithm (Thomas algorithm) given the charge distribution from the plasma species (electrons and ions), and in turn computes the self-consistent azimuthal electric field. The positions and the velocities of the plasma particles are updated at each time step of the simulation according to the electromagnetic forces at the particles' location [68][69]. The details of the simulation setup and conditions are described in the following.

The domain of the "ground-truth" simulation represents a 0.5×10^{-2} m-long extent of the azimuthal coordinate of a Hall thruster. A uniform radial magnetic field (B_y) with an intensity of 20 mT is applied. Also, an axial electric field (E_x) with the magnitude of $2 \times 10^4\ Vm^{-1}$ is imposed with a uniform distribution across the domain. The simulation's initialization involves sampling the electrons and the ions from their initial Maxwellian distributions at the temperatures of 2 eV and 0.1 eV, respectively. The sampled particles are then loaded throughout the domain with a uniform density of $1 \times 10^{17}\ m^{-3}$. The domain is discretized along the azimuthal (z) coordinate using $N_g = N_i = 100$ nodes. At the simulation's start, the number of macroparticles per cell for either the electron or the ion species is 200. The time step is $5 \times 10^{-12} s$, and the total simulated time is 10 μs. The plasma properties are averaged over every 500-time steps (2.5 ns) and recorded in our dataset. Thus, the total number of plasma snapshots is 4000.

A fictitious axial extent with the length of 0.01 m is considered. This axial extent provides a means to represent the axial convection of the instability waves, which is important to properly capture their saturation [23]. The axial extent also enables limiting the growth of the charged particles' energy [23]. The electrons or the ions that leave the domain from one of the axial boundaries are resampled from their initial distributions and are reloaded at the opposite axial end of the domain, retaining their azimuthal position.

Along the azimuthal direction, a periodic boundary condition is applied on the particles. No boundary condition is applied on the particles along the radial coordinate, and the particles move freely in this direction. Note that, despite assuming a finite axial length, the simulation still represents a 1D problem, since the Poisson's equation is only solved along the azimuthal coordinate. An azimuthally periodic solution for the plasma potential is ensured by imposing zero-volt Dirichlet conditions at both azimuthal ends of the domain.

The test case is collisionless. This assumption is to simplify the involved dynamics and, hence, to limit the number of candidate terms that may need to be accounted for in the Phi Method's library.

For the present test case, we aim to use Phi Method to discover a system of discretized PDEs that describes the spatiotemporal evolution of multiple plasma properties simultaneously. These plasma properties, which constitute the output vector of the Phi Method ROM, are the electron number density (n_e), the electrons' azimuthal and axial drift velocity components ($V_{d,ez}$ and $V_{d,ex}$), and the azimuthal electric field (E_z).

The library of candidate terms consists of the state variables in the output vector themselves (the above quantities) in addition to a few other nonlinear functions of them, namely, $n_e V_{d,ez}$, $n_e V_{d,ez}^2$, $n_e E_z$, and $n_e T_{ez}$. The last term, $n_e T_{ez}$, represents the electron pressure, $\Pi_{zz} = k_B n_e T_{ez}$, where k_B is the Boltzmann constant, and T_{ez} is the azimuthal electron temperature.

Accordingly, for the plasma test case, $\Phi \in R^{24\times 4}$, and Eq. 5.2 is expressed as

$$\begin{bmatrix} n_{e_i} \\ V_{d,ez_i} \\ E_{z_i} \\ V_{d,ex_i} \end{bmatrix}^{\mathrm{T}}_{k+1} = \begin{bmatrix} \begin{bmatrix} n_{e_{i-1}} \\ n_{e_i} \\ n_{e_{i+1}} \end{bmatrix}^T \begin{bmatrix} V_{d,ez_{i-1}} \\ V_{d,ez_i} \\ V_{d,ez_{i+1}} \end{bmatrix}^T \begin{bmatrix} E_{z_{i-1}} \\ E_{z_i} \\ E_{z_{i+1}} \end{bmatrix}^T \begin{bmatrix} V_{d,ex_{i-1}} \\ V_{d,ex_i} \\ V_{d,ex_{i+1}} \end{bmatrix}^T \begin{bmatrix} (n_e V_{d,ez})_{i-1} \\ (n_e V_{d,ez})_i \\ (n_e V_{d,ez})_{i+1} \end{bmatrix}^T \begin{bmatrix} (n_e V_{d,ez}^2)_{i-1} \\ (n_e V_{d,ez}^2)_i \\ (n_e V_{d,ez}^2)_{i+1} \end{bmatrix}^T \begin{bmatrix} (n_e E_z)_{i-1} \\ (n_e E_z)_i \\ (n_e E_z)_{i+1} \end{bmatrix}^T \begin{bmatrix} (n_e T_{ez})_{i-1} \\ (n_e T_{ez})_i \\ (n_e T_{ez})_{i+1} \end{bmatrix}^T \end{bmatrix}_k \Phi. \quad \text{Eq. 5.17}$$

The choice of the considered dynamical library terms in Eq. 5.17 is informed by the conservation equations for the electrons, namely, the continuity equation (Eq. 5.18) and the 1D momentum equations along the axial and the azimuthal directions (Eq. 5.19 and Eq. 5.20). Additionally, the chosen library reflects a balance between capturing the dominant dynamics and maintaining reasonable model complexity. As will be demonstrated in subsection 5.3.2, the model as defined per Eq. 5.17 exhibited acceptable performance for the utility demonstration purposes of this thesis regarding Phi Method. However, it is acknowledged that the omission of certain terms, such as $n_e V_{d,ex} V_{d,ez}$ as well as other possible dynamical terms, leaves open the possibility of further improving the predictions. Further investigations along this line to maximize the performance of Phi Method ROMs are left for future work.

$$\frac{\partial n_e}{\partial t} = -\frac{\partial n_e V_{d,ez}}{\partial z}, \quad \text{Eq. 5.18}$$

$$q B_y n_e V_{d,ez} = m_e \frac{\partial (n_e V_{d,ex})}{\partial t} + m_e \frac{\partial \left(n_e V_{d,ex} V_{d,ez}\right)}{\partial z} - q n_e E_x + \frac{\partial \Pi_{xz}}{\partial z}, \quad \text{Eq. 5.19}$$

$$-q B_y n_e V_{d,ex} = m_e \frac{\partial \left(n_e V_{d,ez}\right)}{\partial t} + m_e \frac{\partial \left(n_e V_{d,ez}^2\right)}{\partial z} - q n_e E_z + \frac{\partial \Pi_{zz}}{\partial z}. \quad \text{Eq. 5.20}$$

In the above equations, q and m_e are the elementary charge and the electron mass, respectively. Π_{zz} and Π_{xz} are the pressure tensor components (Π_{ij}), which are defined as the second moments of the electrons' velocity distribution function ($f_e(v)$) according to

$$\Pi_{ij} = m_e \int_{-\infty}^{\infty} \left(v - V_{d,ei}\right)\left(v - V_{d,ej}\right) f_e(v) d^3 v\,. \quad \text{Eq. 5.21}$$

Similar to what was done for the fluid flow test case, the data corresponding to each plasma variable is normalized here with respect to its maximum value. This ensures the order-of-magnitude consistency across the involved terms. The initial 2 μs of the data is disregarded as this time period represents the transient phase of the system corresponding to the excitation and the growth of the ECDI. This was to

enable the application of OPT-DMD to the data for the purpose of comparisons between the OPT-DMD-derived and the Phi-Method-derived ROMs.

To train the Phi Method model, the output vector and the terms in the library matrix were evaluated on several data points during the training interval that spanned from 2 to 6 μs. The Φ matrix was obtained following a regression on the resulting data matrices (on the left-hand and the right-hand sides of Eq. 5.17). The rest of the simulation data corresponding to the timeframe of 6 – 10 μs were reserved for testing the model.

With regard to the OPT-DMD model, in order for it to be comparable with the Phi Method ROM, we used the same terms in the Phi Method's library to construct the augmented state vectors ($\boldsymbol{\mathcal{X}}_k$) for OPT-DMD (refer to subsection 3.2.1.1). The $\boldsymbol{\mathcal{X}}_k$s comprise the rearranged snapshots of the normalized state variables, in addition to their relevant nonlinear observables, according to Eq. 3.16. OPT-DMD was applied to the augmented matrix X (Eq. 3.14), which contains the state vectors $\boldsymbol{\mathcal{X}}_k$, with $k \in [800, 2400)$ corresponding to the training interval of 2 – 6 μs. The leading 25 DMD modes were retained for the OPT-DMD ROM.

5.3.2. Results and discussions

Fig. 5.9 shows the time evolution signals of the output state variables (n_e, $V_{de,z}$, E_z, and $V_{de,x}$) at the mid-location of the domain from the Phi Method and the OPT-DMD ROMs. The time variations of these variables are reconstructed during the training interval and are forecasted within the test period. The predicted signals are compared against the ground-truth from the PIC simulation.

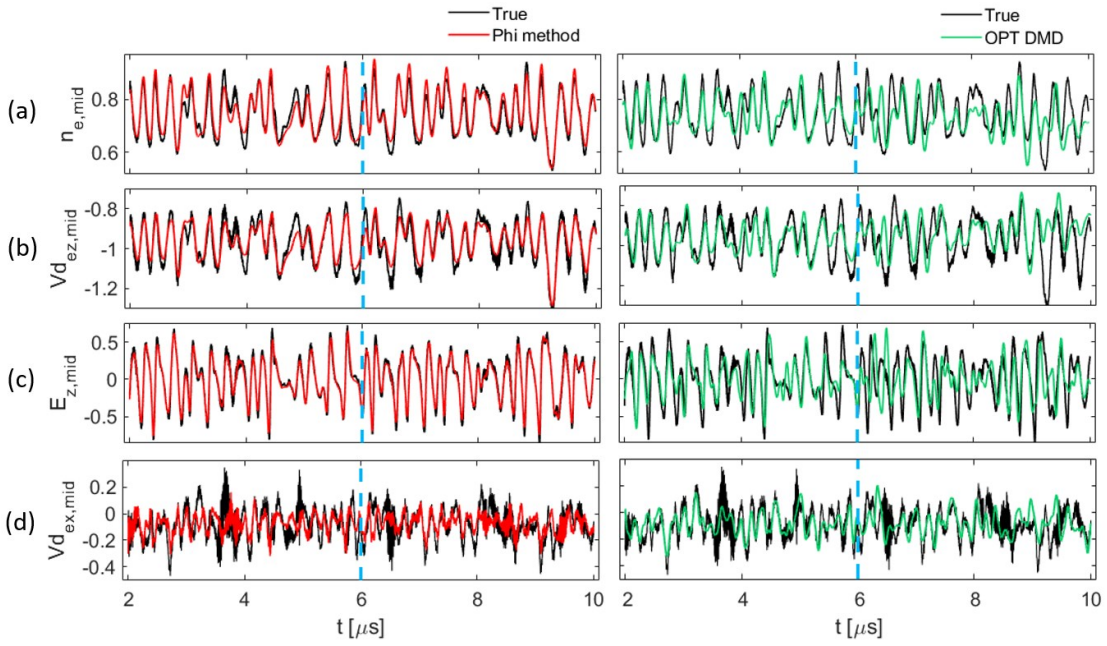


Fig. 5.9 Comparison of the predictions from the Phi Method (**left column**) and the OPT-DMD (**right column**) ROMs against the ground-truth data for plasma test case; time evolutions of the local values at the mid-location within the simulation domain of (a) normalized electron number density (n_e), (b) normalized electrons' azimuthal drift velocity ($V_{d,ez}$), (c) normalized azimuthal electric field (E_z), and (d) normalized electrons' axial drift velocity ($V_{d,ex}$). Dashed blue lines indicate the end of the training interval.

Moreover, in Fig. 5.10, the predictions of the data-driven ROMs regarding the full spatiotemporal evolution of the output plasma state variables are provided and compared against the original data.

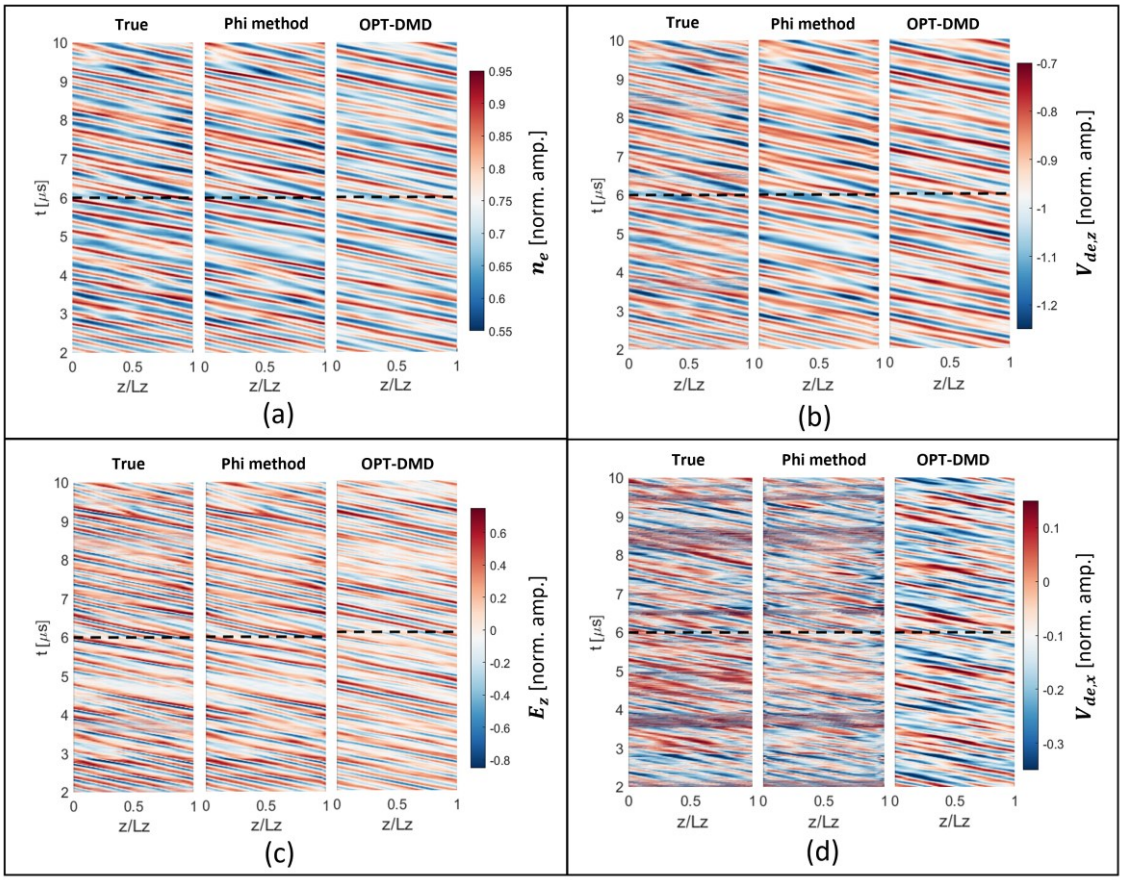


Fig. 5.10 Comparison of the predictions from the Phi Method and the OPT-DMD ROMs against the ground-truth data for the plasma test case; spatiotemporal evolution plots of (a) normalized electron number density (n_e), (b) normalized electrons' azimuthal drift velocity ($V_{d,ez}$), (c) normalized azimuthal electric field (E_z), and (d) normalized electrons' axial drift velocity ($V_{d,ex}$). Dashed black lines indicate the end of the training interval.

We would highlight that, in advancing the Phi Method ROM from one time step to the next, all the *input* variables named in subsection 5.3.1, i.e., the library terms in Eq. 5.17, were obtained from the model itself except for the $n_e T_{ez}$, which was fed into the ROM from the PIC simulation data. With this exception, the model-predicted values were used to update the other library terms for the prediction of the *output* state vector at the subsequent time step.

In theory, achieving a closed system of equations is possible by also including T_{ez} as a state variable and finding an equation for its dynamics as well along with the other variables. Prediction using a closed system means that all variables are predicted simultaneously by the model itself. However, in practice, advancing in time with a fully self-consistent system of equations derived by Phi Method (or any data-driven method) poses challenges. Minor errors in the learned coefficients, likely and primarily due to noise in the data, amplify the predictions' error propagation across variables and over time. Any inaccuracies in the prediction of one state variable directly impact the predictions of all others. Over time, these compounded errors can grow, leading to an accumulation that destabilizes the solution. Therefore, for the purpose of the Phi Method's demonstrations in this thesis, fully self-consistent

predictions were not attempted, leaving this as a subject for future work (refer also to subsection 7.2.1.2).

Besides the above remarks, to reflect the azimuthal periodicity in the data, Eq. 5.2 was used with the following redefinition of the $\boldsymbol{y}$ vectors at the boundary nodes

$$\boldsymbol{y}_i^k = \begin{bmatrix} \boldsymbol{x}_{N_i-1}^k \\ \boldsymbol{x}_i^k \\ \boldsymbol{x}_{i+1}^k \end{bmatrix}^T, \qquad i = 1, \tag{Eq. 5.22}$$

$$\boldsymbol{y}_i^k = \begin{bmatrix} \boldsymbol{x}_{i-1}^k \\ \boldsymbol{x}_i^k \\ \boldsymbol{x}_2^k \end{bmatrix}^T, \qquad i = N_i. \tag{Eq. 5.23}$$

Therefore, the boundary values are computed within the Phi Method ROM itself given the initial conditions.

The OPT-DMD ROM is fully self-consistent, i.e., the values of the complete state vector (X) are computed within the model at each time step.

As was mentioned in subsection 4.2.3.2, the ability of a data-driven ROM to provide reliable predictions of the *local* values of the state variables of a plasma system is a challenging task. This is because the local values of the plasma properties are affected by an array of intricate plasma phenomena and interactions. In this respect, Fig. 5.9 and Fig. 5.10 clearly illustrate that Phi Method reliably manages to handle this task as it is evidenced by the remarkable agreement between the Phi Method ROM's predictions with the corresponding ground-truth data.

Furthermore, when evaluating the performance of the OPT-DMD model, one must consider its inherent self-consistency as well as the challenging aspect of predicting the system's dynamics in the absence of a fully periodic behavior, which we extensively discussed in Chapter 4. Despite these factors, OPT-DMD is seen in Fig. 5.9 and Fig. 5.10 to provide a reasonably accurate prediction of the system's state vector.

For a more quantitative comparison between the accuracy of predictions from the Phi Method and the OPT-DMD ROMs, we refer to Fig. 5.11. This figure shows the time variations of the loss factor associated with the data-driven models' predictions of the various plasma properties. The loss factor is computed with respect to ground-truth data and over the entire domain in accordance with Eq. 4.1.

From the plots in Fig. 5.11, we observe that, first of all, the mean of the loss over time corresponding to the Phi Method ROM is smaller than the mean loss corresponding to the OPT-DMD ROM for all the plasma properties except for the $V_{d,ex}$ (Fig. 5.11(d)), where the models from both methods exhibit comparable mean loss values.

Second, the Phi Method's loss in the forecasting window remains at a similar level to that during the training. In contrast, the OPT-DMD's predictions experience a slight increase in loss factor during the test window. This difference in the loss factor behavior between the two models stems from their inherently distinct nature as was also explained in subsection 5.2.2.2. The fundamental assumption of

Phi Method is that the dynamics of a system can be characterized by a unique set of (partial) differential equations throughout the evolution of the system. Once these equations are identified through the best-fit operator Φ, they are presumed to effectively capture the system's evolution, whether reconstructing the observed data during the training phase or forecasting the unseen data beyond the training interval. OPT-DMD also assumes that the spatial modes and their dynamics remain constant throughout time. However, while the assumption of the constancy of PDEs is a realistic assumption for physical systems, the time-invariability of the DMD modes is only strictly valid for a fully periodic system. Accordingly, due to slight variations in the existing DMD's spatiotemporal modes from the training to the test interval, the observed increase in the OPT-DMD's loss beyond the training point is expected.

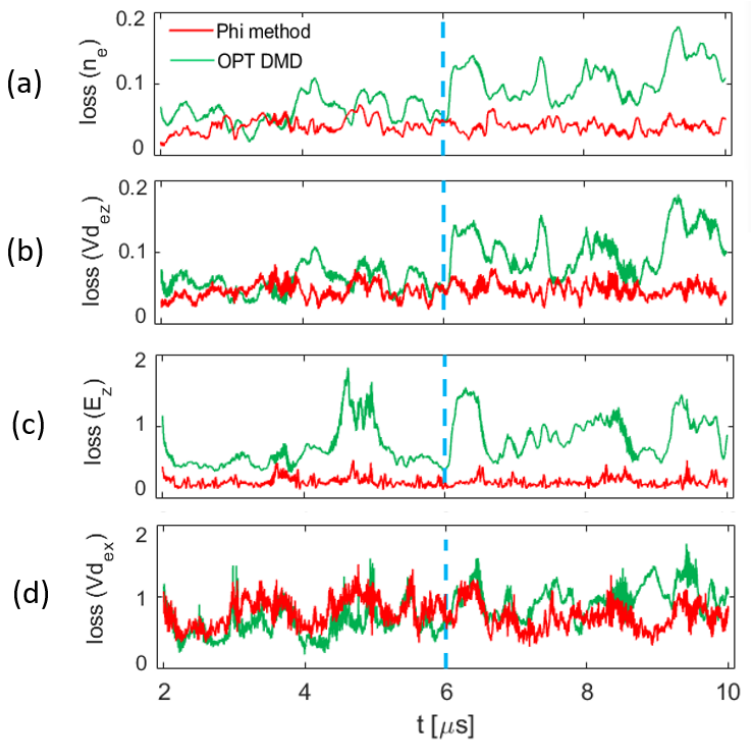


Fig. 5.11 Comparison of the ROMs from Phi Method and OPT-DMD for the plasma test case; time evolutions of the loss factor calculated over the entire domain corresponding to (a) normalized electron number density (n_e), (b) normalized electrons' azimuthal drift velocity ($V_{d,ez}$), (c) normalized azimuthal electric field (E_z), and (d) normalized electrons' axial drift velocity ($V_{d,ex}$). Dashed blue lines indicate the end of the training interval.

Third, we observe from Fig. 5.11 that some plasma properties are predicted with a higher accuracy than the others. Furthermore, the relative accuracy between the Phi Method and the OPT-DMD ROMs varies across different plasma properties. In particular, the loss associated with the Phi Method's predictions of the n_e is the smallest among the various state properties and remains below 0.05 at all times. The Phi Method and the OPT-DMD models show the largest disparity in their loss levels toward predicting the E_z, with the OPT-DMD's loss being more than three times that of the Phi Method's. In addition, both models exhibit nearly identical loss levels (below 1 on average) for the $V_{d,ex}$. The loss factor for the $V_{d,ex}$ represents the highest value relative to the loss factors associated with the other predicted plasma properties.

5.3.2.1. Impacts of the size of the library on the Phi Method ROM

In this subsection, we study the impacts that the size of the library of candidate terms may have on Phi Method, assessing the extent to which the number of candidate library terms can affect the learned operator Φ.

For the purpose of the above analysis, we focused on a particular plasma property, namely, the electrons' axial current density ($J_{ex} = -qn_eV_{d,ex}$). This quantity is of particular interest since it represents the aggregate contributions of various effects to the phenomenon of electrons' transport across the magnetic fields. The lack of an accurate prediction of this quantity has constrained the self-consistency of the fluid-based plasma simulations so far (refer to section 1.3 for more details).

We have trained two distinct Phi Method models for the J_{ex} property using the simulation data, with each model being represented by a different set of library terms. The first model is characterized by a small library of terms that includes only the four essential terms that appear in the governing PDE (Eq. 5.20). These are $n_eV_{d,ez}$, n_eE_z, n_eT_{ez}, and $n_eV_{d,ez}^2$.

The second model features a large library that consists of the relevant plasma state variables and their nonlinear observables, including the product of the linear and the quadratic functions of the plasma variables. For this model, the terms in the library are the following: n_e, $V_{d,ez}$, T_{ex}, E_z, $n_eV_{d,ez}$, n_eE_z, n_eT_{ez}, $n_eV_{d,ez}^2$, $n_eE_z^2$, $n_eT_{ez}^2$, $n_e^2V_{d,ez}$, $n_e^2E_z$, $n_e^2T_{ez}$, $V_{d,ez}E_z$, $V_{d,ez}T_{ez}$, E_zT_{ez}, $V_{d,ez}^2E_z$, $V_{d,ez}E_z^2$, $V_{d,ez}^2T_{ez}$, $V_{d,ez}T_{ez}^2$, $E_z^2T_{ez}$, and $E_zT_{ez}^2$.

Accordingly, the matrices of the coefficients are represented by $\Phi_{\mathrm{SL}} \in R^{12\times1}$ for the Phi Method model with a small library (SL) and by $\Phi_{\mathrm{LL}} \in R^{66\times1}$ for the model with an expanded large library (LL).

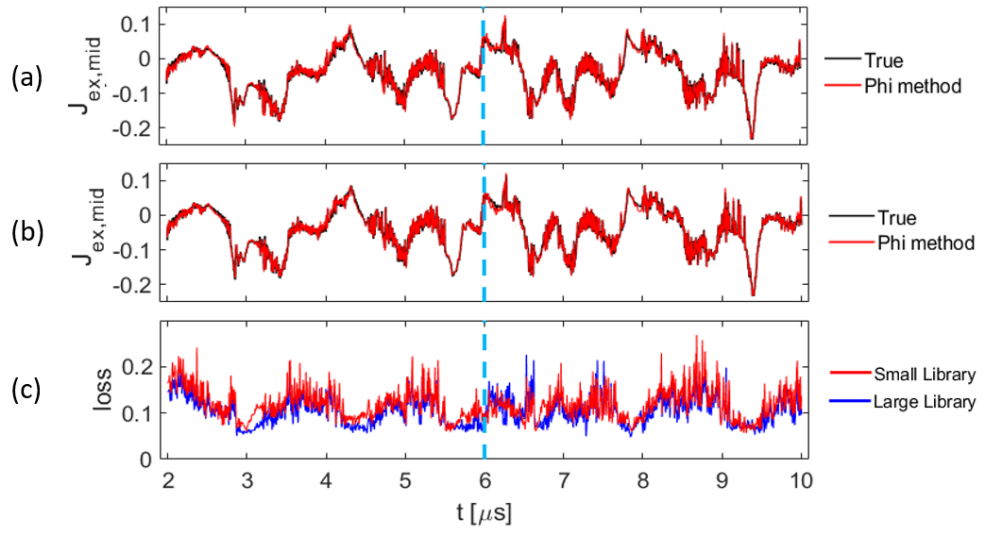


Fig. 5.12 Comparison of the Phi method ROMs for the normalized electrons' axial current density (J_{ex}) when regressing over a small and a large library of terms in the plasma test case; (a) &(b) time evolutions of the local values of the J_{ex} at the mid-location within the domain from the ground-truth simulation and from the Phi Method models, plot (a) with a small library, plot (b) with a large library of terms. (c) Time evolutions of the loss factor calculated over the entire domain associated with each ROM. Dashed blue lines indicate the end of the training interval.

The predicted time evolution signals of the J_{ex} at the midpoint of the domain from the above two Phi Method ROMs are compared against the true data in Fig. 5.12(a) and (b). In addition, Fig. 5.12(c) presents the temporal variations of the loss factors associated with the two models. The loss factors are calculated over the entire domain. It is evident from the plots in Fig. 5.12 that the predictions of the two ROMs are nearly identical.

In Fig. 5.13, the rearranged representations of the Φ_{SL} and the Φ_{LL} matrices (operators) are illustrated. Detailed interpretations of the obtained coefficients and their comparison with the analytical values from the discretization of the governing PDEs are discussed in Chapter 6.

Here, we would point out from Fig. 5.13 that, even in the presence of numerous dynamical terms in the library, Phi Method can identify the dominant term(s) which, in this special case, is the $n_e T_{ez}$. Also, the relative magnitude of the coefficients of the terms that are shared between the two models are comparable. Nonetheless, for the model with the larger library, the weight of the pressure contribution (Π_{zz}) is distributed among all observables that have been functions of both the n_e and the T_{ez}. In other words, a portion of the pressure contribution is captured within the $n_e T_{ez}^2$ and the $n_e^2 T_{ez}$ terms, apart from the main share, which is attributed to the $n_e T_{ez}$. As a result, the algorithm assigns non-zero coefficients to the terms $n_e T_{ez}^2$ and $n_e^2 T_{ez}$, even though they are not essential for the representation of the dynamics. This observation is a consequence of the least-square regression ($L2$-norm minimization) corresponding to the current Phi Method's implementation. The $L2$-norm minimization promotes uniformity among the optimized coefficients of the various terms.

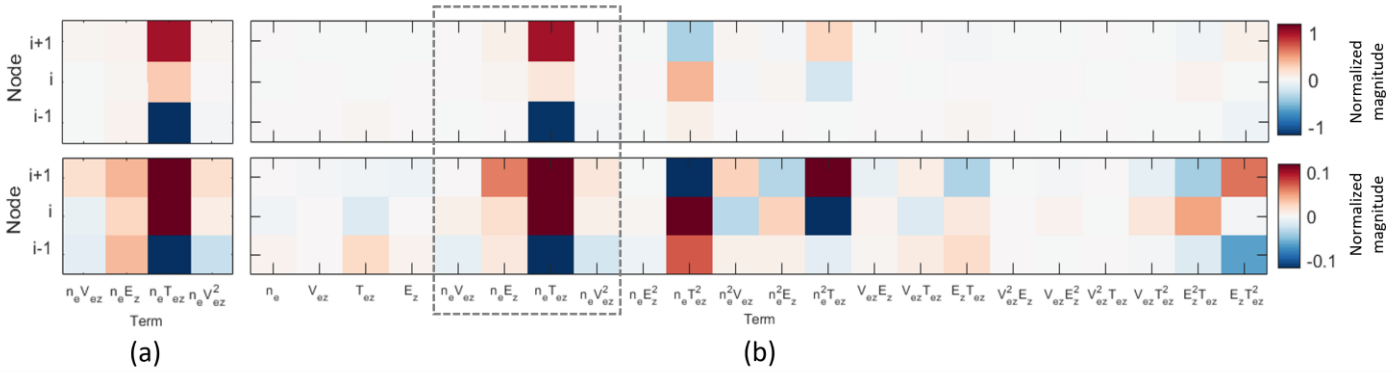


Fig. 5.13 Rearranged normalized representations of the coefficient matrices (Φ) from the Phi method's application to the J_{ex} data of the plasma test case; (a) Φ_{SL} visualization (small library of terms), (b) Φ_{LL} visualization (large library of terms). The bottom row is a rescaled version of the top row. The dashed box encompasses the terms shared between the two Phi Method models.

The uniformity-promoting behavior can be avoided by promoting sparsity in the discovered model, which can be achieved by imposing constraints that encourage the algorithm to incorporate as few terms as required in the model. While the presented case here suggests that both models have equivalent predictive capabilities (from Fig. 5.12), the model with fewer terms is not only more interpretable but also likely to demonstrate better generalizability across a diverse range of scenarios by minimizing the risk of overfitting to the data. Thus, the main purpose of encouraging sparsity is to improve the interpretability and generalizability of the model.

Incorporating a sparsity-promoting regression in the Phi Method's algorithm will be a subject for research beyond this thesis and is, thus, discussed in some more detail in section 7.2.1.

5.4. Noise robustness of Phi Method: comparative analysis against OPT-DMD

The effects of noise in the data on the performance of the ROMs from Phi Method are investigated in this section for the fluid flow and the plasma test cases (sections 5.2.2 and 5.3, respectively). The noise robustness of Phi Method is assessed comparatively against that of OPT-DMD, which was demonstrated in section 4.3 to exhibit strong noise robustness characteristics.

To perform the noise robustness analyses, artificial Gaussian noise was added to the simulation data of the fluid flow and the plasma test cases, and the data-driven models were trained on the resulting noisy data.

5.4.1. Fluid flow around a cylinder test case

For this test case, we added Gaussian noise with the standard deviation of $\sigma = 0.03$ to all the flow properties. The characteristics of the Phi Method and the OPT-DMD ROMs used here are kept the same as those described in subsection 5.2.2.1. The "nonlinear" Phi Method ROM (Eq. 5.14), whose library featured both linear and nonlinear terms, is assessed in this subsection.

Fig. 5.14 and Fig. 5.15 present the predictions of the data-driven ROMs trained on the noisy data. The predictions are shown in terms of the time evolutions of the spatially averaged vorticity and the local value at the mid-domain-location of the vorticity field in Fig. 5.14. Fig. 5.15 presents the predicted snapshots of the entire vorticity field at a few sample time instants within the test window. The true data in these figures represents the simulation data after the introduction of noise. In Fig. 5.16, the absolute-difference plots between the true and the predicted vorticity field snapshots are additionally illustrated.

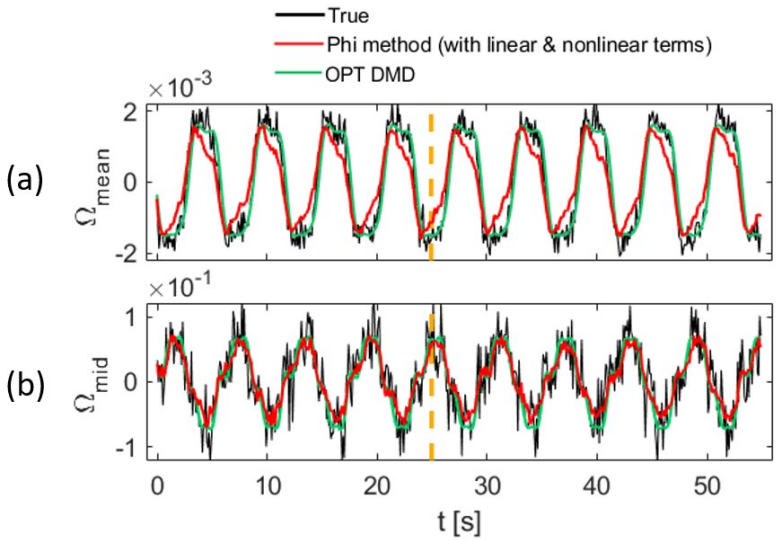


Fig. 5.14 Comparison between the predictions of the Phi Method and the OPT-DMD models with respect to the ground-truth data in the presence of a Gaussian white noise in the fluid flow test case; time evolutions of (a) spatially averaged normalized vorticity, (b) local value of the vorticity at the mid-location within the simulation domain. Yellow lines indicate the end of the training interval.

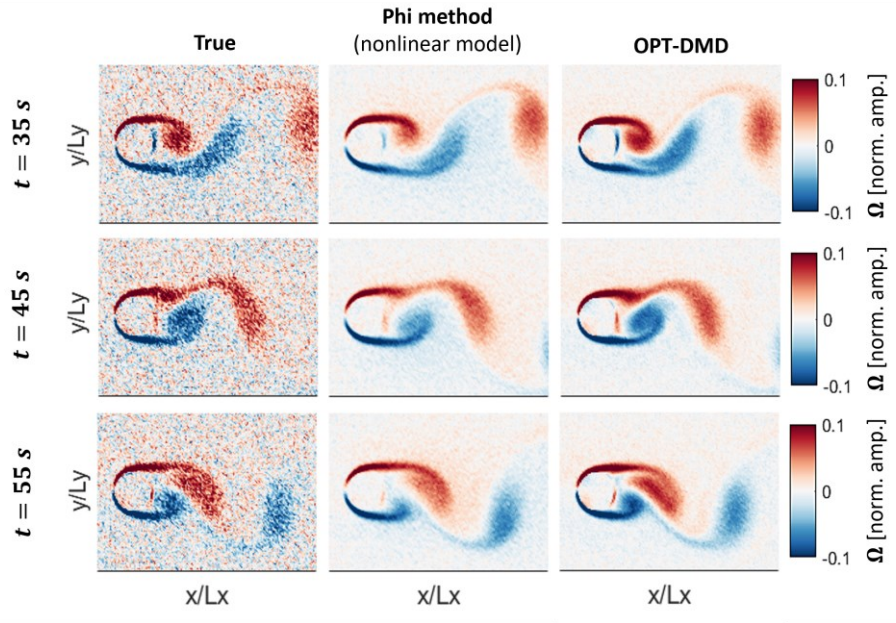


Fig. 5.15 Comparison between the predictions of the Phi Method and the OPT-DMD models with respect to the ground-truth data in the presence of a Gaussian white noise in the fluid flow test case; snapshots of the normalized vorticity field at three different time instants during the test interval.

These results show that both the Phi method and the OPT-DMD ROMs can effectively learn the underlying dynamics in the presence of noise. This is not a surprising outcome for OPT-DMD because the noise is largely excluded after performing a low-rank ($r = 14$) truncation of the POD expansion of the data as part of the DMD algorithm. However, it is a notable observation that Phi Method is also able to maintain its predictive capability without featuring a significant noise level in its predictions.

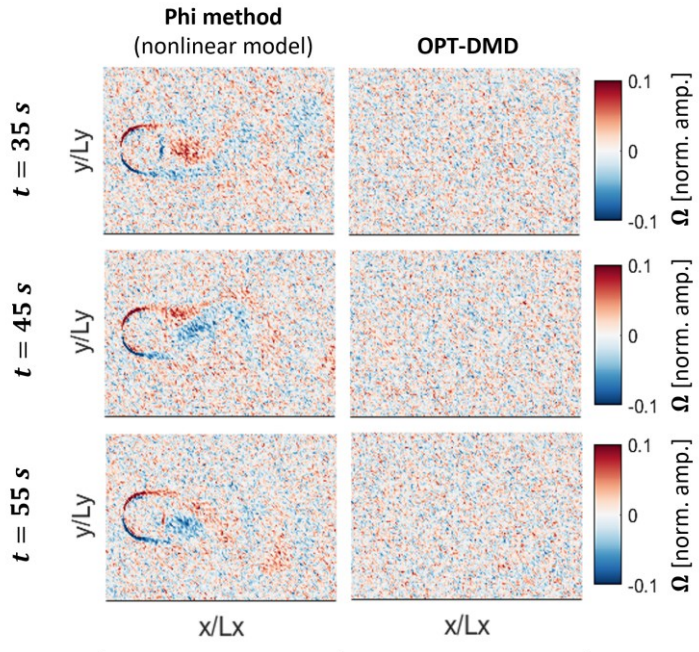


Fig. 5.16 Comparison between the Phi Method and the OPT-DMD models in the presence of a Gaussian white noise in the fluid flow test case; plots of absolute difference between the true and the predicted normalized vorticity snapshots at three different time instants during the test interval.

5.4.2. Plasma discharge test case

Gaussian noise with a standard deviation of $\sigma = 0.05$ is added to the full dataset corresponding to all the relevant plasma properties obtained from the PIC simulation of the 1D azimuthal plasma discharge test case. It is emphasized that the original PIC data inherently contains a certain level of (statistical) noise, and the Gaussian noise is added on top of that pre-existing noise. As a result, this case represents a more extreme scenario than the one analyzed in the previous subsection, noting also that the introduced noise has a higher amplitude.

The Phi Method and the OPT-DMD models, whose details were presented in subsection 5.3.1, were trained on the dataset with augmented noise. The time variations of the predicted values of the plasma properties at the midpoint of the domain, and the complete spatiotemporal maps of the plasma properties, from the Phi Method and the OPT-DMD ROMs are provided in Fig. 5.17 and Fig. 5.18, respectively. We observe that OPT-DMD effectively filters the noise and is able to provide a relatively consistent representation of the system. For the case of Phi Method, however, the presence of high-level noise in the data affects the obtained ROM, resulting in the prediction of lower-amplitude fluctuations in the plasma properties' signals compared to the ground-truth. This is because certain dynamical information within the data is obscured by the noise. Nevertheless, apart from the amplitude, the main traces of the signals and the frequencies of the fluctuations across the different properties are consistently captured by the Phi Method ROM.

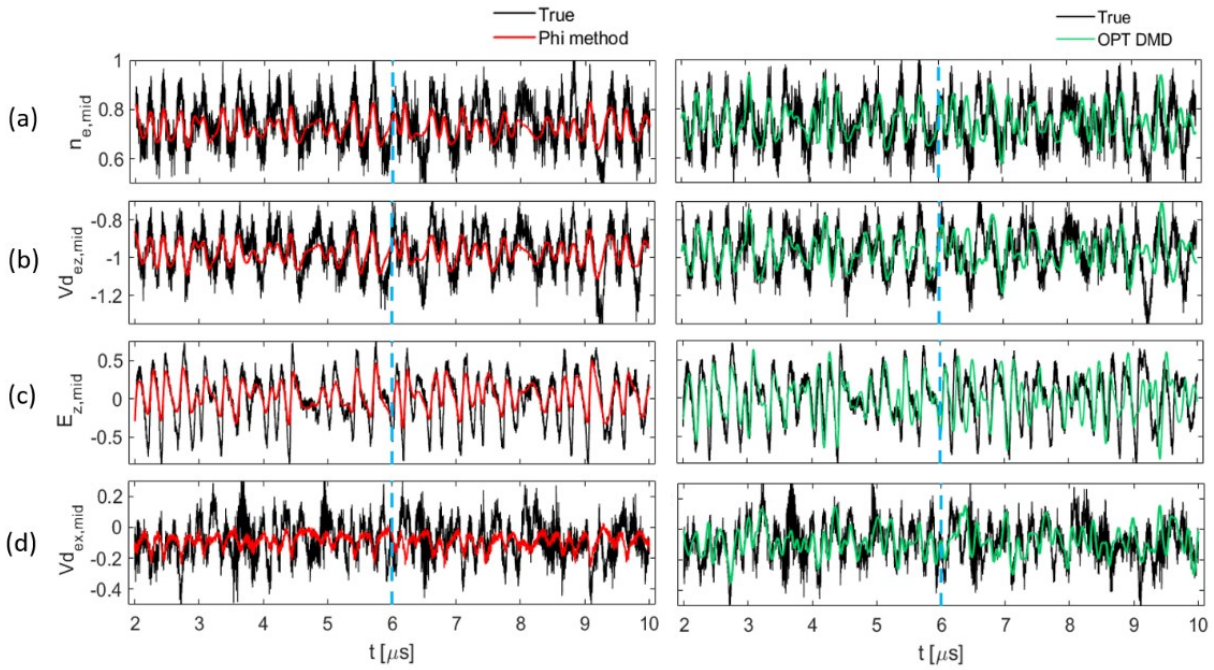


Fig. 5.17 Comparison of the predictions from the Phi Method (**left column**) and the OPT-DMD (**right column**) ROMs against the ground-truth data in the presence of a Gaussian white noise in the plasma test case; time evolution of the local values at the mid-location within the simulation domain of (a) normalized electron number density (n_e), (b) normalized electrons' azimuthal drift velocity ($V_{d,ez}$), (c) normalized azimuthal electric field (E_z), and (d) normalized electrons' axial drift velocity ($V_{d,ex}$). Dashed blue lines indicate the end of the training interval.

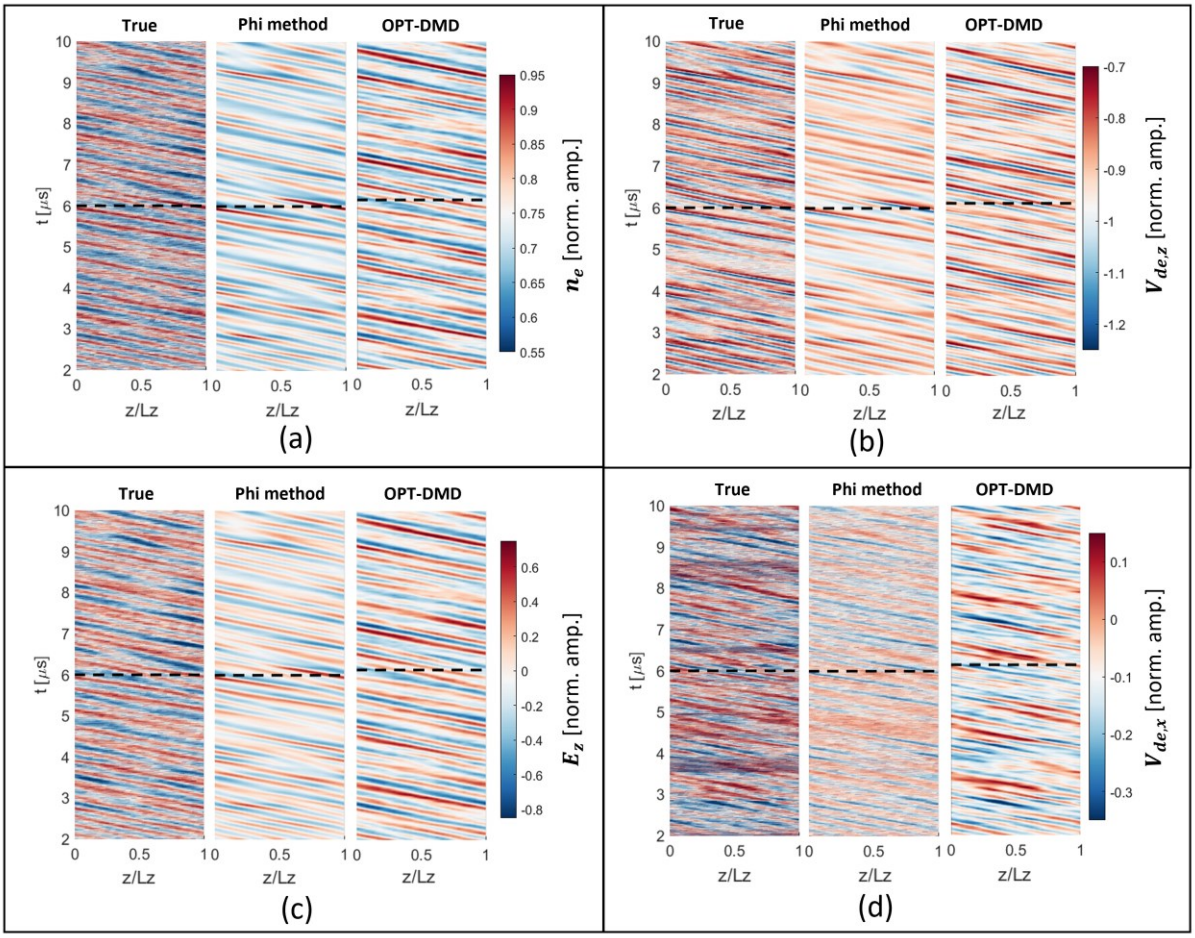


Fig. 5.18 Comparison of the predictions from the Phi Method and the OPT-DMD ROMs against the ground-truth data in the presence of a Gaussian white noise in the plasma test case; spatiotemporal evolution plots of (a) normalized electron number density (n_e), (b) normalized electrons' azimuthal drift velocity ($V_{d,ez}$), (c) normalized azimuthal electric field (E_z), and (d) normalized electrons' axial drift velocity ($V_{d,ex}$). Dashed black lines indicate the end of the training interval.

5.5. Chapter conclusions

We introduced in this Chapter a novel data-driven local operator finding algorithm termed "Phi Method". Phi Method is developed to enable the discovery of the discretized form of differential equations describing a system's dynamics and, thus, to allow straightforward and fast time forecasting of the system state variables.

The underlying concept and formulation of Phi Method was presented in detail. It was pointed out that Phi Method belongs to the category of the ML/DD algorithms that identify the local dynamics operators and, hence, discover the differential equation(s) that describe the time evolution (dynamics) of a system. An example of such data-driven algorithms was mentioned to be SINDy. It was highlighted that Phi Method is, in essence, an extension to SINDy algorithm to enable the simultaneous discovery of the dynamics and the optimum discretization stencil for the involved dynamics variables.

We demonstrated the remarkable potentials and generalizability of the Phi Method approach to enable the development of accurate and predictive data-driven ROMs across three test cases: (1) a canonical Lorenz attractor problem, (2) a widely adopted canonical 2D fluid benchmark problem corresponding to the case of viscous flow around a cylinder, and (3) a 1D $\mathrm{E} \times \mathrm{B}$ plasma problem representative of the azimuthal dynamics of a Hall thruster discharge. The involved dynamics in the first test case is governed by an ODE, whereas the evolution of the system in the fluid and the plasma test cases are governed by a system of PDEs.

To have a comparative reference for the predictive performance of the Phi-Method-derived ROMs, we applied OPT-DMD as well to the data of the test cases 2 and 3 and presented the predictions of the resulting ROMs.

Across all the three test cases, Phi Method was shown to reliably learn the involved dynamics, and the obtained ROMs from this approach featured a great predictive capacity. With regard to the most notable observations and outcomes, in the Lorenz test case, we verified the interpretability of the Phi Method's learned operator (the Φ matrix). We showed that the dominant coefficients of the Φ matrix had values that compared with a difference of less than 1 % against the analytical coefficients from the discretized ODE that describes the dynamics of the Lorenz attractor.

In the fluid flow test case, it was shown that a linear Phi Method ROM, which excludes the nonlinear terms that are present in the PDE describing the evolution dynamics of the vorticity field, provides a reasonably well representation of the dynamics and provides predictions that compare closely with the ground-truth data, especially in terms of the time variations of the local vorticity values.

It was also demonstrated that Phi Method can learn the underlying dynamics from the data of the transient state of the system, and the resulting model also provides accurate predictions. In this sense, Phi Method exceeds the capabilities of DMD-based data-driven techniques, which are inherently limited in their ability to properly represent the transient phenomena. The capability of Phi Method to work seamlessly with the transient data is particularly beneficial for the dynamics discovery of systems that may not exhibit a quasi-steady state or that for which the generation of statistically representative high-fidelity data of the quasi-steady state of the system may be computationally burdensome, such as the case of a Hall thruster discharge.

In the plasma discharge test case, the ROMs from Phi Method and OPT-DMD were both successfully demonstrated toward a simultaneous forecasting of several plasma properties. Phi Method was overall observed to provide a more accurate prediction of the variations in the local values of the plasma state variables compared to OPT-DMD throughout the training and test period, an observation that was linked to the inherently local learning of the dynamics associated with the Phi Method algorithm.

The noise robustness of Phi Method was assessed for both the fluid flow and plasma discharge test cases, again in comparison against OPT-DMD. In the fluid flow test case, both Phi Method and OPT-DMD demonstrated effective noise robustness, accurately capturing the dynamics in the presence of added noise. The fact that Phi Method maintained its predictive accuracy without significant noise

interference in the results was particularly notable. In the plasma discharge test case, where higher levels of Gaussian noise were added, OPT-DMD continued to filter noise effectively, providing consistent predictions. Phi Method, however, displayed slightly reduced amplitude in its predictions, as the higher noise levels partially obscured dynamical information. Despite this, Phi Method accurately captured the key patterns and frequency characteristics, demonstrating its capacity to handle substantial noise without significant performance degradation.

Chapter 6 Phi Method for the data-driven learning of parametric dynamics

The dynamics of many real-world systems is dependent on a set of parameters. These parameters are either intrinsic to the system and inherently evolve over time, or that are external and can be modified to control the system's behavior. A fully generalizable model (data-driven or otherwise) shall be able to reliably represent these parametric behaviors.

Plasma technologies serve as an example of parametric dynamical systems. For instance, in a Hall thruster, the dominant plasma phenomena governing the device's global dynamics change as the "self-sustained electric field" parameter varies over one characteristic cycle of the system. In addition, changing the control parameters, such as the intensity of the externally applied magnetic field or the applied discharge voltage can majorly alter the dynamics of the thruster.

In the present Chapter, the aim is to demonstrate that Phi Method can effectively learn the parametric dynamics so as to provide accurate predictions of systems' behavior over unseen parameter spaces. We describe how the Phi Method's implementation from Chapter 5 can be adapted for the purpose of learning parametric dynamics. These adaptations to the Phi Method's formulation yield the "parametric Phi Method" and the "ensemble Phi Method" techniques. The test cases of fluid flow past a cylinder and the Hall thruster plasma discharge, discussed in Chapter 5, are adopted here as well. For the fluid flow test case, the performance of parametric Phi Method is compared, for reference, against that of parametric OPT-DMD, which was introduced in subsection 3.2.3.

Section 6.1 describes the parametric Phi Method and its variant, ensemble Phi Method. Sections 6.2 and 6.3 evaluate the performance of parametric/ensemble Phi method in the fluid flow and the plasma test cases, respectively. Section 6.2 also presents a comparison between the performance of the ROMs from parametric Phi Method and parametric OPT-DMD. The main takeaways from the discussions in this Chapter are provided in section 6.4.

6.1. Description of the Phi Method's implementation for parametric dynamics learning

Like most dynamical systems, plasmas can be described by a set of partial differential equations. The plasma PDEs capture the evolution of key physical quantities, such as the densities, velocities, and temperatures of the plasma species (electrons and ions) as well as possibly the electromagnetic fields. A key challenge in accurately describing the plasma dynamics through PDEs arises from the inherent variability in plasma behavior under different conditions and operating regimes, which serve as additional parameters on top of the system's state variables. Parametric PDEs address this by introducing parameters to represent diverse conditions. These parameters can represent physical or geometric properties, the environmental factors, and/or the external control inputs. Particularly in plasma devices, the important parameters can be the operating conditions such as the applied voltage, and the magnitude and topology of the externally applied magnetic field, or the intrinsic state variables like the average operating density and temperature, the ions' type (hence, their mass, ionization energy, and collision frequencies), and the material properties of the surfaces of the device that are in contact with the plasma.

Despite the near real-time capabilities of most data-driven ROMs to generate forecasts, their often-limited adaptability to parametric variations presents a significant challenge to achieving broader generalizability for the models. As a result, parameter embedding emerges as a crucial consideration in data-driven ROMs. Otherwise, when faced with variation in parameters, the model needs to be rebuilt through training on the data related to the new parameter, which serves as a major drawback for any practical reduced-order model intended for real-world applications.

In subsection 3.2.3, we presented a parametric extension to OPT-DMD, obtaining the "parametric OPT-DMD" method. Here, we describe how Phi Method can be adapted for data-driven reduced-order modeling of parametric dynamical systems. Recalling from subsection 3.2.3, a parametric dynamical system is a system described by the following general-form differential equation

$$\frac{d\boldsymbol{x}(t)}{dt} = \boldsymbol{\mathcal{F}}(\boldsymbol{x}(t); \boldsymbol{\mu}). \qquad \text{Eq. 6.1}$$

In Eq. 6.1, $\boldsymbol{x}$ is the state vector and $\boldsymbol{\mu}$ is the "parameter vector" containing the individual parameters (μ_j with $j = 1, 2, \dots, n_p$) that the system's dynamics is dependent on, hence, $\boldsymbol{\mu} = \left[\mu_1, \mu_2, .., \mu_{n_p}\right]^T$.

We would remind from section 5.1 that Phi Method discovers discretized differential equations describing a system's evolution ($\boldsymbol{\mathcal{F}}$ in Eq. 6.1). The discretized dynamics is encapsulated within the linear operator Φ

$$\boldsymbol{x}_i^{k+1} = \boldsymbol{F}(\boldsymbol{y}_i^k)\Phi. \qquad \text{Eq. 6.2}$$

In Eq. 6.2, $\boldsymbol{x}_i^{k+1}$ represents the state vector of the system on node i at time step $k+1$, $\boldsymbol{y}_i^k$ encompasses the state vectors on node i and on its neighboring nodes at time step k, and $\boldsymbol{F}$ is the

library of candidate terms that can include linear and nonlinear observables ($\boldsymbol{f}_l(\boldsymbol{y_i})$, $l = 1, 2, \dots L$) of the state vector variables. As was discussed in detail in section 5.1, Eq. 6.2 is assumed to be valid for every node within the computational domain ($i = 2, 3, \dots, N_g - 1$, for a 1D domain) and for all times ($k = 1, 2, \dots.$).

To train a Phi Method model, we pointed in section 5.1 that we evaluate the $\boldsymbol{x}_i^{k+1}$ and the $\boldsymbol{F}(\boldsymbol{y}_i^k)$ for all the nodal points i and at all the time steps k within the training interval and store the values in two matrices X and Θ, respectively (Eq. 5.7). In case the number of training data points is very large, we can consider a randomly sampled subset of the points in space and time for the model training. Matrix Φ is determined by performing a least-squares regression as per Eq. 6.3.

$$\Phi = \underset{\Phi}{\arg\min} \|\mathrm{X} - \Theta\Phi\|_2 \approx \Theta^{\dagger} X \qquad \text{Eq. 6.3}$$

The parametric extension to Phi Method ("**parametric Phi Method**" implementation), which enables the discovery of the dynamical systems given by Eq. 6.1, is straightforward. The only modification to what was described above involves incorporating additional observables into the library of candidate terms ($\boldsymbol{F}$) such that

$$\boldsymbol{x}_i^{k+1} = \boldsymbol{F}(\boldsymbol{y}_i^k, \boldsymbol{\mu})\Phi_{\mathrm{Param}}. \qquad \text{Eq. 6.4}$$

The additional observables can be expressed as various functions $\boldsymbol{h}_j(\boldsymbol{y_i}, \boldsymbol{\mu})$, $j = 1,2, \dots, J$, of both the parameters and the state variables of the system. Therefore,

$$\boldsymbol{F}(\boldsymbol{y_i}, \boldsymbol{\mu}) = \left[\boldsymbol{f}_1(\boldsymbol{y_i}),\ \boldsymbol{f}_2(\boldsymbol{y_i}), \dots, \boldsymbol{f}_L(\boldsymbol{y_i}),\ \boldsymbol{h}_1(\boldsymbol{y_i}, \boldsymbol{\mu}),\ \boldsymbol{h}_2(\boldsymbol{y_i}, \boldsymbol{\mu}), \dots, \boldsymbol{h}_J(\boldsymbol{y_i}, \boldsymbol{\mu})\right]. \qquad \text{Eq. 6.5}$$

Another approach to use Phi Method for reduced-order modelling of dynamical systems is the so-called **"ensemble Phi Method"** approach. Ensembling techniques are commonly used for data-driven and machine-learning-based ROM developments [168][183][235]. Ensembling is the practice of aggregating multiple individual models to achieve an improved performance. By leveraging the diversity among the individual learned models, the ensembling approach aims to mitigate overfitting, to enhance generalization, and to increase the models' robustness to varying data patterns.

In the ensemble Phi Method, instead of training a single model on the complete dataset that covers the whole parameter space, we train individual models on the data that correspond to each set of values of the parameter vector within our training set ($\boldsymbol{\mu}_s$, subsection 3.2.3). This leads to S independent models, with S being the total number of realizations (sets of values) of the parameter vector $\boldsymbol{\mu}$ in the training dataset. To aggregate the resulting models, we compute the average over S number of coefficient matrices, Φ_s, $s = 1, 2, \dots S$, associated with each model, i.e., $\Phi_{\mathrm{Ens}} = \frac{1}{S}\sum_{s=1}^{S} \Phi_s$. The average coefficient matrix (operator) Φ_{Ens} represents the ensemble Phi Method model, which is then used for the predictions as per Eq. 6.4, with Φ_{Param} replaced with Φ_{Ens}.

6.2. Comparative evaluation of Phi Method's performance against parametric OPT-DMD in the canonical problem of a flow past a cylinder

The predictive performance of parametric ROMs from Phi Method are assessed in this section in comparison against the performance of the ROMs from parametric OPT-DMD. Before proceeding further, however, it is important to highlight an inherent difference between the parametric extensions to Phi Method and OPT-DMD.

In the PDE-discovery techniques, to which Phi Method belongs, the parameters are directly incorporated into the model. This explicit treatment of parameters allows PDE-based models to naturally adapt to changes in parameters. It also provides a more direct and generalizable representation of how the system may respond to different parameter values. On the contrary, whereas POD-based models, such as those from OPT-DMD, have proven successful in capturing dominant modes in the data and reducing the data dimensionality [207][210][236], they may not be the optimal choice when the focus would be on explicitly modeling parametric dependencies. Indeed, in extending the DMD algorithm to accommodate parametric variations, a common approach entails employing interpolation techniques between the modes associated with the neighboring parameter values [166][227] (see also subsection 3.2.3). Nonetheless, the interpolation between the modes assumes a smooth transition in the parameter space among the different parameter values. Accordingly, in the cases where a system's behavior undergoes significant changes across the parameter space, this interpolation approach to realize parametric POD-based ROMs will become impractical because of the substantial training dataset that will be needed to adequately sample the parameter space. This remark will be further clarified through the discussions in this section, as well as those in section 6.3.

6.2.1. Description of the problem setup

The fluid flow test case represents the evolution of a 2D flow around a stationary cylinder, which is characterized by the recurring shedding of vortices in the downstream wake of the cylinder. The vortex shedding phenomenon is strongly influenced by the Reynolds number of the flow [237] – beyond a certain Reynolds number, the flow becomes unstable as it passes over the cylinder and small perturbation of the flow leads to the wake instability and the periodic formation of the vortices. In the intermediate range of Reynolds numbers, the flow is laminar, whereas it transitions to a turbulent state once the Reynolds number exceeds a specific threshold [237].

The dynamics of the vorticity field (Ω) in the flow-past-a-cylinder problem is governed by the PDE given by Eq. 5.12, provided below again for convenience.

$$\frac{\partial \Omega}{\partial t} + (\boldsymbol{V}.\boldsymbol{\nabla})\Omega = \frac{1}{Re}\nabla^2\Omega, \qquad \text{Eq. 5.12 (repeated)}$$

Eq. 5.12 represents a parametric dynamical system with the parameter being the freestream's Reynolds number (Re).

As was pointed out in subsection 5.2.2.1, the ground-truth simulation data are generated for this test problem using the "ViscousFlow.jl" software package [234]. Particularly, for the data-driven modelling of the parametric dynamics, flow simulations were performed across an intermediate range of Reynolds numbers, where the flow exhibits the oscillatory vortex-shedding pattern while remaining laminar.

The simulations' setup and conditions are overall the same as those described in subsection 5.2.2.1. Briefly, the computational domain is 0.06×0.04 m 2D box discretized using 300×200 nodes. A cylinder of 0.01-m diameter is included in the domain, with its center positioned 0.01 m away from the left boundary. The free stream of the flow enters the domain at a specific pre-determined Reynolds number. Each simulation was run for a total duration of 100 s, with the flow data collected at 0.1 s intervals.

The simulations were performed over a broad set of Reynolds numbers, namely, $Re \in$ {100, 150, 175, 200, 250, 275, 300, 350, 400, 425, 450, 500}. All of the data corresponding to this broad range of Reynolds numbers were divided into the training and the testing sets in two fashions: one, $Re_{train} \in$ {100, 200, 300, 400, 500} and $Re_{test} \in$ {150, 250, 350, 450}, and two, $Re_{train} \in$ {100, 150, 200, 250, 300, 350, 400, 450, 500} and $Re_{test} \in$ {175, 275, 425}. The first division of the data is referred to as "**dataset 1**" or **DS1**. The second division, which represents finer increments in the parameter space for the training, is called "**dataset 2**" or **DS2**. In general, we have used DS1 for the training and testing of the ROMs in the fluid flow test case unless it is explicitly stated otherwise.

For either dataset, the initial 45 seconds of the simulations, representing the transient state of the fluid system (subsection 5.2.2.1), were excluded from either the training or the test data. This was to enable deriving reliable ROMs from the OPT-DMD application to the data. It is recalled from the discussions in subsections 3.6.2 and 5.2.2.2 that the POD-based methods, including OPT-DMD, work reliably only when applied to the (quasi) steady-state data from a system.

The parametric OPT-DMD was trained on the time-series snapshots of the vorticity field belonging to the training parameter set. We would highlight that, in the parametric DMD, determining the optimal number of ranks in the underlying SVD (or, equivalently, the number of DMD modes to retain) involves a compromise between two aspects: (i) the percentage of information captured in the resulting low-rank representation on the training data points, and (ii) the accuracy of the interpolation in the parameter space on the test data points. In this regard, retaining a higher number of DMD modes improved the closeness of the predictions of an individual OPT-DMD ROM to the original training data corresponding to each specific parameter. However, we observed that the quality of the interpolation between the DMD bases degraded as the number of DMD modes increased. Having identified the optimal trade-off between the above two effects, we retained 10 DMD modes for this test case example.

In the case of Phi Method, the library of terms (Θ) can include both linear and nonlinear observables of the system's state vector as well as the involved parameter(s), which is the Re for the present test case. The library terms considered for the fluid flow test case are seen from Eq. 6.6. In subsection 5.3.2.1, we presented a thorough analysis of the impacts of the size of the library and the inclusion of extra terms. Hence, we have focused here only on the minimum set of dynamical terms that are essential to properly represent the dynamics of fluid flow system. Accordingly, Eq. 6.4 becomes

$$\Omega_{i,j}\big|_{k+1} = \begin{bmatrix} \begin{bmatrix}\Omega_{i,j-1}\\ \Omega_{i-1,j}\\ \Omega_{i,j}\\ \Omega_{i+1,j}\\ \Omega_{i,j+1}\end{bmatrix}^T & \begin{bmatrix}V_{x_{i,j-1}}\\ V_{x_{i-1,j}}\\ V_{x_{i,j}}\\ V_{x_{i+1,j}}\\ V_{x_{i,j+1}}\end{bmatrix}^T & \begin{bmatrix}V_{y_{i,j-1}}\\ V_{y_{i-1,j}}\\ V_{y_{i,j}}\\ V_{y_{i+1,j}}\\ V_{y_{i,j+1}}\end{bmatrix}^T & \begin{bmatrix}(\Omega V_x)_{i,j-1}\\ (\Omega V_x)_{i-1,j}\\ (\Omega V_x)_{i,j}\\ (\Omega V_x)_{i+1,j}\\ (\Omega V_x)_{i,j+1}\end{bmatrix}^T & \begin{bmatrix}(\Omega V_y)_{i,j-1}\\ (\Omega V_y)_{i-1,j}\\ (\Omega V_y)_{i,j}\\ (\Omega V_y)_{i+1,j}\\ (\Omega V_y)_{i,j+1}\end{bmatrix}^T & \begin{bmatrix}\frac{1}{Re}\Omega_{i,j-1}\\ \frac{1}{Re}\Omega_{i-1,j}\\ \frac{1}{Re}\Omega_{i,j}\\ \frac{1}{Re}\Omega_{i+1,j}\\ \frac{1}{Re}\Omega_{i,j+1}\end{bmatrix}^T \end{bmatrix}_k \Phi_{\text{Param}}. \quad \text{Eq. 6.6}$$

Before training the Phi Method models, each flow property was normalized with respect to its spatiotemporally maximum value over the *entire* dataset that corresponds to all the simulated parameters. The left-hand side of Eq. 6.6 as well as the terms in the library were then evaluated on various nodes ($i = 2, 3, \dots, N_i - 1,\ j = 2, 3, \dots, N_j - 1$) and time steps ($k = 1, 2, \dots m - 1$) using the normalized data across the training parameter space. $\frac{1}{Re}$, included in the last term within the library, captures the parametric dependency in the resulting PDE that describes the evolution of the vorticity. The optimal matrix of coefficients Φ_{Param} associated with the parametric Phi Method ROM was obtained by performing a regression on both sides of Eq. 6.6.

We also derived an ensemble Phi Method ROM for the fluid flow test case. For the ensemble ROM, the individual models were trained separately on the simulation data corresponding to each Reynolds number of dataset 2 (DS2), i.e., $Re_{train} \in$ {100, 150, 200, 250, 300, 350, 400, 450, 500}. This resulted in S = 9 independent models. Taking an average over the coefficient matrices of the individual models, the coefficient matrix (operator), Φ_{Ens}, of the ensemble ROM was obtained. The library terms used to drive the ensemble ROM were the same as those shown in Eq. 6.6.

6.2.2. Results

The predictions of the trained parametric ROMs on the test sets of the Re parameter are presented and discussed in this subsection.

Fig. 6.1 shows the forecasts of the parametric Phi Method and the parametric OPT-DMD ROMs over the test Re parameters of DS1 ($Re_{test} \in$ {150, 250, 350, 450}). The results are presented in terms of the temporal evolution of the spatially averaged vorticity (Ω_{mean}) and the local value of the vorticity field (Ω_{mid}) in comparison to the ground-truth data from the simulations. The local vorticity value corresponds to the value at the mid-location of the domain (x = 0.03 m and y = 0.02 m). Fig. 6.1(right column) additionally presents the time variations of the ROMs' loss factor. The loss factor quantifies the extent of error (or deviation) in the models' predictions relative to the true data. The loss is calculated as was described in subsection 5.2.2.2.

The plots of Ω_{mid} (Fig. 6.1(middle column)) evidently show that the predictions of the parametric Phi Method ROM closely align with the true data. In contrast, the predictions of the parametric OPT-DMD ROM exhibit a relatively high level of discrepancy. According to Fig. 6.1(right column), the loss associated with the forecasts of the parametric Phi Method ROM is, on average, about four times smaller than that of the parametric DMD ROM across the test parameters.

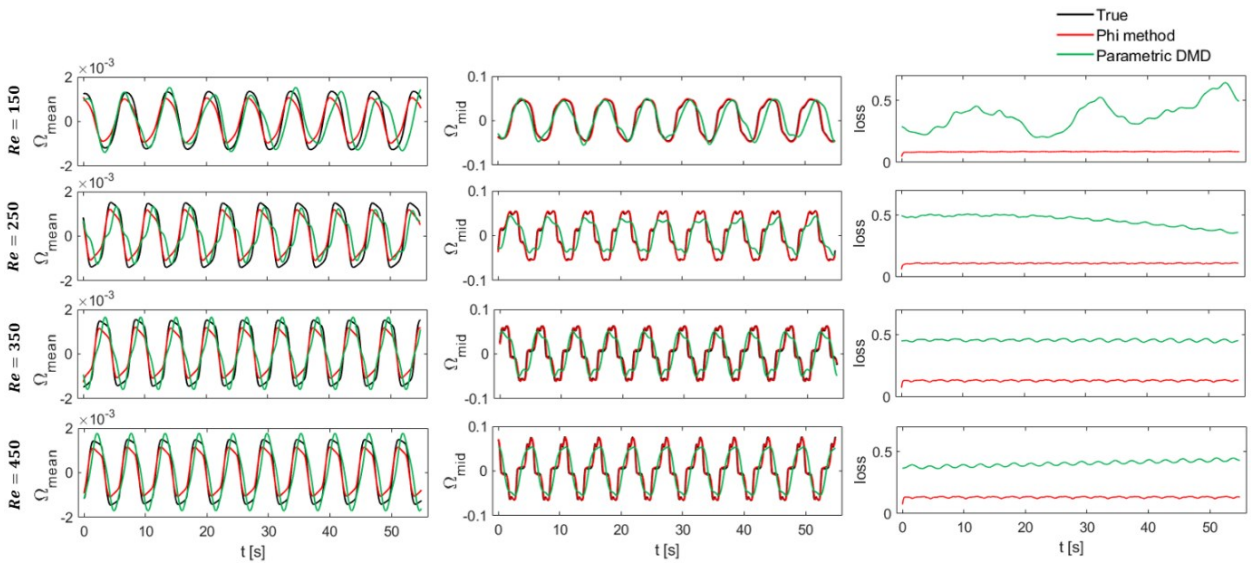


Fig. 6.1 Comparison of the predictions from the parametric Phi Method and the parametric OPT-DMD ROMs against the ground-truth data for the fluid flow test case across the test Reynolds numbers belonging to DS1; time evolutions of the spatially averaged normalized vorticity (**left column**), and local values of the normalized vorticity at the mid-location within the simulation domain (**middle column**). (**right column**) Time evolutions of the loss factor calculated over the entire domain.

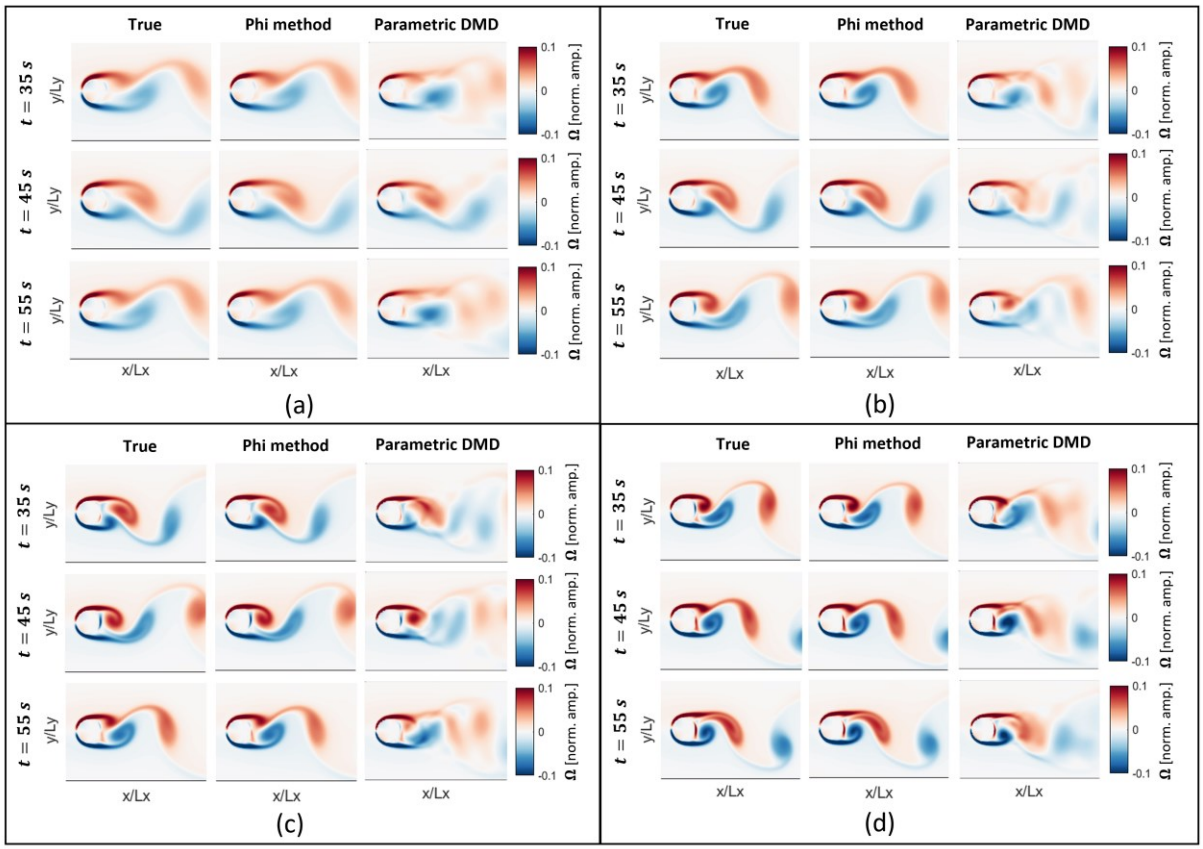


Fig. 6.2 Comparison of the predictions from the parametric Phi Method and the parametric OPT-DMD ROMs against the ground-truth data for the fluid flow test case across the test Reynolds numbers belonging to DS1; 2D snapshots of the normalized vorticity field at three different sample time instants for the test parameters of (a) Re = 150, (b) Re = 250, (c) Re = 350, and (d) Re = 450.

The fully reconstructed 2D snapshots of the normalized vorticity field from the parametric Phi Method and the parametric OPT-DMD ROMs at three different time instants are illustrated in Fig. 6.2 for the test Reynolds numbers of DS1. Furthermore, in Fig. 6.3, the absolute-difference maps between the predicted snapshots and the respective “true” snapshots are plotted.

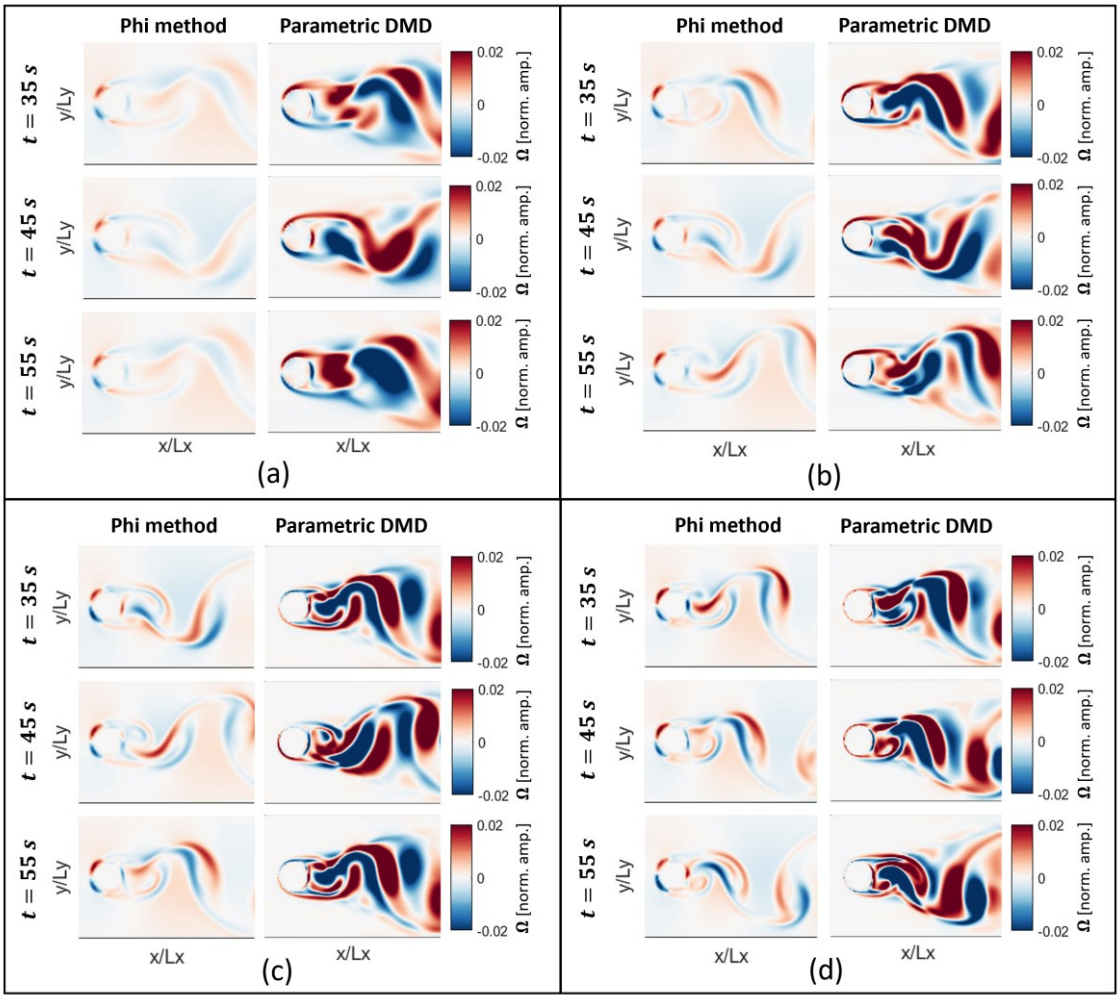


Fig. 6.3 Comparison of the predictions from the parametric Phi Method and the parametric OPT-DMD ROMs for the fluid flow test case; absolute-difference plots between the true and the predicted normalized vorticity snapshots at three time instants for the test parameters of DS1, namely, (a) Re = 150, (b) Re = 250, (c) Re = 350, and (d) Re = 450.

The close resemblance between the Phi Method ROM’s snapshots and the ground-truth further demonstrates the ability of Phi Method to reliably capture the parametric dependency in the dynamics and, hence, to provide successful predictions of the desired quantity across the parameter space. Contrarily, the parametric OPT-DMD does a rather unsatisfactory job in embedding the parametrization underlying the dynamics. This is evident from the relatively significant discrepancies observed between its predicted snapshots and the ground-truth. In this respect, although we noticed in subsection 5.2.2.2 that the OPT-DMD exhibited a slightly better performance compared to Phi Method toward time forecasting of the fluid flow system at a fixed parameter value, the present results from Fig. 6.2 and Fig. 6.3 underline that the parametric Phi Method provides substantially more accurate predictions when the defining parameter of a system undergoes variations.

In Fig. 6.4, we have compared the true and the interpolated spatial modes corresponding to the first four leading DMD bases across the test Re parameters of DS1. The true bases (modes) are obtained by applying OPT-DMD directly to the data associated with each specific test Reynolds number. It is evident from the plots in Fig. 6.4 that, in some cases, the interpolated modes are significantly different from their true counterparts.

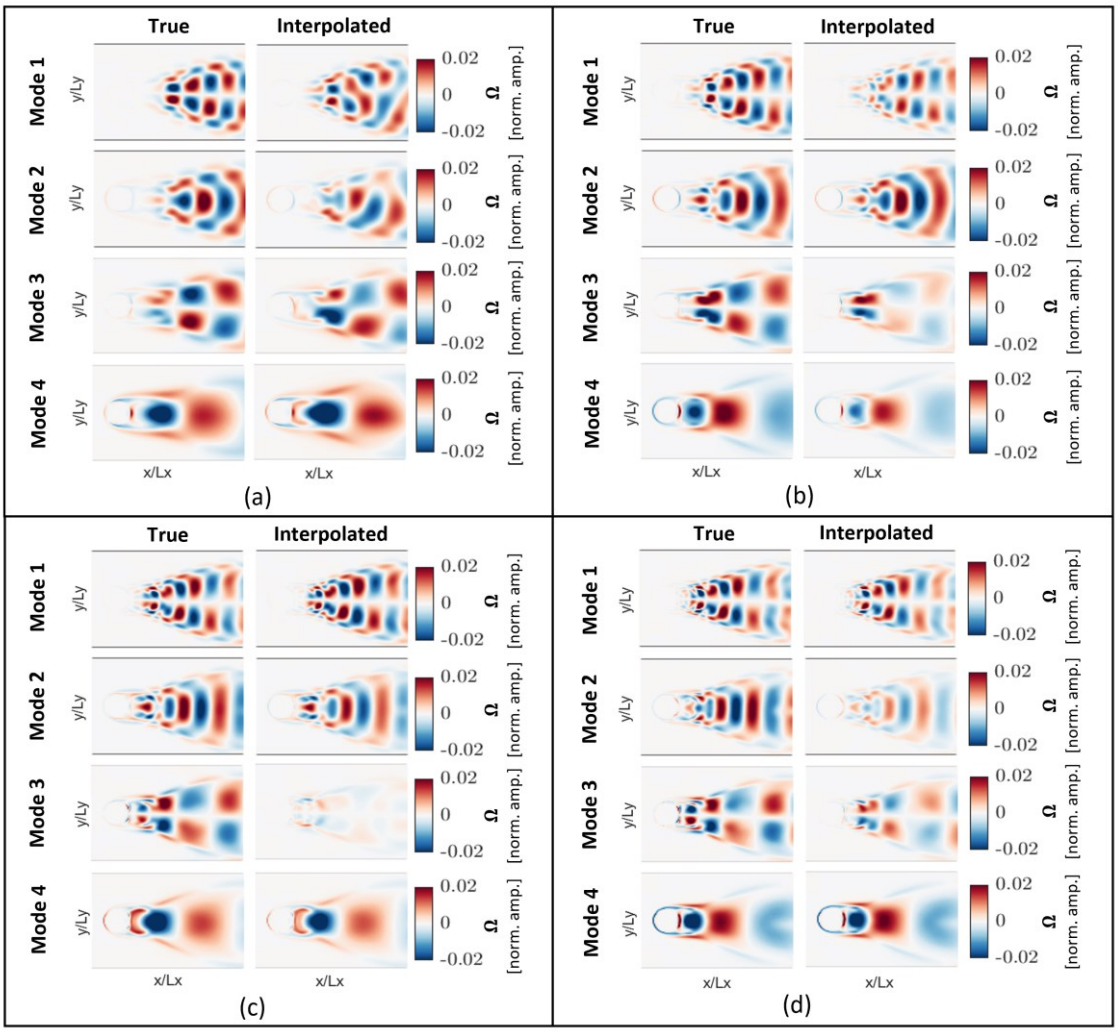


Fig. 6.4 Visualization of the first four interpolated DMD modes associated with the parametric OPT-DMD for the fluid flow test case over the test Reynolds numbers of DS1. The interpolated modes are compared against the corresponding true DMD modes. (a) Re = 150, (b) Re = 250, (c) Re = 350, and (d) Re = 450.

We would emphasize that the “poor” performance of the parametric OPT-DMD in the fluid flow test case is not a limitation of the OPT-DMD itself as a data-driven method, but rather it is related to the limited accuracy of the interpolation between the DMD bases. In this regard, it should be noted that the optimality of the DMD modes is not preserved through the interpolation. In other words, the resulting interpolated modes may no longer represent the best low-rank approximation of the system that satisfies the linearity of the time dynamics.

When the variation of the DMD modes in the parameter space occurs on a non-smooth and strongly nonlinear manifold, or when the two parameters between which we are interpolating the DMD modes

are too far apart, the interpolated modes can exhibit a larger extent of deviation from their optimality condition. This argument is further supported by the results shown through Fig. 6.5 to Fig. 6.7. To obtain these results, the parametric OPT-DMD, in addition to the parametric Phi Method, are trained on dataset 2 (DS2). As was pointed out in subsection 6.2.1, DS2 corresponds to more samples of the parameter space associated with the fluid system. The training set of DS2 includes the simulation data for $Re_{train} \in$ {100, 150, 200, 250, 300, 350, 400, 450, 500}. The ROMs are tested on the set of Reynolds numbers corresponding to $Re_{test} \in$ {175, 275, 425}.

Fig. 6.5 shows the predicted time evolutions of the Ω_{mean} and the Ω_{mid} in comparison against the ground-truth. The figure also presents the time evolutions of the loss factors associated with the models' predictions. Fig. 6.6 compares the predicted and the true snapshots of the vorticity field at three sample time instants. Fig. 6.7 shows a comparison between the true and the interpolated spatial modes corresponding to the first four leading DMD bases across the test Re parameters of DS2.

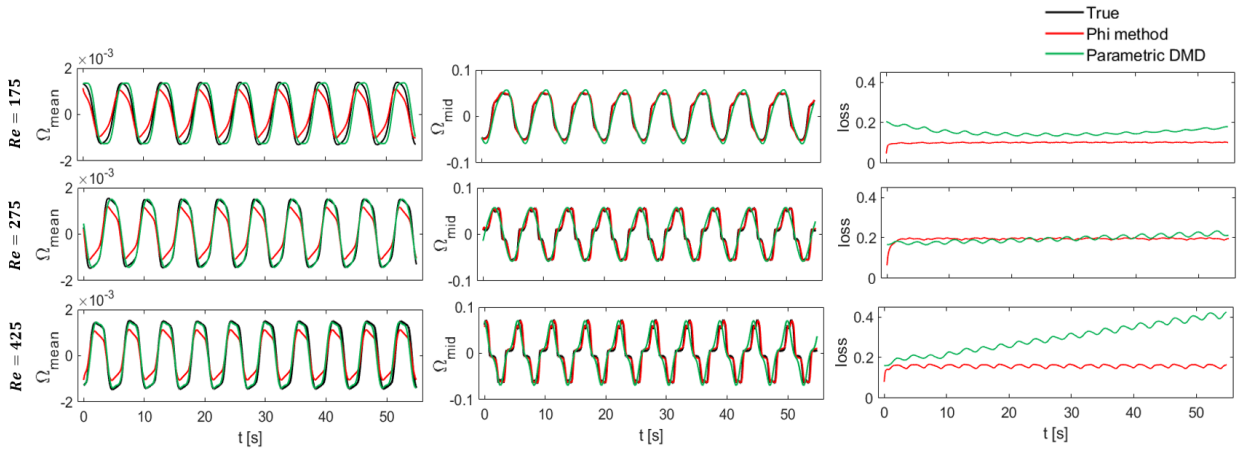


Fig. 6.5 Comparison of the predictions from the parametric Phi Method and the parametric OPT-DMD ROMs against the ground-truth data for the fluid flow test case across the test Reynolds numbers belonging to DS2; time evolutions of the spatially averaged normalized vorticity (**left column**), and local values of the normalized vorticity at the mid-location within the simulation domain (**middle column**). (**right column**) Time evolutions of the loss factor calculated over the entire domain.

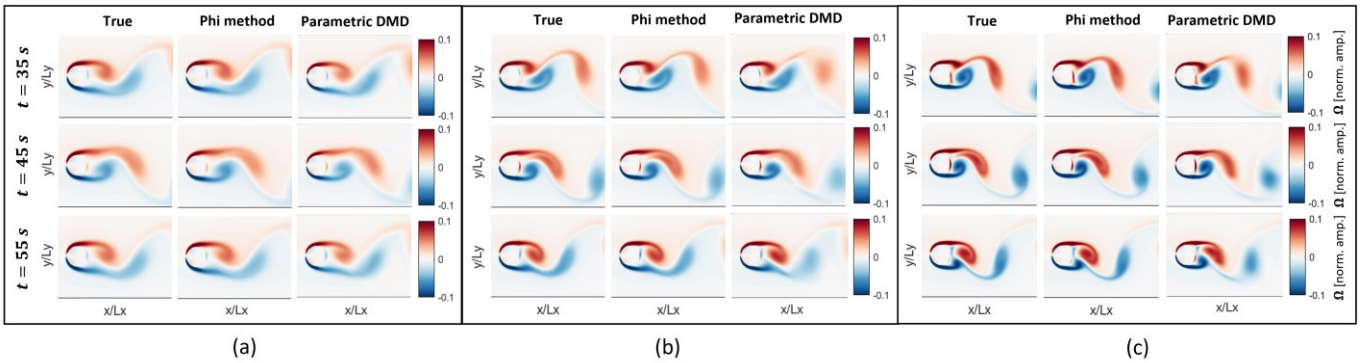


Fig. 6.6 Comparison of the predictions from the parametric Phi Method and the parametric OPT-DMD ROMs against the ground-truth data for the fluid flow test case across the test Reynolds numbers belonging to DS2; 2D snapshots of the normalized vorticity field at three different time instants for the test parameters of (a) $Re = 175$, (b) $Re = 275$ and (c) $Re = 425$.

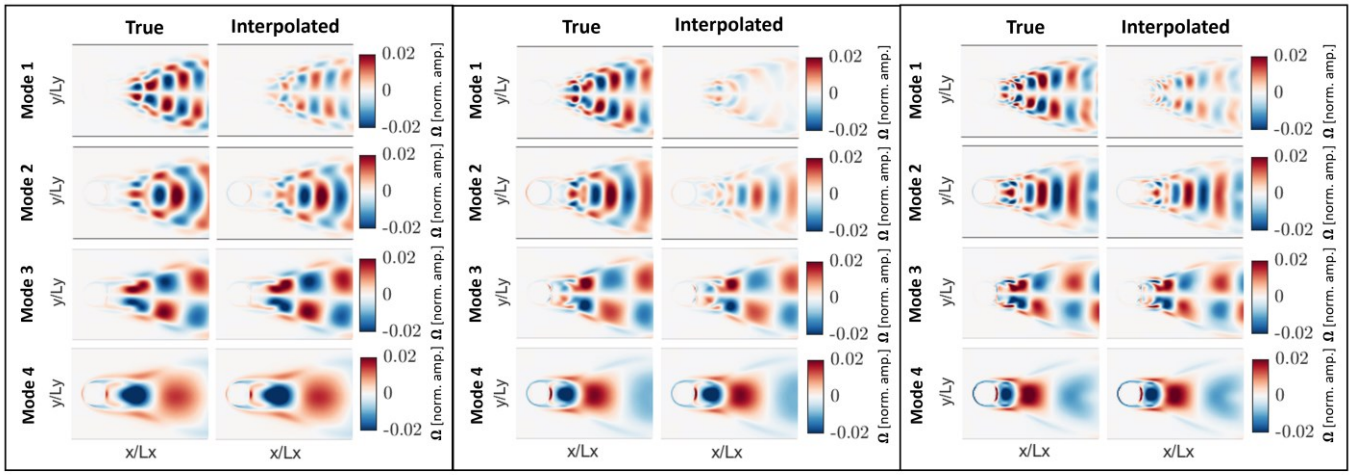


Fig. 6.7 Visualization of the first four interpolated DMD modes associated with the parametric OPT-DMD for the fluid flow test case over the test Reynolds numbers of DS2. The interpolated modes are compared against the corresponding true DMD modes. (a) $Re = 175$, (b) $Re = 275$ and (c) $Re = 425$.

Focusing on the parametric OPT-DMD ROM, the above outcomes from the method's training and testing on DS2 show that a smaller increment between the training parameters that sample the parameter space of the system improves the quality of the model. Referring to Fig. 6.7, in particular, and comparing the plots against those in Fig. 6.4, we see that the discrepancy between the true and the interpolated spatial DMD modes is reduced when the model is trained on DS2. We underline that a "sufficient" number of parameter samples is case-specific and depends on the extent to which the DMD modes vary within the parameter space of a certain system.

Besides expanding the training dataset, employing more advanced interpolation schemes may also improve the quality of the parametric OPT-DMD ROM by allowing for the interpolation to be reliably performed between more widely spaced parameters. These advanced interpolation schemes include techniques that utilize an intermediary linear subspace called "Grassman manifold" to perform the interpolation and then transfer back to the subspace of the POD modes [238].

In Fig. 6.8, the spatiotemporally averaged losses of the predictions from the parametric Phi Method and the parametric OPT-DMD ROMs with respect to the ground-truth data of the fluid flow test case are plotted for various Reynolds numbers. The results are presented for the ROMs trained on DS1 and on DS2.

Referring to Fig. 6.8, the spatiotemporally averaged loss of the Phi Method ROM is seen to remain identical between the training on DS1 and DS2 across the range of Re values. Additionally, for Phi Method, the ROM's loss level remains similar among the training and the test data. However, in the case of the parametric OPT-DMD, the loss in predicting the test data is considerably higher than the loss associated with the reconstruction of the training data. Moreover, it is again observed that the number of samples of the parameter space that are included in the training greatly influences the disparity level between the loss of the parametric OPT-DMD ROM on the training and on the test data.

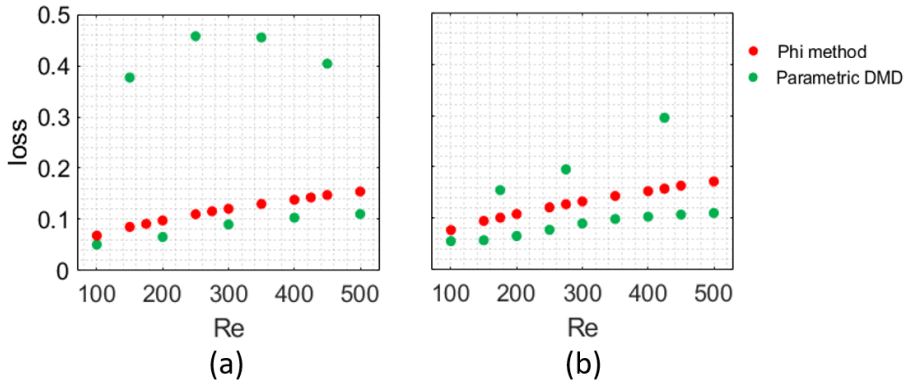


Fig. 6.8 Comparison of the predictions from the parametric Phi Method and the parametric OPT-DMD ROMs for the fluid flow test case; plots of the time-averaged (mean) values of the loss factor vs. the Reynolds number from the ROMs trained on (a) dataset 1 (DS1), and (b) dataset 2 (DS2). The loss factor is calculated over the entire domain.

The difference between the accuracy of the parametric Phi Method ROM and the parametric DMD ROM lies in a fundamental difference between these two techniques. In fact, by identifying the local relationships and correlations within the data, Phi Method approximates the underlying PDE describing the data. Consequently, this method, like any other data-driven PDE-discovery technique, enables the explicit incorporation of the parametric dependencies present in the system into the model. In contrast, POD-based approaches, such as DMD, do not explicitly account for the parametric dependencies in their formulation. They identify the modes based on the covariance of the dataset without differentiating between the different parameter values. Accordingly, for the parametric DMD, although a distinct set of modes is associated with each parameter, there is no explicitly quantified relationship between the specific parameter regimes and their corresponding modes. This “implicit” treatment in parametric DMD limits the model’s ability to capture the intricate relationships between the parameters and the system’s response.

We have illustrated in Fig. 6.9 the rectangular rearrangement of the matrices of coefficients ($\Phi_{\text{Param}} \in \mathrm{R}^{30\times1}$, or $\Phi_{\text{Param}} \in \mathrm{R}^{5\times6}$ after rearrangement) for the parametric Phi Method ROM when trained on DS1 and on DS2. It is noticed that the matrices (operators) of the two models are almost identical.

The last column of the matrices in Fig. 6.9 captures the parametric dependency of the dynamics on the Reynolds number, as was represented by the right-hand-side term in the theoretical relation describing the dynamics of the vorticity (Eq. 5.12). Discretizing this term in Eq. 5.12 using a finite difference scheme, we obtain

$$\frac{1}{Re}\nabla^2\Omega = \frac{1}{\Delta x^2}\left(\frac{\Omega^k_{i+1,j}}{Re} - \frac{2\Omega^k_{i,j}}{Re} + \frac{\Omega^k_{i-1,j}}{Re}\right) + \frac{1}{\Delta y^2}\left(\frac{\Omega^k_{i,j+1}}{Re} - \frac{2\Omega^k_{i,j}}{Re} + \frac{\Omega^k_{i,j-1}}{Re}\right) \qquad \text{Eq. 6.7}$$

where, Δx and Δy denote the cell sizes of the grid discretizing the domain along the x and y directions, respectively.

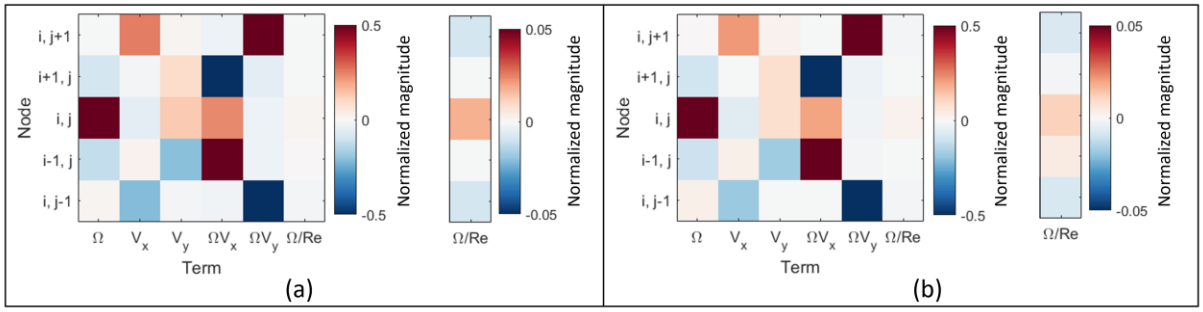


Fig. 6.9 Rearranged normalized representations of the coefficients matrices (Φ_{Param}s) of the parametric Phi Method ROM for the fluid flow test case; (a) from training on DS1, and (b) from training on DS2.

Given that $\Delta x = \Delta y$ for the data used to train the ROMs, we could expect a nearly uniform distribution of coefficients for the $\frac{\Omega}{Re}$ term across the surrounding nodes of the node (i, j). However, looking at Fig. 6.9, it is interesting that this is not the case. In fact, the magnitude of the coefficients for $\frac{\Omega^k_{i,j-1}}{Re}$ and $\frac{\Omega^k_{i,j+1}}{Re}$ are larger than those corresponding to $\frac{\Omega^k_{i-1,j}}{Re}$ and $\frac{\Omega^k_{i+1,j}}{Re}$. This disagreement between the coefficients matrices of the two Phi Method ROMs visualized in Fig. 6.9 and what is prescribed by the finite differencing can imply that the identified stencil by Phi Method is an alternative optimal stencil for the discretization of the governing PDE. This must, of course, be considered in conjunction with the coefficients of the other library terms and their associated discretization stencils.

We also point out that the rest of the columns of the visualized matrices in Fig. 6.9 (those that do not reflect the parametric dependence on the Re) are closely aligned with the Φ matrices obtained from application of "basic" Phi Method to a single-parameter dataset as was discussed in subsection 5.2.2.2 (Fig. 5.8(b)).

As a final point of discussion, we recall from subsection 6.2.1 that we also developed an ensemble Phi Method ROM for the fluid flow test case. This was done by averaging individual Φ matrices that were obtained by training Phi Method on single datasets associated with the individual Reynolds number values within DS2. We present in the following the rearranged representations of characteristic matrices corresponding to the ensemble Phi Method model.

Fig. 6.10(a) shows the Φ_{Ens} matrix of the ensemble ROM, which is the mean of the Φ matrices of the individual ROMs. Fig. 6.10(b) illustrates the normalized standard deviations of the coefficients of the library terms across the individual models. From Fig. 6.10(a), the Φ_{Ens} operator is very similar to the Φ_{Param} associated with the parametric Phi Method ROM (Fig. 6.9) in terms of the coefficients learned for the library terms. This implies that, for the fluid flow test case, the ensemble ROM is rather equivalent to the "standard" parametric ROM.

In addition, from Fig. 6.10(b), we notice that, across the individual Phi Method models constituting the ensemble ROM, the coefficients of the library terms that have been found as dominant and important for the dynamics (larger absolute values/darker shades in Fig. 6.10(a)) show very small standard deviations (values toward -1 on logarithmic scale/ colormap toward darker blue shades in Fig. 6.10(b)). This means that the coefficients of the dominant terms are determined with very high confidence.

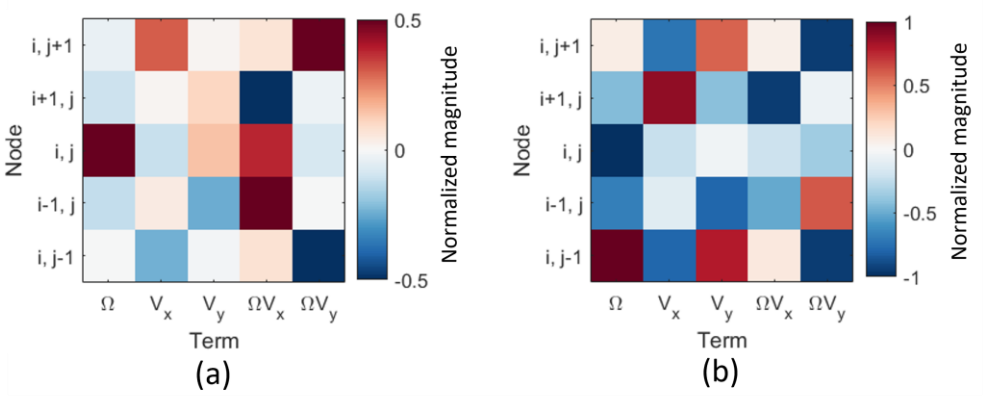


Fig. 6.10 Rearranged normalized representations of the characteristic matrices corresponding to the ensemble Phi Method ROM for the fluid flow test case. (a) Φ_{Ens} matrix (mean of the Φ matrices of the individual constituent models), (b) normalized (by the absolute values of the Φ_{Ens} matrix) standard deviations of the coefficients of the library terms across the individual models. The individual models constituting the ensemble ROM were trained on the dataset 2. The colormap in plot (b) is in logarithmic scale.

Referring to Fig. 6.10, it is highlighted that the term $\frac{\Omega}{Re}$, which captures the dependency of the dynamics on the Re parameter is not shown in the rearranged matrix representations. This is because the coefficients of this terms were found to be much smaller than the coefficients of the other terms in the Phi Method models' libraries.

6.3. Assessment of the cross-parameter predictive performance of Phi Method in the plasma discharge problem

We now look at the performance of Phi Method toward predicting the parametric dynamics in a plasma discharge test case. We will follow the same order of discussions as that in section 6.2: first describing the problem setup, and then presenting and discussing the results. The discussions will revolve around the predictive performance of the parametric ROMs and the interpretability of the associated coefficients matrices (operators).

Before proceeding further, we underline that, for the fluid flow test case of section 6.2, the variability of the problem's dynamics across the parameter space as represented by the Reynolds number was sufficiently smooth so that the interpolation associated with the parametric OPT-DMD would produce meaningful results. For the plasma discharge problem, however, the parametric OPT-DMD was found to not be a suitable choice due to the strong variations observed in the system's behavior (dynamics) across the parameter space. As a result, the parametric OPT-DMD method failed to provide reasonable interpolations of the modes between the parameters. Consequently, in this section, we exclusively present the results from Phi Method.

6.3.1. Description of the problem setup

The adopted test problem represents a 1D azimuthal geometry of a Hall thruster. The plasma in this test case evolves in the presence of externally applied, mutually perpendicular electric (E) and magnetic (B) fields. As was pointed out in subsection 1.1.3 as well, $\mathrm{E} \times \mathrm{B}$ plasma configurations like that in a Hall

thruster are of remarkable scientific interest across various plasma science domains. This is largely because of the rich underlying physics of the $\mathrm{E} \times \mathrm{B}$ plasmas that is characterized by a broad spectrum of instabilities and turbulence that interact with the plasma species. In particular, a long-standing universal question in the physics of $\mathrm{E} \times \mathrm{B}$ plasmas regards the "anomalous" transport of the electrons across the magnetic field (see also section 1.2). Anomalous transport refers to the enhanced mobility of the plasma particles compared to what is predicted by the classical theories. In recent years, numerous studies have pointed to the significant contributions of plasma instabilities to the anomalous transport of the electrons [15][20]. However, a universally applicable model to predict the instabilities and their induced transport is yet to be achieved [72][239][240].

The 1D setup of the plasma test case serves as a simplified but representative geometry that captures instabilities and fluctuations as well as the interactions of these with the plasma particles. The characteristics of the instabilities and their impacts on the plasma vary with the magnitudes of the applied electromagnetic field. As a result, the applied axial electric field (E_x) and the radial magnetic field (B_y) act as the parameters of the dynamical system represented by the adopted plasma discharge configuration. Our plasma discharge problem provides an ideal plasma test case to demonstrate that the Phi Method ROM can capture these parametrizations and to also illustrate the generalizability of the derived relationship(s) from Phi Method to a broad range of parameters.

IPPL-1D PIC code [229][230] was used to simulate the plasma in this adopted test case and so to generate a dataset corresponding to various values of the E_x and the B_y. The simulations' setup follows what was described in subsection 5.3.1 except for a few modifications. An overview of the setup here is provided below.

The simulation domain is a Cartesian section along the azimuthal (z) direction of a Hall-thruster-representative geometry. The z-direction is perpendicular to the directions of the applied electric field, that is along the axial coordinate (x), and the applied magnetic field, that is along the radial coordinate (y).

To ensure that the domain size remains larger than the characteristic length of the instabilities across the entire simulated range of the E_x and the B_y parameters, the azimuthal extent of the domain is increased to 0.02 m compared to the 0.005-m azimuthal extent used for the ground-truth simulation in section 5.3. The simulations here have N_i = 400 computational nodes, with a time step of $\Delta t = 5 \times 10^{-12} s$. The total simulated time for all the simulations is 10 μs. The collected data from the simulations are the relevant plasma properties averaged over every 500 timesteps (2.5 ns). The initial 2 μs of the simulation data during which the system exhibits highly nonlinear transient behaviors is discarded.

Along the azimuthal direction, a periodic boundary condition is applied to the particles motion as well as to the electric potential in order to mimic the periodicity along the azimuth. While the electric potential is resolved only along the azimuthal direction (1D problem setup), the particles' trajectories are traced along all the three dimensions. A finite extent of 0.01 m is assumed along the axial direction to limit the growth of the energy of the system [23]. The particles crossing an axial boundary of the domain are

resampled from their initial distributions and reintroduced into the domain from the opposite axial boundary. The initialization conditions of the simulations are identical to those presented in subsection 5.3.1.

We carried out PIC simulations over a wide range of electric fields, $E_x \in$ {10, 12.5, 15, 17.5, 20, 22.5, 25, 27.5, 30, 32.5, 35, 37.5, 40} kVm^{-1}, at a fixed magnetic field of B_y = 20 mT. Similarly, multiple simulations were performed for various magnetic field values, $B_y \in$ {10, 12.5, 15, 17.5, 20, 22.5, 25, 27.5, 30, 32.5, 35, 37.5, 40} mT, with the electric field being fixed at $E_x = 20\ kVm^{-1}$. The data corresponding to these simulations were used to train and test both parametric and ensemble Phi Method ROMs.

The aim of the test case is to assess the Phi Method's ability to re-discover from the simulations' data the parametric PDE given by Eq. 6.8. This equation describes the spatiotemporal evolution of the electrons' axial flux density (J_{ex})

$$-qB_yJ_{ex} = m_e\frac{\partial(n_eV_{d,ez})}{\partial t} + m_e\frac{\partial(n_eV_{d,ez}^2)}{\partial z} - qn_eE_z + \frac{\partial\Pi_{zz}}{\partial z}. \qquad \text{Eq. 6.8}$$

In Eq. 6.8, $J_{ex} = n_eV_{d,ex}$, where n_e and $V_{d,ex}$ are the electrons' number density and axial drift velocity, respectively. The terms on the right-hand side of Eq. 6.8 represent different force terms that contribute to the electrons' axial flux density. The first and the second terms represent, respectively, the contributions of the temporal inertia and the convective inertia. The third term captures the contribution of the azimuthal instabilities [229], and the fourth term corresponds to the pressure, with $\Pi_{zz} = k_Bn_eT_{ez}$.

According to Eq. 6.8, we have defined the library terms for Phi Method to be $n_eV_{d,ez}$, n_eE_z, n_eT_{ez} and $n_eV_{d,ez}^2$. Although the underlying physics of this plasma test case and, hence, the distribution of the plasma properties in general depends on both the E_x and the B_y, the E_x does not appear explicitly in Eq. 6.8. To capture the explicit parametric dependency on the B_y, nonetheless, we represent the term B_yJ_{ex} on left-hand side of Eq. 6.8 as a variable for which we would like to develop a ROM using Phi Method. In this sense, once the ROM is trained, to obtain the J_{ex} at a certain value of the B_y, we first evaluate the B_yJ_{ex} term from the model and then divide by the B_y value.

We have developed two "standard" parametric Phi Method models with slightly different libraries. The first model, called "parametric model 1 (PM1)", is described by the relation in Eq. 6.9

$$\left[\left(B_y\,J_{ex}\right)_i\right]_k = \left[\begin{bmatrix}(n_eV_{d,ez})_{i-1}\\(n_eV_{d,ez})_i\\(n_eV_{d,ez})_{i+1}\end{bmatrix}^T \begin{bmatrix}(n_eE_z)_{i-1}\\(n_eE_z)_i\\(n_eE_z)_{i+1}\end{bmatrix}^T \begin{bmatrix}(n_eT_{ez})_{i-1}\\(n_eT_{ez})_i\\(n_eT_{ez})_{i+1}\end{bmatrix}^T \begin{bmatrix}(n_eV_{d,ez}^2)_{i-1}\\(n_eV_{d,ez}^2)_i\\(n_eV_{d,ez}^2)_{i+1}\end{bmatrix}^T\right]_k \Phi_{\mathrm{PM1}}. \qquad \text{Eq. 6.9}$$

It is underlined regarding Eq. 6.9 that, for PM1, the left-hand side of the equation and the library on the right-hand side are both at the same time step k, which is in line with Eq. 6.8. This means that the derived relation from PM1 ROM establishes the correlations that exist at each time step among the specified terms in the model.

Whereas PM1 includes the term $n_e V_{d,ez}$, it does not account for the time derivative of this quantity, which does appear in Eq. 6.8. To address this, we include the term $n_e V_{d,ez}$ at both the current time step (k) and at the preceding time step ($k-1$). This modification yields a variant of PM1, denoted as "parametric model 2, PM2" and given by Eq. 6.10

$$\left[\left(B_y J_{ex}\right)_i\right]_k = \begin{bmatrix} \begin{bmatrix} (n_e V_{d,ez})_{i-1} \\ (n_e V_{d,ez})_i \\ (n_e V_{d,ez})_{i+1} \end{bmatrix}_k^T \begin{bmatrix} (n_e V_{d,ez})_{i-1} \\ (n_e V_{d,ez})_i \\ (n_e V_{d,ez})_{i+1} \end{bmatrix}_{k-1}^T \begin{bmatrix} (n_e E_z)_{i-1} \\ (n_e E_z)_i \\ (n_e E_z)_{i+1} \end{bmatrix}_k^T \begin{bmatrix} (n_e T_{ez})_{i-1} \\ (n_e T_{ez})_i \\ (n_e T_{ez})_{i+1} \end{bmatrix}_k^T \begin{bmatrix} \left(n_e V_{d,ez}^2\right)_{i-1} \\ \left(n_e V_{d,ez}^2\right)_i \\ \left(n_e V_{d,ez}^2\right)_{i+1} \end{bmatrix}_k^T \end{bmatrix} \Phi_{\mathrm{PM2}}. \qquad \text{Eq. 6.10}$$

In subsection 5.3.2.1, we assessed the sensitivity of the predictions from Phi Method ROMs to the size of the library for the same plasma test case as adopted here. We observed that the inclusion of additional terms in the library does not significantly affect the model's predictive capability as long as the essential terms are included in the library of candidate terms. However, the presence of unnecessary terms can deteriorate the interpretability of the Phi Method model because the model will no longer be sparse. The inclusion of extra terms can also alter the values of the coefficients found for the essential terms. A sparse model is more likely to be generalizable to various scenarios beyond the training dataset. In any case, it is also noted that when an expert knowledge might be lacking and, hence, it would not be possible to identify the relevant (essential) dynamical terms a-priori, the library size will inevitably need to be large. The expansion of the library in such cases ensures that sufficient flexibility is embedded within the model so that it will be able to effectively capture the correlations present in the data. The lack of model sparsity in the cases of large library terms can then be addressed by using sparsity-promoting techniques that aim to optimize the coefficient matrix Φ in the sparsest possible way so as to effectively eliminate the non-essential terms [101][215]. This aspect will be also discussed in section 7.2.1, where we outline the next steps to further mature Phi Method.

Considering the above arguments, and for the sake of the demonstrations in this section within the plasma discharge test case, we chose to include in the Phi Method's libraries the minimum set of terms which was needed to represent the dynamics.

The parametric Phi Method models PM1 and PM2 were trained on a dataset comprised by the simulation data corresponding to the parameters set of $E_{x,train} \in \{15, 20, 25, 30, 35\}\ kVm^{-1}$ and $B_{y,train} \in \{15, 20, 25, 30, 35\}$ mT. The remaining data from the simulations with the other values of the E_x and the B_y were reserved to be compared against the ROMs' predictions for testing purposes.

For the test case here, ensemble Phi Method ROMs were also developed. Two ensemble models with the libraries corresponding to those for the PM1 and the PM2 were trained on the exact training dataset as that for the "standard" parametric ROMs. The ensemble ROMs were derived using the approach detailed in subsection 6.1. The coefficients matrices (operators) associated with the two ensemble ROMs are denoted as Φ_{EM1} and Φ_{EM2}.

6.3.2. Results

Once the Phi Method models were trained, the derived data-driven relations were used to predict the spatiotemporal evolution of the J_{ex} over the unseen (test) parameter set. The predictions of the PM1

and PM2 as were defined by Eq. 6.9 and Eq. 6.10, respectively, exhibited indistinguishable similarity. Moreover, PM1 predictions were found to be closely representative of the predictions of the ensemble model EM1 as well. This will be evidenced later in this subsection. Therefore, to avoid redundancy in presenting the results and for the clarity of discussions, Fig. 6.11 to Fig. 6.13 present the predictions of the PM1 only.

The predicted time evolutions of the normalized local J_{ex} value at the center of the domain for various values of the applied E_x and B_y belonging to the test parameter set are provided in Fig. 6.11. The predicted signals are compared against the ground-truth data from the PIC simulations. Additionally, in Fig. 6.12 and Fig. 6.13, the complete spatiotemporal maps of the J_{ex} as predicted by the PM1 are compared against the "true" 2D maps from the PIC simulations for the various test values of the E_x and the B_y, respectively. For all the results shown, the normalization factor for J_{ex} is $6.0625 \times 10^{24}\ m^{-2}s^{-1}$, corresponding to the maximum electron flux density across various E_x and B_y parameter values.

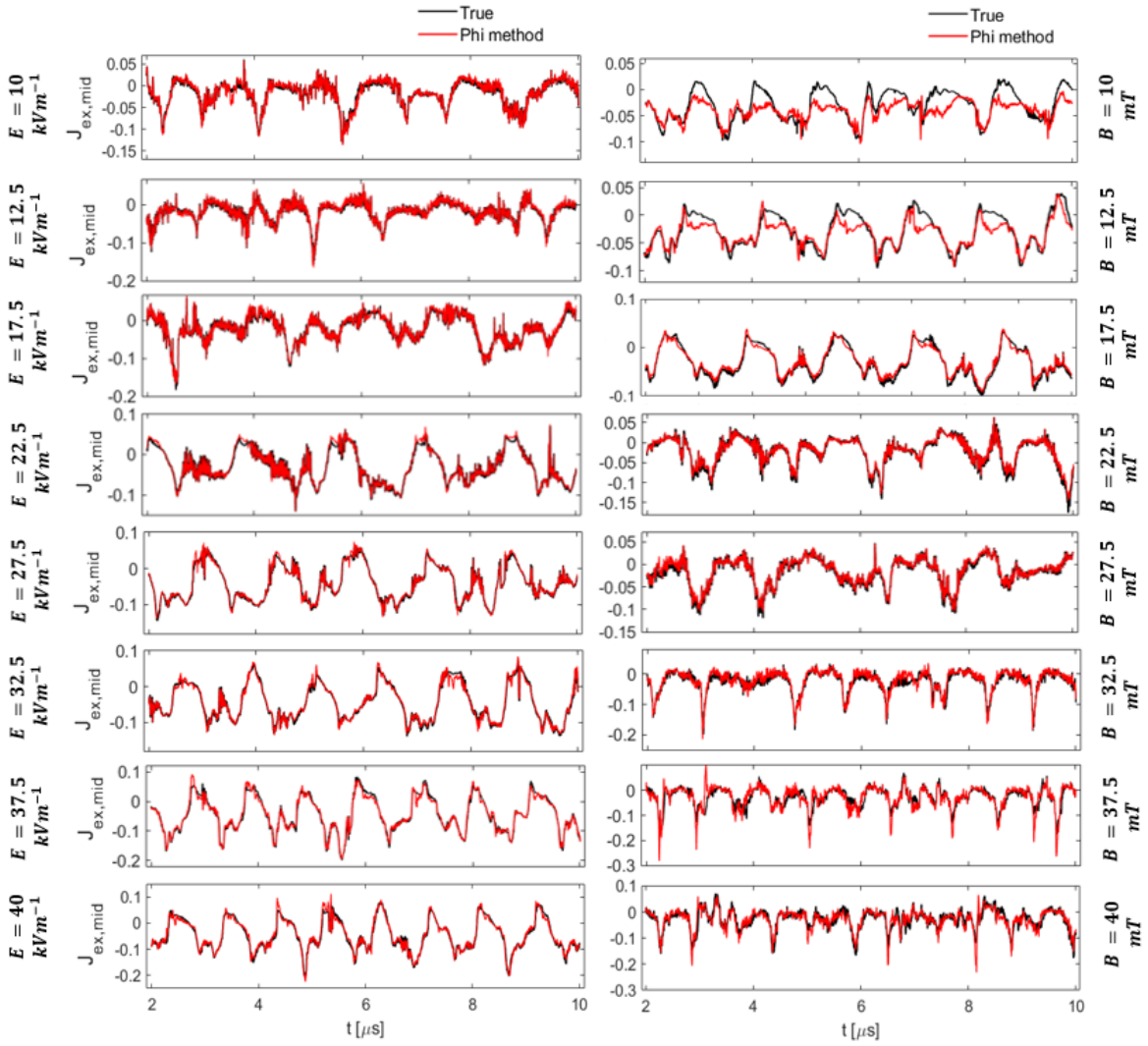


Fig. 6.11 Predictions of the parametric Phi Method ROM (PM1) for the plasma discharge test case compared against the ground-truth data; time evolutions of the local value of the normalized electrons' axial flux density (J_{ex}) at the mid-location within the simulation domain for various test values of the axial electric field (E_x, **left column**) and for various test values of the radial magnetic field (B_y, **right column**).

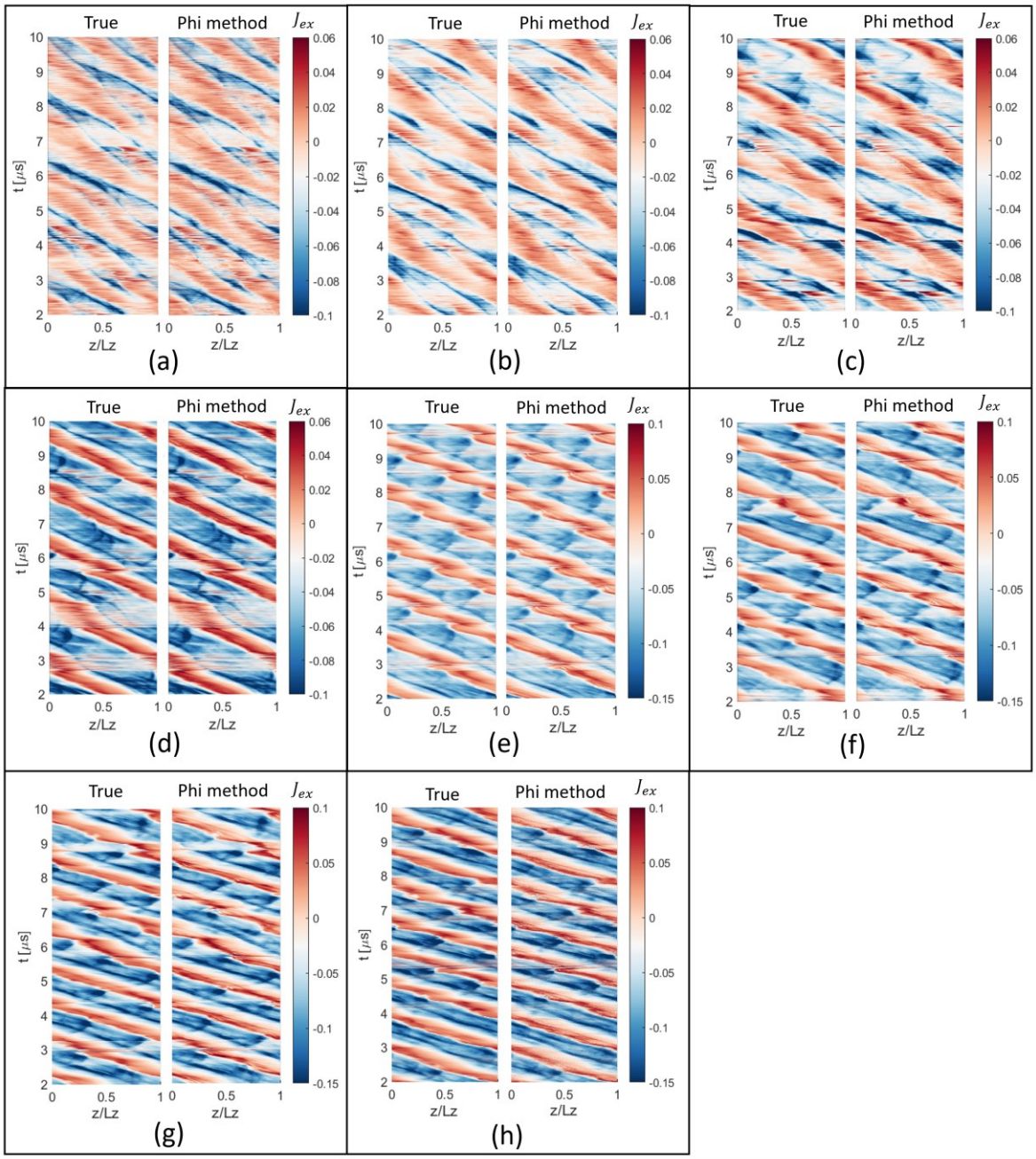


Fig. 6.12 Predictions of the parametric Phi Method ROM (PM1) for the plasma discharge test case compared against the ground-truth data; spatiotemporal maps of the normalized electrons' axial flux density (J_{ex}) for the test values of the axial electric field (E_x): (a) $E_x = 10\ kVm^{-1}$, (b) $E_x = 12.5\ kVm^{-1}$, (c) $E_x = 17.5\ kVm^{-1}$, (d) $E_x = 22.5\ kVm^{-1}$, (e) $E_x = 27.5\ kVm^{-1}$, (f) $E_x = 32.5\ kVm^{-1}$, (g) $E_x = 37.5\ kVm^{-1}$, and (h) $E_x = 40\ kVm^{-1}$.

Looking at Fig. 6.11 to Fig. 6.13, it is evident that the parametric ROM is able to predict the time evolution of the J_{ex} as a function of the plasma properties included in the Phi Method's library. In addition, the parametric data-driven relationship recovered between the target quantity – J_{ex} – and the candidate terms in the library is applicable to a wide parameter space, which emphasizes the generalizability of the model.

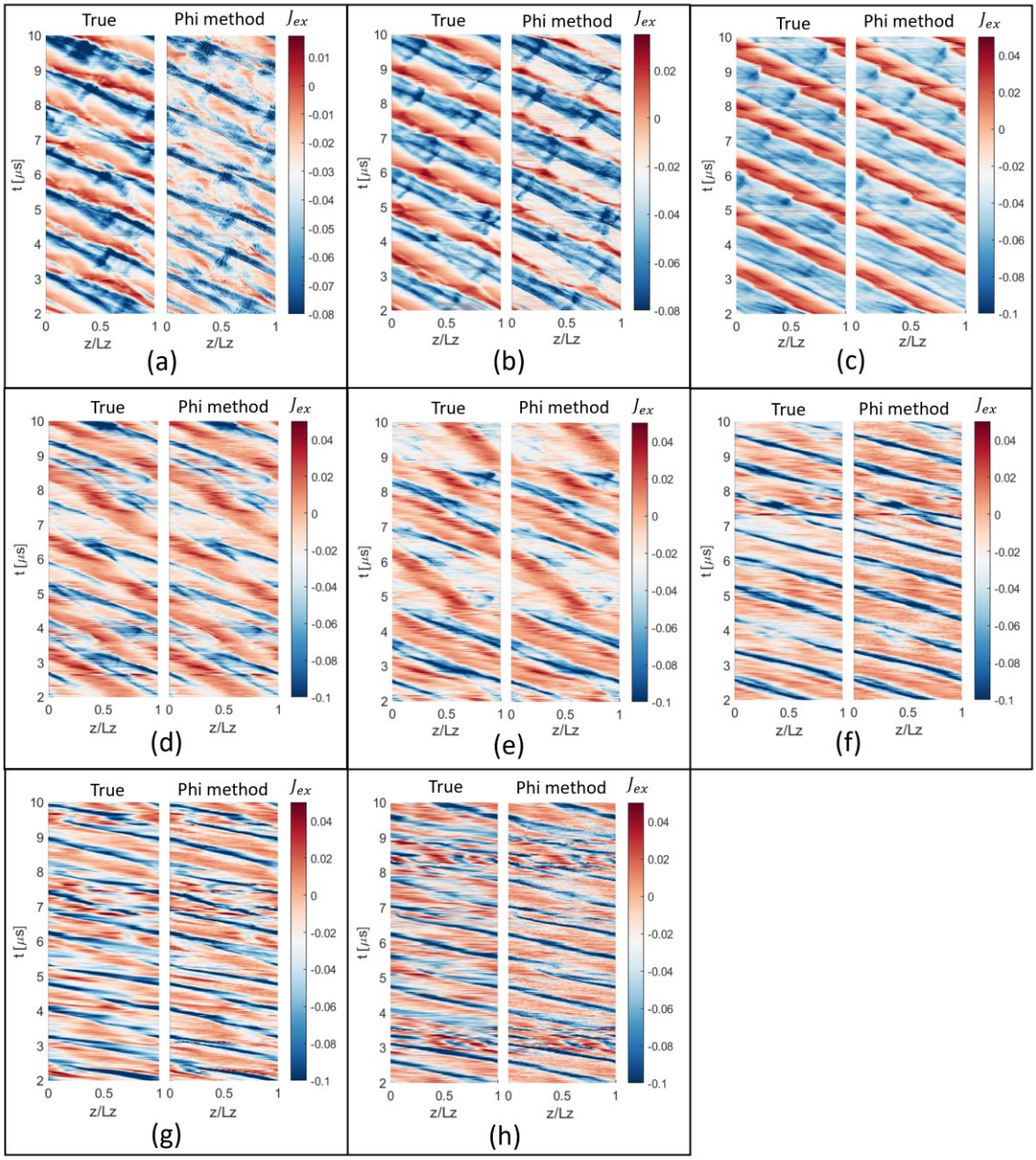


Fig. 6.13 Predictions of the parametric Phi Method ROM (PM1) for the plasma discharge test case compared against the ground-truth data; spatiotemporal maps of the normalized electrons' axial flux density (J_{ex}) for the test values of the radial magnetic field (B_y): (a) $B_y = 10$ mT, (b) $B_y = 12.5$ mT, (c) $B_y = 17.5$ mT, (d) $B_y = 22.5$ mT, (e) $B_y = 27.5$ mT, (f) $B_y = 32.5$ mT, (g) $B_y = 37.5$ mT, and, (h) $B_y = 40$ mT.

In Fig. 6.11 to Fig. 6.13, the eight extreme cases corresponding to the lowest and the highest values of the applied electric and magnetic fields, namely, E_x = 10, 12.5, 37.5, and 40 kVm^{-1}, and B_y = 10, 12.5, 37.5, and 40 mT, fall outside the range of the parameter space over which PM1 had been trained. These cases, thus, represent extrapolation scenarios. In this regard, toward the lowest and the highest magnetic field values, the PM1's accuracy degrades, with the deviations being mainly noticeable around the signals' peaks and dips. However, the predictions are still reasonably aligned with the ground truth. For the extreme electric field cases, the accuracy of the PM1 predictions remains at the same level as that in the intermediate-parameter regimes.

To have an overall assessment of the variations in the ROMs' accuracy across the investigated parameter space, the spatiotemporally averaged predicted values of the J_{ex} from the PM1 and the EM1 models are plotted in Fig. 6.14 against the E_x and the B_y. Fig. 6.14(a) and (c) show the variations vs the E_x and the B_y for the PM1. Fig. 6.14(b) and (d) show the same variations for the EM1. The corresponding "true" J_{ex} values from the PIC simulations are also shown in these plots as red dots. We have superimposed on the plots of Fig. 6.14 the spatiotemporally averaged losses of the J_{ex} predictions with respect to the true values from the simulations.

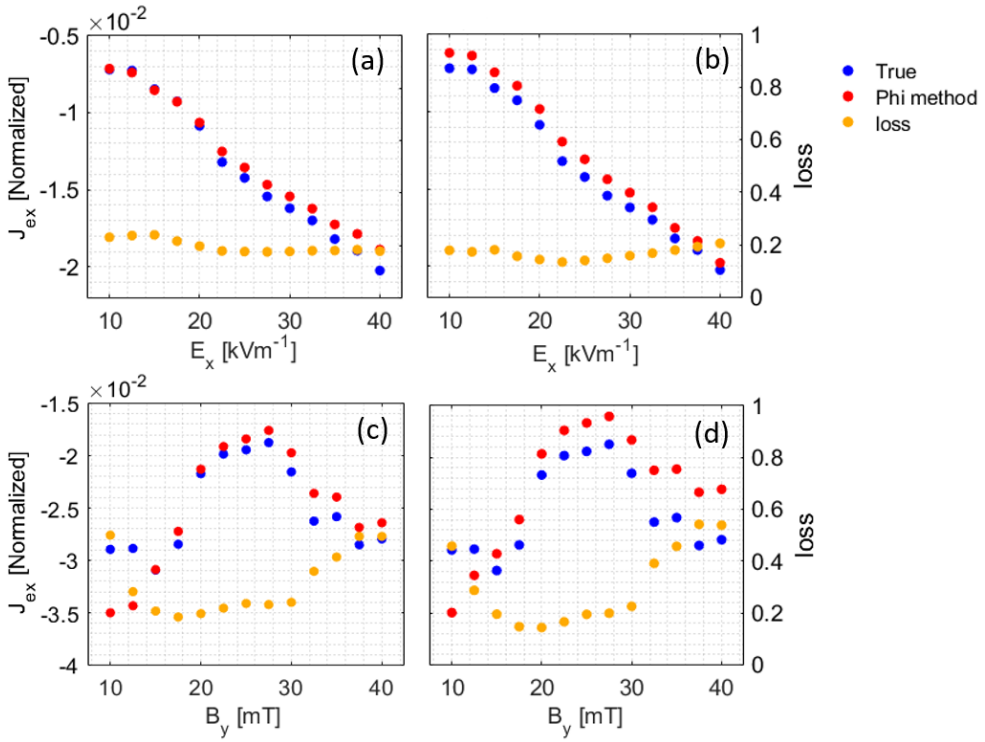


Fig. 6.14 Predictions of the PM1 (**left column**) and the EM1 (**right column**) for the plasma discharge test case; spatially and temporally averaged normalized electrons' axial flux density (J_{ex}) values from the ROMs (predictions) and the simulations (ground-truth) against: (a) and (b) the various electric field values, (c) and (d) the various magnetic field values. The associated predictions' loss factors are also plotted. The normalization factor for J_{ex} is $6.0625 \times 10^{24}\ m^{-2}s^{-1}$.

The plots in Fig. 6.14 indicate that, across various values of the E_x, the loss factors associated with the PM1 and the EM1 models remain almost constant, whereas they exhibit notable variations across the different B_y values. The PM1's and the EM1's loss increases toward the two bounds of the investigated B_y range.

Comparing the plots (b) and (d) in Fig. 6.14 against plots (a) and (c), we observe that the predictions' accuracy of the EM1 is quite comparable to that of the PM1 overall. Accordingly, we can conclude that, for this 1D plasma discharge test case, either training a single model over a large dataset or combining multiple models trained over subsets of the data lead to nearly equivalent ROMs in terms of their predictive performance.

6.3.2.1. Characteristics and interpretability of the coefficients matrices for the parametric and the ensemble Phi Method ROMs

Here, we examine closely the obtained Φ matrices (operators) corresponding to the different DD ROMs discussed so far, namely, the PM1 and PM2 ("standard" parametric Phi Method ROMs), and the EM1 and EM2 (ensemble Phi Method ROMs). We recall from subsection 6.3.1 that Φ_{PM1} and $\Phi_{\mathrm{EM1}} \in R^{12\times1}$, whereas $\Phi_{\mathrm{PM2}}, \Phi_{\mathrm{EM2}} \in R^{15\times1}$. For the demonstration purposes in this subsection, these Phi method operators are rearranged into matrices with the dimensions of 3×4 and 3×5 for the PM1/EM1 and the PM2/EM2, respectively. The rearranged Φ matrices are illustrated in Fig. 6.15.

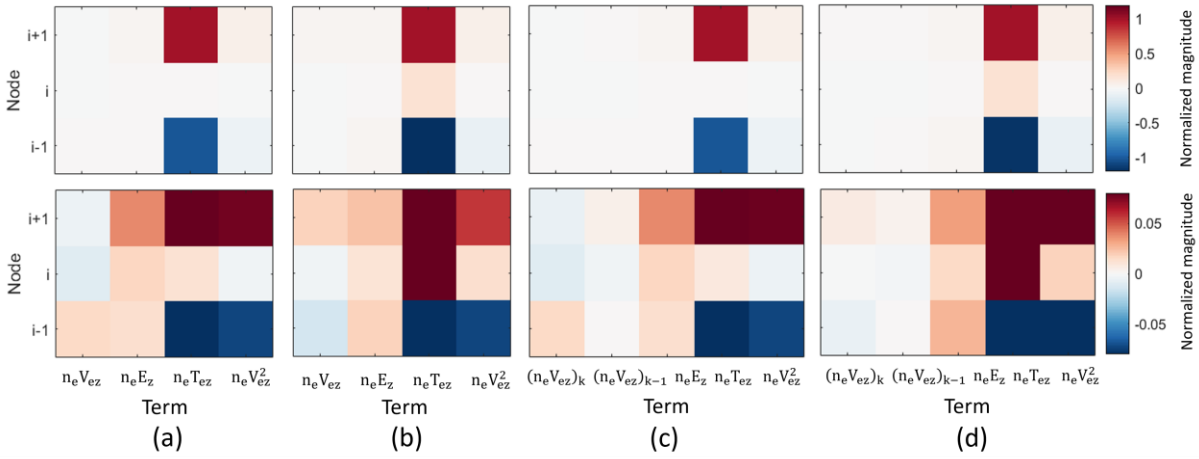


Fig. 6.15 Rearranged normalized representations of the coefficients matrices (Φ) corresponding to the parametric Phi Method ROMs (PM1 and PM2) and the ensemble Phi Method ROMs (EM1 and EM2) for the plasma discharge test case; (a) Φ_{PM1}, (b) Φ_{EM1}, (c) Φ_{PM2}, (d) Φ_{EM2}. The bottom row plots are the rescaled visualizations of the top row plots.

It is observed that the Φ matrices of the four models exhibit notable similarity in terms of the relative significance of their respective entries. In particular, the variations of the coefficients' values along the horizontal axis, which shows the relative importance of the dynamical terms, present a common trend among the matrices of the various ROMs.

Along the vertical axis, i.e., for each term, the relative magnitudes of the coefficients represent the optimal discretization stencil learned by Phi Method for that particular term. In this regard, the Φ_{EM1} and the Φ_{EM2} matrices (Fig. 6.15(b) and (d)) are seen to have identified slightly different stencils for certain terms compared to the parametric ROMs (Fig. 6.15(a) and (c)). The differences are the most distinguishable for the n_eT_{ez} and $n_eV_{d,ez}$ terms.

Furthermore, the coefficients associated with the term $\left(n_eV_{d,ez}\right)_{k-1}$ in PM2/EM2, which is absent in PM1/EM1, are vanishingly small. The similarity of the magnitudes of the coefficients shared among the various models suggests that the inclusion of the additional term $\left(n_eV_{d,ez}\right)_{k-1}$ does not have a notable impact on the Phi Method ROM.

Fig. 6.16 presents the normalized standard deviations of the coefficients of the library terms across the individual models that yielded the ensemble Phi Method ROMs. The standard deviation of each coefficient is normalized with respect to the magnitude of the ensemble mean of the corresponding coefficient over the individual models.

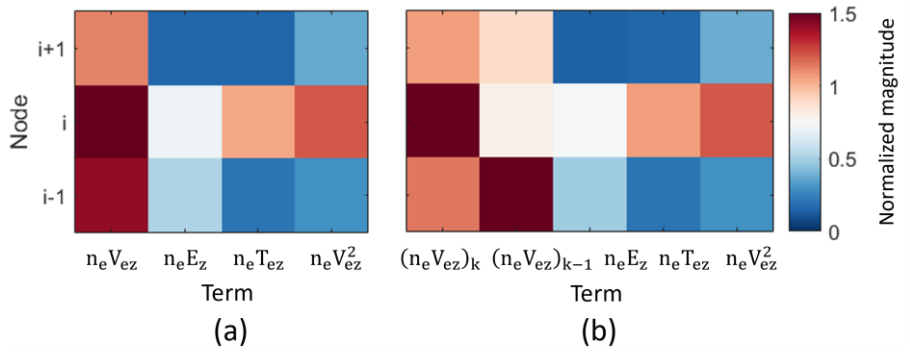


Fig. 6.16 Normalized standard deviations of the coefficients of the library terms across the individual models constituting the ensemble ROMs; (a) standard deviations across the individual models of EM1, (b) standard deviations across the individual models of EM2. The normalization is performed by the absolute value of the mean coefficients matrices (Φ_{EM1} and Φ_{EM2}).

From Fig. 6.16, the largest variability (standard deviation) among the coefficients of the matrices of the individual models is observed for the coefficients of the $n_eV_{d,ez}$ term. The rest of the terms show lower levels of variability in their coefficients' values over the individual models, especially for the neighboring nodes of $i-1$ and $i+1$. This analysis shows that the coefficients associated with the more dominant terms are determined with a higher level of confidence. Whereas, the determination of the coefficients related to terms that contribute minimally to the dynamics involves the highest degree of uncertainty. This observation had also been pointed out from a similar analysis in the fluid flow test case in subsection 6.2.2. In the plasma test case here, the least contributing terms are the $n_eV_{d,ez}$ for EM1 and the $\left(n_eV_{d,ez}\right)_k$ and $\left(n_eV_{d,ez}\right)_{k-1}$ for EM2.

As the last discussion, we aim to compare the data-driven coefficients of the Phi Method ROMs against the analytical counterparts from the discretization of Eq. 6.8. To this end, we rewrite Eq. 6.8 in a discretized form using the finite differencing scheme

$$\begin{aligned} -qn_{e,max}B_{max}V_{d,ex_{max}}\left(\hat{n}_e\hat{B}\hat{V}_{d,ex}\right) \\ = \frac{m_e}{\Delta t}n_{e,max}V_{d,ez_{max}}\,\Delta_t\left(\hat{n}_e\hat{V}_{d,ez}\right) - qn_{e,max}E_{z,max}\left(\hat{n}_e\hat{E}_z\right) \\ + \frac{K}{2\Delta z}n_{e,max}T_{ez,max}\,\Delta_z\left(\hat{n}_e\hat{T}_{ez}\right) + \frac{m_e}{2\Delta z}n_{e,max}V_{d,ez_{max}}^2\,\Delta_z\left(\hat{n}_e\hat{V}_{d,ez}^2\right) \end{aligned} \qquad \text{Eq. 6.11}$$

In Eq. 6.11, Δ_t denotes first-order-accurate temporal discretization of the time derivative (i.e., $\Delta_t\left(\hat{n}_e\hat{V}_{d,ez}\right) = \left(\hat{n}_e\hat{V}_{d,ez}\right)_k - \left(\hat{n}_e\hat{V}_{d,ez}\right)_{k-1}$). Δ_z is the second-order-accurate discretized spatial derivative (e.g., $\Delta_z\left(\hat{n}_e\hat{T}_{ez}\right) = \hat{n}_e\hat{T}_{ez}|_{i+1} - \hat{n}_e\hat{T}_{ez}|_{i-1}$). Δz represents the size of the cells used for the discretization along the z direction, and Δt denotes the timestep between two consecutive data snapshots. Any quantity labeled with the subscript "*max*" represents the spatiotemporal maximum of that quantity across

the entire dataset. The quantities marked with the caret symbol (^) on the top indicate the normalized properties with respect to their maximum value. Rearranging Eq. 6.11 as in Eq. 6.12 results in a theoretical relation among the normalized properties, which is equivalent to the data-driven relation from the Phi Method models

$$\hat{B}\hat{J}_{ex} = C_1\Delta_t\left(\hat{n}_e\hat{V}d_{ez}\right) + C_2\left(\hat{n}_e\hat{E}_z\right) + C_3\Delta_z\left(\hat{n}_e\hat{T}_{ez}\right) + C_4\Delta_z\left(\hat{n}_e\hat{V}_{d,ez}^2\right). \quad \text{Eq. 6.12}$$

The coefficients in Eq. 6.12 are defined as

$$C_1 = -\frac{m_e V_{d,ez_{max}}}{qB_{max}V_{d,ex_{max}}\Delta t}, \quad \text{Eq. 6.13}$$

$$C_2 = \frac{E_{z,max}}{B_{max}V_{d,ex_{max}}}, \quad \text{Eq. 6.14}$$

$$C_3 = -\frac{KT_{ez,\,max}}{2qB_{max}V_{d,ex_{max}}\,\Delta z}, \quad \text{Eq. 6.15}$$

$$C_4 = -\frac{m_e V_{d,ez_{max}}^2}{2qB_{max}V_{d,ex_{max}}\Delta z}. \quad \text{Eq. 6.16}$$

In order to directly compare the above theoretical coefficients with the Phi Method's learned coefficients, we needed to derive the equivalent Phi Method's approximations of the theoretical coefficients C_1, C_2, C_3, and C_4 from the Φ matrices. To approximate the coefficients of the terms that involve spatial derivatives, i.e., $\Delta_z\left(\hat{n}_e\hat{T}_{ez}\right)$ and $\Delta_z\left(\hat{n}_e\hat{V}_{d,ez}^2\right)$, we computed the average of the coefficients of the nodes $i-1$ and $i+1$. For the terms not involving derivatives such as $\hat{n}_e\hat{E}_z$, we considered the average of the coefficients over the three neighboring nodes of $i-1$, i, and $i+1$ as a representative of the terms' coefficient. For the temporal derivative term, $\Delta_t\left(\hat{n}_e\hat{V}_{d,ez}\right)$ that was included in the library of PM2/EM2, the respective coefficient was approximated as the average difference between the coefficients of $\left(\hat{n}_e\hat{V}_{d,ez}\right)_k$ and $\left(\hat{n}_e\hat{V}_{d,ez}\right)_{k-1}$ across all the three neighboring nodes $i-1$, i, and $i+1$.

According to the above paragraph, the relations between the approximated coefficients from PM1/EM1 and PM2/EM2 (denoted by superscript 1 and 2, respectively) and the entries of the respective rearranged matrices of the models (Φ^1 for PM1/EM1 and Φ^2 for PM2/EM2) are defined as

$$C_1^2 \approx \frac{1}{3}\left(\sum_{i=1}^{3}\left(\Phi_{i,1}^2 - \Phi_{i,2}^2\right)\right), \quad \text{Eq. 6.17}$$

$$C_2^1 \approx \frac{1}{3}\left(\sum_{i=1}^{3}\left(\Phi_{i,2}^1\right)\right), \qquad C_2^2 \approx \frac{1}{3}\left(\sum_{i=1}^{3}\left(\Phi_{i,3}^2\right)\right), \quad \text{Eq. 6.18}$$

$$C_3^1 \approx \frac{1}{2}\left(\Phi_{1,3}^1 + \Phi_{3,3}^1\right), \qquad C_3^2 \approx \frac{1}{2}\left(\Phi_{1,4}^2 + \Phi_{3,4}^2\right), \quad \text{Eq. 6.19}$$

$$C_4^1 \approx \frac{1}{2}\left(\Phi_{1,4}^1 + \Phi_{3,4}^1\right), \qquad C_4^2 \approx \frac{1}{2}\left(\Phi_{1,5}^2 + \Phi_{3,5}^2\right). \quad \text{Eq. 6.20}$$

In the above equations, $\Phi_{i,j}$ indicates the entry at the i-th row and the j-th column of the rearranged Φ matrix for PM1/EM1 or PM2/EM2.

In Fig. 6.17(a) and (b), the evaluations of the coefficients C_1 to C_4 from PM1, PM2, EM1, and EM2 are compared against the corresponding analytical values of the same coefficients. Moreover, in Fig. 6.17(c) and (d), the errors defined as the normalized differences between the Phi Method coefficients' values and the corresponding analytical ones are plotted. The normalization factor has been the analytical values of the coefficients.

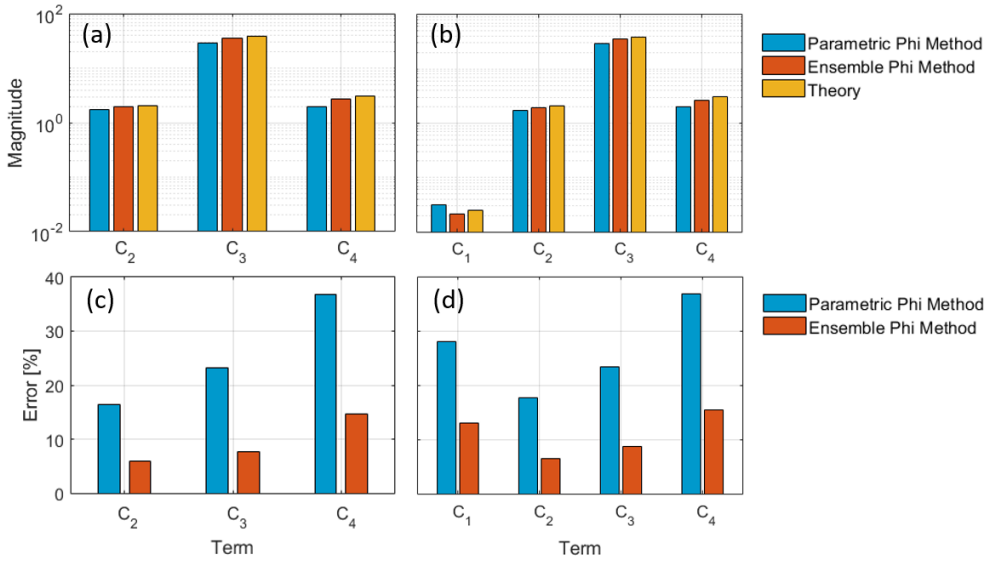


Fig. 6.17 Comparison of the learned coefficients of the terms in Eq. 6.12 from the parametric and the ensemble Phi Method against the respective analytical values of the coefficients for the plasma discharge test case. Plots (a) and (b) show the magnitudes of the coefficients for the PM1/EM1 and the PM2/EM2 ROMs, respectively. Plots (c) and (d) show, respectively, the normalized error of the obtained coefficients relative to the analytical values for the PM1/EM1 and the PM2/EM2 ROMs.

The plots in Fig. 6.17 indicate that, first of all, all the studied Phi Method ROM variants have identified the coefficients C_1 to C_4 within a 40 % range of the corresponding analytical values. It shall be noted that, while we have termed the difference between the analytical and the learned coefficients as "error", this does not necessarily imply an inherent error in the model. In fact, in order to derive the analytical coefficients themselves, as well as to obtain the approximations of the equivalent coefficients from the Φ matrices, we assumed the finite difference discretization stencil. However, the Phi Method's premise is that it finds the optimal discretization stencil, which might be different from that of the finite difference method. Furthermore, it is important to note that we used PIC simulations for the dataset generation in the plasma test case. The PIC method does not explicitly solve Eq. 6.8. This means that the finite difference scheme was not part of the solution that led to the generation of the dataset. The absence of the finite difference scheme in the solution process underlying the dataset supports the hypothesis that this scheme does not necessarily serve as the optimal discretization stencil for the present test case. Nonetheless, we would also mention that the presence of noise in the PIC simulation data can slightly impact the inference of the optimal coefficient matrix, which can also partly account for the observed disparity between the learned and the analytically calculated coefficients.

As the second point, the coefficients obtained from the ensemble models appear to align more closely with the "theory". However, this apparently higher level of agreement might be misleading,

considering our above argument regarding the caveat of assuming the finite differencing stencil to translate the coefficients of the Φ matrices back to the coefficients of Eq. 6.12. This is especially true for the terms involving spatial derivatives because, during the translation of the learned coefficients based on the finite difference stencil, the contribution of the middle node (node i) would be ignored. In this regard, and as an example, we can see from Fig. 6.15(b) and (d) that, while the middle node's coefficient for the $n_e T_{ez}$ term is as large as that on the neighboring nodes, there is no obvious way to include its contribution in the approximation of the C_3 coefficient.

Despite the above remark, the improved agreement between the ensemble ROMs' coefficients and the analytical ones can also be, at least partly, due to the statistical benefits of the ensembling approach. In general, the ensembling can average out the biases and errors of the individual models, leading to a more representative estimation of the underlying data patterns. Therefore, despite the equivalent prediction accuracy observed between the "standard" parametric Phi Method ROMs and the ensemble ROMs, the ensembling could have improved the estimation of the coefficients matrix.

6.4. Chapter conclusions

In this Chapter, we presented the application of Phi Method toward the discovery of parametric PDEs – equations that describe the time evolution of systems with parametric dependency of the dynamics. This capability of Phi method was demonstrated in two test cases: (1) a 2D fluid flow around a cylinder with parametric dependence on the Reynolds number of the freestream, and (2) a 1D Hall-thruster-representative $\mathrm{E} \times \mathrm{B}$ plasma discharge, in which the underlying phenomena and the global dynamics are largely dependent on the values of the externally imposed electric and magnetic fields.

It was discussed that, to learn a data-driven ROM for the dynamics of a parametric system using Phi Method, two distinct approaches are possible: one approach is that of the "standard" parametric Phi Method, which involves learning a single model trained over a range of values of the parameter(s) associated with the system. The other approach – the "ensemble" Phi Method – consists of developing individual data-driven models for each specific parameter value and then obtaining an aggregate model by averaging over the individually trained models. This aggregate (ensemble) model can then be used for the predictions of the dynamics over the unseen parameter space.

In each test case investigated, the ROMs from the parametric and the ensemble implementations of Phi Method were well able to recover the governing parametric PDE from the data of the simulations performed over a training set of values of the relevant parameter(s) (Reynolds number for the fluid system, and the electric and the magnetic field intensities for the plasma system).

The parametric Phi Method ROMs also presented a remarkable performance in predicting the systems' dynamics over the unseen sets of parameters, confirming the generalizability of the learned dynamics over the parameter space.

In the plasma test case, it was shown that the predictive performance of the ROMs from the parametric and the ensemble Phi Method is quite comparable. The Φ matrices obtained from either

approach were also very similar, particularly in terms of the identified relative importance of the dynamical terms. This observation was also confirmed in the fluid flow test case.

The equivalent learned coefficients of the PDE describing the time evolution of the electrons' axial flux density in the plasma test case from the parametric and the ensemble Phi Method approaches compared rather closely with the analytical coefficients obtained from the finite-difference-discretization of the same PDE.

It was highlighted, nonetheless, that the ensemble Phi Method approach provides certain benefits over the "standard" parametric Phi Method. In this respect, the ensemble Phi Method provides, on the one hand, statistical insights into the derived ROMs. For instance, from both the fluid flow and the plasma test cases, we understood using the ensemble Phi Method that the coefficients of the terms that dominantly contribute to the dynamics of a system are determined with a high confidence level (small standard deviations across the individual models). On the other hand, the aggregation technique that underpins the ensemble Phi Method can lower the biases and/or errors of the individual models, hence, improving the reliability and robustness of the resulting ensemble ROM.

In the fluid flow test case, we also applied the parametric extension of the OPT-DMD algorithm (subsection 3.2.3) in order to compare the performance of this POD-based approach in representing the parametric dynamics against the performance of Phi Method. The parametric OPT-DMD provided unsatisfactory predictions of the dynamics over the test parameter set. However, the predictions improved in the case where the algorithm was trained on an expanded dataset corresponding to a larger set of training Reynolds numbers. In contrast, the parametric Phi Method was seen to not require a large dataset representing many samples of the parameter space so as to be able to properly represent the parametric dynamics. This observation highlights a strong merit of Phi Method as a local operator-finding algorithm for parametric dynamics discovery over POD-based approaches like DMD.

It is finally emphasized that, even though for the adopted test cases in this Chapter, the parametric dynamics and the relationship among the involved state variables and parameters were known, the great performance of Phi Method in recovering the involved parametric dependencies of the dynamics and the demonstrated generalizability of the associated ROMs have important applied implications. Indeed, the results underline that Phi Method can be reliably applied to the scenarios where the underlying parametric PDE(s) may not be fully or partially known, or the cases where one may be interested in finding correlations between a certain state variable and the quantities of the system that can be readily experimentally accessible.

Chapter 7 Conclusions and future work

7.1. Overview and summary of the thesis

Plasmas, ubiquitous from cosmic to laboratory scales, and pivotal in various applications ranging from fusion energy to in-space propulsion, have remained shrouded with many mysteries. The intricate dynamics of plasmas, characterized by multifaceted underlying phenomena across various scales, has long posed major challenges toward understanding and predicting them, impeding us from harnessing plasmas for fusion energy or from leveraging them for revolutionary propulsion technologies.

Against this backdrop, the primary question of the present thesis was: "*How can machine learning (ML) establish innovative pathways to address the persisting challenges in plasma physics, thus, facilitating our ability to understand, predict, and control plasmas?*"

Toward answering this research question, Chapter 1 presented an overview of the basics of plasma physics so as to aid understanding the complexities of plasmas and the ensuing challenges for their predictive modeling. By examining the limitations of existing simulation methodologies and tools, primarily in terms of computational cost and/or self-consistency, the crucial need for novel, accurate, and efficient simulation models was underscored – models that are not only capable of resolving the persisting unknowns surrounding plasmas behavior but that can also be effectively integrated into the design and development processes of advanced plasma technologies. The overall promises of machine learning to this end were reviewed, including the acceleration of high-fidelity simulations and the development of reduced-order models. We followed by viewing a picture of the existing opportunities to cross-transfer well-demonstrated ML applications in fluid mechanics to plasma physics. Importantly, it was emphasized that cost-efficient kinetic simulations play a critical, pre-requisite role in generating extensive high-fidelity datasets for ML, the lack of which has been a main contributing factor to the limited advance of ML in plasma physics. The breakthrough reduced-order PIC scheme by Reza [83] was highlighted as a notably promising enabler in this regard.

Chapter 2 delved deeper into the ML applications for the analysis and modeling of physical systems. The ML's capacity to transform data from simulations (and experiments) into actionable insights, thus enriching domain knowledge was highlighted. The significant uptake of ML approaches across various scientific domains, especially in the adjacent field to plasma physics – fluid mechanics – paves the way for a rapid adoption and integration into plasma research of the most promising ML methods. Establishing this mapping was a principal aim of Chapter 2. Important fundamental ML concepts were discussed and the three main categories of learning by machines, i.e., supervised, unsupervised, and semi-supervised were reviewed. An introduction of several main algorithms belonging to each category was next presented. The areas where ML has been integrated into computational fluid dynamics were assessed, with a particular emphasis on dimensionality reduction and dynamics discovery as two crucial

components for developing ROMs. The POD approach and the autoencoders were discussed as prominent ML (data-driven) dimensionality-reduction techniques and the DMD (more generally Koopman) method, SINDy, and SHRED were presented as important examples of dynamics discovery (system identification) algorithms. Finally, recent ML applications in plasma systems modeling were analyzed, which offered a glimpse into the evolving landscape of ML-driven research in plasma physics.

Following on the state-of-the-art review in Chapter 2, the work started from Chapter 3 to investigate different ML methodologies across various plasma configurations and setups, which covered a broad range of complex plasma dynamics.

Chapter 3 looked closely at the applications of DMD alongside its foundational method, POD/SVD, to plasma systems' analysis and mode identification. The Chapter was kicked off by describing in detail the theory behind POD/SVD and then demonstrated its role in understanding the low-dimensional representability of data, in data compression and de-noising, and finally to reconstruct unseen data based on the learned reduced coordinates. Various DMD variants were then introduced, including the basic (Exact) DMD, and the Optimized DMD (OPT-DMD). The performance of these two DMD methods was compared toward reproducing the ground-truth datasets using the identified dominant coherent structures. OPT-DMD exhibited much greater reliability in capturing low-rank dynamics representation due to its ability to constrain the eigenvalues associated with the derived modes to near the imaginary axis, hence, ensuring stability of the modes over arbitrary time durations. Evaluating OPT-DMD's convergence against rank number, a trade-off was pointed out between reconstructing training data accurately and predicting test data reliably, emphasizing the importance of selecting an optimal number of ranks. Additionally, a novel application of OPT-DMD for spectral analyses in complementarity to FFT techniques was introduced. The consistency of the OPT-DMD results with those of FFT was noted and the benefits of OPT-DMD over the FFT method in yielding a simultaneous spatial-temporal characterization of plasma instabilities were emphasized. This advantage of DMD hugely aides the experimental investigation of instabilities. Lastly, the robustness of OPT-DMD's identification of spatial structures and frequencies of dominant instability modes with respect to rank inflation issue was affirmed.

Chapter 4 assessed the predictive capabilities of OPT-DMD-derived ROMs across various plasma configurations and test cases. The ROM's performance in forecasting the plasma state over both short and long timeframes was evaluated, observing the stability and reliability of predictions with bounded errors. Despite the inherent assumption of temporally invariant patterns associated with the DMD method, OPT-DMD ROMs demonstrated great predictive potential, particularly in scenarios featuring quasi-periodic plasma behavior. Enhancing the state space of OPT-DMD with the spatiotemporal data of multiple plasma properties, a remarkable improvement was observed for long-term forecasts. This was linked to the improved learning of OPT-DMD on multi-property state vectors because such state space enhancement enables the algorithm to capture the interrelations among the plasma properties. OPT-DMD ROMs exhibited robustness to noise as well, reliably identifying the dominant modes even in noisy datasets. It shall be kept in mind that, in the context of ROM development, the accuracy and the computational cost come as a trade-off. In this regard, the OPT-DMD models are of negligible

computational cost compared to the PIC simulations, enabling very fast predictions of the plasma behavior. However, their accuracy was noticed to be changing across various plasma configurations and conditions depending on the complexity of underlying dynamics. Regardless, the results showed that, in general, the OPT-DMD serves as a reliable method to derive models for applications such as operational control and digital twin development, where the models are first and foremost intended to be predictive over short timeframes – timeframes that correspond to the acquisition and receipt of new datasets from the physical system that enable updating the ML model.

Within Chapter 5, “Phi Method” was introduced – a novel data-driven local operator finding algorithm that was developed within this PhD research. By learning the discretized form of differential equations governing the dynamics via a constrained regression on data that is informed by well-established numerical discretization schemes, Phi Method enables straightforward time forecasting of systems’ state. Contrary to OPT-DMD, which provides a global representation of the dynamics in terms of the time evolution of dominant structures in data, Phi Method learns the “local” spatiotemporal dynamics, leading to the discovery of the governing system of equations. Through detailed demonstrations across three distinct test cases – the Lorenz attractor problem, a flow-past-a-cylinder problem, and a 1D Hall-thruster-representative plasma problem – the Phi Method's performance in constructing reliable and robust ROMs was showcased. Most notably from the method’s demonstrations in this Chapter, it was shown that Phi Method enables learning the dynamics from the transient data of a system, exceeding the capabilities of OPT-DMD and similar methods that are inherently limited in working with transient data. This advantage of Phi Method is highly valuable for systems that exhibit unsteady/transient state evolutions, or that for which generating representative data of the (quasi-)steady state can be computationally challenging, for instance, plasma systems like Hall thrusters. The interpretability of the Phi-Method ROMs was demonstrated in the Lorenz test case by showing that the coefficients of the learned operator by Phi Method compares very closely against the analytical coefficients of discretized Lorenz system. The noise robustness characteristics of Phi Method were also analyzed, for the fluid flow and the plasma discharge test cases that involved different levels of added Gaussian noise.

The application of Phi Method was expanded in Chapter 6 to learning parametric PDEs – equations that describe the time evolution of systems with parametric dependency of the dynamics. Two test cases were examined: a 2D fluid flow around a cylinder with parametric dependence on the Reynolds number, and a 1D Hall-thruster-representative plasma discharge, where the dynamics is heavily affected by the imposed electric and magnetic fields. We explored two pathways for learning parametric ROMs using Phi Method: the “standard” parametric Phi Method, and the "ensemble" Phi Method. In adopted both test cases, ROMs from parametric and ensemble Phi Method effectively recovered the governing parametric PDEs from simulation data. They demonstrated remarkable predictive performance over unseen parameter sets, indicating the generalizability of learned dynamical relationships across parameter space. Comparative analysis between the two approaches revealed comparable predictive performance and overall similar learned operators. The ensemble Phi Method offered statistical insights into the derived ROMs and reduced biases/errors, enhancing reliability. For the fluid test case, Phi Method’s performance was compared against the parametric extension of OPT-DMD. Whereas parametric OPT-DMD had relatively good predictions only with expanded datasets, Phi

Method did not require extensive training data, which underscored its superior efficacy in learning parametric dynamics over POD-based approaches. The results and discussions in Chapter 6 demonstrated Phi Method's capability in recovering parametric dependencies of the dynamics and the generalizability of the models, making the approach highly promising for scenarios where the underlying parametric PDEs are partially known or that correlations between system variables and experimentally accessible quantities are of interest.

7.2. Future works and research directions

Machine learning and data-driven techniques represent an exciting landscape, ripe with potentials, for advancing computational plasma physics, in general, and to develop predictive and cost-effective reduced-order models, in particular. ML/DD applications in plasma physics are still limited in scope and in extent, meaning that there is significant room for follow-on research on the basis of the results and outcomes of this thesis. Many high-potential use-cases of ML (explored in Chapter 2) can be imported into the field from fluid mechanics with regards to both simulations and experiments. Of course, we did not touch upon ML for experimental plasma physics in this thesis although this area stand to equally benefit from ML approaches by the extension of the methods employed in experimental fluid mechanics.

With the above in mind, a roadmap is sketched in this Chapter for how the outcomes of this thesis can be built upon. The Chapter also suggests some promising directions toward which the research on ML in plasma physics for scientific and applied purposes can be headed.

In Section 7.2.1, the next steps to further improve the Phi Method implementation are discussed, specifically with respect to augmenting its noise robustness as well as incorporating parsimonious learning into the algorithm. Section 7.2.2 shares the author's perspectives on how two highly promising ML methods of SINDy and SHRED (subsection 2.3.1) can be used toward enhancing scientific discovery in plasma physics and revamped plasma modelling capabilities. Finally, Section 7.2.3 sets out a vision for the encapsulation of the ML research in plasma physics within the groundbreaking framework of the digital twin technology. The digital twin concept will be briefly introduced and its revolutionary promises for plasma science and engineering will be described, highlighting that the realization of these promises is largely entangled to advances in ML-empowered plasma simulations and reduced-order models.

7.2.1. Toward fast and predictive plasma models using Phi Method

Two pathways of development for Phi Method beyond this thesis are envisioned: one consists of improving the robustness of the algorithm to noise in the data, and the other involves embedding sparsity in the Phi Method's implementation. The following subsections discuss each of these development/improvement pathways.

7.2.1.1. Optimization-based regression for augmented robustness to noise

Section 5.4 showed that the presence of noise in the data can affect the quality of the reduced-order models obtained from the Phi Method algorithm. This was particularly seen to be the case for the plasma

discharge test case (subsection 5.4.2), where the noise content were rather substantial. It is important to note that, for a time-forecasting ML algorithm, even small inaccuracies in the learned coefficients of the dynamics model can accumulate to large discrepancies when stepping forward in time. This can lead to the instability and divergence of the predictions.

To improve the robustness of the predictions to noise, the model's prediction window during training can be extended. This approach involves minimizing a loss function that considers the model's predictions over longer time horizons, rather than focusing solely on the immediate next step. By doing so, the model learns to account for the cumulative effects of errors over extended periods. This can in turn enhance the model's stability despite the presence of noise.

As was explained in section 5.1, in the baseline Phi Method implementation, the Φ matrix is derived by performing a least-squares regression on the data such that

$$\Phi = \underset{\Phi}{\operatorname{argmin}} \ \|\mathrm{X} - \ \Theta\Phi\|_2 = \Theta^{\dagger} X. \qquad \text{Eq. 7.1}$$

The involved data matrices in Eq. 7.1 were described in section 5.1. The Φ solution obtained from this equation can be taken as the initial guess for a subsequent optimization problem which aims to minimize the error between the model's prediction and the ground-truth data over an extended time period, e.g., the entire training interval

$$\Phi = \underset{\Phi}{\operatorname{argmin}} \sum_{k=1}^{m_t} \sum_{i=1}^{N_g} \left\| x_i^k - \theta_i^0 \Phi^{\mathrm{k}} \right\|_2 . \qquad \text{Eq. 7.2}$$

In Eq. 7.2, x_i is the ground-truth data on the i-th node and at the time step k, θ_i^0 represents the input library terms corresponding to node i evaluated at the initial time step, N_g is the total number of grid nodes, and m_t is the number of training time steps. $\theta_i^0 \Phi^{\mathrm{k}}$ represents the Phi-Method model's prediction at the k-th time step (note that Φ^{k} means Φ to the k-th power).

As a way of assessing for the future the efficacy of the above optimization approach, it was tested preliminarily for the case of the 1D azimuthal $\mathrm{E} \times \mathrm{B}$ discharge problem, which was described in subsection 5.3.1. The aim here has been to predict the time evolutions of the electron number density (n_e) as a function of various sets of library terms.

Fig. 7.1 compares the results from baseline models obtained using Eq. 7.1 and models obtained from solving the optimization problem as given by Eq. 7.2. It is evident from the various model comparisons in Fig. 7.1 that accounting for an extended prediction window in the optimization problem can significantly improve the models' robustness to noise and remarkably enhance the stability of their predictions across different scenarios. These are highly encouraging results that underline the potential effectiveness of the optimization-based implementation of Phi Method for augmenting the stability and robustness of the models from this approach.

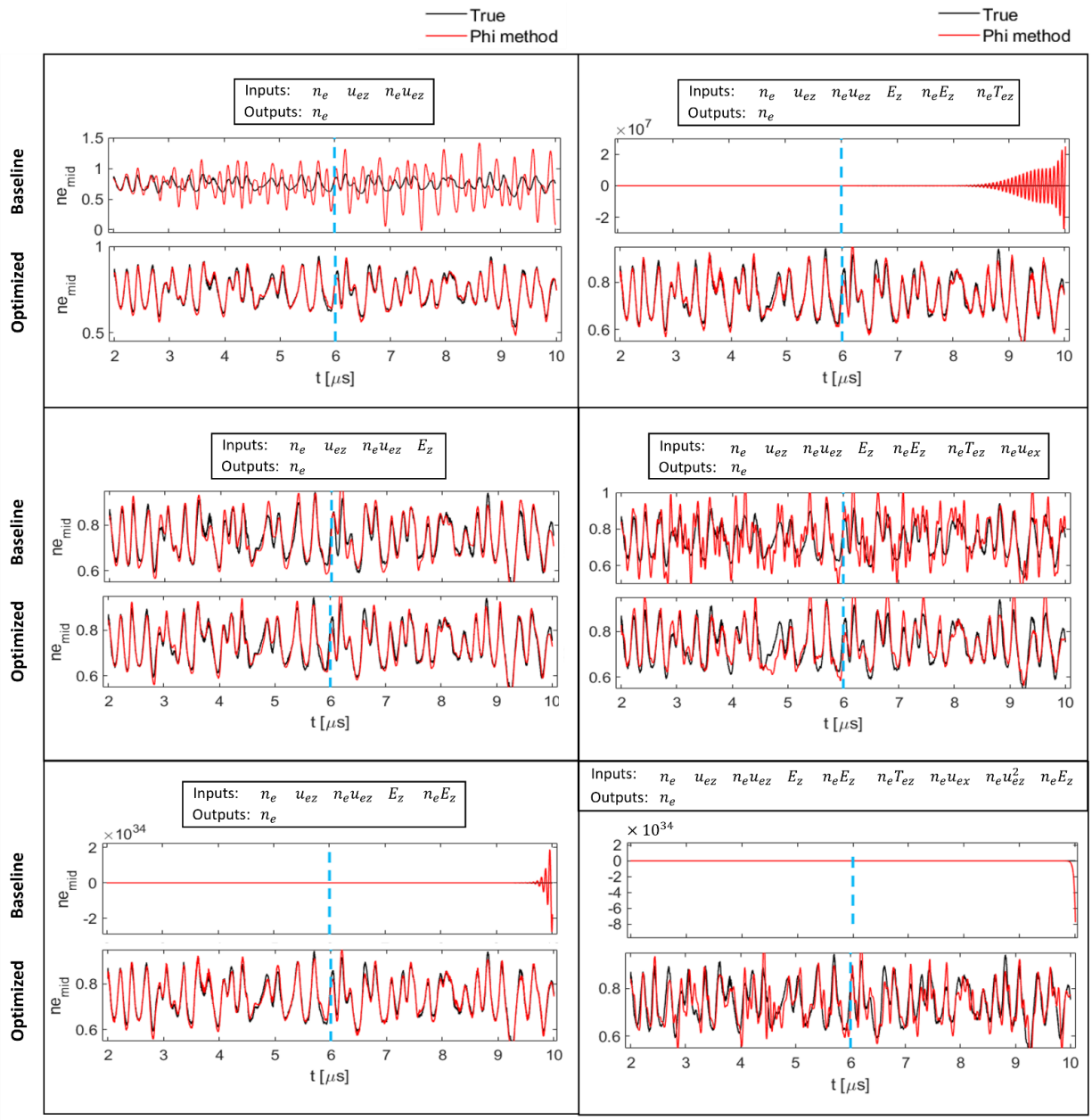


Fig. 7.1 Comparison of the Phi Method models for the electron number density (n_e) in the 1D azimuthal E × B discharge problem (subsection 5.3.1). For various input library terms, the comparisons are made between baseline Phi Method models obtained from solving Eq. 7.1 and "optimized" Phi Method models obtained from solving Eq. 7.2.

7.2.1.2. Sparse regression for closure model discovery and self-consistent predictiveness

The general importance of the concept of sparsity (parsimony) in the context of physics-inspired data-driven model discovery, particularly toward increasing the interpretability and generalizability of the learned models, was discussed in subsections 2.1.2.3 and 2.3.1. With regards to Phi Method, in particular, it was pointed out in subsection 5.3.2.1, as well as in subsection 6.3.1, that sparsity-promoting optimization can enhance the interpretability of the learned operators, leading to models that only include the essential terms to represent the systems' dynamics even in the presence of expanded libraries of candidate dynamical terms. This is important with respect to both data-driven closure model

discovery (discrepancy modelling) and self-consistent predictiveness of the discovered system of governing equations. Accordingly, future work on Phi Method to further mature the approach will involve the incorporation of the sparsity-promoting optimization within the algorithm.

The common approaches to incorporate sparsity in the model include $L1$ regularization (LASSO) and thresholding techniques, such as Sequential Thresholding Least Squares (STLS) [182] and the Sparse Relaxed Regularized Regression (SR3) [241]. In L1 regularization, an additional penalty term is added to the standard loss function of the model, which is proportional to the sum of absolute values of the model's coefficients. This penalty term encourages some of the coefficients to be zero, effectively selecting a subset of the library terms. The thresholding techniques involve eliminating the terms with coefficients below a certain threshold [182]. Hard thresholding sets the coefficients below a pre-defined value to zero, whereas soft thresholding shrinks the coefficients' values toward zero.

7.2.2. Perspectives on the applications of SINDy and SHRED

Among the various types and flavors of ML algorithms for physical systems modelling, which we extensively reviewed in Chapter 2, some perspectives for future applications are shared in this section for two specific architectures – SINDy and SHRED. Both of these architectures have demonstrated high versatility in various use cases and represent a strong promise and utility potential for plasma applications for both scientific and applied purposes.

We start by SINDy. As it was pointed out in subsection 2.3.1, in addition to the inherent power of SINDy for developing generalizable and interpretable dynamics models, which largely roots in the underlying sparse regression, many variants of this approach have emerged over the last years to robustify the algorithm and facilitate its working with limited and noisy data. Of particular note are Ensemble SINDy [183] and Bayesian SINDy [242] (also Bayesian-autoencoder SINDy [192]), all of which provide pathways for uncertainty quantification as well. Moreover, it is demonstrated that enforcing physics-informed constraints on top of SINDy's sparse learning further improves the generalizability and physics consistency of the learned dynamics. Learning the weak-form of the dynamics has additionally proven beneficial to address the numerical issues associated with spatial-temporal differentiation of noisy data [216]. All these factors and progressive improvements contribute to making SINDy a promising candidate with which to explore the possibilities of addressing plasma modelling challenges, including the self-consistency of plasma fluid system of equations and the realization of computationally fast, fully predictive plasma models.

Prior applications of SINDy to high-fidelity data from plasma and fluid systems has demonstrated its applicability to recover *known* nonlinear PDEs, such as the Vlasov's equation [216] and the Navier-Stokes equation [158], based solely on the data. This promise can be extended to the discovery of the unknown plasma dynamics and system of governing equations that obviate the reliance of the current-generation fluid-based simulations on experimentally defined, ad-hoc, and highly case-specific transport coefficients.

The SINDy-derived PDEs can potentially feature rigorous closure models. In this respect, there can be the flexibility to determine the moment of the plasma species' distribution function at which to close

the system of PDEs (e.g., pressure, heat flux, etc.). Additionally, the SINDy algorithm can provide a trade-off by construction between the complexity (number of appearing terms) of the discovered closure models and error over test data. Hence, at the rather low complexity limit, the DD closure models are likely to recover the known terms from the classical theory and/or from prior research efforts. Increasing the models' complexity, the discovered closure may include previously unknown nonlinear terms that represent the missing kinetic and nonclassical effects in the conventional plasma fluid models. PDE plasma descriptions has the general advantage of being rather easy to interpret and enabling to distinguish various physical mechanisms through the terms appearing in the system of equations. PDE "parametrization" can be applied to SINDy as well (similar to parametric Phi Method, section 6.1), which may provide the data-driven system of equations the flexibility to account for the variation in the dominant phenomena over various behavioral regimes of the plasma.

The plasma PDEs will be highly effective to advance fundamental studies in plasma physics. However, for the engineering applications, such as optimization-based design and predictive control, computational efficiency (ultimately, real-time solutions) is key. This is where ODEs (and generally ROMs) come into play. Coupling SINDy with autoencoder [182] or Bayesian autoencoder networks [192] can enable an effective and joint discovery of the dominant modes underlying the data and the time evolution of these modes. Parameter embedding can also be applied in the autoencoder-SINDy architecture to improve the generalizability of the derived ODEs. Besides their engineering value, the parameterized ODEs can aid scientific discovery by allowing us to understand the evolution and interactions of global coherent structures across various operating/plasma conditions characterized by the embedded parameters, e.g. the electric and the magnetic field intensities.

We now turn our focus to SHRED. Prior demonstrations of this architecture [193]-[195] have proven that it can play a central role in several areas of ML-enabled plasma modelling.

The first possibility is to reconstruct and/or to forecast the high-dimensional spatiotemporal evolution of several plasma properties using sensor trajectories of a different property that is dynamically coupled to them. For example, we can measure the electron number density using plasma probes (which can be rather straightforward/inexpensive) and use SHRED to reconstruct the fields related to the electron temperature, electron velocities, etc. The dynamic coupling among the properties is key for this kind of mapping. This roots in the fact that a system of n coupled PDEs (each describing the spatiotemporal evolution of one property) can be represented as a single nonlinear PDE of order n, to solve which SHRED can be applied. It is recalled from subsection 2.3.1 that the SHRED architecture is a nonlinear generalization of separation of variables for linear PDE solving, which makes use of the time-history measurements rather than full initial conditions over entire space to make the solution unique.

The capability to reconstruct and/or forecast the often-extended state variables of plasma systems in the way that is offered by SHRED can be highly advantageous in plasma research because usually a specific plasma field is easier and/or more accessible to measure.

Another area of application for SHRED is toward super-resolution, i.e., the mapping of a low-resolution field (e.g., from a coarse-grid simulation) to a high-resolution one by filling in the missing details. We recall the discussions in subsection 2.1.2.1 on how super-resolution and subgrid modelling

frameworks have been followed to enhance/accelerate high-fidelity simulations. Often, the CNNs have been used for flow-field enhancement purposes, but SHRED also stands as a promising candidate architecture to this end. Applying SHRED for super-resolution amounts to simply increasing the number of sensors to match the number of grid points of the coarse mesh.

Since SHRED can map local sensor-measured data to full high-dimensional states, it may be applied toward discrepancy modelling as well (see subsection 2.1.2.2). The real-world measurements can be used within SHRED to update and possibly correct a plasma simulation that gives predictions of high-dimensional state variables, potentially identifying a model for the source(s) of discrepancy. As discrepancy modelling is, in general, an important application area, exploring the use of SHRED's capabilities along this direction is warranted.

Lastly, the features and characteristics of SHRED position it as a highly promising architecture for plasma digital twins, particularly because of the unique interface that it offers between real-world sensor data and the computational models underlying the twin.

7.2.3. Digital twins: the next game-changer for plasma science and technology

One of the most relevant – and perhaps strong motivators for – industrial applications of physics-informed ML and the ML/DD ROMs is the transformative technology of digital twins (DTs). A DT is a concept that has been around under different names and terminologies since the 1990s [243]-[247], and some authors have even traced it back in some forms to the Apollo era [248]. However, the DTs have only emerged as a viable and practically revolutionary technology in the last decade. This digital technology has seen a surge in its appeal across the industries over the past few years along with the significant increase in computational power and the rapid advances in ML algorithms and diagnostics/sensing capabilities that all serve to strongly empower the DTs.

Inspired by a definition proposed by the Digital Engineering Integration Committee of the American Institute of Aeronautics and Astronautics (AIAA) in 2020 [249], a digital twin is a virtual representation of a real-world (physical) counterpart that is able to mimic the behavior of the physical asset (twin) accurately, has predictive capability, is "personalized" to the specific device, and is dynamically updated by receiving data from the physical twin throughout the device's life cycle. The predictiveness characteristic of a DT, i.e., its ability to forecast behavior beyond the available spatiotemporal span of data, enables the digital twin to inform decisions that in turn realize value.

The DT concept extends beyond the conventional notion of simulation and modelling, which are considered static, by the merit of a bidirectional interaction between the digital and the physical twin. This bidirectional interaction allows the dynamic evolution of the DT and comprises feedback flows of information from the physical system to the virtual representation and from the virtual back to the physical system to enable decision-making and control. A DT is particularly powerful for the "what-if scenario" analysis in a way that far exceeds what is feasible/practical to do through experiments alone.

The DT concept has generated an immense interest in recent years across fields of science and engineering due to the many benefits that DTs can offer to the applications involving complex, multifaceted, and interconnected systems (engineering, health, societal, etc.). Focusing on the

engineering applications, the use of DTs is motivated by three general objectives: (**i**) to drive down the cost and time of design and development/manufacture, (**ii**) to increase the fidelity, efficiency, and monitorability of the entire development process before product delivery, and (**iii**) to enable effective asset management during the entire product's life cycle.

Today, prominent intended use cases of DTs within engineering can be divided into "pre-delivery" uses and "operational/lifecycle management" uses [250]. The pre-delivery usage refers to the role that DTs can play in product design and manufacturing processes [250][251]. The asset management application includes damage detection and structural/system health monitoring. Of particular relevance, the use of DTs for asset management have been proposed for nuclear fusion reactors in Ref. [252].

The promises of the DT technology for the plasma industry revolve around two pillars: (**i**) streamlined, efficient, and methodical computer-aided design and development processes, (**ii**) robust autonomous operation and control of the plasma system. In this regard, recalling the discussions presented in section 1.2, the DT technology is closely aligned with and in fact build upon the predictive plasma models. Because of the continual exchange of data that exists between the physical and the digital twin, however, the DT can transcend and expand the capabilities and use cases of predictive plasma models. Moreover, as the DT aims to replicate the behavior of its physical twin as closely as possible (ideally, in an identical manner) and because the integration of physics-based/informed predictive models under the hood of a DT in principle enables extensive interrogation and probing of the processes ongoing during the device's operation, DTs also open novel, unprecedented pathways for scientific inquiry.

Accordingly, the reason why the last section of this thesis has been discussing the DT technology is that, in the author's opinion, the pursuit of fully fledged DTs for plasma technologies provides an optimal overarching framework of both academic and industrial relevance for the ML research in plasma physics. In fact, ML/DD ROMs that can cost-effectively and accurately describe the plasma system's time dynamics by learning from the simulation and/or experimental data are essential enabling elements of the DT technologies. In this regard, the concept of "discrepancy modelling" [114] (subsection 2.1.2.2) and the role of ML in increasing the representativeness between the digital representation models and the physical asset based on a stream of real-world data serve as crucially important and pertinent lines of research. Even beyond the immediate scope of ML studies, the continual verification/validation (V&V) and uncertainty quantification (UQ) that are necessary to ensure the trustworthiness and reliability of the digital twin predictions for decision making can promote further progress and maturation of the current V&V and UQ practices and methodologies in the field, benefiting the broader plasma physics research.

This PhD research represented a significant step forward in leveraging ML and DD techniques to advance plasma modelling and simulation, including toward DT's realization. The findings underscored the enabling role that ML can play in realizing robust and predictive ROMs for plasma systems, which can help address fundamental plasma questions, on one side, and revamp the plasma engineering practices, on the other. While there are still challenges for ML in physical systems modelling and analysis that need to be addressed, transformative advancements in plasma science and technology

using ML is within reach via a continued research and innovation at the cross-section of plasma physics and machine learning.

Bibliography

[1] Langmuir I, "Oscillations in Ionized Gases," *Proc. Natl Acad. Sci.* **14** 627-637 (1928)

[2] Chen FF, "Plasma Physics and Controlled Fusion," 2nd ed., Springer, New York, 2006.

[3] Dendy RO (ed.), "Plasma Physics: An Introductory Course," Cambridge University Press, Cambridge, 1993.

[4] Northrop TG, "The guiding center approximation to charged particle motion," *Annals of Physics*, Vol. **15**, Issue 1, pp 79-101 (1961)

[5] Gibbon P, "Introduction to Plasma Physics," Published by CERN in the Proceedings of the CAS-CERN Accelerator School: Plasma Wake Acceleration, Geneva, Switzerland, 23–29 November 2014, edited by B. Holzer, CERN-2016-001 (CERN, Geneva, 2016)

[6] Krall NA, Trivelpiece AW, "Principles of Plasma Physics," McGraw-Hill, 1973.

[7] Spitzer L, "Physics of Fully Ionized Gases," Interscience Publishers, 1962.

[8] Bittencourt JA, "Fundamentals of Plasma Physics," 3rd ed., Springer New York, NY, 2004.

[9] Lieberman MA, Lichtenberg AJ, "Principles of Plasma Discharges and Materials Processing," Wiley-Interscience, 2005.

[10] Meek JM, Craggs JD, "Electrical Breakdown of Gases," Oxford University Press, 1978.

[11] Lu X, Bruggeman PJ, Reuter S, Naidis G, Bogaerts A, Laroussi M, Keidar M, et al., "Grand challenges in low temperature plasmas," *Frontiers in Physics* **10** (2022)

[12] Patil BS, Wang Q, Hessel V, Lang J, "Plasma N2-fixation: 1900–2014," *Catalysis Today*, Vol. **256**, Part 1, pp 49-66 (2015)

[13] URL to "Contemporary Physics Education Project" website: https://www.cpepphysics.org/plasma-physics-fusion/, last accessed in May 2024.

[14] Adamovich I, Agarwal S, Ahedo E, Alvez LL, Baalrud S, et al., "The 2022 Plasma Roadmap: low temperature plasma science and technology," *J. Phys. D: Appl. Phys.* **55** 373001 (2022)

[15] Kaganovich ID, Smolyakov A, Raitses Y, Ahedo E, Mikellides IG, Jorns B, Taccogna F, et al., "Physics of E × B discharges relevant to plasma propulsion and similar technologies," *Phys. Plasmas* **27**, 120601 (2020)

[16] Hopwood J, "Review of inductively coupled plasmas for plasma processing," *Plasma Sources Sci. Technol.* **1** 109 (1992)

[17] Sarakinos K, Alami J, Konstantinidis S, "High power pulsed magnetron sputtering: A review on scientific and engineering state of the art," *Surface and Coatings Technology*, Vol. **204**, Issue 11, pp. 1661-1684 (2010)

[18] Goebel DM, Katz I, "Fundamentals of Electric Propulsion: Ion and Hall Thrusters," JPL Space Science and Technology Series, Wiley, 2008

[19] Boeuf JP, "Tutorial: Physics and modeling of Hall thrusters", *Journal of Applied Physics* **121** 011101 (2017)

[20] Boeuf JP, Smolyakov A, "Physics and instabilities of low-temperature E × B plasmas for spacecraft propulsion and other applications," *Physics of Plasmas* **30**, 050901 (2023)

[21] Heron A, Adam JC, "Anomalous conductivity in Hall thrusters: Effects of the non-linear coupling of the electron-cyclotron drift instability with secondary electron emission of the walls," *Phys. of Plasmas* **20**, 082313 (2013)

[22] Lazurenko A, Vial V, Bouchoule A, Prioul M, Adam JC, Heron A, Laval G, "Characterization of microinstabilities in Hall thruster plasma: experimental and PIC code simulation results," IEPC-2003-0218, *In Proceedings of International Electric Propulsion Conference*, Toulouse, France (2003)

[23] Lafleur T, Baalrud SD, Chabert P, "Theory for the anomalous electron transport in Hall effect thrusters. I. Insights from particle-in-cell simulations," *Phys. Plasmas* **23**, 053502 (2016)

[24] Boeuf JP, Garrigues L, "E × B electron drift instability in Hall thrusters: Particle-in-cell simulations vs. theory," *Phys. Plasmas* **25**, 061204 (2018)

[25] Reza M, Faraji F, Andreussi T, Andrenucci M, "A Model for Turbulence-Induced Electron Transport in Hall Thrusters," IEPC-2017-367, *35th International Electric Propulsion Conference*, Atlanta, Georgia (2017)

[26] Jorns B, "Predictive, data-driven model for the anomalous electron collision frequency in a Hall effect thruster," *Plasma Sources Sci. Technol.* **27** 104007 (2018)

[27] Chapurin O, Smolyakov AI, Hagelaar G, Raitses Y, "On the mechanism of ionization oscillations in Hall thrusters," *J. Appl. Phys.* **129** (23): 233307 (2021)

[28] Kim V, "Main physical features and processes determining the performance of stationary plasma thrusters," *J. Propul. Power* **14**, 5 (1998)

[29] Zhurin VV, Kaufman HR, Robinson RS, "Physics of closed drift thrusters," *Plasma Sources Sci. Technol.* **8** R1 (1999)

[30] Choueiri EY, "Anomalous resistivity and heating in current-driven plasma thrusters," *Phys. Plasmas* **6** 2290 (1999)

[31] Hasegawa A, "Plasma Instabilities and Nonlinear Effects," *Physics and chemistry in space, Vol. 8, Springer-Verlag*, (1975)

[32] Koshkarov O, Smolyakov A, Raitses Y, Kaganovich I, "Self-organization, structures, and anomalous transport in turbulent partially magnetized plasmas with crossed electric and magnetic fields," *Phys. Rev. Lett.* **122**, 185001 (2019)

[33] Lakhin VP, Ilgisonis VI, Smolyakov AI, Sorokina EA, "Nonlinear excitation of long-wavelength modes in Hall plasmas," *Phys. Plasmas* **23**, 102304 (2016)

[34] Choueiri EY, "Plasma oscillations in Hall thrusters," *Phys. Plasmas* **8**, 1411–1426 (2001)

[35] Boeuf JP, Garrigues L, "Low frequency oscillations in a stationary plasma thruster," *J. Appl. Phys.* **84**, 3541–3554 (1998)

[36] Barral S, Makowski K, Peradzyński Z, Dudeck M, "Transit-time instability in Hall thrusters," *Phys. Plasmas* **12**, 073504 (2005)

[37] Fernandez E, Scharfe MK, Thomas CA, Gascon N, Cappelli MA, "Growth of resistive instabilities in E×B plasma discharge simulations," *Phys. Plasmas* **15**, 012102 (2008)

[38] Gary SP, Sanderson JJ, "Longitudinal waves in a perpendicular collisionless plasma shock: I. Cold ions," *Journal of Plasma Physics* **4**(4):739-751 (1970)

[39] Cavalier J, Lemoine N, Bonhomme G, Tsikata S, Honore C, Gresillon D, "Hall thruster plasma fluctuations identified as the E×B electron drift instability: Modeling and fitting on experimental data," *Phys. Plasmas* **20**, 082107 (2013)

[40] Tsikata S, Honore C, Gresillon DM, "Collective Thomson scattering for studying plasma instabilities in electric thrusters," *J. Instrum.* **8**, C10012 (2013)

[41] McBride JB, "Theory and Simulation of Turbulent Heating by the Modified Two-Stream Instability," *Physics of Fluids* 15.12 (1972)

[42] Petronio F, Tavant A, Charoy T, Alvarez-Laguna A, Bourdon A, Chabert P, "Conditions of appearance and dynamics of the Modified Two-Stream Instability in E×B discharges," *Phys. Plasmas* **28**, 043504 (2021)

[43] Villafana W, Petronio F, Denig AC, Jimenez MJ, Eremin D et al., "2D radial-azimuthal particle-in-cell benchmark for E×B discharges," *Plasma Sources Sci. Technol.* **30** 075002 (2021)

[44] Smolyakov A, Zintel T, Couedel L *et al.,* "Anomalous Electron Transport in One-Dimensional Electron Cyclotron Drift Turbulence," *Plasma Phys. Rep.* **46**, 496–505 (2020)

[45] Adam JC, Héron A, Laval G, "Study of stationary plasma thrusters using two-dimensional fully kinetic simulations", *Phys. of Plasmas* **11**, 295 (2004)

[46] Janhunen S, Smolyakov A, Sydorenko D, Jimenez M, Kaganovich I, Raitses Y, "Evolution of the electron cyclotron drift instability in two dimensions," *Phys. Plasmas* **25**, 082308 (2018)

[47] Taccogna F, Minelli P, Asadi Z, Bogopolsky G, "Numerical studies of the ExB electron drift instability in Hall thrusters," *Plasma Sources Sci. Technol.*, **28**(6):064002 (2019)

[48] Reza M, Faraji F, Knoll A, "Parametric investigation of azimuthal instabilities and electron transport in a radial-azimuthal E×B plasma configuration", *Journal of Applied Physics* **133**, 123301 (2023)

[49] Reza M, Faraji F, Knoll A, "Influence of the magnetic field curvature on the radial-azimuthal dynamics of a Hall thruster plasma discharge with different propellants", *Journal of Applied Physics*, **134**, 233303 (2023)

[50] Reza M, Faraji F, Knoll A, "Effects of the applied fields' strength on the plasma behavior and processes in E×B plasma discharges of various propellants: I. Electric field", *Phys. Plasmas* **31**, 032120 (2024)

[51] Reza M, Faraji F, Knoll A, "Effects of the applied fields' strength on the plasma behavior and processes in E×B plasma discharges of various propellants: II. Magnetic field", *Phys. Plasmas* **31**, 032121 (2024)

[52] Powis AT, Carlsson JA, Kaganovich ID, Raitses Y, Smolyakov A, "Scaling of spoke rotation frequency within a Penning discharge", *Physics of Plasmas*, **25**(7), 072110 (2018)

[53] Janes GS, Lowder RS, "Anomalous Electron Diffusion and Ion Acceleration in a Low-Density Plasma," *Phys. Fluids* **9**, 1115 (1966)

[54] McDonald MS, Gallimore AD, "Comparison of Breathing and Spoke Mode Strength in the H6 Hall Thruster Using High Speed Imaging," IEPC-2013-353, *33rd International Electric Propulsion Conference* (2013)

[55] Charoy T, Lafleur T, Laguna Alvarez A, Bourdon A, Chabert P, "The interaction between ion transit-time and electron drift instabilities and their effect on anomalous electron transport in Hall thrusters," *Plasma Sources Sci. Technol.* **30** 065017 (2021)

[56] Petronio F, Charoy T, Alvarez Laguna A, Bourdon A, Chabert P, "Two-dimensional effects on electrostatic instabilities in Hall thrusters. I. Insights from particle-in-cell simulations and two-point power spectral density reconstruction techniques," *Physics of Plasmas* 30, 012103 (2023)

[57] Janhunen S, Smolyakov A, Chapurin O, Sydorenko D, Kaganovich I, Raitses Y, "Nonlinear structures and anomalous transport in partially magnetized E×B plasmas," *Phys. Plasmas* **25**, 011608 (2018)

[58] Meezan N, Hargus W, Cappelli M, "Anomalous electron mobility in a coaxial Hall discharge plasma," *Phys. Rev. E* **63**, 026410 (2001)

[59] Adam JC, Boeuf JP, Dubuit N, Dudeck M, Garrigues L, Gresillon D et al., "Physics, simulation and diagnostics of Hall effect thrusters," *Plasma Phys. Control. Fusion* **50** 124041 (2008)

[60] Balescu R, "Aspects of Anomalous Transport in Plasmas," 1st Edition, CRC Press, 2019

[61] URL to a description of digital engineering and digital tools: https://insights.sei.cmu.edu/blog/what-digital-engineering-and-how-it-related-devsecops/, last accessed in May 2024.

[62] Smucker B, Krzywinski M, Altman N, "Optimal experimental design," *Nat Methods* **15**, 559–560 (2018)

[63] Piragino A, Faraji F, Reza M, Ferrato F et al. "Background Pressure Effects on the Performance of a 20 kW Magnetically Shielded Hall Thruster Operating in Various Configurations," *Aerospace*. **8**(3), 69 (2021)

[64] Byrne MP, Roberts PJ, Jorns BA, "Coupling of Electrical and Pressure Facility Effects in Hall Effect Thruster Testing," IEPC-2022-377, *In Proc. of 37th International Electric Propulsion Conference*, Cambridge, MA (2022)

[65] Taccogna F, Garrigues L, "Latest progress in Hall thrusters plasma modelling," *Reviews of Modern Plasma Physics*, Springer Singapore, **3** (1) (2019)

[66] Hara K, "An overview of discharge plasma modeling for Hall effect thrusters," *Plasma Sources Sci. Technol.* **28** 044001 (2019)

[67] Kuldinow DA, Yamashita Y, Mansour AR, Hara K, "Ten-moment fluid model with heat flux closure for gasdynamic flows," *Journal of Computational Physics*, Vol. **508**, 113030 (2024)

[68] Birdsall CK, Langdon AB, "Plasma Physics via Computer Simulation," CRC Press (1991)

[69] Taccogna F, Cichocki F, Eremin D, Fubiani G, Garrigues L, "Plasma propulsion modeling with particle-based algorithms," *J. Appl. Phys.* **134**, 150901 (2023)

[70] Katz I, Chaplin VH, Ortega AL, "Particle-in-cell simulations of Hall thruster acceleration and near plume regions," *Phys. Plasmas* **25**, 123504 (2018)

[71] Mikellides IG, Jorns B, Katz I, Ortega AL, "Hall2De Simulations with a First-principles Electron Transport Model Based on the Electron Cyclotron Drift Instability," *52nd AIAA/SAE/ASEE Joint Propulsion Conference*, Salt Lake City, Utah (2016) DOI: 10.2514/6.2016-4618

[72] Mikellides IG, Lopez Ortega A, "Challenges in the development and verification of first-principles models in Hall-effect thruster simulations that are based on anomalous resistivity and generalized Ohm's law," *Plasma Sources Sci. Technol.* **28**, 014003 (2019)

[73] Lafleur T, Baalrud SD, Chabert P, "Theory for the anomalous electron transport in Hall effect thrusters. II. Kinetic model," *Physics of Plasmas* **23**(5), 053503 (2016)

[74] Taccogna F, Minelli P, "Three-dimensional particle-in-cell model of Hall thruster: The discharge channel," *Phys. Plasmas* **25**, 061208 (2018)

[75] Charoy T, "Numerical study of electron transport in Hall thrusters", Plasma Physics [physics.plasm-ph], Institut Polytechnique de Paris. English. ⟨NNT: 2020IPPAX046⟩ ⟨tel-02982367⟩ (2020)

[76] Coche P, Garrigues L, "A two-dimensional (azimuthal-axial) particle-in-cell model of a Hall thruster," *Phys. Plasmas* **21**, 023503 (2014)

[77] Eremin D, "An energy- and charge-conserving electrostatic implicit particle-in-cell algorithm for simulations of collisional bounded plasmas," *Journal of Computational Physics* **452**, 110934 (2022)

[78] Sun H, Banerjee S, Sharma S, Powis AT, Khrabrov AV, Sydorenko D, Chen J, Kaganovich ID, "Direct implicit and explicit energy-conserving particle-in-cell methods for modeling of capacitively coupled plasma devices," *Phys. Plasmas* **30**, 103509 (2023)

[79] Juhasz Z, Ďurian J, Derzsi A, Matejčík S, Donkó Z, Hartmann P, "Efficient GPU implementation of the Particle-in-Cell/Monte-Carlo collisions method for 1D simulation of low-pressure capacitively coupled plasmas," *Computer Physics Communications*, **263**, 107913 (2021)

[80] Ricketson LF and Cerfon AJ, "Sparse grid techniques for particle-in-cell schemes," *Plasma Phys. Control. Fusion* **59** 024002 (2017)

[81] Reza M, Faraji F, Knoll A, "Concept of the generalized reduced-order particle-in-cell scheme and verification in an axial-azimuthal Hall thruster configuration," *J. Phys. D: Appl. Phys.* **56** 175201 (2023)

[82] Reza M, Faraji F, Knoll A, "Generalized reduced-order particle-in-cell scheme for Hall thruster modeling: concept and in-depth verification in the axial-azimuthal configuration," *ArXiv preprint*, arXiv:2208.13106 (2022)

[83] Reza M, "Reduced-order particle-in-cell scheme for computationally efficient kinetic plasma modelling," PhD thesis, Imperial College London (2024)

[84] Faraji F, Reza M, Knoll A, "Verification of the generalized reduced-order particle-in-cell scheme in a radial-azimuthal E×B plasma configuration," *AIP Advances* **13**, 025315 (2023)

[85] Reza M, Faraji F, Knoll A, "Latest verifications of the reduced-order particle-in-cell scheme: Penning discharge and axial-radial Hall thruster case," AIAA-2024-2712, *In Proc. of 2024 SciTech Forum conference*, Orlando, Florida (2024)

[86] Charoy T, Boeuf JP, Bourdon A, Carlsson JA, Chabert P, Cuenot B, Eremin D et al., "2D axial-azimuthal particle-in-cell benchmark for low-temperature partially magnetized plasmas," *Plasma Sources Sci. Technol.* **28**, 105010 (2019)

[87] URL to details of the Penning Discharge Benchmark activity "https://jpb911.wixsite.com/landmark/copie-de-test-case-3-fluid-hybrid", last accessed in May 2024.

[88] Reza M, Faraji F, Knoll A, Piragino A, Andreussi T, Misuri T, "Reduced-order particle-in-cell simulations of a high-power magnetically shielded Hall thruster," *Plasma Sources Sci. Technol.* **32**, 065016 (2023)

[89] Samuel AL, "Some Studies in Machine Learning Using the Game of Checkers," *IBM Journal of Research and Development*, vol. **3**, no. 3, pp. 210-229 (1959)

[90] Wiener N, "Cybernetics or Control and Communication in the Animal and the Machine," re-issue of the 1961 Second Edition, The MIT Press (2019)

[91] Rosenblatt, F, "The perceptron: A probabilistic model for information storage and organization in the brain," *Psychological Review*, **65**(6), 386–408 (1958)

[92] URL to Lighthill's report, https://www.chilton-computing.org.uk/inf/literature/reports/lighthill_report/p001.htm, last accessed in May 2024.

[93] Rumelhart D, Hinton G, Williams R, "Learning representations by back-propagating errors," *Nature* **323**, 533–536 (1986)

[94] Hopeld JJ, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Natl. Acad. Sci.* **79**:2554-2558 (1982)

[95] Gardner E, "The space of interactions in neural network models," *J. Phys. A* **21**:257 (1988)

[96] Brunton SL, Noack BR, Koumoutsakos P, "Machine Learning for Fluid Mechanics," *Annu. Rev. Fluid Mech.* **52**:477–508 (2020)

[97] Vinuesa R, Brunton SL, "Enhancing computational fluid dynamics with machine learning," *Nat Comput Sci* **2**, 358–366 (2022)

[98] Karniadakis GE, Kevrekidis IG, Lu L, Perdikaris P et al., "Physics-informed machine learning," *Nat Rev Phys* **3**, 422–440 (2021)

[99] Pollard A, Castillo L, Danaila L, Glauser M (eds.), "Whither Turbulence and Big Data in the 21st Century?" Springer Cham, 2017.

[100] Cherkassky V, Mulier FM, "Learning from Data: Concepts, Theory, and Methods," 2nd Edition, Wiley-IEEE Press, 2008.

[101] Brunton SL, Kutz JN, "Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control," Cambridge University Press, 2019.

[102] Nelder JA, Wedderburn RWM, "Generalized Linear Models," *Royal Statistical Society. Journal. Series A: General*, Vol. **135**, Issue 3, pp 370–384 (1972)

[103] Schölkopf B, Smola AJ, "Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond," The MIT Press, 2018.

[104] Breiman L, "Random Forests," *Machine Learning* **45**, 5–32 (2001)

[105] Hornik K, Stinchcombe M, White H, "Multilayer feedforward networks are universal approximators," *Neural Networks*, Vol. **2**, Issue 5, pp. 359-366 (1989)

[106] Krizhevsky A, Sutskever I, Hinton GE, "ImageNet classification with deep convolutional neural networks," *Commun. ACM* **60**, 6 84–90 (2017)

[107] Hochreiter S, Schmidhuber J, "Long Short-Term Memory," *Neural Comput* **9** (8): 1735–1780 (1997)

[108] Graves A, Fernández S, Schmidhuber J., "Multi-dimensional Recurrent Neural Networks," In: de Sá, J.M., Alexandre, L.A., Duch, W., Mandic, D. (eds) Artificial Neural Networks – ICANN 2007. ICANN 2007. Lecture Notes in Computer Science, vol 4668. Springer, Berlin, Heidelberg (2007)

[109] Pathak J, Hunt B, Girvan M, Lu Z, Ott E, "Model-Free Prediction of Large Spatiotemporally Chaotic Systems from Data: A Reservoir Computing Approach," *Phys. Rev. Lett.* **120**, 024102 (2018)

[110] Kingma DP, Welling M, "Auto-Encoding Variational Bayes", *ArXiv*, arXiv:1312.6114v11 (2022)

[111] Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S et al., "Generative adversarial networks," *Commun. ACM* **63**, 11 139–144 (2020)

[112] Sutton RS, Barto AG, "Reinforcement Learning," 2nd edition, Adaptive Computation and Machine Learning series, The MIT Press, 2018.

[113] Schaul T, Horgan D, Gregor K, Silver D, "Universal Value Function Approximators," *In Proc. of the 32nd International Conference on Machine Learning* PMLR **37**:1312-1320 (2015)

[114] Ebers MR, Steele KM, Kutz JN, "Discrepancy Modeling Framework: Learning Missing Physics, Modeling Systematic Residuals, and Disambiguating between Deterministic and Random Effects," *SIAM Journal on Applied Dynamical Systems* **23**:1, 440-469 (2024)

[115] Raissi M, Perdikaris P, Karniadakis GE, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, Vol. **378**, pp. 686-707 (2019)

[116] Cai S, Mao Z, Wang Z, Yin M, Karniadakis GE, "Physics-informed neural networks (PINNs) for fluid mechanics: a review," *Acta Mech. Sin.* **37**, 1727–1738 (2021)

[117] Eivazi H, Tahani M, Schlatter P, Vinuesa R, "Physics-informed neural networks for solving Reynolds-averaged Navier–Stokes equations," *Physics of Fluids* **34** (7): 075117 (2022)

[118] Kim Y, Choi Y, Widemann D, Zohdi T, "A fast and accurate physics-informed neural network reduced order model with shallow masked autoencoder," *Journal of Computational Physics*, Vol. **451**, 110841 (2022)

[119] Markidis S, "The Old and the New: Can Physics-Informed Deep-Learning Replace Traditional Linear Solvers?" *ArXiv*, arXiv:2103.09655v2 (2021)

[120] Eivazi H, Wang Y, Vinuesa R, "Physics-informed deep-learning applications to experimental fluid mechanics," *Meas. Sci. Technol.* **35** 075303 (2024)

[121] Sanderse B, Stinis P, Maulik R, Ahmed SE, "Scientific machine learning for closure models in multiscale problems: a review," *ArXiv*, arXiv:2403.02913 (2024)

[122] Pope SB, "Turbulent Flows," *Cambridge: Cambridge University Press*, 2000.

[123] Freidberg JP, "The ideal MHD model," In: Ideal MHD. *Cambridge: Cambridge University Press*, 2014.

[124] Hara K, Hanquist K, "Test cases for grid-based direct kinetic modeling of plasma flows," *Plasma Sources Sci. Technol.* **27**, 065004 (2018)

[125] Bar-Sinai Y, Hoyer S, Hickey J, Brenner MP, "Learning data-driven discretizations for partial differential equations," *PNAS*, **116** (31) 15344-15349 (2019)

[126] Zhuang J, Kochkov D, Bar-Sinai Y, Brenner MP, Hoyer S, "Learned discretizations for passive scalar advection in a two-dimensional turbulent flow," *Phys. Rev. Fluids* **6**, 064605 (2021)

[127] Stevens B, Colonius T, "Enhancement of shock-capturing methods via machine learning," *Theor. Comput. Fluid Dyn.* **34**, 483–496 (2020)

[128] Stevens B, Colonius T, "FiniteNet: A fully convolutional LSTM network architecture for time-dependent partial differential equations," *arXiv preprint*, arXiv:2002.03014 (2020)

[129] Jeon J, Lee J, Kim SJ, "Finite volume method network for the acceleration of unsteady computational fluid dynamics: Non-reacting and reacting flows," *Int J Energy Res.* 46(8): 10770-10795 (2022)

[130] Kochkov D, Smith JA, Alieva A, Wang Q, Brenner MP, Hoyer S, "Machine learning–accelerated computational fluid dynamics," *PNAS*, **118** (21) e2101784118 (2021)

[131] Fukami K, Fukagata K, Taira K, "Super-resolution reconstruction of turbulent flows with machine learning," *Journal of Fluid Mechanics,* **870**:106–20 (2019)

[132] Li Z, Kovachki N, Azizzadenesheli K, Liu B, Bhattacharya K, Stuart A, Anandkumar A, "Fourier neural operator for parametric partial differential equations," *arXiv preprint*, arXiv:2010.08895 (2020)

[133] Tang W, Shan T, Dang X, Li M, Yang F, et al., "Study on a Poisson's equation solver based on deep learning technique," *In Proc. 2017 IEEE Electrical Design of Advanced Packaging and Systems Symposium (EDAPS)*, Haining, China, pp. 1-3 (2017)

[134] Ajuria Illarramendi E, Alguacil A, Bauerheim M, Misdariis A, Cuenot B, Benazera E., "Towards a hybrid computational strategy based on Deep Learning for incompressible flows," AIAA 2020-3058, *AIAA AVIATION 2020 FORUM* (2020)

[135] Levine M, Stuart A, "A framework for machine learning of model error in dynamical systems," *Communications of the American Mathematical Society*, **2**(07):283-344 (2022)

[136] Smolyakov AI, Chapurin O, Frias W, Koshkarov O, Romadanov I et al., "Fluid theory and simulations of instabilities, turbulent transport and coherent structures in partially-magnetized plasmas of discharges," *Plasma Physics and Controlled Fusion* **59**(1):014041 (2016)

[137] Alvarez Laguna A, Esteves B, Bourdon A, Chabert P, "A regularized high-order moment model to capture non-Maxwellian electron energy distribution function effects in partially ionized plasmas," *Phys. Plasmas* **29** (8): 083507 (2022)

[138] MacArt JF, Sirignano J, Panesi M, "Deep Learning Closure of the Navier-Stokes Equations for Transitional Flows," AIAA 2022-1703, *AIAA SCITECH 2022 Forum* (2022)

[139] Duraisamy K, Iaccarino G, Xiao H, "Turbulence modeling in the age of data," *Annual review of fluid mechanics* **51**:357-77 (2019)

[140] Ling J, Kurzawski A, Templeton J, "Reynolds averaged turbulence modelling using deep neural networks with embedded invariance," *Journal of Fluid Mechanics* **807**:155–66 (2016)

[141] Novati G, de Laroussilhe HL, Koumoutsakos P, "Automating turbulence modelling by multi-agent reinforcement learning," *Nat Mach Intell* **3**, 87–96 (2021)

[142] Taira K, Brunton SL, Dawson STM, Rowley CW, Colonius T, McKeon BJ et al., "Modal Analysis of Fluid Flows: An Overview," *AIAA Journal* **55**:12, 4013-4041 (2017)

[143] Murata T, Fukami K, Fukagata K, "Nonlinear mode decomposition with convolutional neural networks for fluid dynamics," *Journal of Fluid Mechanics* **882**:A13 (2020)

[144] Eivazi H, Le Clainche S, Hoyas S, Vinuesa R, "Towards extraction of orthogonal and parsimonious non-linear modes from turbulent flows," *Expert Systems with Applications*, **202**:117038 (2022)

[145] Lee K, Carlberg KT, "Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders," *Journal of Computational Physics* **404**:108973 (2020)

[146] Benner P, Gugercin S, Willcox K, "A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems," *SIAM Review* **57**:4, 483-531 (2015)

[147] Cenedese M, Axas J, Bauerlein B et al., "Data-driven modeling and prediction of non-linearizable dynamics via spectral submanifolds," *Nat Commun* **13**, 872 (2022)

[148] Kaiser E, Noack BR, Cordier L, Spohn A, Segond M, Abel M et al., "Cluster-based reduced-order modelling of a mixing layer," *Journal of Fluid Mechanics* **754**:365–414 (2014)

[149] Amsallem D, Zahr MJ, Farhat C, "Nonlinear model order reduction based on local reduced-order bases," *Int. J. Numer. Meth. Engng* **92**: 891-916 (2012)

[150] Schmid PJ, "Dynamic mode decomposition of numerical and experimental data," *Journal of Fluid Mechanics*, **656**:5–28 (2010)

[151] Kutz JN, Brunton SL, Brunton BW, Proctor JL, "Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems," Philadelphia, PA: Society for Industrial and Applied Mathematics; Series: Other titles in applied mathematics; 149 (2016)

[152] Rowley CW, Mezic I, Bagheri S et al, "Spectral analysis of nonlinear flows," *Journal of Fluid Mechanics*, **641**:115–27 (2009)

[153] Vlachas PR, Byeon W, Wan ZY, Sapsis TP, Koumoutsakos P, "Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks", *Proc. R. Soc. A.* **474**:20170844 (2018)

[154] Wu H, Mardt A, Pasquali L, Noe F, "Deep generative Markov state models," *Advances in Neural Information Processing Systems*, **31** (2018)

[155] Kim B, Azevedo VC, Thuerey N, Kim T, Gross M, Solenthaler B, "Deep Fluids: A Generative Network for Parameterized Fluid Simulations," *Computer Graphics Forum*, **38**: 59-70 (2019)

[156] Coscia D, Demo N, Rozza G, "Generative adversarial reduced order modelling," *Sci Rep* **14**, 3826 (2024)

[157] Peherstorfer B, Willcox K, "Data-driven operator inference for nonintrusive projection-based model reduction," *Computer Methods in Applied Mechanics and Engineering*, **306**:196-215 (2016)

[158] Brunton SL, Proctor JL, Kutz JN, "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," *PNAS* **113** (15) 3932-3937 (2016)

[159] Mezic I, "Analysis of fluid flows via spectral properties of the Koopman operator," *Annu. Rev. Fluid Mech.* **45**:357-378 (2013)

[160] Lungo GV, Santoni-Ortiz C, Abkar M, Porté-Agel F, Rotea MA, Leonardi S, "Data-driven reduced order model for prediction of wind turbine wakes," *Journal of Physics: Conference Series* **625**:012009 (2015)

[161] Huang C, Anderson WE, Harvazinski ME, Sankaran V, "Analysis of self-excited combustion instabilities using decomposition techniques," *In 51st AIAA Aerospace Sciences Meeting*, pages 1–18 (2013)

[162] Sayadi T, Schmid PJ, Nichols JW, Moin P, "Reduced-order representation of near-wall structures in the late transitional boundary layer," *Journal of Fluid Mechanics* **748**:278–301 (2014)

[163] Schmid PJ, Violato D, Scarano F, "Decomposition of time-resolved tomographic PIV," *Experiments in Fluids* **52**:1567–1579 (2012)

[164] Kutz JN, Fu X, Brunton SL, "Multiresolution Dynamic Mode Decomposition," *SIAM Journal on Applied Dynamical Systems* **15**:2, 713-735 (2016)

[165] Le Clainchey S, Vegay JM, "Higher Order Dynamic Mode Decomposition," SIAM *Journal on Applied Dynamical Systems* **16**:2, 882-925 (2017)

[166] Andreuzzi F, Demo N, Rozza G, "A dynamic mode decomposition extension for the forecasting of parametric dynamical systems," *SIAM Journal on Applied Dynamical Systems* **22**(3):2432-58 (2023)

[167] Askham T, Kutz JN, "Variable projection methods for an optimized dynamic mode decomposition," *SIAM Journal on Applied Dynamical Systems* **17**:1, 380-416 (2018)

[168] Sashidhar D, Kutz JN, "Bagging, optimized dynamic mode decomposition for robust, stable forecasting with spatial and temporal uncertainty quantification," *Phil. Trans. R. Soc. A* **380**:20210199 (2022)

[169] Baddoo PJ, Herrmann B, McKeon BJ, Kutz JN, Brunton SL, "Physics-informed dynamic mode decomposition," *Proceedings of the Royal Society A* **479**, no. 2271: 20220576 (2023)

[170] Koopman BO, "Hamiltonian systems and transformation in Hilbert space," *Proceedings of the National Academy of Sciences* **17**(5):315–318 (1931)

[171] Yeung E, Kundu S, Hodas N, "Learning deep neural network representations for Koopman operators of nonlinear dynamical systems," In 2019 American Control Conference (ACC), *IEEE*, pp. 4832-4839 (2019)

[172] Takeishi N, Kawahara Y, Yairi T, "Learning Koopman invariant subspaces for dynamic mode decomposition," *Advances in neural information processing systems* **30** (2017)

[173] Lusch B, Kutz JN, Brunton SL, "Deep learning for universal linear embeddings of nonlinear dynamics," *Nature Communications*, vol. **9**, no. 1, p. 4950 (2018)

[174] Mardt A, Pasquali L, Wu H, Noe F, "VAMPnets: Deep learning of molecular kinetics," *Nature Communications*, vol. **9**, no. 5 (2018)

[175] Williams MO, Kevrekidis IG, Rowley CW, "A data-driven approximation of the Koopman operator: Extending dynamic mode decomposition," *Journal of Nonlinear Science* **25** 1307-1346 (2015)

[176] Li Q, Dietrich F, Bollt EM, Kevrekidis IG, "Extended dynamic mode decomposition with dictionary learning: A data-driven adaptive spectral decomposition of the Koopman operator," *Journal of Nonlinear Science*, **27**(10):103111 (2017)

[177] Murari A, Peluso E, Spolladore L, Rossi R, Gelfusa M, "Upgrades of Genetic Programming for Data-Driven Modeling of Time Series," *Evol. Comput.* **31** (4): 401–432 (2023)

[178] Schmidt M, Lipson H, "Distilling Free-Form Natural Laws from Experimental Data," *Science* **324**, 81-85 (2009)

[179] Rudy SH, Brunton SL, Proctor JL, Kutz JN, "Data-driven discovery of partial differential equations," *Sci. Adv.* **3**, e1602614 (2017)

[180] Hayden S, "Learning partial differential equations via data discovery and sparse optimization," *Proc. R. Soc. A.* **473**: 20160446 (2017)

[181] Loiseau J-C, Brunton SL, "Constrained sparse Galerkin regression," *Journal of Fluid Mechanics*, **838**:42–67 (2018)

[182] Champion K, Zheng P, Aravkin AY, Brunton SL, Kutz JN, "A Unified Sparse Optimization Framework to Learn Parsimonious Physics-Informed Models From Data," *in IEEE Access*, vol. **8**, pp. 169259-169271 (2020)

[183] Fasel U, Kutz JN, Brunton BW, Brunton SL, "Ensemble-SINDy: Robust sparse model discovery in the low-data, high-noise limit, with active learning and control," *Proc Math Phys Eng Sci*. **478**(2260):20210904 (2022)

[184] Messenger DA, Bortz DM, "Weak SINDy for partial differential equations," *Journal of Computational Physics* **443**:110525 (2021)

[185] Kaheman K, Kutz JN, Brunton SL, "SINDy-PI: a robust algorithm for parallel implicit sparse identification of nonlinear dynamics," *Proc. R. Soc. A.* **476**: 20200279 (2020)

[186] Naozuka GT, Rocha HL, Silva RS et al., "SINDy-SA framework: enhancing nonlinear system identification with sensitivity analysis," *Nonlinear Dyn* **110**, 2589–2609 (2022)

[187] Kaheman K, Brunton SL, Kutz NJ, "Automatic differentiation to simultaneously identify nonlinear dynamics and extract noise probability distributions from data," *Mach. Learn.: Sci. Technol.* **3** 015031 (2022)

[188] Wentz J, Doostan A, "Derivative-based SINDy (DSINDy): Addressing the challenge of discovering governing equations from noisy data," *Computer Methods in Applied Mechanics and Engineering*, **413**:116096 (2023)

[189] Beetham S, Capecelatro J, "Formulating turbulence closures using sparse regression with embedded form invariance," *Physical Review Fluids*, **5**(8):084611 (2020)

[190] Callaham JL, Rigas G, Loiseau J-C, Brunton SL, "An empirical mean-field model of symmetry-breaking in a turbulent wake", *Science Advances*, **8**(19) (2022)

[191] Kaiser E, Kutz JN, Brunton SL, "Sparse identification of nonlinear dynamics for model predictive control in the low-data limit," *Proceedings of the Royal Society A.*, **474**: 20180335 (2018)

[192] Mars Gao L, Kutz JN, "Bayesian autoencoders for data-driven discovery of coordinates, governing equations and fundamental constants," *Proceedings of the Royal Society A*, **480**: 20230506 (2024)

[193] Williams JP, Zahn O, Kutz JN, "Sensing with shallow recurrent decoder networks," *Proc. R. Soc. A.* **480**:20240054 (2024)

[194] Ebers MR, Williams JP, Steele KM, Kutz JN, "Leveraging arbitrary mobile sensor trajectories with shallow recurrent decoder networks for full-state reconstruction," *ArXiv preprint*, arXiv:2307.11793 (2023)

[195] Kutz JN, Reza M, Faraji F, Knoll A, "Shallow Recurrent Decoder for Reduced Order Modeling of Plasma Dynamics," *ArXiv preprint*, arXiv:2405.11955 (2024)

[196] Anirudh R, Archibald R, Asif MS, Becker MM, Benkadda S, Bremer PT, Bude RH, Chang CS, Chen L, Churchill RM, Citrin J et al., "2022 review of data-driven plasma science," *IEEE Transactions on Plasma Science*, vol. **51**, no. 7, pp. 1750-1838 (2023)

[197] Trieschmann J, Vialetto L, Gergs T, "Review: Machine learning for advancing low-temperature plasma modeling and simulation," *Journal of Micro/Nanopatterning, Materials, and Metrology*, Vol. **22**, Issue 4, 041504 (2023)

[198] Bonzanini AD, Shao K, Graves DB, Hamaguchi S, Mesbah A, "Foundations of machine learning for low-temperature plasmas: methods and case studies," *Plasma Sources Science and Technology*, **32** 024003 (2023)

[199] Trieschmann J, Gergs T, Liu Y, Mussenbrock T, "A machine learning approach to the solution of Poisson's equations for plasma simulations," In: Bulletin of the American Physical Society. Vol **65**. *American Physical Society* (2020)

[200] Aguilar X, Markidis S, "A Deep Learning-Based Particle-in-Cell Method for Plasma Simulations," *in 2021 IEEE International Conference on Cluster Computing (CLUSTER)*, Portland, OR, USA, pp. 692-697 (2021)

[201] Kube R, Churchill RM, Sturdevant B, "Machine learning accelerated particle-in-cell plasma simulations," *ArXiv preprint*, arXiv:2110.12444 (2021)

[202] Nicolini JL, Na D-Y, Teixeira FL, "Model Order Reduction of Electromagnetic Particle-in-Cell Kinetic Plasma Simulations via Proper Orthogonal Decomposition," *in IEEE Transactions on Plasma Science*, vol. **47**, no. 12, pp. 5239-5250 (2019)

[203] Nayak I, Teixeira FL, "Dynamic Mode Decomposition for Prediction of Kinetic Plasma Behavior," *2020 International Applied Computational Electromagnetics Society Symposium (ACES)*, Monterey, CA, USA, pp. 1-2 (2020)

[204] Nayak I, Kumar M, Teixeira FL, "Detection and prediction of equilibrium states in kinetic plasma simulations via mode tracking using reduced-order dynamic mode decomposition," *Journal of Computational Physics*, **447**:110671 (2021)

[205] Badiali C, Bilbao PJ, Cruz F, Silva LO, "Machine-learning-based models in particle-in-cell codes for advanced physics extensions," *Journal of Plasma Physics*, **88**(6):895880602 (2022)

[206] Zhong L, Gu Q, Wu B, "Deep learning for thermal plasma simulation: Solving 1-D arc model as an example," *Computer Physics Communications*, **257**:107496 (2020)

[207] Sasaki M, Kawachi Y, Dendy RO, Arakawa H, Kasuya N et al., "Using dynamical mode decomposition to extract the limit cycle dynamics of modulated turbulence in a plasma simulation," *Plasma Physics and Controlled Fusion*, **61** 112001 (2019)

[208] Shivamoggi BK, "Kelvin-Helmholtz instability of a plasma in a magnetic field: finite-resistivity effects," *Physica Scripta*. **24** 49 (1981)

[209] Taylor R, Kutz JN, Morgan K, Nelson BA, "Dynamic mode decomposition for plasma diagnostics and validation," *Rev Sci Instrum* **89**, 053501 (2018)

[210] Kaptanoglu AA, Morgan KD, Hansen CJ, Brunton SL, "Characterizing Magnetized Plasmas with Dynamic Mode Decomposition," *Physics of Plasmas* **27**, 032108 (2020)

[211] Maddaloni D, Domínguez-Vázquez A, Terragni F, Merino M, "Data-driven analysis of oscillations in Hall thruster simulations," *Plasma Sources Sci. Technol.* **31** 045026 (2022)

[212] Köhne S, Boella E, Innocenti ME, "Unsupervised classification of fully kinetic simulations of plasmoid instability using self-organizing maps (SOMs)," *Journal of Plasma Physics*, **89**(3):895890301 (2023)

[213] Comisso L, Lingam M, Huang YM, Bhattacharjee A, "General theory of the plasmoid instability," *Physics of Plasmas*. **23**, 100702 (2016)

[214] Kaptanoglu AA, Morgan KD, Hansen CJ, Brunton SL, "Physics-constrained, low-dimensional models for magnetohydrodynamics: First-principles and data-driven approaches," *Physical Review E*, **104** 015206 (2021)

[215] Kaptanoglu AA, Hansen C, Lore JD, Landreman M, Brunton SL, "Sparse regression for plasma physics," *Physics of Plasmas*, **30** 033906 (2023)

[216] Alves EP, Fiuza F, "Data-driven discovery of reduced plasma physics models from fully kinetic simulations," *Physical Review Research*, **4**, 033192 (2022)

[217] Abramovic I, Alves EP, Greenwald M, "Data-driven model discovery for plasma turbulence modelling," *Journal of Plasma Physics*, **88**(6):895880604 (2022)

[218] Camargo SJ, Biskamp D, Scott BD, "Resistive drift-wave turbulence," *Physics of Plasmas*, **2**(1):48-62 (1995)

[219] Donaghy J, Germaschewski K, "In search of a data-driven symbolic multi-fluid ten-moment model closure," *Journal of Plasma Physics*, **89**(1):895890105 (2023)

[220] Van de Plassche KL, Citrin J, Bourdelle C, Camenen Y, Casson FJ, Dagnelie VI, Felici F, Ho A, Van Mulders S, JET Contributors, "Fast modeling of turbulent transport in fusion plasmas using neural networks," *Phys. Plasmas* **27**(2): 022310 (2020)

[221] Heinonen RA, Diamond PH, "Turbulence model reduction by deep learning," *Phys. Rev. E* **101**, 061201(R) (2020)

[222] Baudat G, Anouar F, "Kernel-based methods and function approximation," *Proceedings of the International Joint Conference on Neural Networks*, **2**:1244–1249 (2001)

[223] Burges CJ, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, **2**:121–167 (1998)

[224] Mika S, Ham J, Lee DD, Schölkopf B, "A kernel view of the dimensionality reduction of manifolds," *Proceedings of the 21st International Conference on Machine Learning*, page 47 (2004)

[225] Schölkopf B, Hofmann, Smola A, "Kernel methods in machine learning," *Annals of Statistics*, **36**:1171–1220 (2008)

[226] Askham T, duqbo/optdmd: optdmdv1.0.0, https://doi.org/10.5281/zenodo.439385 (2017)

[227] Huhn QA, Tano ME, Ragusa JC, Choi Y, "Parametric dynamic mode decomposition for reduced order modeling," *Journal of Computational Physics*, vol. **475** 111852 (2023)

[228] Sayadi T, Schmid PJ, Richecoeur F, Durox D, "Parametrized data-driven decomposition for bifurcation analysis, with application to thermo-acoustically unstable systems," *Physics of Fluids* **27**, 037102 (2015)

[229] Faraji F, Reza M, Knoll A, "Enhancing one-dimensional particle-in-cell simulations to self-consistently resolve instability-induced electron transport in Hall thrusters," *J. Appl. Phys.* **131**, 193302 (2022)

[230] Reza M, Faraji F, Knoll A, "Resolving multi-dimensional plasma phenomena in Hall thrusters using the reduced-order particle-in-cell scheme," *J Electr Propuls* **1**, 19 (2022)

[231] Gavish M, Donoho DL, "The Optimal Hard Threshold for Singular Values is $4/\sqrt{3}$," *in IEEE Transactions on Information Theory*, vol. 60, no. 8, pp. 5040-5053, (2014)

[232] Petronio F, "Plasma instabilities in Hall Thrusters: a theoretical and numerical study," PhD dissertation, Paris Polytechnic Institute, NNT: 2023IPPAX030 (2023)

[233] Lorenz EN, "Deterministic Non-periodic Flow," *Journal of the Atmospheric Sciences*, vol. **20**, pg. 130-140 (1963)

[234] URL to the "ViscousFlow.jl" tutorial: https://juliaibpm.github.io/ViscousFlow.jl/latest/, last accessed in March 2024.

[235] Goel T, Haftka RT, Shyy W, Queipo NV, "Ensemble of surrogates," *Struct Multidisc Optim* **33**, 199–216 (2007)

[236] Brooks JW, Kaptanoglu AA, McDonald MS, "A comparison of Fourier and POD mode decomposition methods for high-speed Hall thruster video," *Frontiers in Space Technologies*, vol. **4** (2023)

[237] Franke R, Rodi W, Schönung B, "Numerical calculation of laminar vortex-shedding flow past cylinders," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. **35**, Pages 237-257 (1990)

[238] Amsallem D, Farhat C, "Interpolation Method for Adapting Reduced-Order Models and Application to Aeroelasticity," *AIAA Journal*, vol. 46, No. 7 (2008)

[239] Marks TA, Jorns BA, "Challenges with the self-consistent implementation of closure models for anomalous electron transport in fluid simulations of Hall thrusters," *Plasma Sources Sci. Technol.* **32** 045016 (2023)

[240] Marks TA, Jorns BA, "Evaluation of several first-principles closure models for Hall thruster anomalous transport," *In Proceedings of AIAA SciTech Forum 2023*, National Harbor, MD & Online (2023)

[241] Zheng P, Askham T, Brunton SL, Kutz JN, Aravkin AY, "A unified framework for sparse relaxed regularized regression: SR3," *IEEE Access*, **7**:1404-23 (2018)

[242] Hirsh SM, Barajas-Solano DA, Kutz JN, "Sparsifying priors for Bayesian uncertainty quantification in model discovery," *R. Soc. Open Sci.* **9**: 211823 (2022)

[243] Singh M, Fuenmayor E, Hinchy EP, Qiao Y, Murray N, Devine D, "Digital Twin: Origin to Future," *Applied System Innovation*. **4**(2):36 (2021)

[244] Gelernter D, "Mirror Worlds: Or: The Day Software Puts the Universe in a Shoebox. How it Will Happen and What it Will Mean," *Oxford University Press*: Oxford, UK (1993)

[245] Grieves M, "Origins of the Digital Twin Concept," (2016) DOI: 10.13140/ RG.2.2.26367.61609.

[246] Grieves M, "Back to the future: Product lifecycle management and the virtualization of product information," *In Product Realization*. Springer: Boston, MA, USA, pp. 1–13, 2009.

[247] Shafto M, Conroy M, Doyle R, Glaessgen E, Kemp C et al., "Draft modeling, simulation, information technology & processing roadmap," *Technol. Area*, **11** (2010)

[248] Rosen R, von Wichert G, Lo G, Bettenhausen KD, "About the Importance of Autonomy and Digital Twins for the Future of Manufacturing," *IFAC-PapersOnLine*, **48**(3), pp. 567–572 (2015)

[249] National Academies of Sciences, Engineering, and Medicine, "Foundational Research Gaps and Future Directions for Digital Twins," Washington, DC: *The National Academies Press* (2023)

[250] Bonney MS, Wagg D, "Historical Perspective of the Development of Digital Twins," In: Epp, D.S. (eds) Special Topics in Structural Dynamics & Experimental Techniques, Volume 5. Conference Proceedings of the Society for Experimental Mechanics Series. *Springer*, Cham (2022)

[251] Wagg DJ, Worden K, Barthorpe RJ, Gardner P, "Digital Twins: State-of-the-Art and Future Directions for Modeling and Simulation in Engineering Dynamics Applications," *ASME. ASME J. Risk Uncertainty Part B*. **6**(3): 030901 (2020)

[252] Iglesias D, Bunting P, Esquembri S, Hollocombe J, Silburn S et al., "Digital twin applications for the JET divertor," *Fusion Engineering and Design*. Vol. **125**, pg. 71-76 (2017)

Appendix

A.1. Nonlinear dimensionality reduction of data using autoencoders

In this appendix, following on the results and discussions in section 3.4, we aim to demonstrate the capabilities of autoencoders in providing a highly low-rank (low-dimensional) representation of data from a complex plasma system in comparison against what was shown to be achievable using SVD in section 3.4.

We have used the same dataset as that described in sections 3.3 and 4.1.3. We recall that this dataset is from a high-fidelity quasi-2D PIC simulation of the radial-azimuthal benchmark test problem [43] (see section 3.3 and the information of test case 1 in Table 3.2). Similar to section 3.4, we have applied the autoencoders to the data of the electrons' axial current density (J_{ey}, representing electrons' cross-magnetic-field transport). The Q2D output data of the J_{ey} were reconstructed on a 2D grid with the size of 256 × 256 along each direction of the domain (radial and azimuthal). Hence, the total dimensionality of the "ground-truth" snapshots is $n = 256 \times 256 = 65536$.

The results of applying SVD to this dataset is identical to those presented and discussed at length in section 3.4. In that section, we observed that a rank-100 truncated representation of the data overall yields highly faithful reconstructions, which compare very closely against the ground-truth data. Considering the dimensionality of the original ground-truth snapshots, a rank-100 representation implies that the dimensionality is reduced by a factor of more than 600.

With this in mind, we compare the performance of SVD against three autoencoder architectures with an increasing number of fully connected hidden layers and a progressively reduced number of latent variables in the autoencoder's bottleneck layer. The three autoencoder architectures are schematically illustrated in Fig. A.1. Through the comparison of the performance, we primarily aim to evaluate the extent to which we can further lower the dimensionality of our data using nonlinear autoencoders without losing the ability to have an accurate reconstruction of the ground-truth.

Before that, however, according to the discussions in section 2.2, the shallow linear autoencoder architecture depicted in Fig. A.1(a) shall be the same as SVD, with the SVD rank number (r) being equivalent to the number of latent variables (n_ζ) in the bottleneck layer of this autoencoder. Attesting this statement, Fig. A.2 illustrates that the reconstructed J_{ey} snapshots from the shallow linear autoencoder with n_ζ = 100 represent an identical resemblance to the SVD-reconstructed snapshots. Moreover, the shallow autoencoder provides an excellent reconstruction of the ground-truth, consistent with the SVD results presented here in Fig. A.2 as well as through the figures in section 3.4.

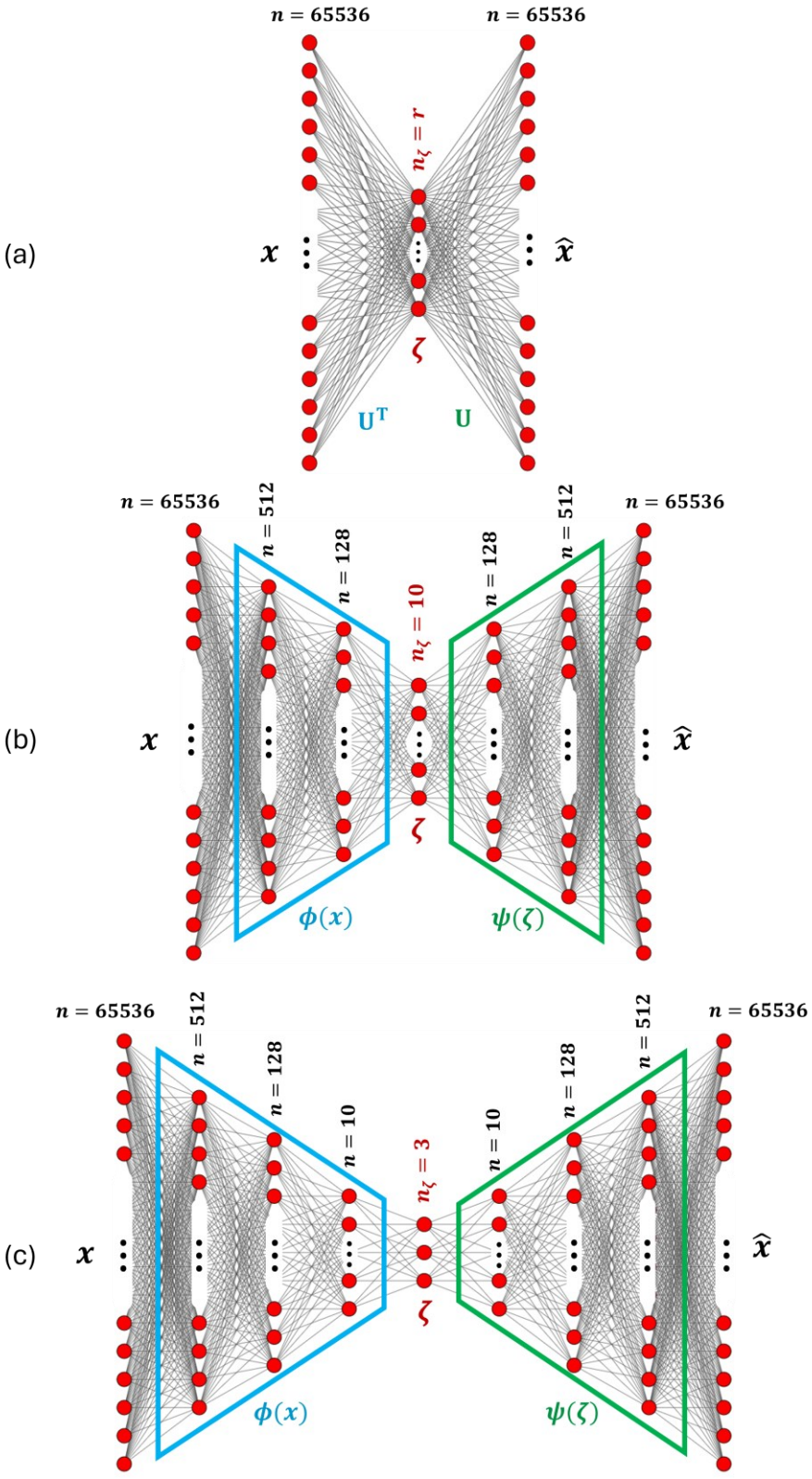


Fig. A.1 Schematic representation of the autoencoder networks applied to the data from the radial-azimuthal plasma test case; (a) shallow linear autoencoder, which is equivalent to SVD, (b) autoencoder with 5 hidden layers and 10 latent variables, (c) autoencoder with 7 hidden layers and 3 latent variables.

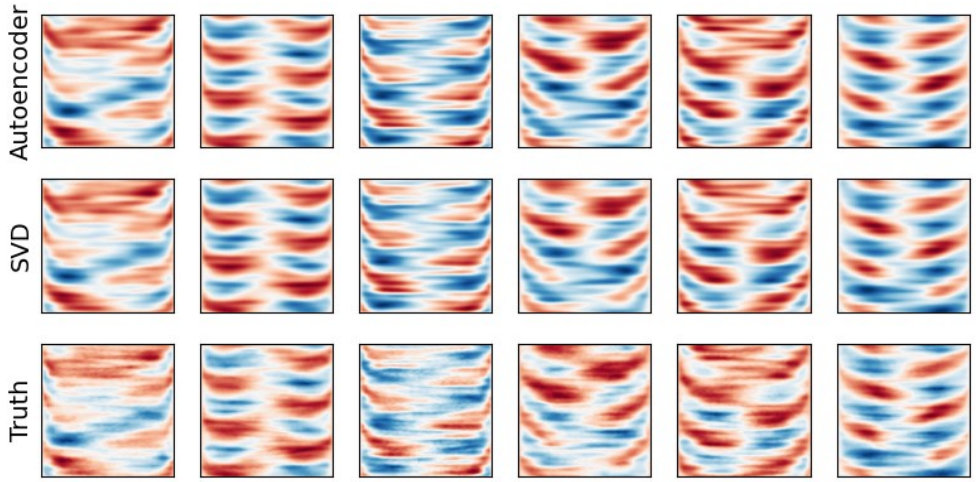


Fig. A.2 Comparison of the reconstructed normalized 2D snapshots of the electrons' axial current density (J_{ey}) from the shallow linear autoencoder (**top row**) and the SVD (**middle row**) with $r = n_\zeta = 100$ against the ground-truth snapshots from the PIC simulation (**bottom row**). Each column represents a random sample time instant within the test window. The horizontal and vertical axes represent the radial and the azimuthal directions, respectively.

We now refer to Fig. A.3, where we show snapshots of the J_{ey} data from the PIC simulation and from the autoencoder architectures of Fig. A.1 at several randomly selected time instants within an adopted test window.

Comparing the reconstructed snapshots from the shallow linear autoencoder with various numbers of latent space variables against the ground-truth ones, we observe that, with $n_\zeta = 10$, substantial loss of detailed information in the data occurs, and with merely 3 modes, the reconstruction is inadequate and not at all similar to the ground-truth. These comparisons clearly show that the reconstruction quality of the shallow liner autoencoder (essentially SVD) is poor with a low number of latent variables, failing to capture the essential details of the plasma data.

To address this, we utilize deeper nonlinear autoencoders. We see from Fig. A.3 that a 5-layer nonlinear autoencoder can reduce the dimensionality to 10 latent variables while maintaining an accurate representation of the data. The reconstruction quality has significantly improved with the 5-layer autoencoder with $n_\zeta = 10$ compared to what was observed with 10 modes from the single-layer architecture. The 5-layer autoencoder with $n_\zeta = 10$ has reduced the data dimensionality by more than 6000 times.

By employing a 7-layer nonlinear autoencoder, we can further reduce the dimensionality to 3 while still achieving a remarkably good representation of the data. This degree of compression corresponds to more than 20,000 times reduction in data dimensionality. Indeed, comparing against the ground-truth, the reconstructed snapshots of J_{ey} from the 7-layer autoencoder are seen to have retained a high level of detail. This demonstrates the capability of sufficiently deep networks to effectively represent complex plasma states in highly compressed latent spaces.

The great efficacy of deep autoencoders in providing a highly reduced nonlinear embedding of the data, as demonstrated here for a sample case of seemingly high-dimensional plasma data, underlines how these architectures can serve as an essential component for developing physics-inspired ML-based ROMs that can be interpretable and readily generalizable (refer to the discussions in subsection 2.1.2.3).

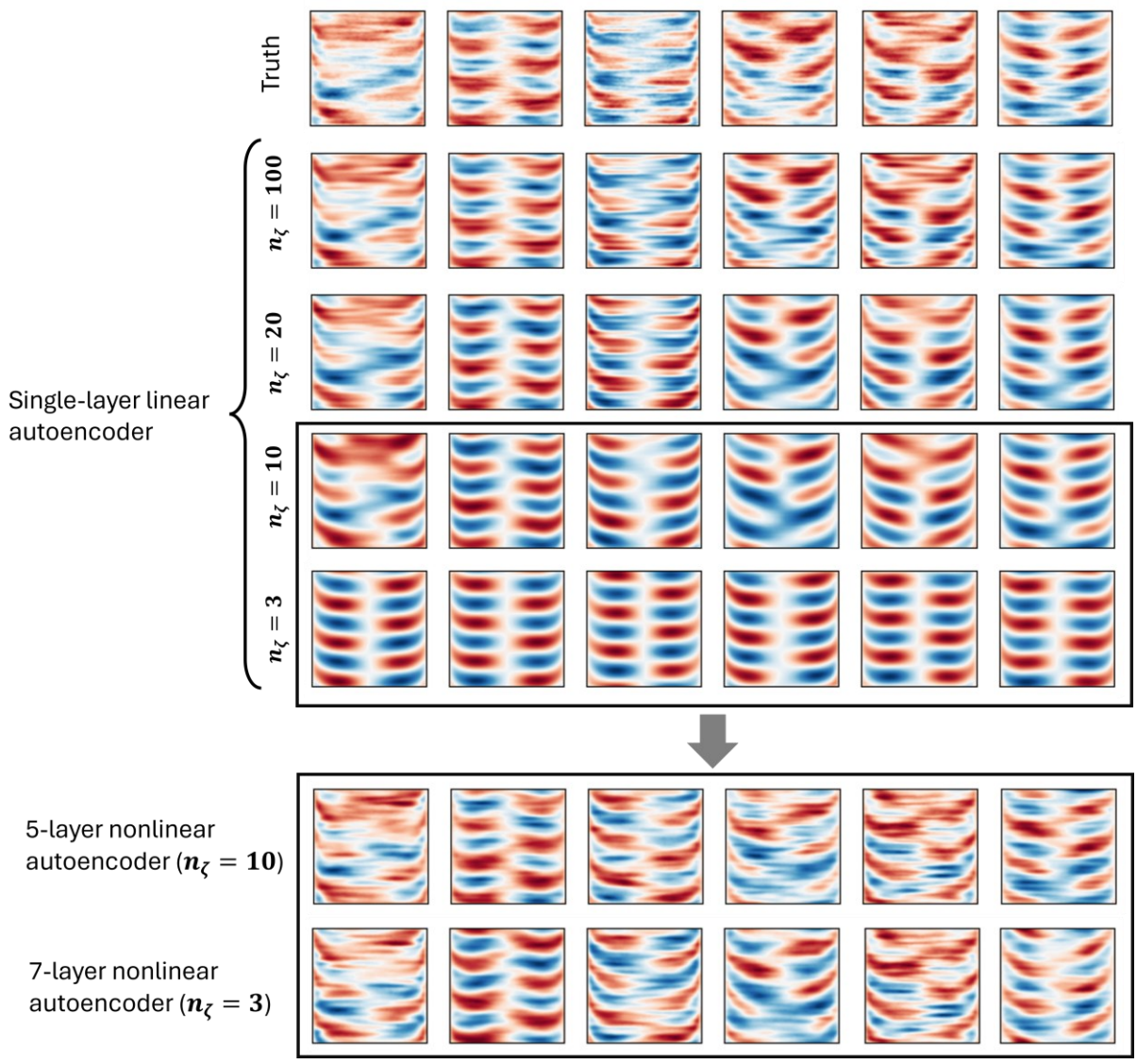


Fig. A.3 Demonstration of the data reconstruction from compressive representations obtained using the different autoencoder architectures shown in Fig. A.1 at several random time instants within the test window; (**top subplot**) reconstructions of the J_{ey} data using a single-layer linear autoencoder with varying number of latent variables compared against the ground-truth data snapshots, (**bottom subplot**) J_{ey} reconstructions using nonlinear autoencoders with different hidden layers and latent variables.