

Structure-preserving approaches and data-driven closure modeling for model order reduction

Présentée le 17 juin 2022

Faculté des sciences de base
Chaire de mathématiques computationnelles et science de la simulation
Programme doctoral en mathématiques

pour l'obtention du grade de Docteur ès Sciences

par

Nicolò RIPAMONTI

Acceptée sur proposition du jury

Prof. D. Kressner, président du jury
Prof. J. S. Hesthaven, directeur de thèse
Prof. E. Sonnendrücker, rapporteur
Prof. B. Haasdonk, rapporteur
Prof. F. Nobile, rapporteur

Acknowledgements

The Ph.D. years represent a long journey marked by inevitable ups and downs. Most people who embark on this adventure will likely experience a wide range of positive and negative emotions, with outcomes that are not always predictable. We can reorganize what is in our heads and around us during the darkest hours, making us ready to face what lies ahead with more strength and determination than expected. Similarly, periods of success and tranquility are often accompanied by feelings of inadequacy, estrangement from what one has done, or not having done our best. Further enriching this picture is the natural mixture of passion, frustration, joy, boredom, fear, love, and curiosity outside the doctoral path. However, I will never forget the extraordinary people I have met during these past years, and the following paragraphs serve to explain how important you have been to me.

My deepest gratitude goes to my thesis director, Prof. Jan S. Hesthaven, for several reasons. First, for believing in me and my abilities as a researcher and offering me the doctoral student position at EPFL. Coming into contact with or directly dealing with the various research projects conducted in the MCSS group has dramatically developed my mathematical knowledge. Second, your continuous support and scientific guidance, combined with the extreme freedom granted to me in my research work, allowed me, whenever I hit walls or stumbled (scientifically speaking), to learn how to deal with these situations, overcome them, and use them as a springboard to get better and better each time. Last but not least, you are a good person, and this is not always a given in research. Nothing grows in the desert, and the success you have achieved and the success those who have trained under your guidance are achieving is a partial reflection of the way you are and relate to others. This research project would not have been possible without you, and I will always keep you as an example in my future.

Special thanks go to the members of the jury defense, starting from Prof. Daniel Kressner, for accepting the role of president. I greatly appreciated the time spent by Prof. Fabio Nobile and Prof. Eric Sonnendrücker in reading my thesis, and your comments helped to ameliorate the dissertation. I want to reserve a special mention for Prof. Bernard Haasdonk: the discussions we had following the private defense of the thesis were of enormous value. Furthermore, the multiple interpretations and research insights offered from the content of my thesis pleasantly surprised me. They made me realize how exciting and flourishing this area of research is one more time.

A sincere acknowledgment goes to those who have contributed to this thesis, Qian, Ceci, and Babak, for being inestimable colleagues, coauthors, and friends. Each in your own way, through stimulating interactions, insights, and suggestions, you have accompanied me and made me grow over the years. Your imprint is evident in each of the chapters of this thesis.

Being a Ph.D. student in mathematics at EPFL also means having teaching duties. In my case, I was fortunate to assist Prof. Michel Cibils in organizing the courses Analysis 3 and 4. His dedication to teaching and his natural way of imparting knowledge to new generations of engineers are vital to EPFL and inspiring to me. Not mentioning you would not have been possible given the time spent together these past four years.

Acknowledgements

I would like to thank Prof. Simone Deparis for introducing me to EPFL 7 years ago with the Polimi-EPFL double degree program. Since then, you have always been a reference point within MATHICSE, and I thank you for the chats exchanged in front of the coffee machine.

My graduate life in EPFL has been more enjoyable because of the many people, colleagues, and friends that I have met here. Thank you Alice (who still believes I don't enjoy hiking), Alessandra (if you need anything, ask her, she has probably taken a course on whatever you need), Andrea (despite the countless "paccate"), Davide (to always be available for coffee at 7:30 am), Edino (for the afternoon breaks at SAT), Francesco (probably the only person in the world who has a Nespresso Pixie cup in "Volluto" color and drives tanks, how stylish!), Giacomo (for the unconditional admiration for Bernardeschi), Luca (bonus points when in mustache mode), Ondine (except when you speak in Italian with a French accent on purpose), Riccardo (the best SIAM president ever), Stefano (for your creativity with memes), Sundar (despite the unforgivable betrayal of MCSS), and all the people who have left their mark on me in the MATHICSE group, past and present. Honorable mention goes to Pego for putting up with me and my character all these years as a flatmate. I would be lying if I said I haven't missed you since you left EPFL.

Although Jan is formally the head of the MCSS group, the field commander who holds it together is Delphine. I am well aware that I have not been as cooperative as possible in bureaucratic matters, and for that, I apologize to you, but I think I have improved over time. You had always given me a helping hand when I needed it, and without you, I would probably still be looking for my office inside the math department after four years.

Over the years, I got to know several people in the MCSS group. Allan, Boris, Mengwu, Fabian, and Hossein, thank you very much for these years together. But there are some people for whom I would like to spend a few more words. Thank you so much, Cate, for always being, probably without you knowing it, a role model. Thank you so much, Niccolò and Qian, the "Chavannes' boys," for the time spent together and for that strong bond of friendship we formed with Deep and Zhen. Thank you very much, Mariella; it was a pleasure to welcome you back to the group and I admire your patience in teaching me how to climb despite my nonprogress. For each one who leaves, new ones join the group. Federico, Junming, Ricardo, Enrico, and Nicola, I wish you all the best in your future.

I also want to thank all my dear friends I have carried with me since my short stay in Stuttgart. Although the last few years have not been the best for traveling and meeting in person, we have managed to keep in touch, and sharing moments with you is always pleasant. Thank you, Alexis, Khouloud, Marko, Nadine, Shi, and Vincent.

It is said that true friendship endures time, space, and silence. I can confirm this. Every time I get a chance to meet my lifelong friends in Lodi, I am surprised at how much having us take different paths and live in different places has changed us. At the same time, however, I realize that deep down, we are still bound together by the same sincere friendship that has endured for 15 years and that I hope will last at least as long. I wish anyone to meet people like Agru, Andre, and Ciulza.

I leave the last paragraph, the most important one, to thank my family. You have helped me through the most challenging times without wanting anything in return, and the happiness you show in seeing me every time I come home is something that should never be taken for granted. Thank you for everything, even beyond your support in writing this thesis.

Lausanne,

Nicolò

Abstract

In this thesis, we explore and propose model order reduction techniques for high-dimensional differential equations that preserve structures and symmetries of the original problems and develop a closure modeling framework that leverages the Mori-Zwanzig formalism and recurrent neural networks. Partial differential equations play an essential role in describing multi-dimensional systems in many disciplines, including engineering, physics, chemistry, and economics. Since high-fidelity approximations of such models often result in a large number of degrees of freedom, the need for iterative evaluations for numerical optimizations and rapid feedback is computationally challenging. The resulting computational requirements are even more critical in the case of nonlinear, time-dependent problems. Among the various model order reduction techniques, the reduced basis method has successfully emerged to produce low-dimensional, stable, and accurate approximations for high-dimensional elliptic and parabolic problems.

The first part of this thesis is devoted to conserving the high-dimensional equation's invariants, symmetries, and structures during the reduction process. Traditional reduction techniques are not guaranteed to yield stable reduced systems, even if the target problem is stable. In the context of fluid flows, the skew-symmetric structure of the problem entails the preservation of the kinetic energy of the system. By recasting the high-fidelity problem in a pure skew-symmetric formulation at the discrete level, we show that the same structure is preserved with minor changes to the reduced basis technique even at the reduced model level. These conservative reduced models offer enhanced stability and accuracy and acquire physical significance by preserving a surrogate of the energy of the original problem. Next, we focus on Hamiltonian systems, which, being driven by symmetry, are a source of great interest in the reduction community. It is well known that the breaking of these symmetries in the reduced model is accompanied by a blowup of the system energy and flow volume. In this thesis, the traditional models proposed within the framework of geometric model reduction for Hamiltonian systems are further developed and combined with the dynamically orthogonal methods, developed by Sapsis and Lermusiaux (2009-2012). In this way, the solution is sought in a low dimensional space that evolves in time and whose rank evolves in time, thus allowing the issue of low reducibility in time of advection-dominated problems to be addressed. The reduced solution is expressed as a linear combination of a finite number of modes and coincides with the symplectic projection of the high-fidelity Hamiltonian problem onto the tangent space of the approximating manifold. An error surrogate is used to monitor the approximation ability of the reduced model and make a change in the rank of the approximating system if necessary. The method is further developed through a combination of discrete empirical interpolation and dynamic mode decomposition to reduce non-polynomial nonlinearities while preserving the symplectic structure of the problem and applied to the Vlasov-Poisson system.

In the second part of the thesis, we consider several data-driven methods to address the under-resolved regime problem in Galerkin reduced models. Trying to maintain the same computational efficiency as traditional reduced models, we introduce a reduced closure term to increase numerical accuracy. The closure term is developed systematically from the Mori-Zwanzig formalism by

Abstract

introducing projection operators on the spaces of retained and truncated modes, thus resulting in an additional memory integral term. The novelty of the approach lies in its application to the case of approximation by reduced basis methods, and it serves as a starting point for studying the influence on the resolved part of the unresolved part of the problem. The interaction turns out to be nonlocal in time and dominated by a high-dimensional orthogonal dynamics equation, which cannot be solved precisely and efficiently. Several classical methods in the field of statistical mechanics are used to approximate the memory term, exploiting the finiteness of the memory kernel support. In the case of reduced models using a reduced basis, we show that approximating and including this interaction in the model leads to a significant improvement in predicting the resolved part of the high-dimensional solution. We conclude this thesis by showing through numerical experiments how long short-term memory networks, i.e., machine learning structures characterized by feedback connections and capable of processing data sequences, represent a valid tool for approximating the memory term introduced through the Mori-Zwanzig formalism.

Keywords: Model order reduction, Reduced basis method, Energy and Structure conservation, Hamiltonian problems, Symplectic manifolds, Dynamical low-rank approximation, Vlasov-Poisson equation, Closure modeling, Mori-Zwanzig formalism, Deep learning.

Sommario

In questa tesi sviluppiamo tecniche di riduzione di modello che preservano strutture e simmetrie per equazioni alle derivate parziali ad alta dimensionalità e proponiamo un quadro di tecniche per la chiusura di modelli ridotti basato sul formalismo di Mori-Zwanzig e reti neurali ricorrenti. Le equazioni alle derivate parziali sono frequentemente utilizzate per la descrizione di sistemi complessi in diversi ambiti, ad esempio in ingegneria, fisica, chimica, ed economia. Approssimare le soluzioni di queste equazioni con accuratezza richiede un grande numero di gradi di libertà, rendendo computazionalmente oneroso risolvere problemi che richiedono una rapida soluzione o approssimazioni per diversi valori dei parametri, come succede in problemi di ottimizzazione. Questo problema risulta ancora più critico nel caso di problemi nonlineari e tempo-dipendenti. Tra le diverse tecniche di riduzione di modello, il metodo delle basi ridotte è emerso come valido strumento per la definizione di approssimazioni di dimensioni ridotte per problemi ellittici e parabolici.

La prima parte della tesi affronta il problema della conservazione di invarianti, simmetrie, e strutture durante il processo di riduzione. Le tecniche tradizionali di riduzione non garantiscono la stabilità del problema ridotto, anche a fronte di un problema da ridurre che risulta stabile. Nel caso di flussi di fluidi, l'antisimmetria dell'operatore usato per descrivere il problema fisico viene usata per dimostrare la conservazione dell'energia cinetica della soluzione. Riscrivendo il problema da ridurre in una forma puramente antisimmetrica a livello discreto, dimostriamo che la stessa struttura viene ereditata dal modello ridotto ottenuto con minime variazioni delle tecniche di riduzione standard. Il modello ridotto conservativo garantisce non solo una maggiore stabilità della soluzione ma acquisisce anche un significato fisico, andando a preservare un surrogato dell'energia del sistema originale. Successivamente ci concentriamo su sistemi Hamiltoniani, i quali, essendo caratterizzati da diverse simmetrie, hanno recentemente suscitato forte interesse nella comunità della riduzione di modello. Diversi studi numerici hanno dimostrato che non conservare queste simmetrie nel modello può comportare forti instabilità, come ad esempio crescite esponenziali dell'energia del sistema. In questa tesi, i metodi di riduzione geometrici per sistemi Hamiltoniani sono combinati con i metodi dinamici ortogonali sviluppati da Sapsis e Lermusiaux (2009-2012). In questo modo, la soluzione ridotta viene cercata in uno spazio di dimensione piccola che evolve nel tempo e il cui rango evolve nel tempo, risolvendo quindi il problema della limitata riducibilità nel tempo per problemi caratterizzati da forte convezione. Le soluzioni ridotte sono rappresentate dalla combinazione lineare di un numero finito di modi e coincide con la proiezione simplettica dell'originale problema Hamiltoniano sullo spazio tangente dello spazio approssimante. Un surrogato dell'errore viene utilizzato per controllare la capacità di approssimazione del modello ridotto e, nel caso, cambiare il rango del sistema approssimante. Il metodo è ulteriormente sviluppato tramite una combinazione dei metodi DEIM e DMD per trattare non linearità non polinomiali preservando allo stesso tempo la struttura simplettica del problema ed è stato applicato all'equazione di Vlasov-Poisson.

Nella seconda parte della tesi, consideriamo diversi metodi basati sulla collezione di dati per

risolvere il problema dei sistemi parzialmente risolti nel contesto delle proiezioni di Galerkin. Per mantenere l'efficienza computazionale dei modelli ridotti, introduciamo un termine di chiusura ridotto per migliorare l'accuratezza dell'approssimazione. Il termine di chiusura viene sviluppato in maniera sistematica a partire dal formalismo di Mori-Zwanzig introducendo operatori di proiezione sugli spazi relativi ai modi risolti e non risolti, i quali portano all'introduzione di un termine integrale di memoria. La novità dell'approccio proposto consiste nell'applicazione di queste tecniche nel caso di basi ridotte, e serve come punto di partenza per studiare l'influenza della parte non risolta sulla dinamica della parte risolta. L'interazione risulta essere dominata dall'equazione ad alta dimensionalità della dinamica ortogonale alla parte risolta ed è non locale in tempo, non potendo quindi essere risolta in maniera precisa ed efficiente. Diversi metodi nell'ambito della statistica meccanica sono utilizzati per approssimare il termine di memoria, sfruttando la finitezza del supporto del nucleo della memoria. Nel caso di modelli ridotti basati su basi ridotte, mostriamo che approssimare ed includere questo termine di memoria porta ad un significativo miglioramento dell'accuratezza della parte risolta del problema. Concludiamo questa tesi mostrando tramite esperimenti numerici come le reti neurali LSTM, caratterizzate da connessioni di feedback e usate per sequenze di dati, riescano ad approssimare in maniera efficace il termine di memoria dato dal formalismo di Mori-Zwanzig.

Parole chiave: Riduzione di modello, Metodo delle basi ridotte, Conservazione dell'energia e della struttura, Problemi Hamiltoniani, Varietà simplettiche, Approssimazioni dinamiche di rango basso, Equazioni di Vlasov-Poisson, Modelli di chiusura, Formalismo di Mori-Zwanzig, Apprendimento profondo.

Contents

Acknowledgements	i
Abstract (English/Italian)	iii
List of Figures	xi
List of Acronyms	xix
1 An Introduction to the Reduced Basis method	1
1.1 Motivations	1
1.2 The reduced basis method	3
1.2.1 Problem formulation and notation	3
1.3 Offline phase	5
1.3.1 Proper orthogonal decomposition	5
1.3.2 Greedy algorithm	8
1.4 Online phase	10
1.5 Efficient treatment of nonlinear term	12
I Structure-preserving Model Order Reduction	15
2 Model order reduction of fluid equations in skew-symmetric form	19
2.1 Skew symmetric and centered schemes for fluid flows	20
2.1.1 Conservation laws	20
2.1.2 Incompressible fluid	22
2.1.3 Compressible fluid	23
2.1.4 Time integration	25
2.2 Model reduction of fluid flow	25
2.2.1 Assembling nonlinear terms and time integration	27
2.3 Numerical experiments	28
2.3.1 Vortex merging	28
2.3.2 2D Kelvin-Helmholtz instability	31
2.3.3 1D Shock problem	32
2.3.4 Continuous variable resonance combustor	34
3 Model order reduction of Hamiltonian systems	43
3.1 Symplectic geometry and Hamiltonian systems	43
3.2 Symplectic Galerkin projection	47
3.3 Proper symplectic decomposition	50

3.3.1	SVD-based methods for orthonormal symplectic basis generation	51
3.3.2	SVD-based methods for non-orthonormal symplectic basis generation . . .	53
3.3.3	Greedy approach to symplectic basis generation	54
3.4	Extension to more general Hamiltonian problems	56
3.4.1	Dissipative Hamiltonian systems	56
3.4.2	Non-canonical Hamiltonian systems	58
4	Symplectic dynamical reduced basis	61
4.1	Problem formulation	63
4.2	Dynamical reduced basis method for Hamiltonian systems	64
4.3	Partitioned Runge-Kutta method	67
4.4	Reduced dynamics under rank-deficiency	69
4.5	Rank-adaptivity	71
4.5.1	Error indicator	72
4.5.2	Criterion for rank update	73
4.5.3	Update of the reduced state	74
4.5.4	Approximation properties of the rank-adaptive scheme	75
4.6	Computational complexity of the rank-adaptive algorithm	76
4.6.1	Efficient treatment of the polynomial nonlinearity	77
4.7	Numerical experiments	79
4.7.1	Shallow water equations	81
4.7.2	Nonlinear Schrödinger equations	89
5	Model order reduction of the Vlasov equation	95
5.1	Introduction	95
5.2	The physical model	97
5.2.1	Geometric particle-based discretization	98
5.3	Model order reduction of the Vlasov-Poisson problem	99
5.3.1	Dynamical structure-preserving MOR	100
5.4	Efficient treatment of nonlinear terms	102
5.4.1	Parameter sampling	104
5.4.2	DMD-DEIM approximation of the Hamiltonian gradient	105
5.4.3	DMD-DEIM reduced dynamics and computational complexity	110
5.5	Numerical experiments	111
5.5.1	Implementation and numerical study	113
5.5.2	Weak Landau damping of 1D Langmuir waves	115
5.5.3	Nonlinear Landau damping of 1D Langmuir waves	120
5.5.4	Two-stream instability	126
6	Conclusion of Part I	131
II	Closure Modeling Framework for Reduced Order Models	133
7	Mori-Zwanzig closure models	137
7.1	Example of interaction between different scales: energy scale identification	138
7.2	Mori-Zwanzig formalism	141
7.2.1	Introductory example	141
7.2.2	Mathematical foundations	142

7.3	Memory approximation	145
7.3.1	Study of the orthogonal dynamics	145
7.3.2	Memory Modelling	149
7.3.3	Estimates of the memory length	158
7.4	Numerical experiments	159
7.4.1	Randomized Cauchy problem	160
7.4.2	1D Burgers' equation	161
8	Recurrent neural network closure of parametric POD-Galerkin reduced-order models	167
8.1	Recurrent neural network memory model	168
8.1.1	Regression of the memory term	168
8.1.2	Conditioned long short-term memory network	168
8.1.3	Training of the network	170
8.1.4	Model selection	171
8.2	Parametric POD-Galerkin with the RNN memory model	172
8.2.1	POD-Galerkin with memory	172
8.2.2	Implicit-explicit Runge-Kutta time integration	172
8.3	Numerical results	174
8.3.1	3D Stokes	175
8.3.2	Kuramoto-Sivashinsky equation	181
8.3.3	Rayleigh-Bénard convection	185
9	Conclusion of Part II	197
	Bibliography	199
	Curriculum Vitae	219

List of Figures

1.1	Examples of samples from from an isotropic (a) and anisotropic (b) bivariate distribution.	7
2.1	VM: (a) Evolution of the kinetic energy K for the advective, divergence and skew-symmetric formulations of the incompressible Euler equation in the vortex merging setting. (b) Singular values of the snapshot matrix of the the solution to the FOM.	29
2.2	VM: Vorticity field at different times obtained from the FOM and the ROM with $n = 17$ and $n = 35$. Starting from three separated vortices, filamentous structures develop as a result of the interactions between the vortices as early as $t = 8$	30
2.3	VM: (a) Evolution of the velocity absolute error, as defined in (2.3.2), for different values of the basis dimension n . (b) Comparison of the time evolution of the reduced kinetic energy K_r for different values of n	31
2.4	KH: Evolution in time of the velocity error (2.3.2) between the high-fidelity solution and the reduced solution of the Kelvin-Helmholtz numerical experiment for different basis dimensions n	32
2.5	KH: Density field at different times obtained from the full-order model and the reduced model for $n = 200$ and $n = 500$. The instability developing from the velocity discontinuity across the interface is properly captured by the ROM solution.	33
2.6	KH: Evolution of the relative errors in the conservation of the mass (a), of the momentum (b), and of the total energy (c) for the solution of the ROM for different values of n	34
2.7	CE: Evolution in time of the mass (a), momentum (c), and energy (e) of the solution to the ROM in case of divergent, advective and skew-symmetric formulations for $n = 102$ and $n = 204$. In (b), (d), and (e) the evolutions of the same quantities are reported only in the case of skew-symmetric formulation for $n = 24$, $n = 102$ and $n = 204$	35
2.8	CE: Evolution in time of the error between the high-fidelity solution and the reduced solution of the compressible Euler numerical experiment for different basis dimensions n	35
2.9	CE: Comparison between the reduced solutions of the divergent and skew-symmetric formulations of the problems in terms of density and pressure at $t = 0.1$ ((a), (b)), $t = 0.3$ ((c), (d)), and $t = 1$ ((e), (f)). Results for the advective formulation are not showed here because the related reduced solutions are unstable after a few time steps.	36
2.10	CE: Singular values of the snapshot matrix S and S_{DEIM}	36

List of Figures

2.11	CE: Comparison between standard POD and POD with DEIM treatment of the nonlinear term in terms of the total error (a), mass error (b), momentum error (c), and kinetic energy error (d).	37
2.12	CVRC: Geometry of quasi-1D CVRC model.	38
2.13	Pressure profile of the steady state (a) and oscillatory mode of pressure located at $x = 0.36$ for the unsteady flow for the combustor model solution (b).	41
2.14	Absolute error between the high-fidelity and approximated pressure profiles (c) and visual comparison of the pressure oscillations at $x = 0.36$ (d).	42
4.1	Comparison of the distribution of computational costs required to solve parameter problems in a multi-query context. In particular, we report the cases of FOM, gROM, and dynamical low-rank model. We underline how in the case in which the problem is not globally reducible, as we have seen in Chapter 2, the efficiency of the classical reduction methods is compromised.	66
4.2	SWE-1D: (a) Singular values of the global snapshots matrix S^{u_h} and time average of the singular values of the local trajectories matrix $S_\tau^{u_h}$. The singular values are normalized using the largest singular values for each case. (b) ϵ -rank of the local trajectories matrix $S_\tau^{u_h}$ for different values of ϵ	82
4.3	SWE-1D: Relative error at time $T = 7$, as a function of the runtime for the complex SVD method (Global ROM), the dynamical RB method (Non adaptive), and the adaptive dynamical RB method for different values of the control parameters r and c . For the sake of comparison, we report the runtime required by the high-fidelity solver to compute the numerical solutions for all values of the parameter $\eta_h \in \Gamma_h$	83
4.4	SWE-1D: On the left column, we report the evolution of the error $E(t)$ for the adaptive and non adaptive dynamical RB methods for different values of the control parameter r and different dimensions $2n_1$ of the approximating manifold of the initial condition. The target error is obtained by solving the full model with initial condition obtained by projecting (4.7.8) onto a symplectic manifold of dimension $2n_1$. On the right column, we report the evolution of the dimension of the dynamical reduced basis over time. The adaptive algorithm is driven by the error indicator (4.5.4), while in the non adaptive setting, the dimension does not change with time. We consider the cases $2n_1 = 6$ (Figures (a)-(b)), $2n_1 = 8$ (Figures (c)-(d)), and $2n_1 = 10$ (Figures (e)-(f)).	85
4.5	SWE-1D: In Figures (a) and (b), we report the evolution of the projection error E_\perp for different values of the initial dimension $2n_1$ of the reduced manifold. In Figures (c) and (d), we report the corresponding evolution of the dimension of the reduced manifolds.	86
4.6	SWE-1D: Relative error (4.7.4) in the conservation of the discrete Hamiltonian (4.7.7) for the dynamical reduced basis method with initial reduced dimensions $2n_1 = 6$ (a), $2n_1 = 8$ (b), $2n_1 = 10$ (c), and $2n_1 = 12$ (d).	86
4.7	SWE-2D: (a) Singular values of the global snapshots matrix S^{u_h} and time average of the singular values of the local trajectories matrix $S_\tau^{u_h}$. The singular values are normalized using the largest singular values for each case. (b) ϵ -rank of the local trajectories matrix $S_\tau^{u_h}$ for different values of ϵ	87
4.8	SWE-2D: High fidelity solution (Figures (a)-(d)) and adaptive dynamical reduced solution (Figures (e)-(h)) for the parameter $(\alpha, \beta) = (\frac{1}{3}, \frac{17}{10})$ and $t = 0, 5, 15$, and 20s. In the adaptive reduced approach, we set $r = 1.1$, $c = 1.3$, and $2n_1 = 6$	88

4.9	SWE-2D: Error (4.7.3), at time $T = 20$, as a function of the runtime for the dynamical RB method and the adaptive dynamical RB method for different values of the control parameters r and c for the simulation of all the sampled parameters in Γ_h . For comparison, the high-fidelity model runtime is $3.3 \cdot 10^5$ s.	89
4.10	SWE-2D: On the left column, we report the evolution of the error $E(t)$ (4.7.3) for the adaptive and non adaptive dynamical RB methods for different values of the control parameters r and c , and for different dimensions $2n_1$ of the initial reduced manifold. The target error is obtained by solving the full model with initial condition obtained by projecting (4.7.10) onto a symplectic manifold of dimension $2n_1$. On the right column, we report the evolution of the dimension of the dynamical reduced basis over time. The adaptive algorithm is driven by the error indicator (4.5.4), while in the non adaptive setting, the dimension does not change with time. We consider the cases $2n_1 = 4$ (Figures (a)-(b)), $2n_1 = 6$ (Figures (c)-(d)), and $2n_1 = 8$ (Figures (e)-(f)).	90
4.11	NLS-2D: (a) Singular values of the global snapshots matrix S^{u_h} and of the time average of the local trajectories matrix $S_\tau^{u_h}$. The singular values are normalized using the largest singular value for each case. (b) ϵ -rank of the local trajectories matrix $S_\tau^{u_h}$ for different values of ϵ	91
4.12	NLS-2D: Evolution of the error in the orthogonality (a) and symplecticity (b) of the reduced basis obtained with the adaptive dynamical RB method for different choices of the control parameters r and r and initial dimension of the reduced manifold $2n_1$	92
4.13	NLS-2D: On the left column, we report the evolution of the error $E(t)$ (4.7.3) for the adaptive and non adaptive dynamical RB methods for different values of the control parameters r and c , and for different dimensions $2n_1$ of the initial reduced manifold. On the right column, we report the evolution of the dimension of the dynamical RB over time. We consider the cases $2n_1 = 6$ (Figures (a)-(b)) and $2n_1 = 8$ (Figures (c)-(d)).	93
5.1	LD: Initial positions and velocity distributions for selected values of the parameter in Γ_h (a) – (b). Exponential time decay of the electrostatic energy $\mathcal{E}(X_\tau^i; \eta_i)$ obtained from the full model solution, for selected values of η_i in Γ_h (c) . Since not all parameters in Γ_h are reported, the black lines in each subplot are used to mark the region where the plotted quantity is contained, for any value of the parameter in Γ_h	115
5.2	LD: Singular values of the global snapshots matrices S_X (a) and S_V (b) compared to the maximum and time average (in τ) of the singular values of the local matrices S_X^τ and S_V^τ	116
5.3	LD: Numerical rank of S_X^τ in (a) and S_V^τ in (b), as a function of τ . Different colors are associated with different values of the threshold, according to the legend. . .	116
5.4	LD: Evolution of the numerical ranks of S_X^τ (a) – (c) and of S_V^τ (d) – (f) for different threshold indicated by ϵ . In each subfigure, the rank behavior for different values of N_x and N , in the discretization of the full-order model, is compared. . .	117

List of Figures

- 5.5 LD: Evolution of the position (*a*) and velocity (*b*) relative errors, as defined in (5.5.7), for different choices of p^* and T . These errors are compared to the target values given by the position component $\varepsilon_{\text{rel},X}^{\text{Target}}$ and the velocity component $\varepsilon_{\text{rel},V}^{\text{Target}}$ of the relative projection errors defined in (5.5.8). The target reduced basis has dimension 4 and is computed, for each time step, using the Complex SVD algorithm, as described in Section 5.5.1. 117
- 5.6 LD: Damping rates of the exponential time decay of the electric energy $\mathcal{E}(X^i; \eta)$, defined in (5.3.2), as a function of the two-dimensional parameter $\eta = (\alpha, \sigma)$. The plots refer to (*a*) the full-order model; (*b*) the dynamical reduced model with $T = 3$ and $p^* = 16$; and (*c*) the dynamical reduced model with $T = 5$ and $p^* = 8$ 118
- 5.7 LD: Evolution of the relative error (5.5.10) of the Hamiltonian (*a*). Evolution of the components $\Delta \mathcal{H}_{\tau-1 \rightarrow \tau}$, $\Delta \mathcal{H}_{\tau-1 \rightarrow \tau}^Z$ and $\Delta \mathcal{H}_{\tau-1 \rightarrow \tau}^{Z,DD}$ of the error bound (5.5.11), (5.5.12) in the local conservation of the reduced Hamiltonian (*b*). The values of the hyper-parameters are set to $p^* = 12$ and $T = 3$, respectively. 119
- 5.8 LD: Comparison of the runtime (in seconds) between the full-order solver and the dynamical reduced basis approach for different hyper-parameter configurations, as function of the parameter sample size p (*a*). Separation of contributions to the running time of the reduced model due to basis evolution (dashed lines) and coefficients evolution (continuous line) (*b*). 120
- 5.9 NLD: Initial position and velocity distributions for selected values of the parameter in Γ_h (*a*) – (*b*). Exponential time decay of the electrostatic energy $\mathcal{E}(X_\tau^i; \eta_i)$ obtained from the full model solution, for selected values of η_i in Γ_h (*c*). Since not all parameters in Γ_h are reported, the black lines in each subplot are used to mark the region where the plotted quantity is contained, for any value of the parameter in Γ_h 121
- 5.10 NLD: Singular values of the global snapshots matrices S_X and S_V compared to the maximum and time average (in τ) of the singular values of the local matrices S_X^τ and S_V^τ 121
- 5.11 NLD: Numerical rank of S_X^τ in (*a*) and S_V^τ in (*b*), as a function of τ . Different colors are associated with different values of the threshold, according to the legend. 122
- 5.12 NLD: Evolution of the position (*a*) and velocity (*b*) relative errors, as defined in (5.5.7), for different choices of p^* and T . These errors are compared to the target values given by the position component $\varepsilon_{\text{rel},X}^{\text{Target}}$ and the velocity component $\varepsilon_{\text{rel},V}^{\text{Target}}$ of the relative projection errors defined in (5.5.8). The target reduced basis has dimension 6 and is computed, for each time step, using the Complex SVD algorithm, as discussed in 5.5.1. 122
- 5.13 NLD: Numerical distribution function for $\eta = (0.4912, 0.9889)$ at different times obtained from (*a*) the full-order model; (*b*) the dynamical reduced model with $T = 3$ and $k_{\text{DEIM}} = 3$; and (*c*) the dynamical reduced model with $T = 5$ and $k_{\text{DEIM}} = 1$. Starting from the perturbed Maxwellian distribution, particles with different energies oscillate with different frequencies leading to the typical filamentation that starts developing at $t = 13.33$. Two trapping vortices, centered at opposite phase velocities, form at $t = 26.66$ and fully develop at $t = 40$ 123
- 5.14 NLD: Evolution of the electric field energy $\mathcal{E}(\cdot; \eta_i)$. The energy is evaluated at the positions X_τ^i computed using the high-fidelity solver and at the positions $X_{r,\tau}^i$ computed using the reduced model, for different values η_i of the parameter. . . . 124

5.15	NLD: Peaks of the electric field energy $\mathcal{E}(X_{r,\tau}^i; \eta_i)$ selected for the computation of the exponential damping and growth rates.	125
5.16	NLD: Contour plots of the damping rate $((a), (c))$ and growth rate $((b), (d))$ of the electric field energy $\mathcal{E}(X_{r,\tau}^i; \eta_i)$ for different values of k_{DEIM} and T	125
5.17	NLD: (a) Comparison of the runtime (in seconds) between the full-order solver and the dynamical reduced basis approach for different hyper-parameter configurations, as a function of the parameter sample size p . (b) Separation of contributions to the running time of the reduced model due to basis evolution (5.8) (dashed lines) and coefficient evolution (5.8) (continuous line).	126
5.18	TSI: (a) – (b) Initial position and velocity distributions for selected values of the parameter in Γ_h . (c) Exponential time decay of the electrostatic energy $\mathcal{E}(X_{r,\tau}^i; \eta_i)$ obtained from the full model solution, for selected values of η_i in Γ_h . Since not all parameters in Γ_h are reported, the black lines in each subplot are used to mark the region where the plotted quantity is contained, for any value of the parameter in Γ_h	127
5.19	TSI: Singular values of the global snapshots matrices S_X and S_V compared to the maximum and time average of the singular values of the local trajectories matrices S_X^τ and S_V^τ . We study position (a) and velocity (b) variables separately. The singular values are normalized using the largest singular value for each case. . . .	127
5.20	TSI: Numerical rank of S_X^τ in (a) and S_V^τ in (b), as a function of τ . Different colors are associated with different values of the threshold, according to the legend. In (c) is reported the evolution of the numerical rank of the electric potential obtained from the full model.	128
5.21	TSI: Evolution of the position (a) and velocity (b) relative errors, as defined in (5.5.7), for different choices of k_{DEIM} and T . These errors are compared to the target values given by the position component $\varepsilon_{\text{rel},X}^{\text{Target}}$ and the velocity component $\varepsilon_{\text{rel},V}^{\text{Target}}$ of the relative projection errors defined in (5.5.8). The target reduced basis has dimension 4 and is computed, for each time step, using the Complex SVD algorithm, as described in Section 5.5.1.	128
5.22	TSI: Evolution of the position (a) and velocity (b) relative errors, as defined in (5.5.7), for different choices of the reduced basis dimension $2n$. The values of the hyper-parameters k_{DEIM} and T are both set to 3.	129
5.23	TSI: Evolution of the electric field energy $\mathcal{E}(\cdot; \eta_i)$. The energy is evaluated at the positions X_r^i computed using the high-fidelity solver and at the positions $X_{r,\tau}^i$ computed using the reduced model, for different values η_i of the parameter. . . .	130
5.24	TSI: (a) Comparison of the runtime (in seconds) between the full-order solver and the dynamical reduced basis approach for different hyper-parameter configurations, as function of the parameter sample size p . (b) Separation of contributions to the running time of the reduced model due to basis evolution (5.8) (dashed lines) and coefficients evolution (5.8) (continuous line).	130
7.1	Normalized $\Pi(i j)$ as function of the distance $i - j$ from three different modes i (a) . Surface representing the logarithm of $\Pi(i j)$, for $(i, j) \in [1, 30] \times [1, 30]$ (b). . . .	141
7.2	BG: Absolute value of the memory contribution for the first $n_2=10$ modes represented as a smooth surface for the exact (a) and approximated (b) memory term as a function of time in case of standard RB ansatz.	148

List of Figures

7.3	BG: Absolute value of the memory contribution for the first $n_2=10$ modes represented as a smooth surface for the exact (left) and approximated (right) memory term as a function of time in case of additional bias term in the RB approximation.	148
7.4	BG: Snapshots of the memory integrand and memory at different time frames in case of standard POD basis. The results showed are related to the third element of the basis.	150
7.5	BG: Snapshots of the memory integrand and memory at different time frames in case of modified POD basis.	150
7.6	BG: Normalized average of the memory kernel over time t and over the entire set of resolved modes for different values of n_2 (a). In (b), the continuous lines represent the average over the resolved set of the memory length, while the dashed lines are the corresponding time averages.	151
7.7	BG: Relation between the average memory length over time and the average reciprocal of the spectral radius of the Jacobian over time for the Burgers' equation in (a). The dashed lines represent the least squares approximation of the data. (b) Behavior of the coefficient C in (7.3.13) as a function of the parameter η for different sizes n_2 of the resolved set of coefficients.	151
7.8	CA: Singular values decay of the snapshots matrix computed from the solution of the test Cauchy problem (7.4.1). The exponential decay suggests that a small basis is sufficient to represent the solution.	160
7.9	CA: Qualitative comparison of the entries 11(a), 34(b), 36(c), and 67(d) of the projection of the exact solution of the problem (7.4.1) onto a space of dimension $n_1 = 16$ (—) and under-resolved ROM solutions for different values of $n_2 \in [2, 3, 4, 5]$. For each value of n_2 , hierarchical models of closure are compared for truncation orders $k = 0$ (—), $k = 1$ (—), $k = 2$ (—), $k = 4$ (—), $k = 7$ (—), $k = 9$ (—), and $k = 11$ (—).	161
7.10	CA: Comparison of the entries 1(a), 2(b), 3(c), and 5(d) of the exact memory term and the hierarchical memory approximations for $n_2 \in [2, 3, 6, 8]$ and truncation orders $k = 0$ (—), $k = 1$ (—), $k = 2$ (—), $k = 4$ (—), $k = 7$ (—), $k = 9$ (—), and $k = 11$ (—).	162
7.11	BG: Error (7.4.3) for POD-Galerkin model solution with (—) and without closure model (—), evaluated for different reduced model sizes and parameters.	164
7.12	BG: Average of the memory contribution over the entire set of n_2 modes in case of exact memory (—), memory approximation with τ -model (—), and truncation (—), evaluated for different reduced model sizes and parameters.	164
7.13	BG: Average energy transfer between the resolved and unresolved part of the simulation, in case of exact memory (—), memory approximation with τ -model (—), and truncation (—), evaluated for different reduced model sizes and parameters.	165
8.1	Conditioned LSTM network. In this sketch, ε represents the system parameter and x_i are the elements of the input sequence.	170
8.2	Comparison between the solution u to the FOM (—), the projection of u onto the approximating manifold (—), the solution u^{PG} to the POD-Galerkin ROM (—), and the expected solution u^{PGL} to the POD-Galerkin ROM with memory closure based on LSTM network (—).	173
8.3	ST: 3D carotid bifurcation geometry model and mesh.	176

8.4	ST: Time-dependent velocity in z -direction, at the inlet boundary, for the training (a) and prediction (b) regimes.	176
8.5	ST: Singular value decay of the snapshot matrix S in logarithmic scale. The spectrum shows a very fast decay, which suggests that a reduced basis with less than 20 elements is enough to represent the high-fidelity solution with a reasonable accuracy.	177
8.6	ST: Test error of the trained networks for the Stokes problem in the metrics defined in (8.3.1) (in (a)) and in (8.3.2) (in (b)). The estimated mean (solid vertical line), minimum and maximum (dashed vertical lines) of the estimated memory length τ over the parameter set are also shown in the plot.	178
8.7	ST: Test errors of the memory model in the metrics defined in (8.3.1) (in (a)) and in (8.3.2) (in (b)), with respect to different number of parameter values in the training data sets.	178
8.8	ST: Evolution of the energy exchange term $2z^\top w$ of the reduced-order models of the Stokes problem for different parameter values in the training and prediction regimes.	179
8.9	ST: Evolution of the errors of the reduced order models for the Stokes problem for different parameter values in the training and prediction regimes.	180
8.10	ST: A sectional view of the velocity magnitude contours of the reduced-order solutions at $t = 2.4$ for $\nu = 3$ for the FOM solution, the projection of the FOM solution onto the reduced space, the solution to the POD-Galerkin model, and the solution to the POD-Galerkin model with the LSTM correction.	181
8.11	ST: Error-cost plot of the reduced-order models for the Stokes problem for the training (a) and prediction (b) intervals.	182
8.12	KS: Test error of the trained networks for the KS equation in the metrics defined in (8.3.1) (in (a)) and in (8.3.2) (in (b)). The estimated mean (solid vertical line), minimum and maximum (dashed vertical lines) of the estimated memory length τ over the parameter set are also shown in the plot.	183
8.13	KS: Test error plots of the trained networks with 128 hidden units for different parameter values in the metrics defined in (8.3.1) (in (a)) and in (8.3.2) (in (b)).	183
8.14	KS: Evolution of the energy exchange term $2z^\top w$ of the reduced-order models of the KS problem for different parameter values in the training and prediction regimes.	184
8.15	KS: Numerical solutions for parameter $\nu = 0.34756$. ----- High-fidelity; — Projection of high-fidelity; — POD-Galerkin; — POD-Galerkin with memory.	186
8.16	KS: Numerical solutions for parameter $\nu = 0.71166$. ----- High-fidelity; — Projection of high-fidelity; — POD-Galerkin; — POD-Galerkin with memory.	187
8.17	KS: Numerical solutions for parameter $\nu = 1.0006$. ----- High-fidelity; — Projection of high-fidelity; — POD-Galerkin; — POD-Galerkin with memory.	188
8.18	KS: Numerical solutions for parameter $\nu = 1.4399$. ----- High-fidelity; — Projection of high-fidelity; — POD-Galerkin; — POD-Galerkin with memory.	189
8.19	KS: Evolution of the numerical solutions of the KS equation with $\eta = 0.64424$	190
8.20	KS: Error-cost plot of the reduced-order models for the KS problem for the training (a) and prediction (b) intervals. The horizontal dashed line represents the projection error.	190
8.21	RB: Simulation setup of the Rayleigh-Bénard convection.	191
8.22	RB: Singular values of the Rayleigh-Bénard convection problem.	192

List of Figures

8.23	RB: Test error of the trained networks for the Rayleigh-Bénard convection in the metrics defined in (8.3.1) (in (a)) and in (8.3.2) (in (b)). The estimated mean (solid vertical line), minimum and maximum (dashed vertical lines) of the estimated memory length τ over the parameter set are also shown in the plot.	192
8.24	RB: Evolution of the energy exchange term $2z^\top w$ of the reduced-order models of the Rayleigh-Bénard problem for different parameter values in the training and prediction regimes.	193
8.25	RB: Temperature evolution of Rayleigh-Bénard convection problem at four points, for $Ra = 14024512.0002$, $Pr = 0.86976$. ----- High-fidelity; — Projection of high-fidelity; — POD-Galerkin; — POD-Galerkin with memory.	194
8.26	RB: Contour of Rayleigh-Bénard convection problem at $t = 10, 25, 70$, and 85 , for $Ra = 14024512.0002$, $Pr = 0.86976$	195
8.27	RB: Error-cost plot of the reduced-order models of the Rayleigh-Bénard convection for the training (a) and prediction (b) intervals. The horizontal dashed line represents the projection error.	195

List of Acronyms

ANN	Artificial neural network
DEIM	Discrete empirical interpolation method
DLR	Dynamical low-rank
DMD	Dynamic mode decomposition
DOF	Degrees of freedom
FEM	Finite element method
FOM	Full order model
FVM	Finite volume method
GLE	Generalized Langevin equation
LSTM	Long short-term memory
MOR	Model order reduction
MZ	Mori-Zwanzig
ODE	Ordinary differential equation
PDE	Partial differential equation
POD	Proper orthogonal decomposition
PSD	Proper symplectic decomposition
RB	Reduced basis
RNN	Recurrent neural network
ROM	Reduced order model

1 An Introduction to the Reduced Basis method

This Chapter introduces the general framework of computational reduction techniques considered in this thesis. After a brief excursus on the necessity and motivations in favor of model order reduction (MOR) in Section 1.1, we target the reduced basis (RB) method for parametrized ordinary differential equations (ODEs), with a particular emphasis, in Section 1.2, on the notations and formulations useful for the description and analysis of the approaches proposed in the following Chapters.

Section 1.3 describes different methods for generating the reduced basis from available information about the original computationally expensive problem we want to reduce. The approximated parametric reduced order model (ROM), obtained using Galerkin projection on the reduced basis, is introduced in Section 1.4, while several approaches addressing the efficiency issue of nonaffine parametric dependence are described in Section 1.5.

1.1 Motivations

High-fidelity simulations have become essential tools for investigating complex problems of scientific interest and industrial value, thanks to a significant increase in the available computational power and more advanced algorithms. The resulting computational speed-up made it possible to solve mathematical problems beyond the reach of previous generations and has cemented the importance of computational sciences in the contemporary technological world. These advancements have gone hand in hand with an ever-increasing demand for more realistic and detailed simulations, requiring finely adapted meshes to achieve pristine precisions and the inclusion of several physical contributions in the modeling phase, leading to multi-scale and multi-physics problems to be solved. In the mathematical formalism, these models consist of parametric partial differential equations (PDEs) that have been discretized by classical methods like finite element (FEM), spectral, or finite volume (FVM) methods leading to dynamical systems with very large state-space dimensions, typically of the order of millions of degrees of freedom to match the target accuracy. Therefore, full order models (FOM) are usually not viable for industrial applications like product design, optimal control, and uncertainty quantification, all of which require repeated model evaluations over a potentially wide range of parameter values.

The goal of computational reduction techniques is to simplify large dimensional dynamical systems, using a limited number of equations and degrees of freedom (DOFs) while retaining the essential features and details of the original solution. Physical insights have been used in the past to

reduce the computational complexity of models, mainly by formulating physical simplifications before starting the computations.

Worth mentioning, in this direction, is the work by Quarteroni [209] on the simulation of blood flow in the human circulatory system. The flow in small vessels is approximated using a one-dimensional model, while arteries require two-dimensional parametrizations. Only the heart's dynamic is described using a three-dimensional model, thus enabling a complete simulation of the blood flow in the human body with the computational power offered by today's processors. A similar idea was exploited before that for the development of the PSP model [100] of electromagnetic effects in MOR transistors, designated as industrywide model for chip design by the Compact Model Council in 2005. Using a large set of measurements and simulations, computationally expensive 3D Maxwell models of transistors have been replaced by cheap parametrized algebraic equivalences. The introduction of these approximating models has represented a breakthrough step for the accurate simulation of integrated circuits comprising millions of semiconductor components.

The two examples cited above share the spirit of a priori simplification of the model at hand and require a deep understanding of the physics of the problem and possible sources of modeling error. They follow the paradigm known in computational reduction as *physical metamodeling*, according to which the new simplified model is nevertheless solved using full-order discretization techniques. Unfortunately, it is not always possible to introduce such simplifications, either for lack of physical insight of the problem or because such simplifications are not easily identifiable. In this thesis we consider a different type of approach, known as *model order reduction*, or *computational reduction technique*, that relies on automatic identification of potential simplifications. Starting from the Truncated Balanced Realization, introduced by Moore [178] in 1981, several other MOR techniques have been developed and flourished during the last 40 years, including the Hankel-norm reduction [104], the Padé-via-Lanczos (PVL) algorithm [88], and the PRIMA method [186]. In the following, we focus on the *reduced basis* method, introduced in the context of nonlinear structural analysis of beams and arches [8], and further developed [20; 22; 92; 215] in the last decades of the last century for more general parametrized PDEs. In particular, we emphasize the significant results obtained for applications in fluid dynamics, starting from the seminal work of Sirovich [232] regarding turbulence theory and arriving at the wide variety of efficient ROMs for Stokes and Navier-Stokes equations [98; 99; 200; 75] proposed in more recent years.

The key components of the RB method (definition of low dimensional reduced subspace, offline/online computational decomposition and a posteriori error estimators) have been introduced and formalized in [130; 208; 109; 167; 168; 71; 256]. The reduction of non-affine and nonlinear problems has been recently addressed [21; 57; 240; 7] and a posteriori error estimators have been developed, for parabolic and elliptic PDEs in [110; 217], with consequent certification of RB methods in those context. Several software packages based on RB methods are now available and usable for a wide range of physical problems. In this regard, we would like to mention *RBmatlab* [117], *RBiCS* [130], and *pymor* [176].

The goal of the following Sections is to provide a general overview of some of the topics mentioned above, supported by appropriate references.

1.2 The reduced basis method

As pointed out in Section (1.1), problems arising in multi-query contexts or involving PDE-constrained optimization entail relevant computational costs and hence demand computational reduction to be tackled.

The reduced basis method is built upon an high fidelity approximation method of the parametric PDE. The goal of the next Section is to present a general formulation of the RB method as an approximation of the solution of parametric PDEs via Galerkin projection on a compact space built using precomputed high-dimensional solutions.

1.2.1 Problem formulation and notation

Let $\mathcal{T} := (t_0, T]$ be a temporal interval and let $\Gamma \subset \mathbb{R}^d$, with $d \geq 1$, be a compact set of parameters. For each $\eta \in \Gamma$, we consider the initial value problem: For $u_0(\eta) \in \mathbb{R}^N$, find $u(\cdot, \eta) \in C^1(\mathcal{T}, \mathbb{R}^N)$ such that

$$\begin{cases} \frac{d}{dt}u(t; \eta) = f(t, u; \eta), & \text{for } t \in \mathcal{T}, \\ u(0; \eta) = u_0(\eta), \end{cases} \quad (1.2.1)$$

where the dot denotes the derivative with respect to time t , $C^1(\mathcal{T}, \mathbb{R}^N)$ denotes continuous differentiable functions in time taking values in \mathbb{R}^N , and $f : \mathcal{T} \times \mathbb{R}^N \times \Gamma \mapsto \mathbb{R}^N$ that satisfies the regularity assumptions required in the Picard-Lindelöf theorem. For a fixed parameter η , additional problem specific properties may be required to f to assume the well-posedness of (1.2.1). Problems described by (1.2.1) often arise from the semidiscrete formulation of systems of PDEs, using a suitable high-fidelity method. Several high-fidelity methods are compatible with the RB framework described in the following, such as the finite element [246], the finite volume [118] and the spectral element approaches [162; 194]. Moreover, in the RB notation, problem (1.2.1) is referred to as full-order model (FOM). To achieve satisfactory results in terms of accuracy of the numerical approximations, these FOMs rely on high-order approximations or on the use of fine meshes, resulting in N being large and, hence, increasing the computational cost of the simulation.

We define the set of all solutions to (1.2.1) for different values of the parameter vector $\eta \in \Gamma$ and $t \in \mathcal{T}$ as

$$\mathcal{M} := \{u(t; \eta) | \eta \in \Gamma, t \in \mathcal{T}\}. \quad (1.2.2)$$

The fundamental assumptions of the RB approach is that \mathcal{M} is *reducible*, i.e., any element of \mathcal{M} can be accurately approximated using a linear combination of a limited number of solutions to (1.2.1). The mathematical measure that encodes the system's linear reducibility is the Kolmogorov n -width [253], a concept from approximation theory that quantifies the maximum possible error that might occur from the projection of the solution trajectory onto an optimal subspace of dimension n . It is defined by

$$D_n(\mathcal{M}) := \inf_{\mathcal{M}_n \subset \mathbb{R}^N} \sup_{v \in \mathcal{M}} \inf_{w \in \mathcal{M}_n} \|v - w\|_2, \quad (1.2.3)$$

where \mathcal{M}_n is a linear subspace of \mathbb{R}^N of dimension n .

Let $\Pi_{\mathcal{M}_n}$ be the Euclidean orthogonal projector operator from \mathcal{M} to \mathcal{M}_n , where we assume that \mathbb{R}^N is equipped with the Euclidean inner product operator $\langle \cdot, \cdot \rangle_N$ and $\Pi_{\mathcal{M}_n}$ is a projection with respect to $\langle \cdot, \cdot \rangle_N$. Being a projector, $\Pi_{\mathcal{M}_n}$ is a linear map satisfying the idempotency condition

$\Pi_{\mathcal{M}_n}^2 = \Pi_{\mathcal{M}_n}$. It trivially follows that also $\mathbb{I}_N - \Pi_{\mathcal{M}_n}$ is a projector, where \mathbb{I}_N is the identity operator, and every element v of \mathbb{R}^N can be written as $v = \Pi_{\mathcal{M}_n} v + (\mathbb{I}_N - \Pi_{\mathcal{M}_n}) v$, entailing that the space \mathbb{R}^N can be defined via the direct sum

$$\mathbb{R}^N = \ker(\Pi_{\mathcal{M}_n}) \oplus \text{range}(\Pi_{\mathcal{M}_n}).$$

The orthogonality of the projector $\Pi_{\mathcal{M}_n}$ implies that

$$\ker(\Pi_{\mathcal{M}_n}) = \text{range}(\Pi_{\mathcal{M}_n})^\perp,$$

or, equivalently, that for any $v \in \mathbb{R}^N$,

$$\Pi_{\mathcal{M}_n} v \in \mathcal{M}_n \quad \text{and} \quad (\mathbb{I}_N - \Pi_{\mathcal{M}_n}) v \in \mathcal{M}_n^\perp.$$

These relations lead to the optimality property of orthogonal projectors stated in the next theorem.

Theorem 1.2.1 ([221, Theorem 1.38, page 39]). *Let $\Pi_{\mathcal{M}_n}$ be the orthogonal projector onto the subspace \mathcal{M}_n . Then for any given vector v in \mathbb{R}^N , the following is true:*

$$\inf_{w \in \mathcal{M}_n} \|v - w\|_2 = \|v - \Pi_{\mathcal{M}_n} v\|_2. \quad (1.2.4)$$

Using (1.2.4), then (1.2.3) simplifies to

$$D_n(\mathcal{M}) := \inf_{\mathcal{M}_n \subset \mathbb{R}^n} \sup_{v \in \mathcal{M}} \|v - \Pi_{\mathcal{M}_n} v\|_2. \quad (1.2.5)$$

From (1.2.5), we can infer that the quality of a reduced-order linear approximation of \mathcal{M} can be judged by how quickly $D_n(\mathcal{M})$ decreases as n increases and, once a user-defined tolerance δ is set, the truncation error is bounded. Once the subspace \mathcal{M}_n has been identified, with n possibly small to ensure computational advantages, we can construct the reduced-order system

$$\begin{cases} \frac{d}{dt} \Pi_{\mathcal{M}_n} u(t; \eta) = \Pi_{\mathcal{M}_n} f(t, u; \eta), & \text{for } t \in \mathcal{T}, \\ \Pi_{\mathcal{M}_n} u(0; \eta) = \Pi_{\mathcal{M}_n} u_0(\eta), \end{cases} \quad (1.2.6)$$

as a result of the projection of (1.2.1) using $\Pi_{\mathcal{M}_n}$. Unless specified otherwise, we assume that $\Pi_{\mathcal{M}_n}$ and the related subspace \mathcal{M}_n are time-independent and we refer to (1.2.6) as *global* reduced-order model (gROM), where we use the term *global* to stress that the approximating subspace is used $\forall t \in \mathcal{T}$. For reducible problems, n can be taken much smaller than N , and thus (1.2.6) has a lower order than (1.2.1).

In the following Section, we summarize a convenient offline/online decomposition that is employed to decouple the operations whose cost depends on the number of degrees of freedom N of the FOM from those independent on it, with the aim of lowering the computational complexity of the reduction algorithm. A complete decomposition is achieved by requiring affine parameter dependence and linearity to the evolution operator f . Several treatments for nonlinear terms, with the aim of restoring this computational splitting, are addressed in Section 1.5.

1.3 Offline phase

The offline step consists of identifying the RB space \mathcal{M}_n , constructing the operator $\Pi_{\mathcal{M}_n}$, and the crucial step of computing reduced matrices and vector associated with $\Pi_{\mathcal{M}_n}f$. Unless stated otherwise, in this work we restrict $\langle \cdot, \cdot \rangle_N$ to the inner product of the Euclidean space. Let $v_1, \dots, v_n \in \mathbb{R}^N$ be a set of basis vectors, orthonormal with respect to $\langle \cdot, \cdot \rangle_N$, such that $\mathcal{M}_n = \text{span}\{v_1, \dots, v_n\}$ and let $U \in \mathbb{R}^{N \times n}$ be the basis matrix collecting these vectors as columns. Given $g \in \mathbb{R}^N$, since $\Pi_{\mathcal{M}_n}g$ belongs to \mathcal{M}_n it can be written in terms of U as $\Pi_{\mathcal{M}_n}g = Uz$, with $z \in \mathbb{R}^n$ expansion coefficients of the representation. Similarly, the orthogonality of the projector translates into

$$U^\top Uz = z, \quad \text{and} \quad U^\top (g - Uz) = \mathbf{0}_N,$$

which yields the matrix representation of $\Pi_{\mathcal{M}_n}$ as

$$\Pi_{\mathcal{M}_n} = UU^\top, \tag{1.3.1}$$

for $g \in \mathbb{R}^N$.

The two problems of formulating the approximation space \mathcal{M}_n and constructing the orthogonal projector $\Pi_{\mathcal{M}_n}$ on it can then be condensed into the search for an optimal basis U , addressed in the following as reduced basis. Two popular and efficient approaches to design the reduced basis U , namely the proper orthogonal decomposition and the greedy method, are detailed in the following sections.

1.3.1 Proper orthogonal decomposition

The POD was developed to obtain low-dimensional representations of turbulent fluid flows [239; 14], after the seminal work of Lumley in the 60s on the study of coherent structures in fully developed turbulent flows [165]. With the aim of simplifying the analysis of complex systems, the POD was subsequently employed in the fields of structural vibrations [72; 9], damage detection [95], and image processing [36; 206]. In the following we will provide a brief description of the application of POD to the case of parametric surrogate modeling [16; 138; 43; 66] for the definition of ROMs. Given any space \mathcal{M} representing the solution manifold of the FOM (1.2.1), the proper orthogonal decomposition method builds an n -dimensional RB space \mathcal{M}_n by first sampling \mathcal{M} and defining the snapshots (or training) set

$$\mathcal{M}_\Delta := \{u(t_i; \eta_j) \mid \eta_j \in \Gamma_h, t_i \in \mathcal{T}_\Delta\}, \tag{1.3.2}$$

with Γ_h discrete subset of Γ of cardinality N_p and \mathcal{T}_Δ discrete subsets of \mathcal{T} of cardinality N_t . In the remainder of this thesis, we will use $N_S = N_p N_t$ to denote the cardinality of the set \mathcal{M}_Δ . Members of \mathcal{M}_Δ are referred to as snapshots of (1.2.1) and one can obtain an approximation $\widetilde{\mathcal{M}}_\Delta$ of this snapshot set by applying a time-integration scheme, e.g., the Runge-Kutta methods, to (1.2.1) for different values $\eta_j \in \Gamma_h$ of the parameter of interest. While sampling in time in (1.3.2) is usually dictated by the chosen numerical integrator and the propagation speed of the information in the physical space, sampling in parameters should satisfy two different criteria. On one hand, the number of ineffective samples must be reduced to avoid unnecessary computational cost and diminishing returns in accuracy. On the other hand, the sampling must be fine enough to ensure that $\widetilde{\mathcal{M}}_\Delta$, and hence \mathcal{M}_Δ , can be considered faithful representations of \mathcal{M} . In practice, an often effective choice, especially in the case of relatively small d , is sampling from a random distribution

or a log-equidistant distribution [194; 219]. In the context of high-dimensional integration, we refer the reader to the techniques described in [234; 154] to mitigate the potential curse of dimensionality effect induced by the dimension of the parameter space. Moreover, throughout this thesis, we assume that we can choose $\widehat{\mathcal{M}}_\Delta$ arbitrary close to \mathcal{M}_Δ and we drop the overscript. The proper orthogonal decomposition (POD) method is a popular approach for the construction of a reduced basis U given the snapshot matrix $S^u \in \mathbb{R}^{N \times N_S}$ having as columns the solution snapshots $u(t_i; \eta_j)$, i.e.,

$$S^u := [u(t_i; \eta_j)], \quad (1.3.3)$$

with $t_j \in \mathcal{T}_\Delta$ and $\eta_j \in \Gamma_h$. The reduced basis is computed as solution to the optimization problem

$$\begin{aligned} \min_{U \in \mathbb{R}^{N \times n}} \|S^u - UU^\top S^u\|_F, \\ \text{subject to } U \in \text{St}(n, \mathbb{R}^N) \end{aligned} \quad (1.3.4)$$

where $\|\cdot\|_F$ is the Frobenius norm and

$$\text{St}(n, \mathbb{R}^N) := \{L \in \mathbb{R}^{N \times n} : L^\top L = \mathbb{I}\}$$

is the set of all orthonormal n -frames in \mathbb{R}^N , also known as Stiefel manifold. Problem (1.3.4) can be viewed as the discrete surrogate of the Kolmogorov n -width problem (1.2.3) for the approximation of the column space of the snapshot matrix S^u , with the l_∞ -norm replaced by the mean-square error. Despite the nonlinear and non-convex nature of problem (1.3.4), the Eckart-Young-Mirsky-Schmidt theorem provides a best approximation result, as stated in the following theorem.

Theorem 1.3.1. *Let $A \in \mathbb{R}^{N \times m}$ has the singular value decomposition (SVD), $A = W\Sigma V^\top$, with $W \in \mathbb{R}^{N \times N}$ and $V \in \mathbb{R}^{m \times m}$ with columns $\{w_i\}_{i=1}^m$ and $\{v_i\}_{i=1}^m$, called left and right singular vectors of A , respectively. $\Sigma \in \mathbb{R}^{N \times m}$ is diagonal in the sense that $\Sigma_{i,j} = 0$ if $i \neq j$, with diagonal entries $\sigma_i = \Sigma_{i,i}$, representing the singular values of A , satisfying $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_{\min(N,m)} \geq 0$. Then, the truncated sum*

$$A_n = U_n \Sigma_n V_n^\top \quad (1.3.5)$$

with $U_n = [w_1, \dots, w_n] \in \mathbb{R}^{N \times n}$, $V_n = [v_1, \dots, v_n] \in \mathbb{R}^{m \times n}$, and $\Sigma_n = [\sigma_1, \dots, \sigma_n] \in \mathbb{R}^{n \times n}$ solves the optimization problem

$$A_n = \min_{\text{rank}(B)=n} \|A - B\|_*, \quad (1.3.6)$$

for $* = \{2, F\}$. In (1.3.5) we have highlighted the dimensions of the involved matrices through the corresponding subscripts. Furthermore

$$\|A - A_n\|_2^2 = \sigma_{n+1}^2 \quad \text{and} \quad \|A - A_n\|_F^2 = \sum_{i=n+1}^{\min(N,m)} \sigma_i^2. \quad (1.3.7)$$

Let

$$\bar{\Sigma}_n = \begin{bmatrix} \Sigma_n & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{N \times m}.$$

It can be verified that

$$A_n = U_n \Sigma_n V_n^\top = W \bar{\Sigma}_n V^\top,$$

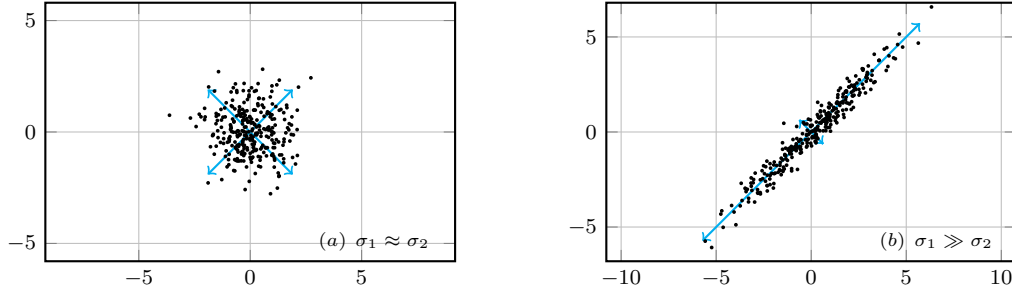


Figure 1.1: Examples of samples from from an isotropic (a) and anisotropic (b) bivariate distribution.

and then it follows that

$$A_n = W \bar{\Sigma}_n V^\top = W \Sigma_n \Sigma^{-1} W^\top A = W_n W_n^\top A.$$

We therefore conclude that the minimization problem (1.3.6) is equivalent to problem (1.3.4), from which it follows that the optimal n -dimensional RB in the Frobenius norm has as columns the first n left singular vectors of the snapshot matrix S^u , i.e. $U = U_n$. The geometric interpretation of Theorem 1.3.1 may provide an even clearer picture of the connection between SVD and POD. The matrix S^u represents a linear operator that maps N dimensional vectors to its columns space. The unit sphere in \mathbb{R}^N is mapped into an hyper-ellipsoid in \mathbb{R}^{N_S} , where the directions of the principal radii are given by the left singular vectors w_i and the corresponding lengths are given by the singular values σ_i . The columns of S^u constitute a set of N_S points in a N -dimensional space. For any $n \leq N_S$, we seek a n -dimensional subspace that minimizes the mean squared distance of the points from the subspace. The best n -dimensional approximation is the n -dimensional ellipsoid having as principal radii the POD modes w_i associated with the largest singular values σ_i . In the principal radii coordinate system, a large anisotropy of the snapshots data suggests a suitable representation in a low-dimensional space (see Figure 1.1). Since the length of each of the radii is related to σ_i , the choice of the dimension n of the POD basis U of the approximating space \mathcal{M}_n is related to the decay of the singular values σ_i . The singular values σ_i are often apostrophized to as energy of the corresponding POD modes, a terminology derived from the early application of POD to incompressible fluid mechanics, where this energy is related to the fluid's kinetic energy. To obtain a representative, low-dimensional basis, following (1.3.7), the modes corresponding to the smallest eigenvalues are discarded. To make the selection procedure more rigorous, let us define the relative information content of an n -dimensional basis U obtained by POD as

$$I(n) = \frac{\sum_i^n \sigma_i}{\sum_i^{N_S} \sigma_i}. \quad (1.3.8)$$

The relative information content $I(n)$ represents the percentage of the energy of the snapshot matrix S^u captured by the first n POD modes and is tightly connected to the Kolmogorov n -width. We refer the reader to [253] for a strong results about the connection between the Kolmogorov n -width and the (Hankel) singular values and related optimization processes. As a reasonable dimension-selection criterion for the POD basis, the dimension n is selected as the smallest integer number satisfying

$$I(n) \geq 1 - \delta_{\text{POD}}^2, \quad (1.3.9)$$

An Introduction to the Reduced Basis method

where δ_{POD}^2 represents the energy contained in the neglected $N_S - n$ modes. Combining (1.3.8) and (1.3.7), it is straightforward to show that criterion (1.3.9) is equivalent to

$$\frac{\|S^u - UU^\top S^u\|_2}{\|S^u\|_2} \leq \delta_{\text{POD}}.$$

The POD algorithm is summarized in Algorithm 1.

Algorithm 1 POD algorithm

- 1: **procedure** POD($u(t_i; \eta_j), \delta_{\text{POD}}$)
- 2: Collect data $u(t_i; \eta_j)$ from the high-fidelity model for selected values of parameters $\eta_j \in \Gamma_h$ and time $t_i \in \mathcal{T}_\Delta$.
- 3: Build the snapshot matrix S^u using the collected data as columns.
- 4: Compute the singular value decomposition of S^u , i.e.,

$$S^u = W\Sigma V^\top.$$

- 5: Find the number of POD basis vectors capturing at least $1 - \delta_{\text{POD}}^2$ of the relative information content of the snapshot matrix.
 - 6: Choose the first n left singular vectors to define the corresponding POD basis functions.
-

Several variations of the method proposed in Algorithm 1 have been suggested over time. We mention here the POD with a weighted inner product [255], where the data used to assemble the snapshot matrix are weighted according to a given probability distribution. In the POD with difference quotients [161], different quotients of the data are added to the standard snapshot matrix to reach optimal pointwise POD projection error bounds and optimal pointwise ROM errors. To overcome certain deficiencies of POD for linear modal analysis, the *smooth* orthogonal decomposition [58] has been proposed in the context of structural vibration analysis.

1.3.2 Greedy algorithm

When d is large, since the computational cost scales with the cardinality of Γ_h , the definition of a compact yet accurate training set becomes a computational challenge. Even though a certain amount of computational complexity is tolerated in the offline stage to obtain a significant speed-up in the online stage, the evaluation of the high-fidelity solution for a large sampling set and the SVD of the corresponding snapshot matrix are often impractical or not even feasible. In the reducible scenario, the fact that the number of relevant modes n extracted from the snapshots matrix is significantly smaller than the number N_S of total snapshots suggests that the amount of snapshots collected can be optimized and the relative basis built using an incremental algorithm. The reduced basis, in which the column space represents the approximating manifold, is improved iteratively by adding basis vectors as columns. By evaluating the high-fidelity solution only once (or few times) per iteration, the SVD of S is no longer required because the optimality over all n -dimensional subspaces is replaced with a local optimality criterion, reducing the computational cost of the offline phase. This summarizes the philosophy of the greedy strategy applied to RB methods [41; 31], which requires the definition of an optimization problem to identify the candidate basis vector at each iteration and an a posteriori error bound estimate to monitor the approximation accuracy of the basis.

Let $\mathfrak{S}_\Delta^u := \Gamma_h \times \mathcal{T}_\Delta$ be a given set of parameters and times for which the high-fidelity solution is

available and $U_k \in \mathbb{R}^{N \times k}$ the orthonormal reduced basis of dimension k produced after k steps of the greedy algorithm, spanning the approximating manifold \mathcal{M}_k . In its idealized form, introduced in [257], the greedy algorithm uses the projection error

$$(t^*, \eta^*) = \operatorname{argmax}_{(t_i, \eta_j) \in \mathfrak{S}_\Delta^u} \|u(t_i; \eta_j) - U_k U_k^\top u(t_i; \eta_j)\|_2 \quad (1.3.10)$$

to identify the snapshot $u^* := u(t^*; \eta^*)$ that is worst approximated by the column space of U_k over the entire sampling set \mathfrak{S}_Δ^u . Let v_{k+1} be the vector obtained by orthonormalizing u^* with respect to U_k . The basis matrix is then updated as $U_{k+1} = [U_k \ v_{k+1}]$. To avoid the accumulation of rounding errors, it is preferable to utilize backward stable orthogonalization processes, such as the modified Gram-Schmidt orthogonalization [33]. The algorithm terminates when the basis reaches the desired dimension, or the error (1.3.10) is below a certain tolerance. In this sense, the basis U_{k+1} is hierarchical because its column space contains the column space of its previous iterations. This process is referred to as *strong greedy method*. Even though introduced as a heuristic procedure, interesting results regarding algebraic and exponential convergence have been formulated in [41; 31], requiring the orthogonality of the basis in the corresponding proofs.

Theorem 1.3.2 ([31, Theorem 3.1, page 8]). *Suppose that $D_0(\mathcal{M}) \leq M$ and*

$$D_n(\mathcal{M}) \leq M n^{-\alpha}, \quad n > 0,$$

for some $M > 0$ and $\alpha > 0$. Then,

$$\max_{u \in \mathcal{M}} \|u - U_n U_n^\top u\|_2 \leq C M n^{-\alpha}, \quad n > 0,$$

with $C := q^{\frac{1}{2}}(4q)^\alpha$ and $q := \lceil 2^{\alpha+1} \rceil^2$.

Theorem 1.3.3 ([31, Theorem 3.2, page 9]). *Suppose that*

$$D_n(\mathcal{M}) \leq M e^{-an^\alpha}, \quad n > 0,$$

for some $M, a, \alpha > 0$ and $\alpha > 0$. Then setting $\beta := \frac{\alpha}{\alpha+1}$, one has

$$\max_{u \in \mathcal{M}} \|u - U_n U_n^\top u\|_2 \leq C M e^{-cn^\beta}, \quad n > 0,$$

whenever for any fixed $0 < \theta < 1$, one takes $c := \min\{|\ln \theta|, (4q)^{-\alpha} a\}$, $C := \max\{e^{cN_0^\beta}, q^{\frac{1}{2}}\}$, $q := \lceil 2\theta^{-1} \rceil^2$ and $N_0 := \lceil (8q)^{\frac{1}{1-\beta}} \rceil = \lceil (8q)^{\alpha+1} \rceil$.

However, in this form, the scheme cannot be efficiently implemented: the cost required to compute (1.3.10) is very high because the greedy procedure requires all the snapshots of the training set to be accessible, thus making the outlined method still computational expensive and relieving the computation only of the cost associated to the SVD. Introduced as adjustment to the shortcomings of the strong greedy algorithm, the *weak greedy algorithm* represents the standard greedy algorithm applied in the RB context. The idea is to replace (1.3.10) with a surrogate indicator $\gamma : \mathfrak{S}_\Delta^u \mapsto \mathbb{R}$ that does not require the computation of the high-fidelity solution for the entire time-parameter domain. In the case of elliptic PDEs, an a-posteriori residual-based indicator requiring a polynomial computational cost in the approximation space dimension n has been introduced in [230], as an application for heat conduction problems. In [219], a comparable algorithm targeting the energy norm of the error is presented. One might also use a goal-oriented

indicator as the driving selection in the greedy process to obtain similar computational benefits. In this direction, in the framework of structure-preserving MOR, we detail in Section 3.3.3 a greedy selection using the Hamiltonian function as a proxy error indicator. More examples of error upper bounds, based on the calculation of the residual and usable as indicators for the greedy approach, are provided in [208]. The substantial computational savings allow the choice of a more refined, and therefore representative, sampling set \mathfrak{S}_Δ^u . The indicator $\gamma(t, \eta)$ is then computed for each element of the set and its evaluations sorted in decreasing order to find the largest one, i.e.,

$$(t^*, \eta^*) = \operatorname{argmax}_{(t_i, \eta_j) \in \mathfrak{S}_\Delta^u} \gamma(t_i, \eta_j).$$

As for the strong form of the approach, the candidate $u(t^*, \eta^*) \in \mathbb{R}^N$ is then appended to the basis U_k and the process is repeated until a stopping criterion is met. For the sake of efficiency, it is crucial that the surrogate indicator is computable with an affordable cost, since it is evaluated (almost) over the entire set \mathfrak{S}_Δ^u for each step of the greedy method. If the cost to evaluate the indicator γ cannot be neglected with respect to the cost of the high-fidelity solver, any form of computational saving is jeopardized.

In [110], it has been noted that once a certain instance of the parameter η is selected for the basis enrichment, taking a single snapshot from the time sequence could lead to a stall in the basis enhancement. Combining a greedy choice in the parameter space and a time compression step based on POD, in [81; 144; 116] a heuristic method, known as POD-greedy, to solve the convergence problem of the pure greedy method has been proposed, and is now the gold standard for the greedy approach to reduced basis generation. Rates of algebraic and exponential convergence are achieved [116], as in Theorems 1.3.2 and 1.3.3, in case of similar decay of the Kolmogorov n -width $D_n(\mathcal{M})$.

1.4 Online phase

Having shown, in Section 1.3, several techniques for generating a reduced basis $U \in \mathbb{R}^{N \times n}$ for the reduced space \mathcal{M}_n , we now explain how this can be used, in combination with (1.2.6), to generate a gROM. An approximation $u_r(t; \eta) \in \mathbb{R}^N$ to the solution to (1.2.1) is provided by means of two approximations. First, we assume $u(t; \eta)$ can be well approximated on the low-dimensional subspace \mathcal{M}_n , leading to the so called RB ansatz

$$u_r(t; \eta) \approx Uz(t; \eta), \tag{1.4.1}$$

where $z : \mathcal{T} \times \Gamma \mapsto \mathbb{R}^n$ represents the generalized coordinates of the approximate solution u_r with respect to U . Substituting (1.4.1) into (1.2.1) leads to the overdetermined system

$$\begin{cases} U \frac{d}{dt} z(t; \eta) = f(t, Uz; \eta) + \mathbf{r}(t; \eta), & \text{for } t \in \mathcal{T}, \\ z(0; \eta) = U^\top u_0(\eta), \end{cases}$$

The quantity \mathbf{r} represents the residual due to (1.4.1) not solving exactly the FOM. In the framework of a Petrov-Galerkin projection, if we consider a second reduced basis W that is orthogonal to

the residual \mathbf{r} and $W^\top U$ is invertible, we recover the reduced system of n equations

$$\begin{cases} \frac{d}{dt}z(t; \eta) = U^\top f(t, Uz; \eta), & \text{for } t \in \mathcal{T}, \\ z(0; \eta) = U^\top u_0(\eta), \end{cases} \quad (1.4.2)$$

In this thesis work, unless stated otherwise, we restrict ourselves to the pure Galerkin framework, i.e. $W = U$. In this context, we report in the following an important result in terms of (local) minimal residual optimality of the Galerkin projection

Theorem 1.4.1 ([48, Theorem 3.2, page 5]). *In case of orthogonal reduced basis U , the Galerkin ROM (1.4.2) is continuous optimal in the sense that the approximated velocity minimizes the norm of the FOM ODE residual (1.2.1) over the column space of U , i.e.,*

$$\frac{d}{dt}u_r(t; \eta) = \underset{v \in \text{range}(U)}{\text{argmin}} \|v - f(t, u_r; \eta)\|_2^2. \quad (1.4.3)$$

Moreover, enriching the approximating reduced space results in a monotonical decrease in the norm of the error in the ROM velocity, as formulated in (1.4.3).

Before stating the accuracy estimate result presented in [210], for the case of POD generated reduced basis, we consider the two different components of the error made in approximating the solution of the FOM problem with the solution of the ROM. Consider the solution $u(t; \eta)$ to the FOM defined in (1.2.1) for a given parameter $\eta \in \Gamma$. We are interested in the time-evolution of the error

$$\varepsilon(t; \eta) := u_r(t; \eta) - u(t; \eta) = \varepsilon_\perp(t; \eta) + \varepsilon_\parallel(t; \eta). \quad (1.4.4)$$

In (1.4.4), we have highlighted the component $\varepsilon_\perp(t; \eta)$ of the error that is orthogonal to the approximating space \mathcal{M}_n , for which $UU^\top \varepsilon_\perp(t; \eta) = 0$, $\varepsilon(0; \eta) = \varepsilon_\perp(0; \eta)$, and that depends only on the approximation capability of the basis U , and the parallel component $\varepsilon_\parallel(t; \eta)$, for which it applies $UU^\top \varepsilon_\parallel(t; \eta) = \varepsilon_\parallel(t; \eta)$.

Theorem 1.4.2 ([210, Proposition 4.2, page 1899]). *Consider solving the initial value problem (1.2.1), approximated using the POD reduced model in the interval \mathcal{T} for a given parameter η . Let $U \in \mathbb{R}^{N \times n}$ be the reduced basis, and let \mathcal{M}_n denote the affine subspace onto which the POD basis projects. The solution $u(t; \eta)$ to the FOM and the solution $u_r(t; \eta)$ to the ROM can be writtern as*

$$u(t; \eta) = Uz_e(t; \eta) + U_c v(t; \eta),$$

and

$$u_r(t; \eta) = Uz_e(t; \eta) + Uw(t; \eta),$$

where $z_e(t; \eta) = U^\top u(t; \eta)$ and $U_c \in \mathbb{R}^{N \times (N-n)}$ is the orthogonal complement to U . Since the projected FOM solution is $u_\perp(t; \eta) = Uz_e(t; \eta)$, the two error components $\varepsilon_\perp(t; \eta)$ and $\varepsilon_\parallel(t; \eta)$ are given by

$$\varepsilon_\perp(t; \eta) = -U_c v(t; \eta),$$

and

$$\varepsilon_\parallel(t; \eta) = Uw(t; \eta).$$

Note that $z_e(t; \eta) \in \mathbb{R}^n$, $w(t; \eta) \in \mathbb{R}^n$, and $v(t; \eta) \in \mathbb{R}^{N-n}$. Let $\gamma \leq 0$ be the Lipschitz constant of $U^\top f(t, u; \eta)$ in the directions orthogonal to \mathcal{M}_n in a region containing $u(t; \eta)$ and $u_\perp(t; \eta)$. In

particular, suppose

$$\|U^\top f(t, u_\perp + U_c v; \eta) - U^\top f(t, u_\perp; \eta)\|_2 \leq \gamma \|v\|_2,$$

for all $(v, t) \in D \subset \mathbb{R}^{N-n} \times \mathcal{T}$, where the region D is such that the associated region $\tilde{D} = \{(t, u_\perp + U_c v) : (v, t) \in D\} \subset \mathbb{R}^N \times \mathcal{T}$ contains (t, u_\perp) and (t, u) for all $t \in \mathcal{T}$. Let $\mu \left(U^\top \frac{\partial f}{\partial u}(t, U z; \eta) \right) \leq \bar{\mu}$ for $(s, t) \in V \subset \mathbb{R}^n \times \mathcal{T}$, where the region V is such that it contains $(z_e(t; \eta), t)$ and $(z_e(t; \eta) + w(t; \eta), t)$ for all $t \in \mathcal{T}$ and μ denotes the logarithmic norm related to the 2-norm, defined as

$$\mu(A) = \lim_{h \rightarrow 0, h > 0} \frac{\|I_n + hA\|_2 - 1}{h},$$

for $A \in \mathbb{R}^{n \times n}$. Then the error ε_\parallel in the ∞ -norm satisfies

$$\|\varepsilon_\parallel\|_\infty \leq \|\varepsilon_\perp\|_2 \frac{\gamma}{\sqrt{2\bar{\mu}}} \sqrt{e^{2\bar{\mu}T} - 1}, \quad (1.4.5)$$

and the 2-norm of the total error satisfies

$$\|\varepsilon\|_2 \leq \|\varepsilon_\perp\|_2 \sqrt{1 + \frac{\gamma^2}{4\bar{\mu}^2} (e^{2\bar{\mu}T} - 1 - 2\bar{\mu}T)}. \quad (1.4.6)$$

Although they represent only upper bounds, the message we take home from (1.4.5) and (1.4.6) is that an inaccurate basis, i.e., large $\varepsilon_\perp(t; \eta)$, can lead to a compounding of the total error $\varepsilon(t; \eta)$ and the parallel error $\varepsilon_\parallel(t; \eta)$ over time. This problem turns out to be particularly relevant, especially in the case of advection dominated problems, and a solution is proposed in Chapters 4 and 5. Even in the case of a small orthogonal component $\varepsilon_\perp(t; \eta)$ of the error, the total error $\varepsilon(t; \eta)$ may increase over time due to the approximation of the FOM dynamics in the ROM. In Chapters 7 and 8, we suggest approaches to improve the full dynamics approximation via a correction term introduced into (1.4.2), thereby reducing the parallel component $\varepsilon_\parallel(t; \eta)$ of the error.

A more detailed analysis of error bounds for time-dependent FOM that also considers other sources of error, such as the dimension n , the exact sampling in the time-parameter domain of the snapshots, and the type of snapshots can be found in [148].

1.5 Efficient treatment of nonlinear term

One crucial assumption for an efficient Offline-Online decomposition is that the dynamics of the FOM (1.2.1) is *affine* in the parameter η and time t , i.e.,

$$f(t, u; \eta) = \sum_{q=1}^Q \Theta^q(\eta, t) f^q(u),$$

where $\Theta^q : \Gamma \times \mathcal{T} \mapsto \mathbb{R}$ are parameter and time dependent functions and $f^q(u)$ are parameter and time independent linear functions in u . In this scenario, the offline-online strategy allows to reduce the computational burden associated with the online phase, which is particularly useful in multi-query and real-time simulations where approximations to the solution to (1.2.1) for new

parameter values are required. In particular, the operational count of the online stage would be independent of the number N of degrees of freedom of the FOM. The linearity requirement on the functions $f^q(u)$ can be relaxed by requiring a more general low-order polynomial dependence on u . By rearranging the order of computation, the tensorial POD technique exploits the structure of polynomial nonlinearities to separate the quantities that depend on the dimension N of the FOM from the reduced variables. We refer the reader to [240] for more details on this approach and we apply this method, with accompanying cost analysis, in Chapter 4. However, the same approach cannot be used in case of general nonlinearity, where additional complexity-reduction mechanisms, more generally known as hyper-reduction techniques, are required. In particular, the computational complexity associated with the repeated computation of the term $U^\top f(t, Uz; \eta)$ in (1.4.2) has to be reduced. Several methods have been proposed in the past: here we mention the gappy POD method [50; 85], the reduced-order quadrature [11], the RB-sparsification technique [51], and different collocation methods [220; 13]. In this thesis we consider the discrete empirical interpolation method (DEIM) [57], a variant of the empirical interpolation method (EIM) introduced in [21] and applied in MOR in [130]. In the same spirit as missing point estimation (MSE) [13], the DEIM method identifies a subset of nonlinearity components to avoid the costly evaluation of the same nonlinearity on all grid points. The approximation of the nonlinearity of the FOM is realized through a coefficient interpolation matrix, whose cost scales proportionally to the size of the set of spatial indices and thus obtaining a computational speedup.

The selected interpolation indices are used to define the row selection operator

$$P = \begin{bmatrix} e_{\rho_1} & \dots & e_{\rho_{n_d}} \end{bmatrix} \in \mathbb{R}^{N \times n_d}, \quad (1.5.1)$$

where $e_{\rho_i} = \begin{bmatrix} 0 & \dots & 0 & 1 & 0 & \dots & 0 \end{bmatrix}^\top \in \mathbb{R}^N$ is the ρ_i -th column of the identity matrix \mathbb{I}_N and $n_d < N$ is the number of selected indices. Let us assume that the manifold

$$\mathcal{M}^f := \{f(u(t; \eta); \eta) \mid \eta \in \Gamma, t \in \mathcal{T}\} \quad (1.5.2)$$

can be accurately approximated via a linear subspace $\mathcal{M}_{\text{DEIM}}^f$ of dimension n_d , i.e.,

$$f(u(t; \eta); \eta) \approx f_{\text{DEIM}}(u(t; \eta); \eta) = \Phi^f \theta(t; \eta), \quad (1.5.3)$$

where $\theta(t; \eta) \in \mathbb{R}^{n_d}$ is a coefficient vector to be computed with respect to the basis $\Phi^f \in \mathbb{R}^{N \times n_d}$. Similarly to the POD method, a basis Φ^f can be computed via a SVD on a set of snapshots obtained by sampling (1.5.2) in time and parameter space. The i -th basis vector is denoted by ϕ_i^f . The coefficient vector $\theta(t; \eta)$ is uniquely determined by imposing the n_d interpolation constraints (1.5.1) in (1.5.3), resulting in

$$P^\top f(u(t; \eta); \eta) = (P^\top \Phi^f) \theta(t; \eta).$$

Assuming $P^\top \Phi^f$ is invertible, we obtain

$$\theta(t; \eta) = (P^\top \Phi^f)^{-1} P^\top f(u(t; \eta); \eta)$$

and the approximation of the nonlinearity becomes

$$f_{\text{DEIM}}(u(t; \eta); \eta) = \Pi_{\mathcal{M}_{\text{DEIM}}^f} f(u(t; \eta); \eta), \quad (1.5.4)$$

where $\Pi_{\mathcal{M}_{\text{DEIM}}^f} := \Phi^f (P^\top \Phi^f)^{-1} P^\top$ is the DEIM oblique projector operator, which satisfies $\Pi_{\mathcal{M}_{\text{DEIM}}^f}^2 = \Pi_{\mathcal{M}_{\text{DEIM}}^f}$ and $\|\Pi_{\mathcal{M}_{\text{DEIM}}^f}\|_2 = \|\mathbb{I}_{n_d} - \Pi_{\mathcal{M}_{\text{DEIM}}^f}\|_2$ if $\Pi_{\mathcal{M}_{\text{DEIM}}^f} \neq \mathbb{I}_{n_d}, \mathbb{O}_{n_d}$. Additional properties of oblique projectors are given in [247]. Approximation (1.5.4) is effectively an interpolation relation, since f_{DEIM} coincides with f at the interpolation points ρ_i , i.e.,

$$P^\top f_{\text{DEIM}}(u(t; \eta); \eta) = P^\top \Phi^f (P^\top \Phi^f)^{-1} P^\top f(u(t; \eta); \eta) = P^\top f(u(t; \eta); \eta).$$

In terms of application, the matrix $\Phi^f (P^\top \Phi^f)^{-1} \in \mathbb{R}^{n \times n_d}$ is assembled during the offline phase and the nonlinear term f is evaluated only in the components specified by P during the online phase. As a result, the computational cost of approximating the nonlinearity depends only on n and n_d but is independent of N . The algorithm to determine the interpolation indices ρ_i has been originally provided in [57] and it is summarized in Algorithm 2. An error bound for f_{DEIM}

Algorithm 2 DEIM algorithm

- 1: **procedure** SELECTION INDICES DEIM($\{\phi_1, \dots, \phi_{n_d}\} \subset \mathbb{R}^N$)
 - 2: Pick ρ_1 as the index corresponding to the largest component in absolute value of ϕ_1^f .
 - 3: Let $\Phi^f \leftarrow [\phi_1^f]$, $P \leftarrow [\rho_1]$.
 - 4: **for** $i = 2$ to n_d **do**
 - 5: Solve $(P^\top \Phi^f) \mathbf{c} = P^\top \phi_i^f$ for \mathbf{c} .
 - 6: Define $\mathbf{r} \leftarrow \phi_i^f - \Phi^f \mathbf{c}$.
 - 7: Let ρ_i be the index of the largest component in absolute value of \mathbf{r} .
 - 8: Let $\Phi^f \leftarrow [\Phi^f \phi_i^f]$, $P \leftarrow [P \rho_i]$.
-

is then given in the following theorem.

Theorem 1.5.1 ([57, Lemma 1, page 5]). *Let f_{DEIM} be the DEIM approximation defined in (1.5.4). Then*

$$\|f - f_{\text{DEIM}}\|_2 \leq \|\Pi_{\mathcal{M}_{\text{DEIM}}^f}\|_2 \|(\mathbb{I}_{n_d} - \Phi^f (\Phi^f)^\top) f\|_2. \quad (1.5.5)$$

In (1.5.5), the quality of the basis Φ chosen to approximate the nonlinearity f directly affects term $\|(\mathbb{I}_{n_d} - \Phi^f (\Phi^f)^\top) f\|_2$ in the error bound. The quantity $\|\Pi_{\mathcal{M}_{\text{DEIM}}^f}\|_2$, known as DEIM error constant, plays the role of the conditioning number of the DEIM approximation and depends on the algorithm chosen for the interpolation indices.

For the classical DEIM algorithm schematized in Algorithm 2, the Φ^f basis not only determine the approximating space, but also the interpolation indices used to sample the nonlinear term, further affecting the quality of the approximation. As pointed out in [57], the justification of Algorithm 2 is given by the limitation, at each step of the iterative process of index selection, of the growth of the approximation error of f . This result is due to the fact that selecting the index of the maximum of the interpolation error \mathbf{r} , means minimizing its reciprocal value, and thus the approximation error (1.5.1) following the proof of Theorem 1.5.1 given in [57]. From this we also get the invertibility of the term $P^\top \Phi^f$.

A variation of the original algorithm for the index subset selection based on a randomized algorithm is proposed in [222], with the aim of reducing the computational cost required and to make the process parallelizable. With similar goals, in [78] the Q-DEIM variant is proposed, which leverages the QR-factorization with column pivoting to further reduce the term $\|\Pi_{\mathcal{M}_{\text{DEIM}}^f}\|_2$. Finally, we mention the adaptive DEIM algorithm [196], which evolves in time the DEIM basis via optimal rank-one updates and will be used in the context of plasma physics in Chapter 5.

Structure-preserving Model **Part I**

Order Reduction

The first part of the thesis is devoted to the conservation, during the reduction process, of invariants and structures characterizing the high fidelity models in the reduction process, which is one of the main original contributions of the thesis.

As stated in Section 1.1, in the past decade, MOR has been successful in reducing the computational complexity of elliptic and parabolic systems of PDEs. In the framework of RB, classical approaches have been presented in Chapter 1. However, the MOR of hyperbolic equations remains a challenge. Symmetries and conservation laws, which are distinctive features of such systems, are often destroyed by conventional MOR techniques, resulting in perturbed, and often unstable reduced systems. The importance of energy conservation is well-known for correct numerical integration of fluid flow. In Chapter 2, we discuss a novel approach in model reduction that exploits skew-symmetry of conservative and centered discretization schemes to recover conservation of energy at the level of the reduced system. Moreover, we argue that the reduced system, constructed with the new method, can be identified by a reduced energy that mimics the energy of the high-fidelity system. Therefore the loss in energy associated with the model reduction remains constant in time. Preserving this physical property of the original problem ensures an overall correct evolution of the numerical fluid that ensures the robustness of the reduced system. We evaluate the performance of the proposed method through numerical simulation of various fluid flows and a numerical simulation of a continuous variable resonance combustor model.¹

Chapter 3 is devoted to the recent developments of projection-based MOR techniques targeting Hamiltonian problems. Hamilton's principle completely characterizes many high-dimensional models in mathematical physics, resulting in rich geometric structures, with examples in fluid dynamics, quantum mechanics, optical systems, and epidemiological models. Unfortunately, as in the case of energy-preserving problems, standard reduction approaches do not guarantee the conservation of the delicate dynamics of Hamiltonian problems, resulting in reduced models plagued by instability or accuracy loss over time. By approaching the reduction process from the geometric perspective of symplectic manifolds, the resulting reduced models inherit the stability and conservation properties of the high-dimensional formulations. We first introduce the general principles of symplectic geometry, including symplectic vector spaces, Darboux's theorem, and Hamiltonian vector fields. These notions are then used as a starting point to describe different structure-preserving RB algorithms, including SVD-based approaches and greedy techniques. We conclude this review Chapter by addressing the reduction of problems that involve a dissipation term or are in a non-canonical Hamiltonian form. Even though the methods presented in this Chapter are not novelties of the thesis work, we deem it necessary to introduce the reader to this part of ROM literature to have a clearer picture of Chapters 4 and 5.²

In Chapter 4, an adaptive structure-preserving model order reduction method for finite-dimensional parametrized Hamiltonian systems modeling non-dissipative phenomena is introduced. To overcome the slowly decaying Kolmogorov n -width (1.2.3) typical of transport problems, the full model is approximated on local reduced spaces adapted in time using dynamical low-rank approximation techniques. The reduced dynamics is prescribed by approximating the symplectic projection of the Hamiltonian vector field in the tangent space to the local reduced space. This step ensures that the canonical symplectic structure of the Hamiltonian dynamics is preserved during the reduction. In addition, accurate approximations with low-rank reduced solutions are obtained by allowing the dimension of the reduced space to change during the time evolution.

¹In accordance with the Springer Copyright Transfer Statement, parts of this chapter are adapted from [4].

²In accordance with the EMS Press Author License Agreement, parts of this chapter are adapted from [127].

Whenever the quality of the reduced solution, assessed via an error indicator, is not satisfactory, the reduced basis is augmented in the worst approximated parameter direction in the current basis. Extensive numerical tests involving wave interactions and nonlinear transport problems demonstrate the superior stability properties and considerable runtime speedups of the proposed method as compared to the global and traditional reduced basis approaches presented in Chapter 3.³

The same Hamiltonian and action principle formulations emerge in the field of plasma physics. High-resolution simulations of particle-based kinetic plasma models typically require a high number of particles and thus often become computationally intractable. This burden is exacerbated in multi-query simulations, where the problem depends on a set of parameters, creating the need for reduction techniques for parametric plasma physics problems. Since the problem's non-dissipative and highly nonlinear nature makes it reducible only locally in time, we adopt the nonlinear reduced basis approach discussed in Chapter 4 where the reduced phase space evolves in time. More in detail, we derive ROMs for the semi-discrete Hamiltonian system resulting from a geometric particle-in-cell approximation of the parametric Vlasov–Poisson equations. This strategy allows a significant reduction in the number of simulated particles, but the evaluation of the nonlinear operators associated with the Vlasov–Poisson coupling remains computationally expensive. In Chapter 5, we propose a novel reduction of the nonlinear terms that combines adaptive parameter sampling and hyper-reduction techniques to address this. The proposed approach allows decoupling the operations having a cost dependent on the number of particles from those that depend on the instances of the required parameters. In particular, in each time step, the electric potential is approximated via dynamic mode decomposition (DMD) and the particle-to-grid map via the discrete empirical interpolation method (DEIM), described in Chapter 5. These approximations are constructed from data obtained from a past temporal window at a few selected values of the parameters to guarantee a computationally efficient adaptation. The resulting DMD-DEIM reduced dynamical system retains the Hamiltonian structure of the full model, provides good approximations of the solution, and can be solved at a fraction of the original computational cost.⁴

A summary of the results and possible insights for future research for each of the chapters are provided in Chapter 6.

Let us observe that, even though the Euler equations considered in Chapter 2 for inviscid and incompressible flow can be put, in its natural Eulerian coordinates, into an Hamiltonian formulation, this is not the *canonical* formulation discussed for most of Chapters 3, 4, and 5. Hence, skipping Chapter 2 does not compromise the comprehension of the remaining Chapters of this block, which, however, requires reading Chapter 1, where RB-based methods are introduced.

³In accordance with the EDP Sciences Copyright Transfer Statement, parts of this chapter are adapted from [131].

⁴In accordance with the AMS Copyright Agreement Form, parts of this chapter are adapted from [129].

2 Model order reduction of fluid equations in skew-symmetric form

MOR, particularly RB methods, has emerged as a powerful approach to coping with the complex and computationally intensive models in engineering and science. As seen in Chapter 1, such techniques construct a reduced ordered representation for the state of a model, which accurately approximates the configuration of the system. The evaluation of this representation is then possible with considerable acceleration.

Although RB methods successfully reduce the computational complexity of models with elliptic and parabolic PDEs, MOR of systems of hyperbolic equations, or models with strong advective terms, remain a challenge. Such models often arise from a set of invariants and conservation laws, some of which are violated by MOR, resulting in a qualitatively wrong and sometimes unstable solution.

Constructing MOR techniques and RB methods that preserve intrinsic structures has recently attracted attention [139; 87; 23]. For example, preserving time symmetries of Lagrangian, Hamiltonian, and port-Hamiltonian systems can be found in the works of [198; 51; 3; 56; 115] and will be the focus of Chapter 3. Conserving inf-sup stability, in the context of finite element methods, can be found in [87; 19]. Furthermore, a flux preserving model reduction for finite volume methods is presented in [49].

Large-scale simulations of fluid flows arise in various disciplines and industries. Therefore, reducing fluid flows, especially when advective terms are dominant, is essential. It is well known that energy conservation, especially kinetic energy, is critical to a qualitatively correct numerical integration of fluid flows. Unfortunately, conventional model reduction techniques often violate the conservation of mass, momentum[49], or energy in fluid flows, resulting in unstable reduced systems, particularly for long time integration.

In this Chapter we discuss how to preserve skew-symmetry of the differential operators at the level of the reduced system. The preservation of the skew-symmetry results in the conservation of quadratic invariants. Furthermore, the conservation of quantities in the proposed method is guaranteed through the mathematical formulation of the reduced system for any orthonormal reduced basis. Therefore, the offline and online computational costs of this method are comparable with conventional MOR techniques introduced in Chapter 1. However, other conservative model reduction methods often require solving multiple nonlinear optimization problems to ensure conservation properties, increasing the computational costs. Furthermore, we show that the reduced system, as a system of coupled differential equations, contains quadratic invariants and associated energy that approximates the high-fidelity system's energy. Therefore, a proper

time stepping scheme preserves the reduced representation of the energy, and therefore, the loss in energy due to model reduction remains constant in time. Furthermore, through numerical experiments, we demonstrate that a quasi-skew-symmetric form of fluid flow, i.e., a formulation where only spacial differential operators are in a skew-symmetric form, offers remarkable stability properties in terms of MOR. This allows an explicit time-integration to be utilized while recovering the robustness of skew-symmetric forms at the reduced level.

The organization of this Chapter adheres to the following scheme. In Section 2.1 we discuss skew-symmetric and conservatives methods for compressible and incompressible fluid flows. Conservative and energy-preserving model reduction of fluid flows is discussed in Section 2.2. We evaluate the performance of the method through numerical simulations of incompressible and compressible fluid flow in Section 2.3. We also apply the method to construct a reduced system for the continuous variable resonance combustor, a one dimensional reaction-diffusion model for a rocket engine.¹

2.1 Skew symmetric and centered schemes for fluid flows

In this Section we summarize the conservation properties of skew-symmetric forms and discretization schemes at the FOM level.

2.1.1 Conservation laws

In the context of fluid flows, transport of conserved quantities can be expressed as

$$\frac{\partial}{\partial t} \rho \varphi + \nabla \cdot (\rho u \varphi) = \nabla \cdot F_\varphi, \quad \text{in } \Omega \subset \mathbb{R}^s. \quad (2.1.1)$$

Here, $s = 1, 2$, or 3 , $\rho : \Omega \rightarrow \mathbb{R}$ is the density, $u : \Omega \rightarrow \mathbb{R}^s$ is the velocity vector field, φ is a measured scalar quantity of the flow, and F_φ is the flux function associated to φ . Integration of (2.1.1) over Ω yields

$$\frac{d}{dt} \int_{\Omega} \rho \varphi \, dx = \int_{\partial\Omega} (F_\varphi - \rho u \varphi) \cdot \hat{n} \, ds, \quad (2.1.2)$$

where $\partial\Omega$ is the boundary of Ω , and \hat{n} is the unit outward normal vector to $\partial\Omega$. This means that the quantity $\rho \varphi$ is explicitly conserved over control volumes. Therefore, (2.1.2) is referred to as the *conservative form* and the convective term in (2.1.1) is referred to as the *divergence form*. However, using the *continuity equation*

$$\frac{\partial}{\partial t} \rho + \nabla \cdot (\rho u) = 0,$$

we can write (2.1.1) as

$$\rho \frac{\partial}{\partial t} \varphi + (\rho u) \cdot \nabla \varphi = \nabla \cdot F_\varphi.$$

The convective term in this formulation is referred to as the *advective form*. The *skew-symmetric* form of the convective term is obtained by the arithmetic average of the divergent and the

¹The author's original contribution for this part of the thesis was to define the research question and develop the method for the case of incompressible fluids. The author participated in the extension of the approach to the case of compressible fluids, for which the main contributor was Dr. Maboudi. The author was primarily responsible for validating the proposed approach by numerical experiments.

advective form:

$$\frac{1}{2} \left(\rho \frac{\partial}{\partial t} \varphi + \frac{\partial}{\partial t} (\rho \varphi) \right) + \frac{1}{2} ((\rho u) \cdot \nabla \varphi + \nabla \cdot (\rho u \varphi)) = \nabla \cdot F_\varphi. \quad (2.1.3)$$

Multiplying (2.1.3) with φ yields

$$\frac{1}{2} \left(\rho \varphi \frac{\partial}{\partial t} \varphi + \varphi \frac{\partial}{\partial t} (\rho \varphi) \right) + \frac{1}{2} ((\rho u \varphi) \cdot \nabla \varphi + \varphi \nabla \cdot (\rho u \varphi)) = \varphi \nabla \cdot F_\varphi. \quad (2.1.4)$$

Using the product rule, we recover

$$\frac{\partial}{\partial t} \rho \varphi^2 + \nabla \cdot (\rho u \varphi^2) = \varphi \nabla \cdot F_\varphi.$$

Therefore, $\rho \varphi^2$ is a conserved quantity for $\nabla \cdot F_\varphi = 0$. Since the divergence, the advective, and the skew-symmetric forms are analytically equivalent at the continuous level, φ^2 is a conserved quantity for all forms. However, the equivalence of these forms is not preserved through a general discretization scheme, and we can not expect φ^2 to be a conserved quantity at the discrete level. This could result in unstable simulations since it is commonly accepted that quadratic invariants conservation is a key feature for the stability of unsteady computations. To motivate the numerical advantages of the skew-symmetric form, consider the operator

$$S_{\rho u}(\cdot) = \frac{1}{2} ((\rho u) \cdot \nabla + [\nabla \cdot \rho u]) (\cdot),$$

with $\nabla \cdot \rho u(\cdot) = \nabla \cdot \rho u$. With a proper set of boundary conditions, this operator is a skew-adjoint operator on L^2 . Here, $[\cdot]$ indicates that the inside of the brackets act as a differential operator. This skew-adjoint property is used later to show the conservation of quadratic quantities in (2.1.1). Similarly, we can define a skew-adjoint operator with respect to the time variable

$$S_{\rho, \partial_t} = \frac{1}{2} \left(\rho \frac{\partial}{\partial t} + \left[\frac{\partial}{\partial t} \rho \right] \right).$$

Here, the subscript ∂_t is to emphasize that S_{ρ, ∂_t} is a differential operator with respect to t . A proper time and space discretization of $S_{\rho u}$ and S_{ρ, ∂_t} can preserve the skewness property, which is the focus of skew-symmetric numerical schemes, with the result of preserving the stability and the conservation properties of the continuous formulation. Numerical time integration of (2.1.4) can be challenging since the time differentiation of different variables is present. Following [179], we rewrite (2.1.4) as

$$\sqrt{\rho} \frac{\partial}{\partial t} (\sqrt{\rho} \varphi) + S_{\rho u}(\varphi) = \nabla \cdot F_\varphi. \quad (2.1.5)$$

Time integration of this form is presented in [179; 213]. Note that one can also generate a quasi-skew-symmetric form [34; 181] of (2.1.1) as

$$\frac{\partial}{\partial t} (\rho \varphi) + \frac{1}{2} (\nabla \cdot (\rho u \varphi) + \rho u \cdot \nabla \varphi + \varphi \nabla \cdot (\rho u)) = \nabla \cdot F_\varphi. \quad (2.1.6)$$

Even though this is not a fully skew-symmetric form (skew-symmetric only in space), the quasi-skew-symmetric form has proved to be more stable than the divergence and the advective forms [179; 34; 181]. Note that this quasi-skew-symmetric form is identical to the skew-symmetric form in the incompressible limit.

We point out that stability is only a necessary condition for a solution to being physical. In the presence of discontinuities and stretched grid nodes, ringing and oscillations do develop if an artificial viscosity is not added, as it will be shown in Section 2.3. For the sake of the understanding of the properties of the fully-discrete formulation, however, explicitly adding dissipation is preferable over relying on the numerical dissipation created by a numerical scheme that is not structure-preserving for two reasons. First, it allows tailoring the viscosity introduced to the problem taken into consideration. Second, while the effect of viscous terms is well understood in the RB framework [6], the same cannot be said for the projection of nonlinear transport terms. However, as we show in Section 2.2, by correctly retaining the neutrality of these terms in the evolution equation for the conserved quantities at the reduced level, the artificial viscosity affects the conservation balance as an isolated and therefore controllable term.

2.1.2 Incompressible fluid

Consider the governing equations of an incompressible fluid with a skew-symmetric convective term:

$$\begin{cases} \nabla \cdot u = 0, \\ \frac{\partial}{\partial t} u + S_{\mathbf{u}}(u) + \nabla p = \nabla \cdot \tau, \end{cases} \quad (2.1.7)$$

defined on Ω . Here, $p : \Omega \rightarrow \mathbb{R}^+$ is the pressure, $\tau : \Omega \rightarrow \mathbb{R}^{s \times s}$ is the viscous stress tensor, and $S_{\mathbf{u}} = \frac{1}{2}([\nabla \cdot u] + u \cdot \nabla)$. It is straightforward to check

$$\frac{\partial}{\partial t} K + \nabla \cdot (Ku) + \nabla \cdot (pu) = \nabla \cdot (\tau u) - (\tau \nabla) \cdot u, \quad (2.1.8)$$

where $K = \frac{1}{2} \sum_{i=1}^s u_i^2$ is the kinetic energy and we used

$$u \cdot S_{\mathbf{u}}(u) = \nabla \cdot (Ku).$$

The only non-conservative term in (2.1.8) is $-(\tau \nabla) \cdot u$, which corresponds to the dissipation of kinetic energy. Therefore, in the absence of the viscous terms, K is a conserved quantity of the system, and $\frac{d}{dt} \int_{\Omega} K dx < 0$ when $\tau \neq 0$. Note that as long as $\nabla \cdot u = 0$, as discussed in Section 2.1.1, the divergence, the convective, and the skew-symmetric forms are identical for the incompressible fluid equation. Thus, kinetic energy is conserved for all forms. However, these forms are not identical for a general discretization scheme, and often conservation of kinetic energy (in the discrete sense) may be violated.

A skew symmetric discretization of (2.1.7) is a scheme that exploits the skew-adjoint property of $S_{\mathbf{u}}$ and ensures conservation of kinetic energy at the discrete level. We uniformly discretize Ω into N points and denote by $\mathbf{u} \in \mathbb{R}^{N \times s}$, $\mathbf{p} \in \mathbb{R}^N$, and $T \in \mathbb{R}^{N \times s \times s}$ the discrete representation of u , p , and τ , respectively. Let D_j be the centered finite difference scheme for $\partial/\partial x_j$, and for $j = 1, \dots, s$. The momentum equation in (2.1.7) is discretized as

$$\frac{d}{dt} \mathbf{u}_i + S_{\mathbf{u}} \mathbf{u}_i + D_i \mathbf{p} = \sum_{j=1}^s D_j T_{i,j}, \quad i = 1, \dots, s \quad (2.1.9)$$

where $S_{\mathbf{u}}$ is the discretization of $S_{\mathbf{u}}$ given by

$$S_{\mathbf{u}} = \sum_{j=1}^d D_j U_j + U_j D_j \quad (2.1.10)$$

and U_j contains components of \mathbf{u}_i on its diagonal. We require D_j to satisfy:

- $D_j = -D_j^\top$
- $D_j \mathbf{1} = \mathbf{0}$, where $\mathbf{1}$ and $\mathbf{0}$ are vectors of ones and zeros, respectively.

The two conditions yield

$$S_{\mathbf{u}} = -S_{\mathbf{u}}^\top, \quad \mathbf{1}^\top S_{\mathbf{u}} \mathbf{u}_i = 0, \quad i = 1, \dots, d.$$

Conservation of momentum in the discrete sense is expressed as

$$\frac{d}{dt} \sum_{i=1}^d \mathbf{1}^\top \mathbf{u}_i = \sum_{i=1}^d \left(-\mathbf{1}^\top S_{\mathbf{u}} \mathbf{u}_i - \mathbf{1}^\top D_i \mathbf{p} \sum_{j=1}^d \mathbf{1}^\top D_j T_{ij} \right) = 0.$$

Similarly, it is verified that

$$\frac{d}{dt} \sum_{i=1}^d \left(\frac{1}{2} \mathbf{u}_i^\top \mathbf{u}_i \right) = - \sum_{i,j=1}^d T_{ij} D_j \mathbf{u}_i \leq 0. \quad (2.1.11)$$

Both conditions for D_j are easily checked for a centered finite differences scheme on a periodic domain. For other types of boundaries, e.g., wall boundary and inflow/outflow, we refer the reader to [76; 180] to construct the proper discrete centered differentiation operator. We note that the finite differences schemes are chosen here for illustration purposes. It is easily checked that any discrete differentiation operator that satisfies discrete integration by parts, e.g., summation by part (SBP) methods and discontinuous Galerkin (DG) methods, also satisfies conditions 1 and 2 and can be used to construct a skew-symmetric discretization.

2.1.3 Compressible fluid

Consider the equations governing the evolution of a compressible fluid in a skew-symmetric form in one spatial dimension

$$\begin{cases} \frac{\partial}{\partial t} \rho + \frac{\partial}{\partial x} (\rho u) = 0, \\ S_{\rho, \partial_t}(u) + S_{\rho u}(u) + \frac{\partial}{\partial x} p = \frac{\partial}{\partial x} \tau, \\ \frac{\partial}{\partial t} \rho E + \frac{\partial}{\partial x} (uE + up) = \frac{\partial}{\partial x} (u\tau - \varphi). \end{cases} \quad (2.1.12)$$

Here $E = e + u^2/2$ is the total energy per unit mass, with $e = p\rho(\gamma - 1)$ being the internal energy, γ the adiabatic gas index, and $\varphi = -\lambda \frac{\partial T}{\partial x}$ is the heat flux, with λ as the heat conductivity. The remaining variables are the same as those discussed in Section 2.1.2. Following [213], the

evolution of the momentum equation is

$$\begin{aligned} \frac{d}{dt} \left(\frac{1}{2} \rho u^2 \right) + \frac{\partial}{\partial x} \left(\rho u \left(\frac{1}{2} u^2 \right) \right) &= \frac{1}{2} u \left(\frac{d}{dt} \rho u + \rho \frac{d}{dt} u \right) + \frac{1}{2} u \left(\left[\frac{\partial}{\partial x} \rho u \right] u + \rho u \frac{\partial}{\partial x} u \right) \\ &= -u \frac{\partial}{\partial x} p + u \frac{\partial}{\partial x} \tau, \end{aligned} \quad (2.1.13)$$

leaving only the pressure work and viscous stress as only possible sources and sinks of kinetic energy. Substituting this into the equation in (2.1.12), while assuming a constant adiabatic index, yields

$$\frac{1}{1-\gamma} \frac{d}{dt} p + \frac{\gamma}{\gamma-1} \frac{\partial}{\partial x} u p - u \frac{\partial}{\partial x} p = -u \frac{\partial}{\partial x} \tau + \frac{\partial}{\partial x} (u \tau - \varphi). \quad (2.1.14)$$

We discretize the real line, uniformly, into N grid points and denote by $\boldsymbol{\rho}, \mathbf{u}, \mathbf{p} \in \mathbb{R}^N$, the discrete representations of ρ, u , and p , respectively. Using the matrix differentiation operator $D \in \mathbb{R}^{N \times N}$ (we omit the subscript "i" for the one dimensional case), introduced in Section 2.1.2, we define the skew-symmetric matrix operator $S_{\boldsymbol{\rho}\mathbf{u}} = \frac{1}{2} (D U R + R U D)$, where R is the matrix that contains $\boldsymbol{\rho}$ on its diagonal. Semi-discrete expression of (2.1.12) and (2.1.14) takes the form

$$\begin{cases} \frac{d}{dt} \boldsymbol{\rho} + D U \boldsymbol{\rho} = 0, \\ S_{\boldsymbol{\rho}, \partial_t}(\mathbf{u}) + S_{\boldsymbol{\rho}\mathbf{u}} \mathbf{u} + D \mathbf{p} = D T, \\ \frac{1}{\gamma-1} \frac{d}{dt} \mathbf{p} + \frac{\gamma}{\gamma-1} D U \mathbf{p} - U D \mathbf{p} = -U D T + D (U T - \varphi). \end{cases} \quad (2.1.15)$$

Recalling the two conditions for D , discussed in Section 2.1.2, it is easily verified that

$$S_{\boldsymbol{\rho}\mathbf{u}}^\top = -S_{\boldsymbol{\rho}\mathbf{u}}, \quad \mathbb{1}^\top S_{\boldsymbol{\rho}\mathbf{u}} \mathbf{u} = -\mathbf{u}^\top D U \boldsymbol{\rho}. \quad (2.1.16)$$

Conservation of the total mass $M(t)$ is expressed as

$$\frac{d}{dt} M = \frac{d}{dt} \mathbb{1}^\top \boldsymbol{\rho} = -\mathbb{1}^\top D R \mathbf{u} = 0.$$

Furthermore, we recover conservation of total momentum $P(t)$ in the discrete sense as

$$\begin{aligned} \frac{d}{dt} P &= \frac{d}{dt} (\boldsymbol{\rho}^\top \mathbf{u}) = \frac{1}{2} \frac{d}{dt} (\boldsymbol{\rho}^\top \mathbf{u}) + \frac{1}{2} \left(\boldsymbol{\rho}^\top \frac{d}{dt} \mathbf{u} + \mathbf{u}^\top \frac{d}{dt} \boldsymbol{\rho} \right) \\ &= \frac{1}{2} \mathbf{u}^\top \frac{d}{dt} \boldsymbol{\rho} + \mathbb{1}^\top S_{\boldsymbol{\rho}, \partial_t}(\mathbf{u}) \\ &= -\frac{1}{2} \mathbf{u}^\top D U \boldsymbol{\rho} - \frac{1}{2} \mathbb{1}^\top S_{\boldsymbol{\rho}\mathbf{u}} \mathbf{u} - \mathbb{1}^\top D \mathbf{p} + \mathbb{1}^\top D T = 0. \end{aligned} \quad (2.1.17)$$

Here we used (2.1.16) and the mass and the momentum equations in (2.1.13). Similarly, for the conservation of the total energy, we have

$$\frac{d}{dt} \left(\frac{1}{\gamma-1} \mathbb{1}^\top \mathbf{p} + \frac{1}{2} (R \mathbf{u})^\top \mathbf{u} \right) = \frac{d}{dt} \left(\frac{1}{\gamma-1} \mathbb{1}^\top \mathbf{p} \right) + \frac{1}{2} \mathbf{u}^\top S_{\boldsymbol{\rho}, \partial_t}(\mathbf{u}) = 0. \quad (2.1.18)$$

In addition to the conservation of the total energy, the skew-symmetric form of (2.1.13) also conserves the evolution of the kinetic energy $K(t)$:

$$\begin{aligned} \frac{d}{dt}K &= \frac{d}{dt} \left(\frac{1}{2} \mathbf{u}^\top R \mathbf{u} \right) = \frac{1}{2} \mathbf{u}^\top S_{\rho, \partial_t}(\mathbf{u}) = -\mathbf{u}^\top S_{\rho \mathbf{u}} \mathbf{u} + \mathbf{u}^\top D \mathbf{p} + \mathbf{u}^\top DT \\ &= \mathbf{u}^\top D \mathbf{p} + \mathbf{u}^\top DT, \end{aligned} \quad (2.1.19)$$

where we have used the skew-symmetry of $S_{\rho \mathbf{u}}$. Therefore, only the pressure and the viscous terms contribute to a change in the kinetic energy.

We point out that there are other methods to obtain a skew-symmetric form for (2.1.12), that result in the conservation of other quantities. An entropy preserving skew-symmetric form can be found in [233]. Furthermore, a fully quasi-skew-symmetric form for (2.1.12), where all quadratic fluxes are in skew-symmetric form, is shown to minimize aliasing errors [134].

2.1.4 Time integration

Following [213; 179] we can construct a fully discrete second order accurate scheme for (2.1.15) as

$$\begin{cases} \frac{1}{2} \sqrt{\rho^{\tau+1/2}} \frac{\sqrt{\rho^{\tau+1}} - \sqrt{\rho^\tau}}{\Delta t} + D U^{\tau+1/2} \rho^\tau = 0, \\ \sqrt{\rho^{\tau+1/2}} \frac{\sqrt{R^{\tau+1}} \mathbf{u}^{\tau+1} - \sqrt{R^\tau} \mathbf{u}^\tau}{\Delta t} + S_{\rho^\tau \mathbf{u}^\tau} \mathbf{u}_\alpha^{\tau+1/2} + D \mathbf{p}^\tau = D T^\tau, \\ \frac{1}{1-\gamma} \frac{\mathbf{p}^{\tau+1} - \mathbf{p}^\tau}{\Delta t} + \frac{\gamma}{\gamma-1} D U^\tau \mathbf{p}^\tau - U^\tau D \mathbf{p}^\tau = -U^\tau D T^\tau + D (U^\tau T^\tau - \varphi^\tau). \end{cases} \quad (2.1.20)$$

Here Δt is the time step, superscript τ denotes evaluating at $t = \tau \Delta t$, superscript $\tau + 1/2$ denotes the arithmetic average of a variable evaluated at $t = \tau \Delta t$ and $t = (\tau + 1) \Delta t$, the square root sign denotes element-wise application of square root and

$$\mathbf{u}_\alpha^{\tau+1/2} = \frac{\sqrt{R^{\tau+1}} \mathbf{u}^{\tau+1} + \sqrt{R^\tau} \mathbf{u}^\tau}{2 \sqrt{\rho^{\tau+1/2}}}.$$

As discussed in [213], this time discretization scheme preserves the symmetries expressed in (2.1.11), (2.1.17), (2.1.18), and (2.1.19). In the incompressible case, the method reduces to the implicit midpoint scheme.

2.2 Model reduction of fluid flow

A straightforward model reduction of (2.1.7) and (2.1.12) does not generally preserve the symmetries and the conservation laws presented in Section 2.1.1. In this Section, we discuss how to exploit the discrete skew-symmetric structure of (2.1.9) and (2.1.15) to recover conservation of mass, momentum, and energy at the level of the reduced system.

Let U_ρ , $U_{\rho \mathbf{u}}$, and $U_{\mathbf{u}_i}$ be the reduced bases for the snapshots of ρ , $R \mathbf{u}$, and \mathbf{u}_i , respectively, with reduced coefficients z_ρ , $z_{R \mathbf{u}}$, and $z_{\mathbf{u}_i}$. The subscript "i" is omitted for the one-dimensional case, and for an incompressible fluid, U_ρ and $U_{\rho \mathbf{u}}$ are not computed. For simplicity, we assume that all bases have the same size n . We seek to project $S_{\mathbf{u}}$, S_{ρ, ∂_t} , and $S_{\rho \mathbf{u}}$ onto the reduced space, such

that the projection preserves the skew-symmetric property. The projected operators, using a Galerkin projection, read

$$S_{\mathbf{u}}^r = U_{\mathbf{u}_i}^\top S_{\mathbf{u}} U_{\mathbf{u}_i}, \quad i = 1, \dots, s, \quad (2.2.1)$$

and

$$S_{\rho, \partial_t}^r = U_{\rho \mathbf{u}}^\top S_{\rho, \partial_t} U_{\mathbf{u}}, \quad S_{\rho \mathbf{u}}^r = U_{\rho \mathbf{u}}^\top S_{\rho \mathbf{u}} U_{\mathbf{u}}. \quad (2.2.2)$$

Note that S_{ρ, ∂_t}^r is not computed explicitly. It is clear that $S_{\mathbf{u}}^r$ is already in a skew-symmetric form. On the other hand, S_{ρ, ∂_t}^r and $S_{\rho \mathbf{u}}^r$ are not, in general, skew-adjoint and skew-symmetric, respectively. The preservation of the skew-symmetric structure can be ensured by requiring $U_{\rho \mathbf{u}} = U_{\mathbf{u}}$. We denote such a basis by $U_{\rho \mathbf{u}, \mathbf{u}}$. Using (2.2.1) and (2.2.2), a Galerkin projection of the momentum equation in (2.1.9) and the governing equations for a compressible fluid in (2.1.15) take the form

$$\frac{d}{dt} z_{\mathbf{u}_i} + S_{\mathbf{u}}^r z_{\mathbf{u}_i} + U_{\mathbf{u}_i}^\top D_i \mathbf{p} = \sum_{j=1}^s U_{\mathbf{u}_i}^\top D_j T_{ij} (U_{\mathbf{u}_i} z_{\mathbf{u}_i}), \quad i = 1, \dots, s \quad (2.2.3)$$

and

$$\begin{cases} \frac{d}{dt} z_{\rho} + \sum_{i=1}^n U_{\rho}^\top D U_i U_{\rho} z_{\rho} = 0, \\ S_{\rho, \partial_t}^r z_{\mathbf{u}} + S_{\rho \mathbf{u}}^r z_{\mathbf{u}} + U_{\rho \mathbf{u}, \mathbf{u}}^\top D U_{\mathbf{p}} z_{\mathbf{p}} = U_{\rho \mathbf{u}, \mathbf{u}}^\top D T, \\ \frac{1}{\gamma - 1} \frac{d}{dt} z_{\mathbf{p}} + \frac{\gamma}{\gamma - 1} U_{\mathbf{p}}^\top D U U_{\mathbf{p}} z_{\mathbf{p}} - U_{\mathbf{p}}^\top U D U_{\mathbf{p}} z_{\mathbf{p}} = -U_{\mathbf{p}}^\top U D T + U_{\mathbf{p}}^\top D (U T - \varphi), \end{cases} \quad (2.2.4)$$

respectively. Note that in (2.2.4), the dependency of T on $U_{\rho \mathbf{u}, \mathbf{u}}$ is not shown for abbreviation. In (2.2.3) and (2.2.4), D_i is always multiplied from the left with a basis matrix or a diagonal matrix. Therefore, the telescoping sum, discussed in Section 2.1.2, cannot be used to show the conservation of mass and momentum. However, POD preserves the linear properties of snapshots. An approximated variable, e.g., density, can be represented as a linear combination of some snapshots as $\rho \approx \rho_r = \sum_{i=1}^{N_S} c_i \rho_i$, for some snapshots ρ_i and some coefficients $c_i \in \mathbb{R}$, for $i = 1, \dots, N_S$. Conservation of the total mass, evaluated by ρ_r and denoted by $M_r(t)$ for the approximated solution, reads

$$\frac{d}{dt} M_r = \frac{d}{dt} \mathbb{1}_N^\top \rho_r = \sum_{i=1}^{N_S} c_i \left(\mathbb{1}_N^\top \frac{d}{dt} \rho_i \right) = - \sum_{i=1}^{N_S} c_i (\mathbb{1}_N^\top D R_i \mathbf{u}_i) = 0, \quad (2.2.5)$$

where we used that $\mathbb{1}_N^\top D = 0_N^\top$. Similarly, we recover conservation of the total momentum $P_r(t)$ for the reduced approximation

$$\begin{aligned} \frac{d}{dt} P_r &= \frac{d}{dt} (\rho_r^\top \mathbf{u}_r) = \frac{1}{2} \frac{d}{dt} (\rho_r^\top \mathbf{u}_r) + \frac{1}{2} \left(\rho_r^\top \frac{d}{dt} \mathbf{u}_r + \mathbf{u}_r^\top \frac{d}{dt} \rho_r \right) \\ &= \sum_{i,j=1}^{N_S} d_i c_j \left(\mathbf{u}_i^\top \frac{d}{dt} \rho_j + \left(\rho_j^\top \frac{d}{dt} \mathbf{u}_i + \mathbf{u}_i^\top \frac{d}{dt} \rho_j \right) \right) = 0. \end{aligned} \quad (2.2.6)$$

Here, $\mathbf{u}_r = \sum_{i=1}^{N_S} d_i \mathbf{u}_i$, for some snapshots \mathbf{u}_i and coefficients $d_i \in \mathbb{R}$. Denoting by $(R\mathbf{u})_r$ the reduced representation of $R\mathbf{u}$ in basis $U_{\rho\mathbf{u},\mathbf{u}}$, the evolution of kinetic energy is expressed as

$$\begin{aligned} \frac{d}{dt} \left(\frac{1}{2} \mathbf{u}_r^\top (R\mathbf{u})_r \right) &= \frac{d}{dt} \left(\frac{1}{2} z_{\mathbf{u}}^\top U_{\rho\mathbf{u},\mathbf{u}}^\top U_{\rho\mathbf{u},\mathbf{u}} R_r z_{\mathbf{u}} \right) = \frac{d}{dt} \left(\frac{1}{2} z_{\mathbf{u}}^\top R_r z_{\mathbf{u}} \right) \\ &= \frac{1}{2} \left(z_{\mathbf{u}}^\top \frac{d}{dt} z_{R\mathbf{u}} + z_{R\mathbf{u}}^\top \frac{d}{dt} z_{\mathbf{u}} \right) \\ &= \frac{1}{2} \left(z_{\mathbf{u}}^\top U_{\rho\mathbf{u},\mathbf{u}}^\top U_{\rho\mathbf{u},\mathbf{u}} \frac{d}{dt} z_{R\mathbf{u}} + z_{R\mathbf{u}}^\top \frac{d}{dt} (U_{\rho,\mathbf{u},\mathbf{u}}^\top U_{\rho\mathbf{u},\mathbf{u}} z_{\mathbf{u}}) \right) \\ &= z_{\mathbf{u}}^\top S_{\rho,\partial_t}^r z_{\mathbf{u}} = z_{\mathbf{u}}^\top U_{\rho\mathbf{u},\mathbf{u}} D U_{\mathbf{p}} \mathbf{P}^r + z_{\mathbf{u}}^\top U_{\rho\mathbf{u},\mathbf{u}}^\top D T. \end{aligned} \quad (2.2.7)$$

In the last line, skew-symmetry of $S_{\rho\mathbf{u}}^r$ is used. Note, that only the reduced pressure and the viscous term contribute to the evolution of the kinetic energy. Furthermore, the quantity $K_r(t) = \frac{1}{2} z_{\mathbf{u}}^\top z_{R\mathbf{u}}$ is the kinetic energy associated with the reduced system (2.2.4), approximating the kinetic energy of the high-fidelity system (2.1.15), and is a quadratic form with respect to the reduced variables. Conservation of kinetic energy for (2.2.3) follows similarly. It is straightforward to check that

$$\frac{d}{dt} \left(\frac{1}{\gamma-1} \mathbb{1}_N^\top \mathbf{p}_r + \frac{1}{2} \mathbf{u}_r^\top (R\mathbf{u})_r \right) = 0, \quad (2.2.8)$$

i.e., the total energy is conserved. We immediately recognize that $z_{\mathbf{p}}/\gamma - 1$ is the internal energy of the reduced system. However, the total internal energy of (2.2.4) is a weighted sum, $b^\top z_{\mathbf{p}}/\gamma - 1$, with $b = U_{\mathbf{p}}^\top \mathbb{1}$ which is an approximation of the total internal energy in (2.1.15). From (2.2.5), (2.2.6), (2.2.7), and (2.2.8) we conclude the following proposition.

Proposition 2.2.1. *The loss in the mass, momentum, and energy associated with the model reduction in (2.2.4) is constant in time, and therefore, bounded.*

2.2.1 Assembling nonlinear terms and time integration

Nonlinear terms that appear in (2.2.3) and (2.2.4) are of quadratic nature. These terms can be evaluated exactly using a set of precomputed matrices as described in Section 1.5. As an example, consider

$$S_{\mathbf{u}}^r = U_{\mathbf{u}}^\top (D U_r + U_r D) U_{\mathbf{u}}^\top.$$

where U_r is the diagonal matrix having the RB approximation u_r of the velocity on its diagonal. We write U_r as a linear combination of matrices as $U_r = \sum_{j=1}^n \mathbf{u}_j^r U_j$, where \mathbf{u}_j^r is the j -th component of \mathbf{u}^r , and U_j contains the j -th column of $U_{\mathbf{u}}$ on its diagonal. It follows

$$S_{\mathbf{u}}^r = \sum_{j=1}^k \mathbf{u}_j^r (U_{\mathbf{u}}^\top (D U_j + U_j D) U_{\mathbf{u}}^\top),$$

The matrices $(U_{\mathbf{u}}^\top (D U_j + U_j D) U_{\mathbf{u}}^\top)$ can be precomputed prior to the time integration of the reduced system. However, the form of the fully discrete system in (2.1.20) introduces quartic terms and support variables. In principle, the same method can be applied to assemble the nonlinear terms. However, the number of precomputed matrices grows exponentially with the order of the nonlinear term, as noted in Section 1.5.

To accelerate the assembly of the nonlinear terms we may approximately evaluate them using the DEIM. Since this is an approximate evaluation, we do not expect conservation of invariants,

as discussed in Section 2.2. However, the numerical experiments in Section 2.3 suggest that an accurate approximation of the invariants is achieved when an accurate DEIM approximation is used for evaluating nonlinear terms. A possible extension of the approach, which exploits a block division of skew-symmetric operators for certain physical problems, is proposed in [269] in the context of noncanonical Hamiltonian problems.

To integrate (2.2.4) in time, the fully discrete system (2.1.20) is modified prior to model reduction, by dividing the mass and momentum equation by $\sqrt{\rho^{\tau+1}}$. Note that since the new form is identical to (2.1.20), it does not affect the conserved quantities. Subsequently, a basis for $\sqrt{\rho}$, denoted by $U_{\sqrt{\rho}}$, is constructed. The nonlinear terms are evaluated exactly using the quadratic expansion or approximated using DEIM.

2.3 Numerical experiments

2.3.1 Vortex merging

Consider the incompressible 2D Euler equation (2.1.7) on a square domain $\Omega = [0, 2\pi]^2$, with periodic boundary conditions. Spatial derivatives are discretized using a Fourier spectral method. To capture the fine details characterizing the solution, 256 modes per dimension are used, for a total of $N = 65536$ degrees of freedom per variable. We consider the evolution of three vortices, with the initial structure given by

$$\omega = \omega_0 + \sum_{i=1}^3 \alpha_i \exp\left(-\frac{(x - x_i)^2 + (y - y_i)^2}{\beta^2}\right).$$

Here, $\omega = \nabla \times u$ is the vorticity of the velocity flow, (x, y) represents the spatial coordinates, (x_i, y_i) is the center of the i -th vortex, α_i its maximum amplitude, and β controls the effective radius of the vortex. In this numerical experiment, the center of three vortices are

$$(x_1, y_1) = (0.75\pi, \pi), \quad (x_2, y_2) = (1.25\pi, \pi), \quad (x_3, y_3) = (1.25\pi, 1.5\pi).$$

Two of the vortices have a positive spin with $\alpha_1 = \alpha_2 = \pi$ and the third rotates in the opposite direction with $\alpha_3 = -0.5\pi$. The effective radius of all the vortices is set to $\beta = 1/\pi$. This arrangement of vortices has been traditionally considered as initial condition to study the process of vortex merging due to fast-moving dipoles with the same spin facing the third vortex of opposite spin [77]. The merging process transfers the vorticity from the initial configuration into long, narrow, and spiral-shaped strips of intense vorticity [142]. Because of aliasing, the formation of such thin vorticity filaments in the fluid may pose numerical challenges.

In the context of MOR, conservation of energy and stability are crucial to capture fine structures. With the absence of natural dissipation, straightforward application of MOR techniques for the Euler equations produces unstable ROMs.

To define the initial conditions in terms of the velocity components u and the pressure p , we define the stream-function Ψ as the solution to the equation

$$-\Delta \Psi = \omega. \tag{2.3.1}$$

The initial velocity is then given by $\nabla \times \Psi$. To solve the stream-function problem (2.3.1), we require $\int_{\Omega} \Psi \, dx = 0$. In our setting, it is easily verified that this requirement implies $\omega_0 = 0.038$.

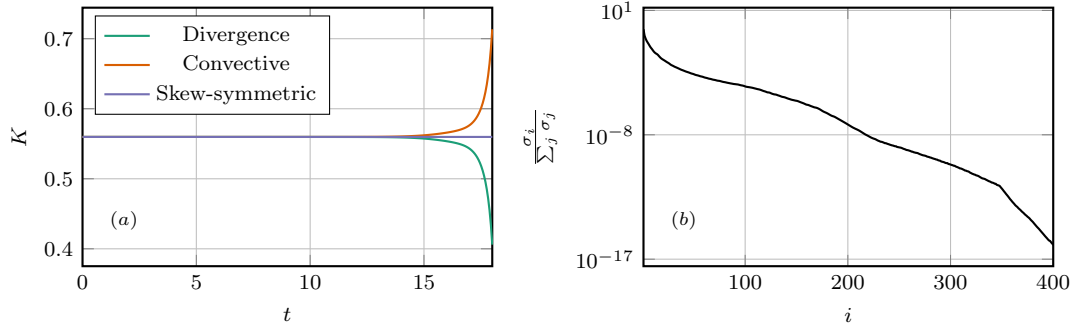


Figure 2.1: VM: (a) Evolution of the kinetic energy K for the advective, divergence and skew-symmetric formulations of the incompressible Euler equation in the vortex merging setting. (b) Singular values of the snapshot matrix of the the solution to the FOM.

The pressure is recovered by solving the related Poisson pressure equation

$$\Delta p = -\nabla \cdot S_{\mathbf{u}}(\mathbf{u}),$$

obtained by applying the divergence operator to (2.1.7) and using the incompressibility condition. We use the numerical integrator defined in (2.1.20), which in this case reduces to the implicit midpoint scheme. The merging phenomenon is simulated for a total of $T = 18$ seconds using a temporal step $\Delta t = 0.004$.

Figure 2.1(a) illustrates the evolution of the kinetic energy K for the advective, the divergence, and the skew-symmetric forms of the high-fidelity system. It is observed that only the skew-symmetric form preserves the kinetic energy, supporting the findings of Section 2.1.1. A total of $N_t = 5000$ temporal snapshots is used to construct a reduced basis, following the process discussed in Section 1.3.1. The decay of the singular values, used as an indicator of the reducibility of the problem, is presented in Figure 2.1(b). The first 35 POD modes contain over 99% of the energy of the high fidelity solution. This suggests that an accurate reduced system can be constructed using fewer basis vectors. Smaller reduced bases are also considered to illustrate the effectiveness and stability of the method.

For qualitative analysis, in Figure 2.2, four solutions at different times are shown for the high-fidelity and the reduced systems with $n = 17$ and $n = 35$ modes. The overall dynamics of the problem, and in particular the formation and development of vorticity filaments, are correctly represented, and even with a moderate number of basis vectors. Although small details are not captured by the reduced system of dimension $n = 17$, the positions and the spreading of the vortices are comparable. Figure 2.3(a) shows the norm of the error between the velocity components of the high-fidelity solution and the reduced solution, defined as

$$\varepsilon_u(t) = \|u(t) - u_r(t)\|_2. \quad (2.3.2)$$

The error decreases consistently as the number n of basis vectors increases. Furthermore, the accuracy is maintained over the entire simulated interval. The conservation of the reduced kinetic energy K_r is presented in Figure 2.3(b). The kinetic energy remains constant even for a small number of basis vectors, where the solution is not well approximated. This is central for the robustness of the reduced system during long time-integration.

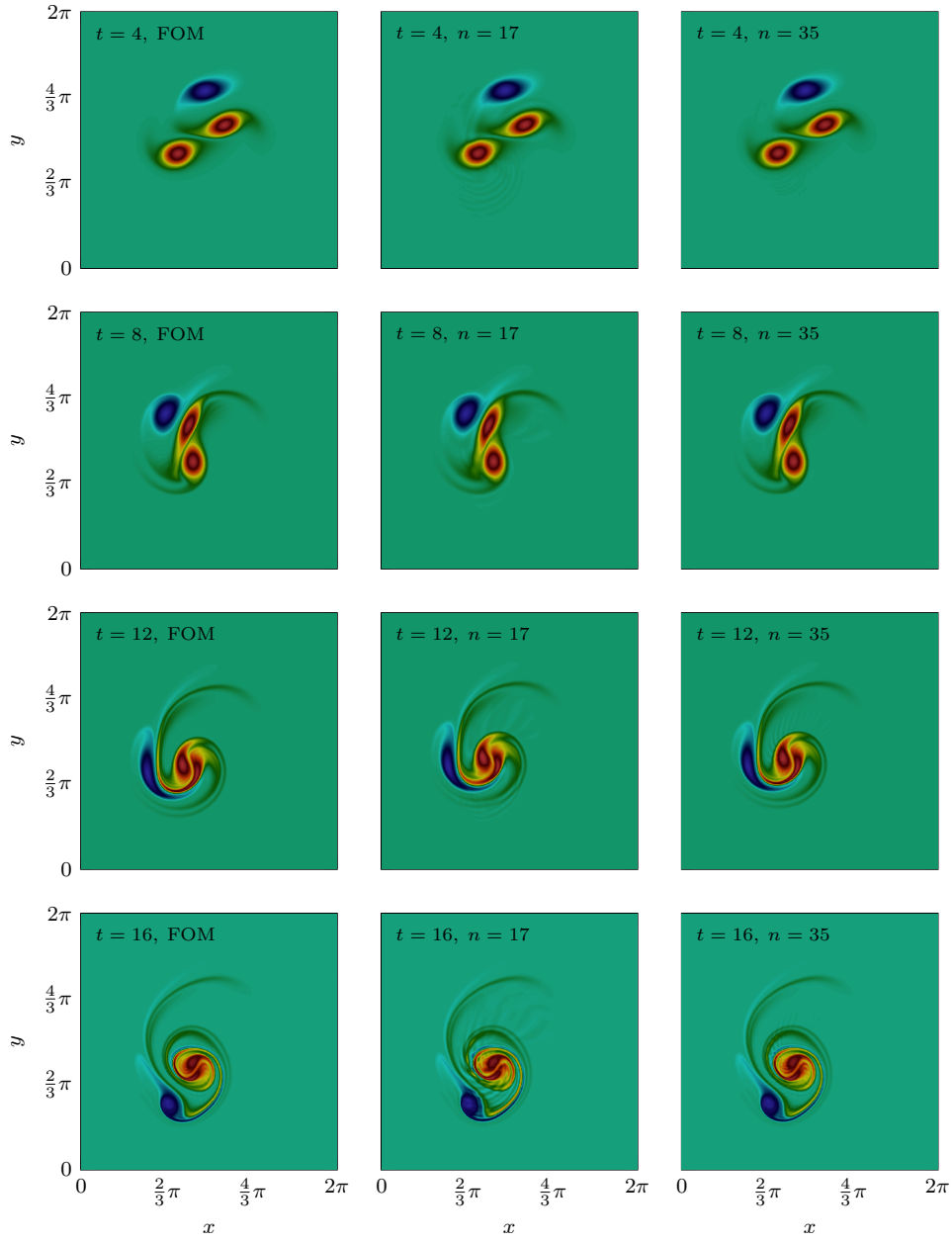


Figure 2.2: VM: Vorticity field at different times obtained from the FOM and the ROM with $n = 17$ and $n = 35$. Starting from three separated vortices, filamentous structures develop as a result of the interactions between the vortices as early as $t = 8$.

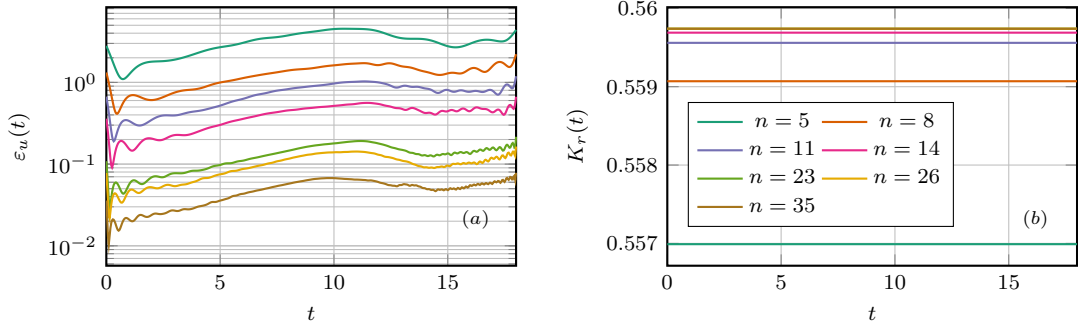


Figure 2.3: VM: (a) Evolution of the velocity absolute error, as defined in (2.3.2), for different values of the basis dimension n . (b) Comparison of the time evolution of the reduced kinetic energy K_r for different values of n .

2.3.2 2D Kelvin-Helmholtz instability

Consider the 2-dimensional compressible Euler equation (2.1.12) in the periodic square box $\Omega = [0, 1]^2$. Unlike the incompressible example in Section 2.3.1, a centered finite difference scheme of fourth order is used to discretize (2.1.12). The physical domain is discretized onto a grid of 256×256 nodes, giving $N = 65536$.

The initial conditions are given by

$$\begin{cases} \rho = \begin{cases} 2, & \text{if } 0.25 < y < 0.75, \\ 1, & \text{otherwise,} \end{cases} \\ \mathbf{u}_1 = a \sin(4\pi y) \left(\exp\left(-\frac{(y-0.25)^2}{2\sigma^2}\right) + \exp\left(-\frac{(y-0.75)^2}{2\sigma^2}\right) \right), \\ \mathbf{u}_2 = \begin{cases} 0.5, & \text{if } 0.25 < y < 0.75, \\ -0.5, & \text{otherwise,} \end{cases} \\ \mathbf{p} = 2.5, \end{cases}$$

where $a = 0.1$ and $\sigma = 5\sqrt{2} \cdot 10^{-3}$. This initial configuration depicts contacting streams of fluid with different densities and velocities. Thin structures and vortices emerge at the interface between the streams for specific choices of parameters describing the initial jets. Such instability is referred to as the Kelvin-Helmholtz instability [54].

As centered schemes are often dissipation free, resolving the discontinuous initial data requires artificial viscosity. Therefore, the method discussed in [270] is used as an artificial viscosity in the high-fidelity model. However, at the level of the reduced system, this is replaced with a low pass filter on the expansion coefficients of POD basis vectors.

The fully discrete skew-symmetric form (2.1.20) is used as time marching scheme with the time step $\Delta t = 5 \cdot 10^{-4}$ over a time frame of $T = 1$. The same time step is adopted for the discretization of the ROM.

Figure 2.4 illustrates that the ROM's accuracy consistently improves as a higher number n of POD modes is considered. Furthermore, the same Figure shows that reducing the skew-symmetric form allows to control the accuracy over the entire integration interval \mathcal{T} . It is observed in Figure 2.5 that all the features of the flow are correctly represented by the solution to the reduced system, even with a small number of basis vectors. Approximations of the mass $M(t)$, the momentum

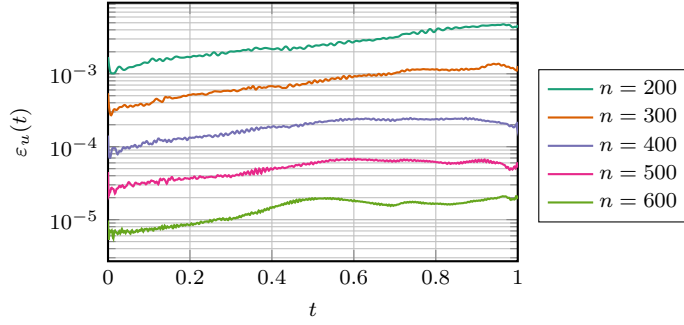


Figure 2.4: KH: Evolution in time of the velocity error (2.3.2) between the high-fidelity solution and the reduced solution of the Kelvin-Helmholtz numerical experiment for different basis dimensions n .

$P(t)$, and kinetic energy $K(t)$ are shown in Figure 2.6. The method's accuracy in approximating these invariants improves as the size of the basis is increased. Furthermore, Figure 2.6(c) shows how the reduced system's kinetic energy mimics the high-fidelity system's kinetic energy. This helps to ensure the correct evolution of kinetic energy, and thus, the internal energy.

2.3.3 1D Shock problem

This section studies the 1-dimensional compressible Euler problem (2.1.12) with a steady-state discontinuous solution without viscous terms. This numerical experiment prepares the ground for Chapter 4, where we consider efficient techniques for the reduction of solutions showing sharp propagating fronts. Here we assess if preserving the skew-symmetric form of (2.1.12) in the reduction process guarantees stability. Consider periodic boundary conditions on the domain $\Omega = [0, 1]$ with the initial condition

$$\begin{cases} \rho = 0.5 + 0.2 \cos(2\pi x), \\ \mathbf{u} = 1.5, \\ \mathbf{p} = 0.5 + 0.2 \sin(2\pi x). \end{cases}$$

The domain is discretized using $N = 2000$ nodes and a centered finite differences scheme is used to assemble the discrete Euler equation in skew-symmetric form, as discussed in Section 2.1.3. The full discrete skew-symmetric form (2.1.20) is used for time integration over the time interval $[0, 0.3]$. To resolve the discontinuous solution we use an artificial viscosity term with $\tau = \mu \frac{\partial u}{\partial x}$, where $\mu = 0.5 \cdot 10^{-4}$.

Figure 2.7 shows the evolution of conserved quantities for the high-fidelity and reduced system. Here, the high-fidelity model is also considered in the divergence and advective form in addition to the skew-symmetric form. It is clear that when the reduced system is not in the skew-symmetric form, it violates the conservation of mass, momentum, and energy. Even while the high-fidelity systems in divergence and advective forms are stable, the constructed reduced system is unstable, independently of the number of basis vectors. On the other hand, the skew-symmetric form yields a stable and conservative reduced system. Note that the energy loss associated with the skew-symmetric form, illustrated in Figure 2.7, is due to the application of an artificial viscosity.

Figure 2.8 shows the evolution of the total error in time. It is observed that the formation of a discontinuity, at $t = 0.16$, affects the method's accuracy. This degradation of the approximation

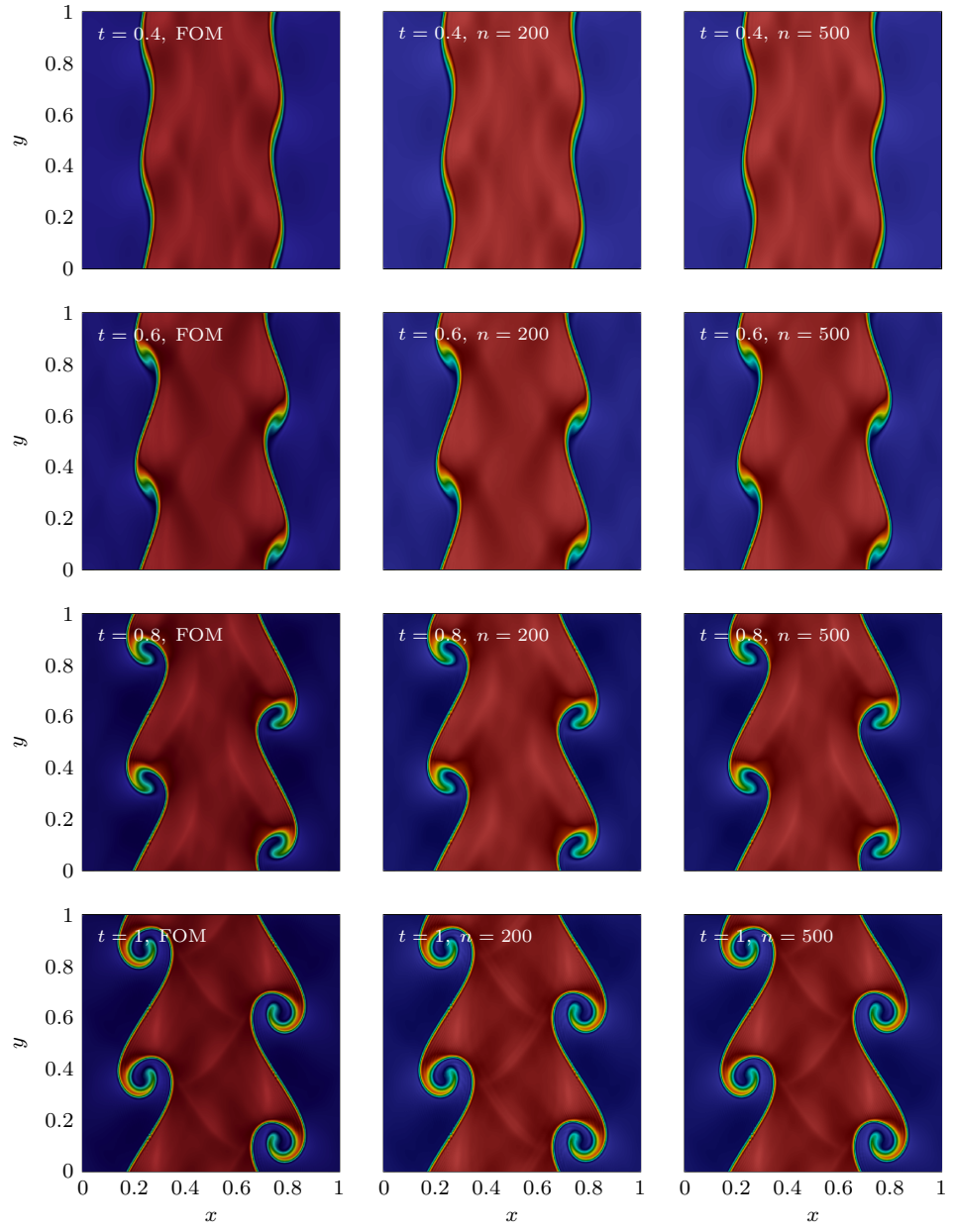


Figure 2.5: KH: Density field at different times obtained from the full-order model and the reduced model for $n = 200$ and $n = 500$. The instability developing from the velocity discontinuity across the interface is properly captured by the ROM solution.

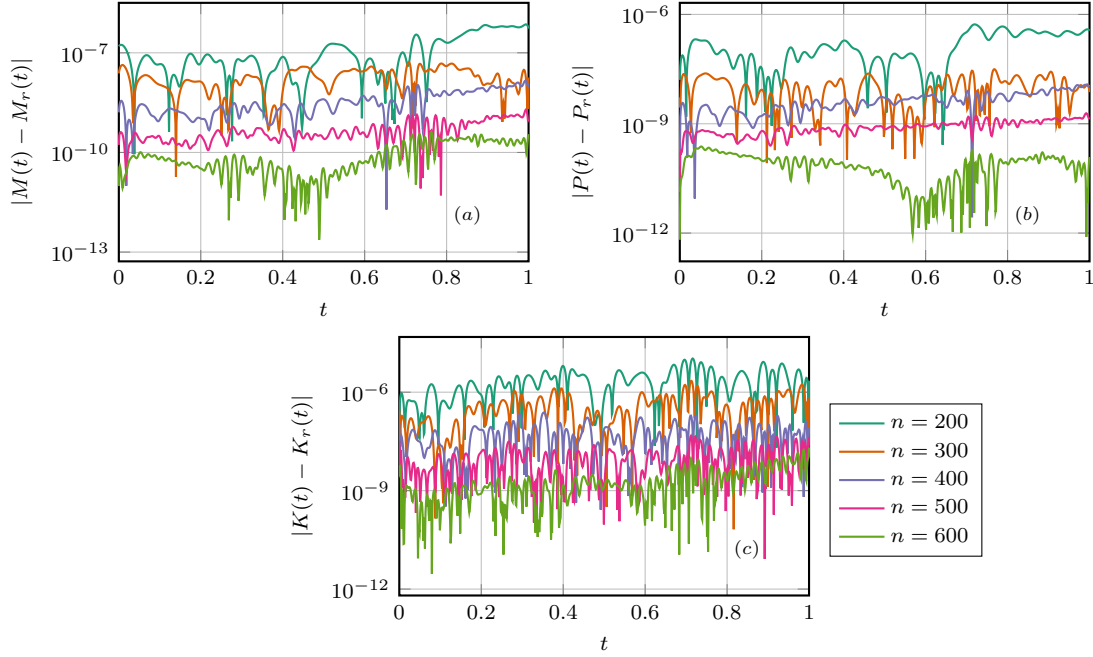


Figure 2.6: KH: Evolution of the relative errors in the conservation of the mass (a), of the momentum (b), and of the total energy (c) for the solution of the ROM for different values of n .

quality is expected because relatively few POD modes are not enough to resolve sharp gradients. However, the method remains robust and stable during the period time of integration. In Figure 2.9 we compare the numerical artifacts of different formulations of the Euler equation. The advective formulation is not shown since it does not yield a stable reduced system. It is observed that the reduced system based on the skew-symmetric formulation accurately represents the overall behavior of the high-fidelity solution. On the other hand, a Gibbs-type error [250] appears near sharp gradients for the reduced system based on the divergence form of the Euler equation. The well-representation of the skew-symmetric form is due to the low aliasing error property of the form, as mentioned in Section 2.1.3. As discussed in Section 2.2.1, the DEIM approximation needed for an efficient evaluation of the nonlinear components of (2.1.12) can affect the conservation properties of the skew-symmetric form. Figure 2.10 shows the decay of the singular values of the nonlinear snapshots. The decay of these snapshots is significantly slower than the temporal snapshots of (2.1.12). This indicates that to maintain the accuracy of the reduced system, the DEIM basis should be chosen richer than the POD basis. Figure 2.11(a) and 2.11(b) present the error and the conservation of total energy when the DEIM is used to approximate the nonlinear term. The energy conservation is recovered once DEIM approximates the nonlinear terms with enough accuracy. In this numerical experiment, evaluation of the nonlinear terms in (2.1.12) using DEIM is four times faster than the high-fidelity evaluation.

2.3.4 Continuous variable resonance combustor

CVRC is a model rocket combustor designed and operated at Purdue University (Indiana, U.S.) to investigate combustion instabilities [271]. This setup is called the Continuously Variable Resonance Combustor (CVRC) because the length of the oxidizer injector can be varied continuously, allowing

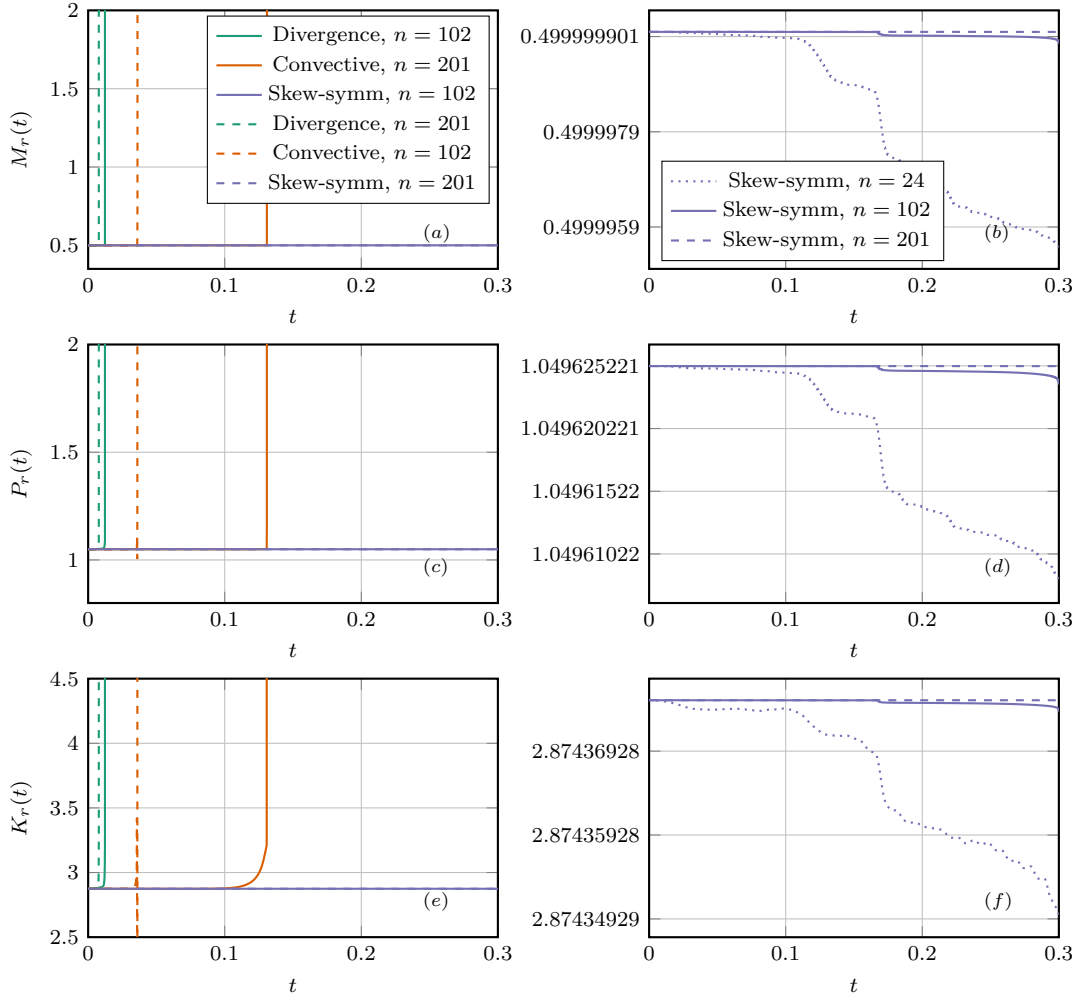


Figure 2.7: CE: Evolution in time of the mass (a), momentum (c), and energy (e) of the solution to the ROM in case of divergent, advective and skew-symmetric formulations for $n = 102$ and $n = 204$. In (b), (d), and (e) the evolutions of the same quantities are reported only in the case of skew-symmetric formulation for $n = 24$, $n = 102$ and $n = 204$.

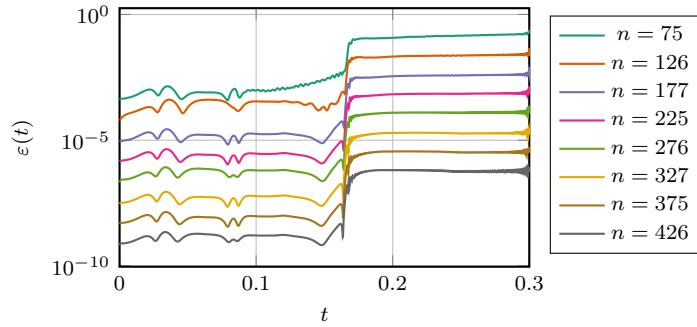


Figure 2.8: CE: Evolution in time of the error between the high-fidelity solution and the reduced solution of the compressible Euler numerical experiment for different basis dimensions n .

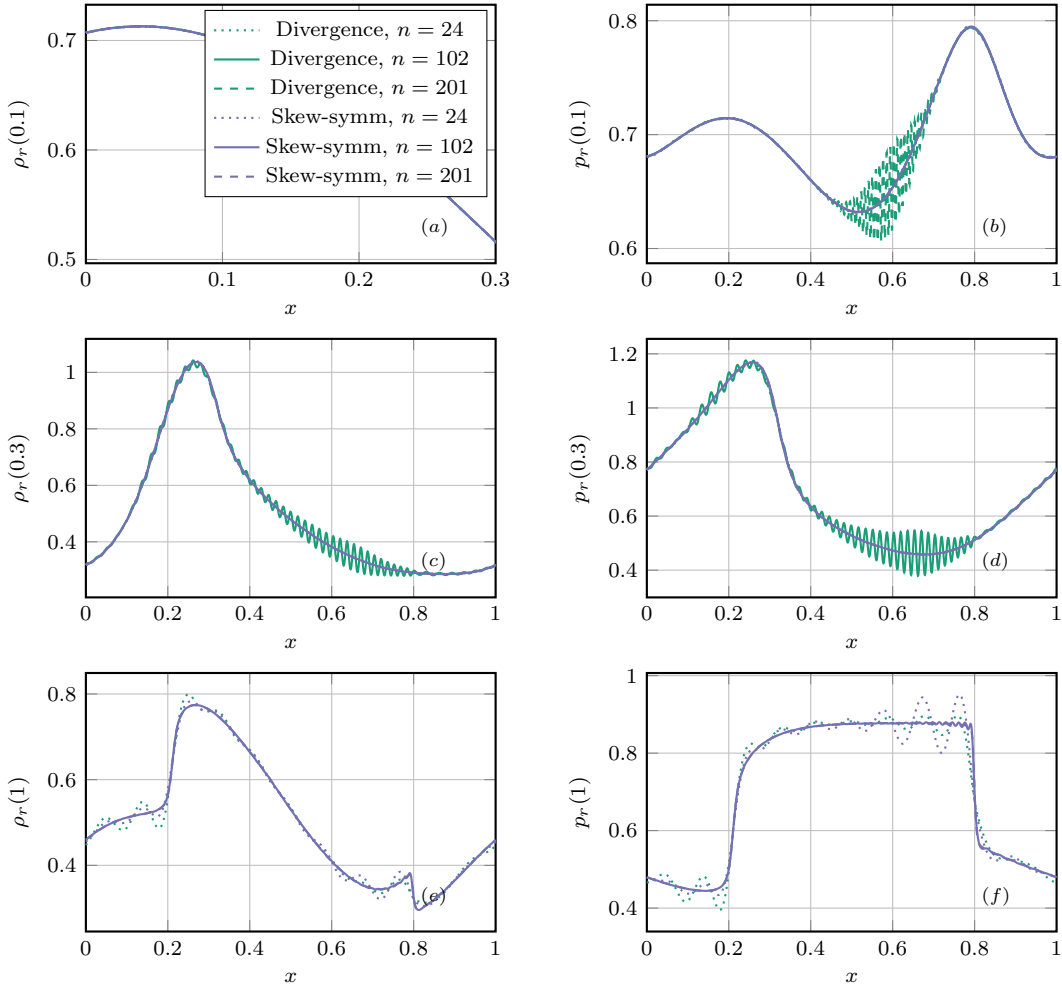


Figure 2.9: CE: Comparison between the reduced solutions of the divergent and skew-symmetric formulations of the problems in terms of density and pressure at $t = 0.1$ ((a), (b)), $t = 0.3$ ((c), (d)), and $t = 1$ ((e), (f)). Results for the advective formulation are not showed here because the related reduced solutions are unstable after a few time steps.

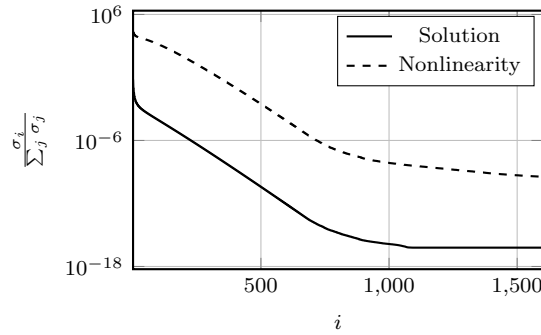


Figure 2.10: CE: Singular values of the snapshot matrix S and S_{DEIM} .

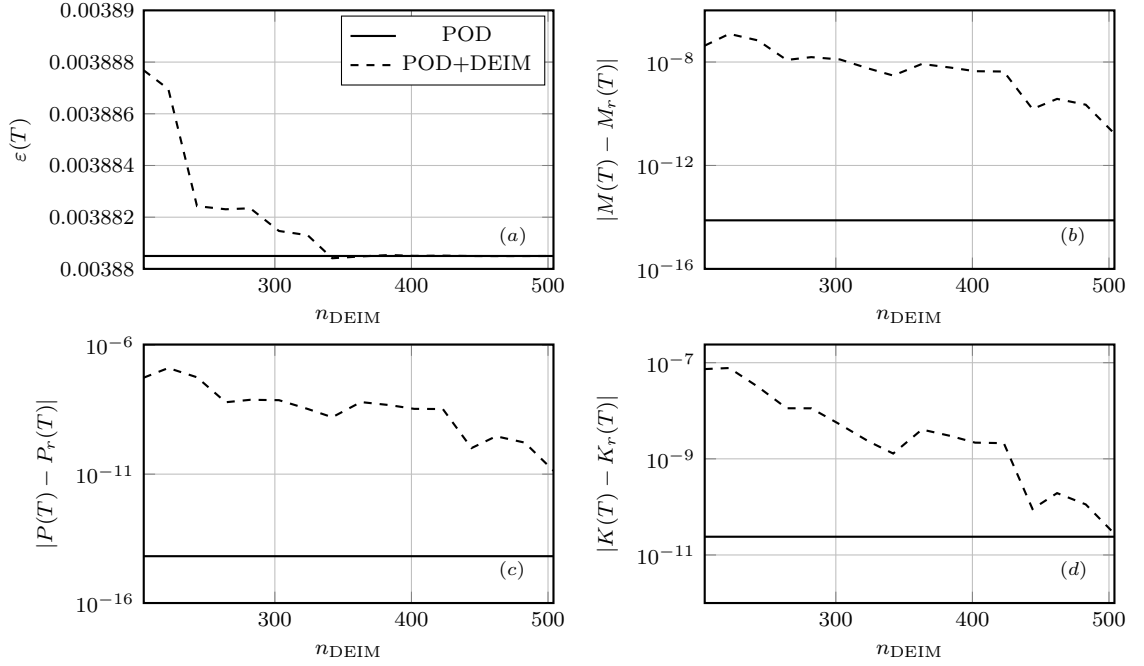


Figure 2.11: CE: Comparison between standard POD and POD with DEIM treatment of the nonlinear term in terms of the total error (a), mass error (b), momentum error (c), and kinetic energy error (d).

Table 2.1: Geometry parameters of the quasi-1D CVRC with an oxidizer post length $L_{\text{op}} = 14$ cm.

Section	Oxidizer post		Chamber	Nozzle	
	injector	back-step		converging part	diverging part
Length (cm)	12.99	1.01	38.1	1.27	3.4
Radius (cm)	1.02	1.02~2.25	2.25	2.25~1.04	1.04~1.95

for a detailed investigation of the coupling between acoustics and combustion in the chamber. However, the 2D/3D high-fidelity simulations of CVRC are expensive. Thus, a quasi-1D model has been proposed in [235] and further developed in [93] to get a fast analysis tool. The CVRC consists of three parts: the oxidizer post, the combustion chamber and the exit nozzle, as shown in Figure 2.12. The oxidizer is injected from the left end of the oxidizer post and meets the fuel, injected through an annular ring around the oxidizer, at the back-step. The combustion products flow through the chamber and exit the system from the nozzle. Both the injector and the nozzle are operated at choked condition during the experiment. The length of the oxidizer post L_{op} of the CRVC can be varied continuously, leading to different dynamics. Here, we focus on the case with $L_{\text{op}} = 14.0$ cm, in which the combustion is unstable. The geometry parameters of the quasi-1D CVRC with an oxidizer post length $L_{\text{op}} = 14.0$ cm are shown in Table 2.1. The back-step and the converging part of the nozzle are sinusoidally contoured to avoid a discontinuity of the radius that will invalidate the quasi-1D governing equations studied here.

The fuel is pure gaseous methane. The oxidizer is a mixture of 42% oxygen and 58% water (per unit mass) injected in the oxidizer post at temperature $T_{\text{ox}} = 1030$ K so that both water and oxygen are in the gaseous phase. The operating conditions are listed in Table 2.2.

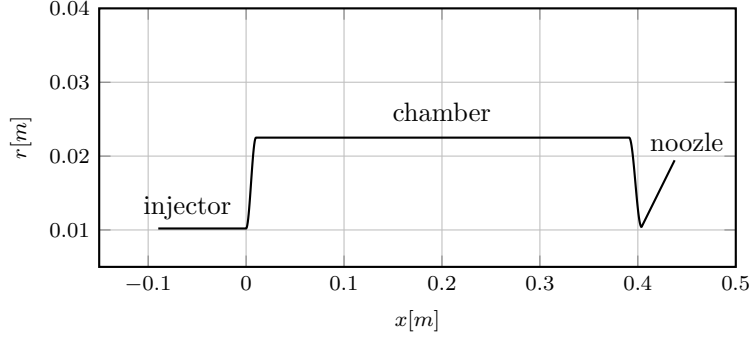
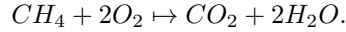


Figure 2.12: CVRC: Geometry of quasi-1D CVRC model.

Table 2.2: CVRC operating conditions.

Parameter	Unit	Value
Fuel mass flow rate, \dot{m}_f	kg/s	0.027
Fuel temperature, T_f	K	300
Oxidizer mass flow rate, \dot{m}_{ox}	kg/s	0.32
Oxidizer temperature, T_{ox}	K	1030
O_2 mass fraction in oxidizer, Y_{O_2}	-	42.4%
H_2O mass fraction in oxidizer, Y_{H_2O}	-	57.6%
Mean chamber pressure	MPa	1.34
Equivalence ratio, E_r	-	0.8

For the combustion, we consider the one-step reaction model



We assume that the fuel reacts instantaneously to form products, allowing us to neglect intermediate species and finite reaction rates. As the equivalence ratio is less than one, some oxidizer is left after the combustion. Therefore, only two species need to be considered: oxidizer and combustion products.

The governing equations that describe the conservation of mass, momentum, and energy of the quasi-1D CVRC flow, are the quasi-1D unsteady Euler equations for multiple species, expressed in conservative form as

$$\frac{\partial}{\partial t} \nu + \frac{\partial}{\partial x} F_\nu = s_A + s_f + s_q. \quad (2.3.3)$$

The conserved variable vector ν and the convective flux vector F_ν are

$$\nu = \begin{pmatrix} \rho A \\ \rho u A \\ \rho E A \\ \rho Y_{ox} A \end{pmatrix}, \quad F = \begin{pmatrix} \rho u A \\ (\rho u^2 + p) A \\ (\rho E + p) u A \\ \rho u Y_{ox} A \end{pmatrix}, \quad (2.3.4)$$

where ρ is the density, u is the velocity, p is the pressure, E is the total energy, Y_{ox} is the mass fraction of oxidizer, and $A = A(x)$ is the cross sectional area of the duct. The pressure p can be

computed using the conserved variables as

$$E = \frac{p}{\rho(\gamma - 1)} + \frac{u^2}{2} - C_p T_{ref},$$

where T_{ref} is the reference temperature and is set at 298.15 K. The temperature T is recovered from the equation of state $p = \rho RT$. The gas properties C_p , R and γ are computed as $C_p = \sum C_p^i Y_i$, $R = \sum R_i Y_i$ and $\gamma = C_p / (C_p - R)$, respectively.

The source terms are

$$s_A = \begin{pmatrix} 0 \\ p \frac{dA}{dx} \\ 0 \\ 0 \end{pmatrix}, \quad s_f = \begin{pmatrix} \dot{\omega}_f \\ \dot{\omega}_f u \\ \dot{\omega}_f (h_0^f + \nabla h_0^{rel}) \\ \dot{\omega}_{ox} \end{pmatrix}, \quad s_q = \begin{pmatrix} 0 \\ 0 \\ q' \\ 0 \end{pmatrix}, \quad (2.3.5)$$

where $\dot{\omega}_f$ is the depletion rate of the fuel, $\dot{\omega}_{ox}$ is the depletion rate of the oxidizer, h_0^f is the total enthalpy of the fuel, ∇h_0^{rel} is the heat of reaction per unit mass of fuel and q' is the unsteady heat release term. The quantity s_A accounts for area variations, s_f and s_q are related to the combustion with the combustion. The quantity s_f represents the addition of the fuel and its combustion with the oxidizer, which in turn results in the creation of the combustion products. The depletion rate of the fuel is

$$\dot{\omega}_f = \frac{k_f \dot{m}_f Y_{ox} (1 + \sin(\xi))}{l_f - l_s},$$

where

$$\xi = -\frac{\pi}{2} + 2\pi \frac{x - l_s}{l_f - l_s}, \quad \forall l_s < x < l_f.$$

The setting of the fuel injection restricts the combustion to the region $l_s < x < l_f$. The reaction constant k_f is selected to ensure that the fuel is consumed within the specified combustion zone. The depletion rate of the oxidizer is computed by

$$\dot{\omega}_{ox} = C_{o/f} \dot{\omega}_f,$$

where $C_{o/f}$ is the oxidizer-to-fuel-ratio.

The unsteady heat release term q' also called the combustion response function, models the coupling between acoustics and combustion. Here, we use the combustion response function designed by Frezzotti et al. [93], which is a function of the velocity, sampled at specific abscissa \hat{x} that is almost coincident with the antinode of the first longitudinal modal shape with a time lag t_0 , i.e.,

$$q'(x, t) = \alpha g(x) A(x) [u(\hat{x}, t - t_0) - \bar{u}(\hat{x})]. \quad (2.3.6)$$

Here \bar{u} is the time averaged velocity, estimated with the steady-state quasi-1D model assuming $q' = 0$, and $g(x)$ is a Gaussian distribution

$$g(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right),$$

where μ is the mean and σ is the standard deviation. The amount of heat released due to velocity oscillations is controlled by the parameter α , in (2.3.6).

The boundary conditions for the quasi-1D CVRC flow include the fixed mass flow rate and the stagnation temperature at the head-end of the oxidizer injector, and the supersonic outflow at the exit of the nozzle.

Prior to the unsteady simulation, the quasi-1D CVRC needs to be excited, which is achieved by adding a perturbation to the steady-state solution. The perturbation is added by forcing the mass flow rate with a multi-sine signal

$$\dot{m}_{ox}(t) = \dot{m}_{ox,0} \left[1 + \delta \sum_{k=1}^K \sin(2\pi k \Delta f t) \right],$$

where \dot{m}_{ox} is the oxidizer mass flow rate in Table 2.2, Δf is the frequency resolution and K is the number of frequencies. In this work, $\Delta f = 50$ Hz and $K = 140$, resulting in a minimal frequency of 50 Hz and a maximal frequency of 7000 Hz. δ is required to be small to control the amplitude of the perturbation and is set as 0.1%.

The procedure of the unsteady simulation of the quasi-1D CVRC flow includes three steps:

- Compute the steady-state solution by setting $\dot{m}_{ox} = \dot{m}_{ox,0}$ and $q' = 0$.
- Excite the system by adding a perturbation to the oxidizer mass flow rate according to (2.3.6) and setting $q' = 0$.
- Perform the unsteady simulation by turning on the combustion response function q' in (2.3.5) and turning off the oxidizer mass flow rate perturbation by setting $\dot{m}_{ox} = \dot{m}_{ox,0}$.

Introduction of an artificial viscosity is essential for a robust and long time-integration of (2.3.3). Common discretization schemes for (2.3.3) are often dissipative, e.g., the Lax-Friedrich scheme used in [259]. Since the skew-symmetric discretization is non-dissipative, we modify (2.3.3) as

$$\frac{\partial}{\partial t} \nu + \frac{\partial}{\partial x} F_\nu = s_A + s_f + s_q + d, \quad d = (0, \frac{\partial}{\partial x} \tau, 0, 0)^\top, \quad (2.3.7)$$

with $\tau = \mu \frac{\partial u_A}{\partial x}$ and $\mu = 6 \cdot 10^{-5}$. This type of artificial viscosity is chosen for its simplicity. This, however, can be replaced with a more moderate and sophisticated method.

Note that the right hand side in (2.3.7) suggests that, in general, mass, momentum, and energy are not conserved. Furthermore, the complex coupling of the variables in (2.3.3) and the non-constant adiabatic gas index prohibits the application of complex and implicit time integration schemes. Therefore, a quasi-skew-symmetric form, introduced in (2.1.6), is used for (2.3.3). It is straightforward to check [233], for $t, s \in \mathbb{R}^N$

$$\frac{1}{2} \delta_x(st)_j + \frac{1}{2} s_j \delta_x(t)_j + \frac{1}{2} t_j \delta_x(s)_j = \frac{1}{4} \delta_x^+(s_j + s_{j-1})(t_j + t_{j-1}). \quad (2.3.8)$$

where $\delta_x(\nu)_j = (\nu_{j+1} - \nu_{j-1})/\Delta x$ is centered finite difference approximation of the space derivative and $\delta_x^+(\nu_j) = (\nu_{j+1} - \nu_j)/\Delta x$ for some $\nu \in \mathbb{R}^N$. Therefore,

$$F_{i+1/2}^\Delta(s_j t_j, s_{j+1} t_{j+1}) = (s_j + s_{j-1})(t_j + t_{j-1}), \quad (2.3.9)$$

can be interpreted as an approximation of a quadratic flux function at the boundary of two adjacent finite volume cells. A better approximation of the flux in (2.3.9) corresponds to a higher order skew-symmetric for a quadratic variable st in (2.3.8). We discretize the real line into N

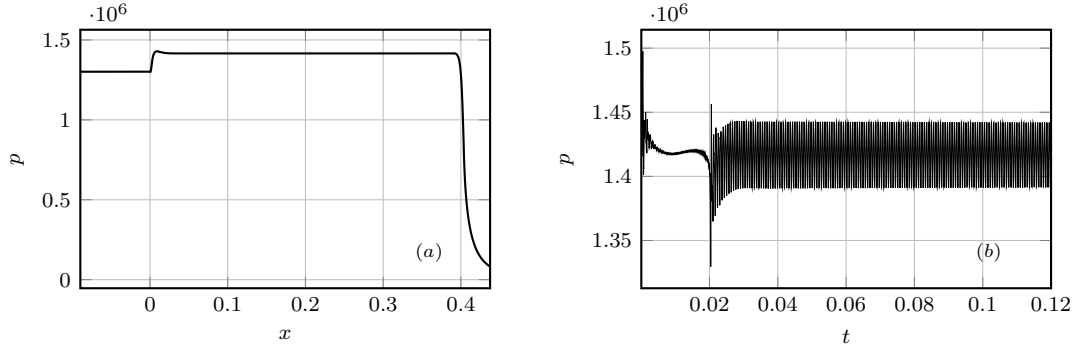


Figure 2.13: Pressure profile of the steady state (a) and oscillatory mode of pressure located at $x = 0.36$ for the unsteady flow for the combustor model solution (b).

uniform cells of size Δx . A quasi-skew-symmetric form for (2.3.7) now takes the form

$$\begin{aligned} \frac{d}{dt} q_j^i + \delta^+ F_{i+1/2}^\Delta(q_j^i r_j^i, q_{j+1}^i r_{j+1}^i) - \delta^+ F_d^\Delta(d_j^i, d_{j+1}^i) + \delta^+ F_p^\Delta(p_j, p_{j+1}) \\ = \int_{c_j} s_A + s_f + s_q dx. \end{aligned}$$

for $j = 1, \dots, N$. Here, c_j is the j -th cell, $q_j^i = \int_{c_j} v^i dx$ is the cell average of the i -th component of v , F_p^Δ is the flux approximation of the pressure term, F_d^Δ is the flux approximation for the viscous term and $r = (u, u, u, u)^\top$.

The three-stage Runge-Kutta (SSP RK3), even though not structure-preserving, is used to integrate (2.3.7) in time. The pressure profile for the steady state, with $q' = 0$, and the pressure oscillatory mode in the unsteady phase is presented in Figures 2.12(a) and 2.12(b), respectively.

The discontinuities that appear in the solution of (2.3.7) suggest that a relatively large basis is required to resolve fine structures. Here, a POD basis is generated with $n = 200$, $n = 300$ and $n = 400$ number of basis vectors. To avoid basis changes in the reduced system, only one POD basis is considered for ρ , ρu , ρE , and ρY_{ox} . The explicit SSP RK3 is then used to integrate the reduced system, for the unsteady system. The source terms are evaluated in the high-fidelity space and projected onto the reduced space. However, in principle, the DEIM can be applied to accelerate the evaluation of this component.

Figure 2.14(a) shows the approximation error of the pressure, due to MOR. It is observed that the approximation is consistently improved as the number of basis vectors increases. Furthermore, the approximate solution maintains high accuracy over a relatively long time-integration. The oscillation of pressure is demonstrated in Figure 2.14(b). The overall behaviour of pressure is well approximated by the reduced system. Similar results are obtained for a POD basis with higher number of modes. We note that the discrete form of (2.3.7) is not in the full skew-symmetric form. Nonetheless, the quasi-skew-symmetric discretization offers remarkable stability preservation.

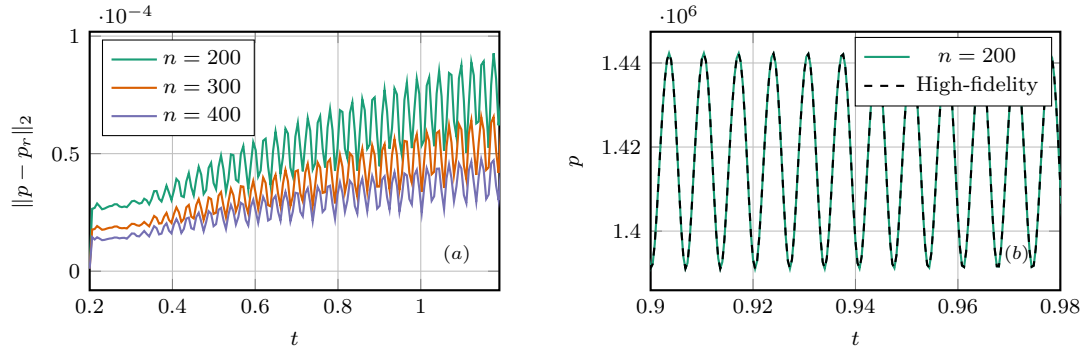


Figure 2.14: Absolute error between the high-fidelity and approximated pressure profiles (c) and visual comparison of the pressure oscillations at $x = 0.36$ (d).

3 Model order reduction of Hamiltonian systems

This Chapter is organized as follows. In Section 3.1, we present the structure characterizing the dynamics of Hamiltonian systems and the concept of symplectic transformations. In Section 3.2, we show that linear symplectic maps can be used to guarantee that the reduced models inherit the geometric formulation from the full dynamics. Different strategies to generate such maps are investigated in Section 3.3, with thoughts on optimality results and computational complexities. Finally, we discuss applications of structure-preserving reduction techniques to two more general classes of problems in Section 3.4.

3.1 Symplectic geometry and Hamiltonian systems

Let us first establish some definitions and properties concerning symplectic vector spaces.

Definition 3.1.1. Let \mathcal{P} be a $2N$ -dimensional vector space. A bilinear map $\omega_{2N} : \mathcal{P} \times \mathcal{P} \rightarrow \mathbb{R}$ is called anti-symmetric (or skew-symmetric) if

$$\omega_{2N}(u, v) = -\omega_{2N}(v, u), \quad \forall u, v \in \mathcal{P}.$$

It is non-degenerate if

$$\omega_{2N}(u, v) = 0, \quad \forall u \in \mathcal{P}. \quad \implies \quad v = 0. \quad (3.1.1)$$

Definition 3.1.2. Let \mathcal{V}_{2N} be a finite-dimensional manifold with ω_{2N} an anti-symmetric bilinear form on $T\mathcal{V}_{2N}$. The pair $(\mathcal{V}_{2N}, \omega_{2N})$ is a *symplectic manifold* if ω_{2N} is non-degenerate. Moreover, \mathcal{V}_{2N} has to be even dimensional.

In Definition 3.1.2, for any $u \in \mathcal{V}_{2N}$ the bilinear form ω_{2N} is the map $\omega_{2N}|_u : T_u\mathcal{V}_{2N} \times T_u\mathcal{V}_{2N} \rightarrow \mathbb{R}$ on the tangent space $T_u\mathcal{V}_{2N}$ of \mathcal{V}_{2N} at u and it varies smoothly with u . Hence, the non-degeneracy of the form implies that for every point of the manifold \mathcal{V}_{2N} the skew-symmetric pairing introduced by ω_{2N} on the tangent space $T_u\mathcal{V}_{2N}$ is non-degenerate in the sense defined in (3.1.1). If ω_{2N} is also closed, i.e., the exterior derivative $d\omega_{2N}$ is zero, then it is possible to define a Lie algebra structure for the symplectic manifold $(\mathcal{V}_{2N}, \omega_{2N})$. In this case, there exists a contravariant skew-symmetric tensor $\mathcal{J}_{2N} : T^*\mathcal{V}_{2N} \rightarrow T\mathcal{V}_{2N}$ of rank 2 on the manifold \mathcal{V}_{2N} such that, for all $\mathcal{F}, \mathcal{G} \in C^\infty(\mathcal{V}_{2N})$, it is possible to define a bracket as

$$\{\mathcal{F}, \mathcal{G}\}_{2N} := \omega_{2N}(\mathcal{J}_{2N}d\mathcal{F}, \mathcal{J}_{2N}d\mathcal{G}), \quad (3.1.2)$$

which takes the name of *Poisson bracket* and the related tensor is commonly referred to as *Poisson tensor*. The Poisson bracket introduces an algebraic structure in the sense given in the following proposition.

Proposition 3.1.3 ([2, Proposition 3.3.17, page 194]). *The real vector space $C^\infty(\mathcal{V}_{2N})$, together with the Poisson bracket $\{\cdot, \cdot\}_{2N}$ defined in (3.1), forms a Lie algebra.*

Since we are interested in structure-preserving transformations, preserving the structure means preserving the anti-symmetric bilinear form, as stated in the following definition.

Definition 3.1.4. Let $(\mathcal{V}_{2N}, \omega_{2N})$ and $(\mathcal{V}_{2n}, \omega_{2n})$ be two symplectic manifolds of finite dimensions with $n \leq N$. The differentiable map $\varphi : (\mathcal{V}_{2N}, \omega_{2N}) \rightarrow (\mathcal{V}_{2n}, \omega_{2n})$ is a symplectic transformation (symplectomorphism) if

$$\varphi^* \omega_{2n} = \omega_{2N},$$

where $\varphi^* \omega_{2n}$ is the pull-back of ω_{2n} with φ or, equivalently, if it satisfies

$$\varphi^* \{\mathcal{F}, \mathcal{G}\}_{2n} = \{\varphi^* \mathcal{F}, \varphi^* \mathcal{G}\}_{2N}, \quad \forall \mathcal{F}, \mathcal{G} \in C^\infty(\mathcal{V}_{2n}).$$

The closedness of the bilinear form ω_{2N} is also important for the local representation of symplectic manifolds as symplectic vector spaces, which are vector spaces over a field equipped with a bilinear form satisfying the requirements introduced in Definition 3.1.1. This result becomes even more relevant in the context of MOR via RB since we are interested in approximating high-dimensional symplectic manifolds, such as solution manifolds of Hamiltonian systems, with low-dimensional linear vector spaces.

Theorem 3.1.5 ([2, Proposition 3.3.21, page 195]). *Let $(\mathcal{V}_{2N}, \omega_{2N})$ be a finite-dimensional symplectic manifold, and (U, Ψ) be an atlas of coordinate chart $\Psi(u) := (q^i(u), p_i(u))_{i=1}^N = (q^1(u), \dots, q^N(u), p_1(u), \dots, p_N(u))$ with $u \in U$. Then (U, Ψ) is a symplectic canonical chart (or Darboux' chart, or canonical basis) if and only if*

$$\{q^i, q^j\}_{2N} = \{p_i, p_j\}_{2N} = 0, \quad \{q^i, p_j\} = \delta_{j,i},$$

on U for $i, j = 1, \dots, N$. Moreover, it holds

$$\omega_{2N} = \sum_{i=1}^N dq^i \wedge dp_i,$$

where it is understood that dq^i and dp_j are elements of the dual basis for the coordinate chart $\Psi(u)$.

One constructive way of proving Theorem 3.1.5 is based on the symplectic Gram-Schmidt procedure [223]. For each $u \in U$, let $u_1, u_2 \in T_u \mathcal{V}_{2N}$ be any elements of the tangent space of \mathcal{V}_{2N} at u . Then, the symplectic canonical chart defines a basis for $T_u \mathcal{V}_{2N}$ and it holds locally

$$\omega_{2N}|_u(u_1, u_2) = \xi^\top J_{2N} \eta, \tag{3.1.3}$$

where $\xi, \eta \in \mathbb{R}^{2N}$ are the expansion coefficients of $u_1, u_2 \in T_u \mathcal{V}_{2N}$ with respect to the canonical basis and

$$J_{2N} = \begin{bmatrix} \mathbb{0}_N & \mathbb{I}_N \\ -\mathbb{I}_N & \mathbb{0}_N \end{bmatrix}, \tag{3.1.4}$$

3.1 Symplectic geometry and Hamiltonian systems

with $\mathbb{0}_N \in \mathbb{R}^{N \times N}$ and $\mathbb{I}_N \in \mathbb{R}^{N \times N}$ denoting the zero and identity matrices, respectively. More generally, using a non-canonical basis, the form retains a structure similar to (3.1.3), with J_{2N} being an invertible skew-symmetric matrix.

One of the essential properties of Euclidean spaces is that all the Euclidean spaces of equal dimensions are isomorphic. By considering Theorem 3.1.5 for the specific case of symplectic vector spaces, a similar result holds since two $2N$ -dimensional symplectic vector spaces are symplectomorphic to one another. In particular, there exists a symplectomorphism between any $2N$ -dimensional symplectic vector space and the symplectic vector space $(\mathbb{R}^{2N}, \omega_0)$, with ω_0 being the canonical symplectic form on \mathbb{R}^{2N} given by (3.1.3). Therefore, symplectic vector spaces are fully characterized by their dimensions.

One of the most relevant applications of the abovementioned concepts of symplectic geometry to the field of dynamical systems concerns the definition of Hamiltonian systems on symplectic manifolds. The gist of the propositions to follow is that to any smooth function on a symplectic manifold, it is associated a vector field, whose flow preserves the smooth function and the symplectic form. The pairing between the given smooth function and the conservative flow stemming from the related vector field represents an instance of the Noether theorem for symplectic manifolds, which affirms that conserved quantities are reflections of symmetries of the system.

Definition 3.1.6. Let $(\mathcal{V}_{2N}, \omega_{2N})$ be a finite dimensional symplectic manifold and $\mathcal{H} : \mathcal{V}_{2N} \rightarrow \mathbb{R}$ a C^∞ function on \mathcal{V}_{2N} . We refer to the vector field $\mathcal{X}_{\mathcal{H}} \in T\mathcal{V}_{2N}$, which satisfies

$$d\mathcal{H} = i(\mathcal{X}_{\mathcal{H}})\omega_{2N},$$

as the *Hamiltonian vector field* related to \mathcal{H} , where $i(\mathcal{X}_{\mathcal{H}})$ denotes the contraction operator and d is the exterior derivative. The function \mathcal{H} is called the *Hamiltonian* of the vector field $\mathcal{X}_{\mathcal{H}}$.

Given a Hamiltonian function \mathcal{H} and a related vector field $\mathcal{X}_{\mathcal{H}}$, a second function \mathcal{H}' is another Hamiltonian for $\mathcal{X}_{\mathcal{H}}$ if and only if $d(\mathcal{H} - \mathcal{H}') = 0$ for all the elements of \mathcal{V}_{2N} . On the other hand, from the non-degeneracy of the bilinear form ω_{2N} it follows the uniqueness of the vector field $\mathcal{X}_{\mathcal{H}}$ associated with \mathcal{H} . Moreover, if $\mathcal{X}_{\mathcal{H}}$ is complete [107], it can be integrated, i.e., there exists a phase flow, which is a one-parameter diffeomorphism $\Phi_{\mathcal{X}_{\mathcal{H}}} : \mathcal{V}_{2N} \rightarrow \mathcal{V}_{2N}$, that satisfies the equation

$$\begin{cases} \frac{d}{dt}\Phi_{\mathcal{X}_{\mathcal{H}}}(t; u) = \mathcal{X}_{\mathcal{H}}(\Phi_{\mathcal{X}_{\mathcal{H}}}(t; u)), & \text{for } t \in \mathcal{T}, \\ \Phi_{\mathcal{X}_{\mathcal{H}}}(0; u) = u. \end{cases} \quad (3.1.5)$$

Equation (3.1.5) is referred to as *Hamilton's equation* of evolution or *Hamiltonian system*.

Let us consider the Darboux' chart defined in Theorem 3.1.5. The Hamiltonian vector field $\mathcal{X}_{\mathcal{H}}$ can be locally written as

$$\mathcal{X}_{\mathcal{H}} = \sum_{i=1}^N \frac{\partial}{\partial q_i} \mathcal{H} \frac{\partial}{\partial p_i} - \frac{\partial}{\partial p_i} \mathcal{H} \frac{\partial}{\partial q_i}. \quad (3.1.6)$$

Moreover, in canonical coordinates on the phase space, given two functions $\mathcal{F}, \mathcal{G} \in C^\infty(\mathcal{V}_{2N})$, the Poisson bracket takes the local form

$$\{\mathcal{F}, \mathcal{G}\}_{2N} = \sum_{i=1}^N \left(\frac{\partial}{\partial q_i} \mathcal{F} \frac{\partial}{\partial p_i} \mathcal{G} - \frac{\partial}{\partial p_i} \mathcal{F} \frac{\partial}{\partial q_i} \mathcal{G} \right). \quad (3.1.7)$$

Using (3.1.6), the local expression of the Hamiltonian system (3.1.5) in the Darboux' chart is

$$\begin{cases} \frac{d}{dt}q_i = \{q_i, \mathcal{H}\}_{2N} = \frac{\partial}{\partial p_i}\mathcal{H}, \\ \frac{d}{dt}p_i = \{p_i, \mathcal{H}\}_{2N} = -\frac{\partial}{\partial q_i}\mathcal{H}, \end{cases} \quad (3.1.8)$$

which is a first order system in the (q, p) -space, or *generalized phase-space*.

Thus, if the state vector $u = (q_1, \dots, q_N, p_1, \dots, p_N)$ is introduced, (3.1.8) takes the form

$$\frac{d}{dt}u(t) = J_{2N}\nabla_u\mathcal{H}(u(t)), \quad (3.1.9)$$

where $\nabla_u\mathcal{H}(u(t))$ is the gradient of \mathcal{H} . The following proposition highlights an important property of the flow $\Phi_{\mathcal{X}_{\mathcal{H}}}$ of Hamiltonian systems.

Proposition 3.1.7. *Let $\Phi_{\mathcal{X}_{\mathcal{H}}}$ be the flow of a Hamiltonian vector field $\mathcal{X}_{\mathcal{H}}$. Then $\Phi_{\mathcal{X}_{\mathcal{H}}} : \mathcal{V}_{2N} \rightarrow \mathcal{V}_{2N}$ is a symplectomorphism.*

We rely on a geometric perspective to highlight the importance of Proposition 3.1.7 in the context of symplectic vector spaces. First, we notice that expression (3.1.3) can be rewritten as the sum of N contributions of the form

$$\omega_{2N}^i|_u(u_1, u_2) = \xi_i\eta_{N+i} - \xi_{N+1}\eta_i, \quad (3.1.10)$$

each representing the oriented area of the orthogonal projection of the 2-dimensional parallelogram in \mathbb{R}^{2N} generated by different subsets of components of the expansion coefficients. The orientation here follows the "right-hand" rule convention for vector multiplication. Given a symplectic map φ , Definition 3.1.4 is recast [171] into the requirement for the Jacobian φ' of the transformation to preserve the bilinear form in the sense that

$$\omega_{2N}(\varphi' u, \varphi' v) = \omega_{2N}(u, v),$$

to be a symplectomorphism. Hence, as a consequence of (3.1.10), every symplectomorphism is a volume-preserving transformation. However, the opposite is not necessarily true, and, in particular, the non-squeezing theorem [112] provides a negative result about the approximability of volume-preserving transformations by symplectic ones. The natural conclusion is that being symplectic is an essentially different and much stringent condition for maps than being volume-preserving. Even though out of the scope of this thesis work, we refer the reader to [12; 73] for conjectures on global invariants for symplectic transformations, known as *symplectic capacities*, that are more sophisticated than the volume defined in (3.1.10).

In addition to the conservation of volume, for the Hamiltonian dynamics it is possible to define constant of motions. The total time differential of a smooth function $\mathcal{I}(q, p, t)$ is given by

$$\begin{aligned} \frac{d}{dt}\mathcal{I} &= \frac{\partial}{\partial t}\mathcal{I} + \sum_{i=1}^N \left(\frac{\partial}{\partial q_i}\mathcal{I} \frac{d}{dt}q_i + \frac{\partial}{\partial p_i}\mathcal{I} \frac{d}{dt}p_i \right) \\ &= \frac{\partial}{\partial t}\mathcal{I} + \sum_{i=1}^N \left(\frac{\partial}{\partial q_i}\mathcal{I} \frac{\partial}{\partial p_i}\mathcal{H} - \frac{\partial}{\partial p_i}\mathcal{I} \frac{\partial}{\partial q_i}\mathcal{H} \right) \\ &= \frac{\partial}{\partial t}\mathcal{I} + \{\mathcal{I}, \mathcal{H}\}_{2N}, \end{aligned}$$

where we used the definition of Hamiltonian system (3.1.8) and the Poisson bracket in canonical form (3.1.7) in the above equalities, leading to the following definition.

Definition 3.1.8. A smooth function $\mathcal{I} : \mathcal{V}_{2N} \times \mathbb{R} \rightarrow \mathbb{R}$ is an *invariant of motion* of the Hamiltonian system (3.1.9) if \mathcal{I} is time-independent and $\{\mathcal{I}, \mathcal{H}\}_{2N} = 0$ for all $u \in \mathcal{V}_{2N}$.

The Hamiltonian, if time-independent, is a constant of motion and as a consequence, it is preserved along the orbits of $\mathcal{X}_{\mathcal{H}}$.

3.2 Symplectic Galerkin projection

The motivation of MOR is to reduce the computational complexity of dynamical systems in numerical simulations. In the context of structure-preserving projection-based reduction, two key ingredients are required to define a reduced model. First, we need a low-dimensional symplectic vector space that accurately represents the solution manifold of the original problem. Then, we have to define a projection operator to map the symplectic flow of the Hamiltonian system onto the reduced space while preserving its delicate geometric properties.

Let us assume that Hamilton's equation can be written in the canonical form

$$\begin{cases} \frac{d}{dt}u(t) = J_{2N} \nabla_u \mathcal{H}(u(t)), \\ u(0) = u_0, \end{cases} \quad (3.2.1)$$

and the related symplectic vector space is denoted by $(\mathcal{V}_{2N}, \omega_{2N})$. Symplectic projection-based model order reduction adheres to the key idea of the general framework of projection-based techniques, described in Chapter 1, with the additional requirement to approximate u in a low-dimensional symplectic vector subspace $(\mathcal{A}_{2n}, \omega_{2N})$ of dimension $2n$. In the following, for the sake of notation, we use \mathcal{A}_{2n} to indicate the reduced symplectic vector space paired with its bilinear form. As for standard reduction techniques, we aim at $n \ll N$ to have a clear reduction, and therefore, significant gains in terms of computational efficiency. Let us consider the linear map $\Phi : (\mathcal{V}_{2N}, \omega_{2N}) \rightarrow (\mathcal{A}_{2n}, \omega_{2N})$ given by

$$z = \Phi(u) = A^+ u,$$

where $z \in \mathbb{R}^{2n}$ and $A^+ \in \mathbb{R}^{2n \times 2N}$ is the corresponding matrix representation of Φ . We require the linear map to be a symplectomorphism in the sense introduced by the following lemma as a result of the definition of symplectic map given in (3.1.4).

Definition 3.2.1 ([189, Lemma 3.1, page 415]). Let $(\mathcal{V}_{2N}, \omega_{2N})$ and $(\mathcal{A}_{2n}, \omega_{2N})$ be two symplectic vector spaces of dimension $2N$ and $2n$, respectively, with $n \leq N$. The linear map $\Phi : (\mathcal{V}_{2N}, \omega_{2N}) \rightarrow (\mathcal{A}_{2n}, \omega_{2N})$ is symplectic if and only its matrix representation $A^+ \in \mathbb{R}^{2n \times 2N}$ satisfies

$$A^+ J_{2N} (A^+)^{\top} = J_{2n}. \quad (3.2.2)$$

The matrix representation A^+ of a symplectic linear map is called *symplectic matrix*.

Given a symplectic matrix $A^+ \in \mathbb{R}^{2n \times 2N}$, its *symplectic inverse* is defined as

$$A = J_{2N} (A^+)^{\top} J_{2n}^{\top}.$$

The symplectic inverse A represents the adjoint operator of the symplectic matrix A^+ with respect to the bilinear form ω_{2N} , i.e. $\omega_{2N}(A^+v, u) = \omega_{2N}(u, Av)$, for all $u \in \mathcal{V}_{2n}$ and $v \in \mathcal{V}_{2N}$. The condition (3.2.2) on the symplecticity of A^+ translates to

$$A^\top J_{2N} A = J_{2n} \quad (3.2.3)$$

for the symplectic inverse A . That is why, with a little abuse of notation, in the following we say that $A \in \mathbb{R}^{2N \times 2n}$ is also symplectic if it belongs to the symplectic Stiefel manifold, defined by

$$\text{Sp}(2n, \mathbb{R}^{2N}) := \{L \in \mathbb{R}^{2N \times 2n} : L^\top J_{2N} L = J_{2n}\}. \quad (3.2.4)$$

It can be easily verified that $A^+(A^+)^\top = \mathbb{I}_{2N}$ if and only if $A = (A^+)^\top$. In the same setting, it is possible to construct an oblique projection operator onto \mathcal{A}_{2n} that reads

$$\Pi'_{\mathcal{V}_{2n}} = A(A^\top A)^{-1} A^\top = A(A^+ A)^{-1} A^+ = AA^+. \quad (3.2.5)$$

In the spirit of RB approximation, the projector (3.2.5), and hence \mathcal{A}_{2n} , should provide a faithful and efficient representation of the original solution manifold. If that is the case, the Hamiltonian system (3.2.1) is amenable of approximation with a Hamiltonian system of dimension $2n$, characterized by the reduced Hamiltonian function $\mathcal{H}_r : \mathcal{A}_{2n} \rightarrow \mathbb{R}$ defined as

$$\mathcal{H}_r(z) = \mathcal{H}(Az),$$

thus preserving the geometric structure of the problem.

In particular, in the framework of Galerkin projection, by considering the RB ansatz

$$\Pi'_{\mathcal{V}_{2N}} u = AA^+ u = Az, \quad (3.2.6)$$

equation (3.2.5) yields

$$A \frac{d}{dt} z = J_{2N} \nabla_u \mathcal{H}(Az) + \mathbf{r}, \quad (3.2.7)$$

with \mathbf{r} being the residual term. By using the chain rule and the properties of A^+ , the gradient of the Hamiltonian in (3.2.7) can be recast as

$$\nabla_u \mathcal{H}(Az) = (A^+)^\top \nabla_z \mathcal{H}_r(z). \quad (3.2.8)$$

Similarly to the approaches described in Chapter 1, by assuming the projection residual \mathbf{r} is either small or orthogonal with respect to the symplectic bilinear form to \mathcal{A}_{2n} , we recover

$$\begin{cases} \frac{d}{dt} z(t) = J_{2n} \nabla_z \mathcal{H}_r(z(t)), & \text{in } \mathcal{T}, \\ z(0) = A^+ u(0). \end{cases} \quad (3.2.9)$$

System (3.2.9) is known as a symplectic Galerkin projection of (3.2.1) onto \mathcal{A}_{2n} . Following the MOR framework described in Chapter 1, the reduction process consists of two stages. The pre-processing stage covers all the computations required to assemble A and corresponds to the *offline* stage. The numerical solution of the low-dimensional problem (3.2.9) represents the *online* stage. Even though the offline stage is possibly computationally expensive, this splitting is beneficial in a multi-query context, when multiple instances of (3.2.1) have to be solved, e.g., for parametric Hamiltonian systems.

The most remarkable result regarding the symplectic Galerkin projection is the guarantee of stability obtained by preserving the symplectic structure of the problem (3.2.1). More in detail, the characterization of symplectic matrices given in (3.2.3) is necessary for reformulating (3.2.8) in terms of the reduced Hamiltonian \mathcal{H}_r and hence preserving the structure in the reduction process. Unfortunately, traditional projection-based reduction techniques, such as those based on Galerkin projection via POD/greedy RB, lack this feature. Therefore, they do not guarantee stable ROM, even if the high-dimensional problem admits a stable solution [210], often resulting in a blowup of system energy. The first result concerning the boundness of energy for symplectic reduction is offered in the following proposition, exploiting the property of the Hamiltonian \mathcal{H} of being an invariant of motion.

Proposition 3.2.2 ([198]). *Let $u(t)$ be the solution of the Hamiltonian FOM (3.2.1) and $z(t)$ be the solution to the Hamiltonian ROM (3.2.9) at the same time t . If the Hamiltonian \mathcal{H} is time-independent, then the error in the Hamiltonian, defined as*

$$\Delta\mathcal{H}(t) = |\mathcal{H}(u(t)) - \mathcal{H}(Az(t))| = |\mathcal{H}(u(0)) - \mathcal{H}(Az(0))|, \quad (3.2.10)$$

is constant for all $t \in \mathcal{T}$.

As a consequence of Proposition 3.2.2, if the initial data $u(0)$ is exactly represented by $Az(0)$ or by introducing a constant bias in the reduced representation, the error in the Hamiltonian is not only constant but equals zero.

Additional stability results regarding symplectic reduction concern Lyapunov stability. Let us consider the dynamical system (3.2.1) and its phase flow $\Phi_{\mathcal{X}_\mathcal{H}}$. A point u_e is called an *equilibrium point* (or *fixed point*) if $\mathcal{X}_\mathcal{H}(u_e) = 0$. However, this property alone does not fully characterize the behavior of the flow $\Phi_{\mathcal{X}_\mathcal{H}}$ in the vicinity of u_e . The point u_e is *Lyapunov stable* if we can choose a neighborhood of u_e as small as desired and all the future states obtained by integrating (3.2.1) will be trapped within this neighborhood, given that the initial condition u_0 is taken in a smaller neighborhood centered in the same equilibrium point u_e . This concept is formalized in the following proposition in terms of Euclidean metric.

Proposition 3.2.3 ([30]). *Let u_e be an equilibrium point for the system (3.2.1) and $\Phi_{\mathcal{X}_\mathcal{H}}$ its phase flow. The point u_e is Lyapunov stable (or nonlinearly stable) if, for any $\varepsilon > 0$, there exists $\delta > 0$ such that, if $\|u_0 - u_e\|_2 \leq \delta$, then $\|\Phi_{\mathcal{X}_\mathcal{H}}(t; u_0) - u_e\|_2 \leq \varepsilon$ for all $t \in \mathcal{T}$.*

For an equilibrium point to be Lyapunov stable, a sufficient condition is provided by the Lyapunov stability theorem.

Proposition 3.2.4 ([30]). *The equilibrium point u_e for the system (3.2.1) is Lyapunov stable if there exists a scalar function $\mathcal{W} : \mathcal{V}_{2N} \rightarrow \mathbb{R}$, such that $\nabla_u \mathcal{W}(u_e) = 0$, the Hessian $\nabla_u^2 \mathcal{W}(u_e)$ is positive definite, and*

$$\nabla_u \mathcal{W}(u)^\top \mathcal{X}_\mathcal{H}(u) \leq 0, \quad \forall u \in \mathcal{V}_{2N}. \quad (3.2.11)$$

The scalar function \mathcal{W} takes the name of Lyapunov function.

Since for a Hamiltonian system $\mathcal{X}_\mathcal{H}(u) = J_{2N} \nabla_u \mathcal{H}(u)$ and J_{2N} is skew-symmetric, condition (3.2.11) is automatically fulfilled for $\mathcal{W} = \mathcal{H}$, making the Hamiltonian \mathcal{H} a natural candidate for the role of Lyapunov function [39]. Thus, if the Hessian of \mathcal{H} is positive definite, then the Lyapunov stability theorem applies. And if the Hessian is negative, one considers $-\mathcal{H}$ as Lyapunov function, leading to the Dirichlet's stability theorem for Hamiltonian systems.

Proposition 3.2.5. *In the setting of Propositions 3.2.3 and 3.2.4, an equilibrium point u_e is Lyapunov stable if it is an isolated local minimum or maximum of the Hamiltonian \mathcal{H} .*

The brief excursus on Lyapunov stability theory for Hamiltonian systems has been propaedeutic for the following result of stability for the solution of Hamiltonian ROMs.

Proposition 3.2.6 ([3]). *Let u_e be an equilibrium point for (3.2.1) and u_e that is a strict local minimum/maximum of \mathcal{H} in the open set $S_{u_e, \mathcal{H}}$ centered in u_e . If $\text{span}(A) \cap S_{u_e, \mathcal{H}} \neq 0$, then there exists a Lyapunov stable equilibrium point for the reduced Hamiltonian system (3.2.9) in $\text{span}(A) \cap S_{u_e, \mathcal{H}}$.*

In the following Section, we describe different strategies to construct the symplectic matrix A as a result of optimization problems, similarly to the POD and greedy procedures.

3.3 Proper symplectic decomposition

Let us consider the snapshot matrix $S^u \in \mathbb{R}^{2N \times N_s}$ introduced in Section 1.3.1 while discussing the POD method. We emphasize that the columns of S^u are the solution vectors $u \in \mathbb{R}^{2N}$ of (3.2.1), obtained for different time instances $t_i \in \mathcal{T}_\Delta$ and parameter instances $\eta_j \in \Gamma_h$.

In Section 3.2, we have shown that an RB ansatz of the form (3.2.6), paired with the symplectic Galerkin projection, leads to a Hamiltonian ROM. To preserve the geometric structure of the original model with the reduction, we consider an optimization problem similar to the POD, known as proper symplectic decomposition (PSD), which represents a data-driven basis generation procedure to extract a symplectic basis from S^u . It is based on the minimization of the projection error of S^u onto the symplectic vector space \mathcal{A}_{2n} , and it results in the following optimization problem for the definition of the symplectic basis $A \in \mathbb{R}^{2N \times 2n}$:

$$\begin{aligned} & \underset{A \in \mathbb{R}^{2N \times 2n}}{\text{minimize}} \quad \|S^u - AA^+ S^u\|_F, \\ & \text{subject to} \quad A \in \text{Sp}(2n, \mathbb{R}^{2N}) \end{aligned} \tag{3.3.1}$$

where $\|\cdot\|_F$ is the Frobenius norm and $\text{Sp}(2n, \mathbb{R}^{2N})$ is the symplectic Stiefel manifold defined in (3.2.4). Problem (3.3.1) is similar to the POD minimization, but with the feasibility set of rectangular orthogonal matrices, also known as Stiefel manifold, replaced by its symplectic counterpart. Regardless of the MOR discussed in this thesis, PSD has relevant implications in different physical applications, such as the study of optical systems [258] and the optimal control of quantum symplectic gates [265]. From a mathematical perspective, we mention applications to optimal control systems [24; 263] and reduction of the Riccati equations [26]. Unfortunately, PSD is significantly more challenging for different reasons with respect to POD. The non-convexity of the feasibility set and the unboundedness of the solution norm preclude standard optimization techniques, which may be plagued by compounding numerical errors [140]. Similarly, iterative solvers are constrained by the $4Nn$ degrees of freedom in the optimization problem, especially for $N \gg 1$. Moreover, most of the attention is focused on the case $2n = 2N$, which is not compatible with the reduction framework of MOR.

Despite the interest in the topic, an efficient optimal solution algorithm has yet to be found for the PSD. Suboptimal solutions have been attained by focusing on the subset of the ortho-symplectic matrices, i.e.,

$$\mathbb{S}(2n, \mathbb{R}^{2N}) := \text{Sp}(2n, \mathbb{R}^{2N}) \cap \text{St}(2n, \mathbb{R}^{2N}). \tag{3.3.2}$$

In [198], while enforcing the additional orthogonality constraint in (3.3.1), the optimization problem is further simplified by assuming a specific structure for A . An efficient greedy method, not requiring any additional block structures to A , but only its orthogonality and simplicity, has been introduced in [3]. More recently, in [38], the orthogonality requirement has been removed, and different solution methods to the PSD problem are explored. In the following, we briefly review the abovementioned approaches.

3.3.1 SVD-based methods for orthonormal symplectic basis generation

In [198], several algorithms have been proposed to directly construct ortho-symplectic bases. Exploiting the SVD decomposition of rearranged snapshots matrices, the idea is to search for optimal matrices in subsets of $\text{Sp}(2n, \mathbb{R}^{2N})$. Consider the more restrictive feasibility set

$$\mathbb{S}_{\text{CL}}(2n, \mathbb{R}^{2N}) := \text{Sp}(2n, \mathbb{R}^{2N}) \cap \left\{ \begin{bmatrix} \Phi & 0 \\ 0 & \Phi \end{bmatrix} \middle| \Phi \in \mathbb{R}^{N \times n} \right\},$$

where CL is used to recall the name method (Cotangent Lift) that we present in the following. Symplecticity condition (3.2.3) is satisfied if and only if $\Phi^\top \Phi = \mathbb{I}_N$, i.e., $\Phi \in \text{St}(n, \mathbb{R}^N)$. This establishes a bijection between $\mathbb{S}_{\text{CL}}(2n, \mathbb{R}^{2N})$ and $\text{St}(n, \mathbb{R}^N)$, leading to

$$\mathbb{S}_{\text{CL}}(2n, \mathbb{R}^{2N}) = \left\{ \begin{bmatrix} \Phi & 0 \\ 0 & \Phi \end{bmatrix} \middle| \Phi \in \text{St}(n, \mathbb{R}^N) \right\}.$$

The PSD problem is then replaced by (3.3.1)

$$\begin{aligned} & \underset{\Phi \in \mathbb{R}^{N \times n}}{\text{minimize}} \quad \|S_{\text{CL}}^u - \Phi \Phi^\top S_{\text{CL}}^u\|_F, \\ & \text{subject to} \quad \Phi \in \text{St}(n, \mathbb{R}^N), \end{aligned} \tag{3.3.3}$$

where $S_{\text{CL}}^u \in \mathbb{R}^{N \times 2N_s}$ is obtained by stacking as separate columns the generalized positions and momenta of the snapshots used to assemble S^u , i.e.,

$$S_{\text{CL}}^u = \begin{bmatrix} p^1 & \dots & p^{N_s} & q^1 & \dots & q^{N_s} \end{bmatrix} \in \mathbb{R}^{N \times 2N_s}.$$

Thus, as a result of the Eckart-Young-Mirsky-Schmidt theorem, (3.3.3) admits a solution in terms of the singular value decomposition of the matrix S_{CL}^u since the optimization problem has the same structure as the POD problem. The main difference with the POD algorithm is the approximation target: while in the POD case we target the entire solution vector, with the approach described above, the resulting optimal basis should fit for both generalized positions and momenta. Thus, by weighting either the positions or the momenta, it is possible to favor the approximation of one of the two quantities, without having, however, a guarantee of optimality of the original PSD functional.

This algorithm, formally known as Cotangent Lift, owes its name to the interpretation of the solution A to (3.3.1) in $\mathbb{S}_{\text{CL}}(2n, \mathbb{R}^{2N})$ as the cotangent lift of linear mappings, represented by Φ and Φ^\top , between vector spaces of dimensions N and n . We refer the reader to [198] for more details on this interpretation. Moreover, this approach constitutes the natural outlet in the field of Hamiltonian systems of the preliminary work of Lall et al. [156] on tangent methods for structure-preserving reduction of Lagrangian systems.

A different strategy, known as Complex SVD, relies on the definition of the complex snapshot matrix

$$S_{\text{CSVD}}^u = \begin{bmatrix} p^1 + iq^1 & \dots & p^{N_s} + iq^{N_s} \end{bmatrix} \in \mathbb{C}^{N \times N_s},$$

with i being the imaginary unit. Let $V = \Phi + i\Psi \in \mathbb{C}^{N \times 2n}$, with $\Phi, \Psi \in \mathbb{R}^{N \times n}$, be the unitary matrix solution to the following accessory problem

$$\begin{aligned} \min_{V \in \mathbb{R}^{N \times 2n}} & \|S_{\text{CSVD}}^u - VV^\top S_{\text{CSVD}}^u\|_F, \\ \text{subject to} & \quad V \in \text{St}(2n, \mathbb{C}^N). \end{aligned} \quad (3.3.4)$$

As for the Cotangent Lift algorithm, the solution to (3.3.4) is known to be the set of the $2n$ left-singular vectors of S_{CSVD}^u corresponding to its largest singular values. In terms of the real and imaginary parts of V , the orthogonality constraint implies

$$\Phi^\top \Phi + \Psi^\top \Psi = \mathbb{I}_{2n}, \quad \Phi^\top \Psi = \Psi^\top \Phi. \quad (3.3.5)$$

Consider the ortho-symplectic matrix, introduced in [198], and given by

$$U = \begin{bmatrix} E & J_{2N}^\top E \end{bmatrix} \in \mathbb{R}^{2N \times 2n}, \quad (3.3.6)$$

with

$$E = \begin{bmatrix} \Phi \\ \Psi \end{bmatrix} \in \mathbb{R}^{2N \times n}$$

that satisfies

$$E^\top E = \mathbb{I}_n, \quad E^\top J_{2N} E = \mathbb{O}_n.$$

Using (3.3.5), it can be shown that such an A is the optimal solution of the optimization problem on the feasibility set

$$\mathbb{S}_{\text{CSVD}}(2n, \mathbb{R}^{2N}) := \text{Sp}(2n, \mathbb{R}^{2N}) \cap \left\{ \begin{bmatrix} \Phi & -\Psi \\ \Psi & \Phi \end{bmatrix} \middle| \Phi, \Psi \in \mathbb{R}^{N \times n} \right\},$$

that consists in minimization, in the Frobenius norm, of the projection of the matrix

$$S_{\text{CSVD}}^u := \begin{bmatrix} S^u & J_{2N} S^u \end{bmatrix},$$

where S^u is the snapshot matrix defined in (1.3.3) in the POD context.

In [38], extending the result obtained in [190] for square matrices, it has been shown that (3.3.6) is a complete characterization of the elements of (3.3.2), meaning that all the ortho-symplectic matrices admit a representation of the form (3.3.6), for a given E and hence $\mathbb{S}(2n, \mathbb{R}^{2N}) \equiv \mathbb{S}_{\text{CSVD}}(2n, \mathbb{R}^{2N})$. In the same work, Haasdonk et al. showed that an ortho-symplectic matrix that solves the minimization problem (3.3.4) in the context of the complex SVD algorithm, is also a minimizer for the PSD problem with an additional orthogonality constraint, and viceversa. This result is achieved using an equivalence argument based on the POD applied to the snapshot matrix S_{CSVD}^u . Thus, combining these two results, the Complex SVD algorithm provides a minimizer of the PSD problem for ortho-symplectic matrices.

3.3.2 SVD-based methods for non-orthonormal symplectic basis generation

In the previous Section, we showed that the basis provided by the Complex SVD method is not only near-optimal in $\mathbb{S}_{\text{CSVD}}(2n, \mathbb{R}^{2N})$, but is optimal for the minimization of the projection error over the entire space of ortho-symplectic matrices. The orthogonality of the resulting basis is beneficial [140], among others, for reducing the condition number associated with the fully discrete formulation of (3.2.9). A suboptimal solution to the PSD problem not requiring the orthogonality of the feasibility set is proposed in [198] as an improvement of the SVD-based generators of ortho-symplectic bases using the Gappy-POD [94], under the name of nonlinear programming approach (NLP). Let $A_* \in \mathbb{S}(2r, \mathbb{R}^{2N})$ be a basis of dimension $2r$ generated using the Complex SVD method. The idea of the NLP is to construct a target basis $A \in \text{Sp}(2n, \mathbb{R}^{2N})$, with $n < r \ll N$, via the linear mapping

$$A = A_* C, \quad (3.3.7)$$

with $C \in \mathbb{R}^{2r \times 2n}$. The symplecticity constraint on A results in C also being symplectic, i.e.,

$$C^\top J_{2r} C = J_{2n}.$$

Using (3.3.7) in (3.3.1) results in the following PSD optimization problem for the coefficient matrix C

$$\begin{aligned} \min_{A \in \mathbb{R}^{2N \times 2n}} & \left\| S^u - A_* C C^\top (A_*)^+ S^u \right\|_F, \\ \text{subject to} & \quad C \in \text{Sp}(2n, \mathbb{R}^{2r}), \end{aligned} \quad (3.3.8)$$

that is characterized by $4nr$ degrees of freedom, a significantly smaller number when compared to the $4nN$ of the original problem. However, no optimality results are available for the NLP method.

A different direction has been pursued in [38], based on the connection between traditional SVD, Schur forms, and the matrix decompositions proposed in the following theorem.

Theorem 3.3.1 ([268, Theorem 1, page 6]). *If $B \in \mathbb{R}^{2N \times N_s}$, then there exists $S \in \text{Sp}(2N, \mathbb{R}^{2N})$, $Q \in \text{Sp}(N_s, \mathbb{R}^{N_s})$ and $D \in \mathbb{R}^{2N \times N_s}$ of the form*

$$D = \begin{bmatrix} b & q & b & n-2b-q \\ \Sigma & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \Sigma & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} b \\ q \\ m-b-q \\ b \\ q \\ m-b-q \end{bmatrix}, \quad (3.3.9)$$

with $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_b)$, $\sigma_i > 0 \forall i = 1, \dots, b$, such that

$$B = SDQ.$$

Moreover, $\text{rank}(B) = 2b + q$ and σ_i are known as symplectical singular values.

Let us apply the SVD-like decomposition to the snapshot matrix S^u , where N_S represents the number of snapshots, and define its weighted singular values as

$$w_i = \begin{cases} \sigma_i \sqrt{\|S_i^u\|_2^2 + \|S_{N_S+i}^u\|_2^2}, & 1 \leq i \leq b, \\ \|S^u\|_2, & b+1 \leq i \leq b+q. \end{cases}$$

with $S_i^u \in \mathbb{R}^{2N}$ being the i -th column of S^u and $\|\cdot\|_2$ the Euclidean norm. The physical interpretation of the classical POD approach characterizes the POD reduced basis as the set of a given cardinality that captures most of the system's energy. The energy retained in the reduced approximation is quantified as the sum of the squared singular values corresponding to the left singular vectors of the snapshot matrix representing the columns of the basis. A similar guiding principle is used in [38], where the energy of the system, i.e., the Frobenius norm of the snapshot matrix, is connected to the weighted symplectic singular values as

$$\|S^u\|_F^2 = \sum_{i=1}^{b+q} w_i^2. \quad (3.3.10)$$

Let \mathcal{I}_{PSD} be the set of indices corresponding to the n largest contributors in (3.3.10),

$$\mathcal{I}_{\text{PSD}} = \{i_j\}_{j=1}^n = \arg \max_{\mathcal{I} \subset \{1, \dots, b+q\}} \left(\sum_{i \in \mathcal{I}} w_i^2 \right).$$

Then the PSD SVD-like decomposition defines a symplectic RB $A \in \text{Sp}(2n, \mathbb{R}^{2N})$ by selecting the pairs of columns from the symplectic matrix S^u corresponding to the indices set \mathcal{I}_{PSD}

$$A = \begin{bmatrix} s_{i_1} & \dots & s_{i_n} & s_{N+i_1} & \dots & s_{N+i_n} \end{bmatrix}.$$

Similarly to the POD, the reconstruction error of the snapshot matrix depends on the magnitude of the discarded weighted symplectic singular values as

$$\|S^u - AA^+ S^u\|_F^2 = \sum_{i \in \{1, \dots, b+q\} \setminus \mathcal{I}_{\text{PSD}}} w_i^2. \quad (3.3.11)$$

Even though there is no proof that the PSD SVD-like algorithm reaches the global optimum in the sense of (3.3.1), the error upper bounds and the numerical investigations offered in [38] suggest that it provides superior results compared to orthonormal techniques.

3.3.3 Greedy approach to symplectic basis generation

In [3], in the framework of structure-preserving model order reduction, a variation of the greedy method described in Section 1.3.2 using the Hamiltonian as a proxy error indicator to assemble a symplectic basis is proposed. Let

$$A_{2k} := \begin{bmatrix} E_k & J_{2N}^\top E_k \end{bmatrix} \in \mathbb{R}^{2N \times 2k},$$

with

$$E_k = \begin{bmatrix} e_1 & \dots & e_k \end{bmatrix} \in \mathbb{R}^{2N \times k},$$

be a given ortho-symplectic basis of dimension $2k$ and consider

$$(t^*, \eta^*) := \operatorname{argmax}_{(t_i, \eta_j) \in \mathfrak{S}_\Delta^u} |\mathcal{H}(u(t_i; \eta_j)) - \mathcal{H}(A_{2k} A_{2k}^+ u(t_i; \eta_j))|. \quad (3.3.12)$$

By (3.2.10), the error in the Hamiltonian depends only on the initial condition. Hence, by assuming that the Hamiltonian function is a good indicator of error, the indicator (3.3.12) does not require integrating in time the FOM (3.2.1) over the entire set Γ_h , but we are allowed to consider only the initial condition, making the procedure faster and more efficient. The parameter space can be explored first to identify the value of the parameter that maximizes the error in the Hamiltonian as a function of the initial condition, i.e.,

$$\eta^* := \operatorname{argmax}_{\eta_j \in \Gamma_h} |\mathcal{H}(u_0(\eta_j)) - \mathcal{H}(A_{2k} A_{2k}^+ u_0(\eta_j))|.$$

This step may fail if $u_0(\eta_j) \in \operatorname{range}(A_{2k})$, $\forall \eta_j \in \Gamma_h$. Then the FOM (3.2.1) is integrated to collect the snapshot matrix

$$S_{\eta^*} = \begin{bmatrix} u(0; \eta^*) & u(t_1; \eta^*) & \dots & u(t_{N_t}; \eta^*) \end{bmatrix}. \quad (3.3.13)$$

Finally, the candidate basis vector $u^* = u(t^*; \eta^*)$ is selected as the snapshot that maximizes the projection error

$$t^* := \operatorname{argmax}_{t_i \in \mathcal{T}_\Delta} \|u(t_i; \eta_j) - A_{2k} A_{2k}^\top u(t_i; \eta_j)\|_2.$$

Standard orthogonalization techniques, such as QR methods, fail to preserve the symplectic structure [44]. In [3], the SR method [224], based on the symplectic Gram-Schmidt, is employed to compute the additional basis vector e_{k+1} that conforms to the geometric structure of the problem. To conclude the $(k+1)$ -th iteration of the algorithm, the basis A_{2k} is expanded in

$$A_{2(k+1)} = \begin{bmatrix} E_k & e_{k+1} & J_{2N}^\top E_k & J_{2N}^\top e_{k+1} \end{bmatrix}.$$

We stress that, with this method, known as symplectic greedy RB, two vectors, e_{k+1} and $J_{2N}^\top e_{k+1}$, are added to the symplectic basis at each iteration, because of the structure of ortho-symplectic matrices. A different strategy, known as PSD-Greedy algorithm and partially based on the PSD SVD-like decomposition, has been introduced in [40], with the feature of not using orthogonal techniques to compress the matrix S_{η^*} . In [3], following the results provided in [41], the exponential convergence of the strong greedy method has been proved.

Theorem 3.3.2. *Suppose that*

$$D_{2n}(\mathcal{M}^*) \leq M e^{-\alpha n}, \quad n > 0,$$

for some $M > 0$, $\alpha > \log 3$, and \mathcal{M}^ a compact subset of \mathcal{M} . Then there exists $\beta > 0$ such that the symplectic basis A_{2n} generated by the symplectic strong greedy algorithm provides exponential approximation properties, i.e.,*

$$\max_{u \in \mathcal{M}^*} \|u - A_{2n} A_{2n}^+ u\|_2 \leq C e^{-\beta n}, \quad n > 0, \quad (3.3.14)$$

Theorem (3.3.2) holds only when the projection error is used as the error indicator instead of the error in the Hamiltonian. However, it has been observed for different symplectic parametric

problems [3] that the symplectic method using the loss in the Hamiltonian converges with the same rate of (3.3.14). The orthogonality of the basis is used to prove the convergence of the greedy procedure. In the case of a non-orthonormal symplectic basis, supplementary assumptions are required to ensure the convergence of the algorithm.

3.4 Extension to more general Hamiltonian problems

Many areas of engineering require a more general framework than the one offered by classical Hamiltonian systems described in Section 3.1, requiring the inclusion of energy-dissipating elements or state-dependent Poisson tensors. The last Section of this Chapter briefly presents structure-preserving reduction techniques for these two special cases.

3.4.1 Dissipative Hamiltonian systems

While the principle of energy conservation is still used to describe the state dynamics, dissipative perturbations must be modelled and introduced in the Hamiltonian formulation (3.2.1). Dissipative Hamiltonian systems, with so-called Rayleigh-type dissipation, are considered as special case of forced Hamiltonian systems, with the state $y = (q, p) \in \mathbb{R}^{2N}$, with $q, p \in \mathbb{R}^N$, following the time evolution given by

$$\begin{cases} \frac{d}{dt}u(t) = J_{2N}\nabla\mathcal{H}(u(t)) + \mathcal{X}_F(u(t)), \\ u(0) = u_0, \end{cases} \quad (3.4.1)$$

where $\mathcal{X}_F \in \mathbb{R}^{2N}$ is a velocity field, introducing dissipation, of the form

$$\mathcal{X}_F := \begin{bmatrix} 0_N \\ f_{\mathcal{H}}(u(t)) \end{bmatrix}. \quad (3.4.2)$$

Following the definition of invariant of motion in Definition 3.1.8, we require \mathcal{X}_F to satisfy $(\nabla_u \mathcal{H})^\top \mathcal{X}_F \leq 0$, $\forall u \in \mathbb{R}^{2N}$, to represent a dissipative term and therefore

$$(\nabla_p \mathcal{H})^\top f_{\mathcal{H}} \leq 0. \quad (3.4.3)$$

In terms of Rayleigh dissipation theory, there exists a symmetric positive semidefinite matrix $R(q) \in \mathbb{R}^{N \times N}$ such that $f_{\mathcal{H}} = -R(q)\dot{q}(p, q)$ and (3.4.3) reads

$$(\nabla_p \mathcal{H})^\top f_{\mathcal{H}} = \dot{q}^\top f_{\mathcal{H}} = -\dot{q}^\top R(q)\dot{q} \leq 0.$$

Several strategies have been proposed to generate stable reduced approximations of (3.4.1), based on Krylov subspaces or POD [123; 203]. In [56], without requiring the symplecticity of the reduced basis, the gradient of the Hamiltonian vector field is approximated using a projection matrix W , i.e., $\nabla_u \mathcal{H}(Uz) \approx W\nabla_z \mathcal{H}_{\text{RB}}(z)$, which results in a non-canonical symplectic reduced form. The stability of the reduced model is then achieved by preserving the passivity of the original formulation. A drawback of such an approach is that, while viable for nondissipative formulations, it does not guarantee the same energy distribution of (3.4.1) between dissipative and null energy contributors. In the following, we show that the techniques based on symplectic geometry introduced in the previous sections can still be used in the dissipative framework described in

3.4 Extension to more general Hamiltonian problems

(3.4.1) with limited modifications to obtain consistent and structured reduced models. Let us consider an ortho-symplectic basis $A \in \mathbb{S}(2n, 2N)$ and the reduced basis representation $u \approx Az$, with $z = (r, s) \in \mathbb{R}^{2n}$ being the reduced coefficients of the representation and $r, s \in \mathbb{R}^n$ being the generalized phase coordinates of the reduced model. The basis A can be represented as

$$A = \begin{bmatrix} A_{qr} & A_{qs} \\ A_{pr} & A_{ps} \end{bmatrix}, \quad (3.4.4)$$

with $A_{qr}, A_{qs}, A_{pr}, A_{ps} \in \mathbb{R}^{N \times n}$ being the blocks, the indices of which are chosen to represent the interactions between the generalized phase coordinates of the two models, such that $q = A_{qr}r + A_{qs}s$ and $p = A_{pr}r + A_{ps}s$. Following [197], the symplectic Galerkin projection of (3.4.1) reads

$$\frac{d}{dt}z = A^+(\mathcal{X}_{\mathcal{H}}(Az) + \mathcal{X}_F(Az)) = J_{2n}\nabla_z \mathcal{H}_{\text{RB}}(z) + A^+ \mathcal{X}_F(Az) = \mathcal{X}_{\mathcal{H}_{\text{RB}}} + A^+ \mathcal{X}_F, \quad (3.4.5)$$

with

$$A^+ \mathcal{X}_F = \begin{bmatrix} A_{ps}^\top & -A_{qs}^\top \\ -A_{pr}^\top & A_{qr}^\top \end{bmatrix} \begin{bmatrix} \mathbb{0}_N \\ f_{\mathcal{H}} \end{bmatrix} = \begin{bmatrix} -A_{qs}^\top f_{\mathcal{H}} \\ A_{qr}^\top f_{\mathcal{H}} \end{bmatrix}. \quad (3.4.6)$$

We note that, in (3.4.5), the reduced dynamics is described as the sum of a Hamiltonian vector field and a term that, for a general choice of the symplectic basis A and hence of A_{qs}^\top , does not represent a dissipative term in the form of a vertical velocity field. The Cotangent Lift method, described in Section 3.3.1, enforces the structure of a vertical velocity field because $A_{qs} = 0$. It can be shown [197] that dissipativity is also preserved since the rate of energy variation of the reduced system is non-positive, i.e.,

$$\nabla_s \mathcal{H}_{\text{RB}}(Az)(A_{qr}^\top f_{\mathcal{H}}) = \dot{r}^\top (A_{qr}^\top f_{\mathcal{H}}) = -(A_{qr}\dot{r})^\top R(A_{qr}s)(A_{qr}\dot{r}) \leq 0. \quad (3.4.7)$$

However, time discretization of the reduced dissipative model is not trivial. Even though the reduction process preserves the dissipative Hamiltonian structure, standard numerical integrators do not preserve the same structure at the fully discrete level.

A completely different approach is proposed in [166], where (3.4.1) is paired with a canonical heat bath, absorbing the energy leakage and expanding the system to the canonical Hamiltonian structure. Consider a dissipative system characterized by the quadratic Hamiltonian $\mathcal{H}(u) = \frac{1}{2}u^\top K^\top K u$. Following [91], such a system admits a time dispersive and dissipative (TDD) formulation

$$\begin{cases} \frac{d}{dt}u = J_{2N}K^\top f(t), \\ u(0) = u_0, \end{cases} \quad (3.4.8)$$

with $f(t)$ being the solution to the integral equation

$$f(t) + \int_0^t \chi(t-s)f(s)ds = Ku, \quad (3.4.9)$$

also known as a *generalized material relation*. The square time-dependent matrix $\chi \in \mathbb{R}^{2N \times 2N}$ is the *generalized susceptibility* of the system, and it is bounded with respect to the Frobenius norm. Physically, it encodes the accumulation of the dissipation effect in time, starting from the initial condition. When $\chi = \mathbb{0}_{2N}$, (3.4.8) is equivalent to (3.3.1). Under physically natural assumptions on χ (see [91, Theorem 1.1, page 975] for more details), system (3.4.8) admits a

quadratic Hamiltonian extension (QHE) to a canonical Hamiltonian system. This extension is obtained by defining an injection $I : \mathbb{R}^{2N} \mapsto \mathbb{R}^{2N} \times \mathcal{H}^{2N}$, where \mathcal{H}^{2N} is a suitable Hilbert space, and reads

$$\begin{cases} \frac{d}{dt}u = J_{2N}K^\top f(t), \\ \partial_t \phi = \theta(t, x), \\ \partial_t \theta = \partial_x^2 \phi(t, x) + \sqrt{2}\delta_0(x) \cdot \sqrt{\chi}f(t), \end{cases} \quad (3.4.10)$$

where ϕ and θ are vector-valued functions in \mathcal{H}^{2N} , δ_0 is the Dirac-delta function, and f solves

$$f(t) + \sqrt{2} \cdot \sqrt{\chi}\phi(t, 0) = Ku(t).$$

It can be shown that system (3.4.10) has the form of a conserved Hamiltonian system with the extended Hamiltonian

$$H_{ex}(u, \phi, \theta) = \frac{1}{2} \left(\|Ku - \phi(t, 0)\|_2^2 + \|\theta(t)\|_{\mathcal{H}^{2N}}^2 + \|\partial_x \phi(t)\|_{\mathcal{H}^{2N}}^2 \right),$$

and can be reduced, while preserving its geometric structure, using any of the standard symplectic techniques. We refer the reader to [166] for a formal derivation of the reduced model obtained by projecting (3.4.10) on a symplectic subspace and for its efficient time integration. The method extends trivially to more general Hamiltonian functions, as long as the dissipation is linear in (3.4.9).

3.4.2 Non-canonical Hamiltonian systems

The canonical Hamiltonian problem (3.2.4) has been defined under the assumption that a canonical system of coordinates for the symplectic solution manifold is given, and the Hamiltonian vector can be represented as (3.1.6). However, many Hamiltonian systems, such as the KdV and Burgers equations, are naturally formulated in terms of a non-canonical basis, resulting in the following description of their dynamics:

$$\begin{cases} \frac{d}{dt}u(t) = J_{2N}\nabla_u \mathcal{H}(u(t)), \\ u(0) = u_0, \end{cases} \quad (3.4.11)$$

with $J_{2N} \in \mathbb{R}^{2N \times 2N}$ being invertible and skew-symmetric, but generally different from that in (3.1.4). A reduction strategy, involving the non-canonical formulation (3.4.11) and based on POD, has been proposed in [105]. Consider the RB ansatz $u \approx Uz$, with $U \in \mathbb{R}^{2N \times n}$ as an orthonormal basis obtained by applying the POD algorithm to the matrix of snapshots collected by solving the full model. The Galerkin projection of (3.4.11) reads

$$\frac{d}{dt}z = U^\top J_{2N} \nabla_u \mathcal{H}(Uz), \quad (3.4.12)$$

with the time derivate of the Hamiltonian function, evaluated at the reduced state, given by

$$\frac{d}{dt}\mathcal{H}(Uz) = \frac{d}{dt}z^\top (\nabla_z \mathcal{H}(Uz)) = (\nabla_u \mathcal{H}(Uz))^\top J_{2N}^\top U U^\top \nabla_u \mathcal{H}(Uz). \quad (3.4.13)$$

3.4 Extension to more general Hamiltonian problems

As expected, the Hamiltonian structure is lost in (3.4.12) and the energy of the system, represented by the Hamiltonian, is no longer preserved in time because $J_{2N}UU^\top$ is not skew-symmetric. Both issues are solved in [105] by considering a matrix W , with the same properties of J_{2N} , such that the relation

$$U^\top J_{2N} = WU^\top \quad (3.4.14)$$

is satisfied. We stress that a condition similar to (3.4.14) naturally holds in the canonical Hamiltonian setting for a symplectic basis and has been used to derive Hamiltonian reduced models using the symplectic Galerkin projection. A candidate W is identified in [105] by solving the normal equation related to (3.4.14), i.e. $W = U^\top J_{2N}U$. For invertible skew-symmetric operators J_{2N} that might depend on the state variables u , Miyatake has introduced in [177] an hyper-reduction technique that preserves the skew-symmetric structure of the J_{2N} operator.

Formulation (3.4.11) is further generalized with the characterization of the phase-space as a Poisson manifold, defined as a $2N_P$ -dimensional differentiable manifold \mathcal{M}_P equipped with a Poisson bracket $\{\cdot, \cdot\} : C^\infty(\mathcal{M}_P) \times C^\infty(\mathcal{M}_P) \mapsto C^\infty(\mathcal{M}_P)$ satisfying the conditions of bilinearity, skew-symmetry, the Jacobi identity, and the Leibniz' rule. Since derivations on $C^\infty(\mathcal{M}_P)$ are represented by smooth vector fields, for each Hamiltonian function $\mathcal{H} \in C^\infty(\mathcal{M}_P)$, there exists a vector field $\mathcal{X}_{\mathcal{H}}$ that determines the following dynamics,

$$\begin{cases} \frac{d}{dt}u(t) = \mathcal{X}_{\mathcal{H}}(u) = J_{2N_P}(u)\nabla_u\mathcal{H}(u(t)), \\ u(0) = u_0, \end{cases} \quad (3.4.15)$$

with the Poisson tensor J_{2N_P} being skew-symmetric, state-dependent, and generally not invertible. The flow of the Hamiltonian vector field $\mathcal{X}_{\mathcal{H}}(u)$, which is a Poisson map and therefore preserves the Poisson bracket structure via its pullback, also preserves the rank $2N$ of the Poisson tensor $J_{2N_P}(u)$. Moreover, $r = 2N_P - 2N$ represents the number of independent nonconstant functions on \mathcal{M}_P that $\{\cdot, \cdot\}$ commutes with all the other functions in $C^\infty(\mathcal{M}_P)$. These functions are known as Casimirs of the Poisson bracket and their gradients belong to the kernel of $J_{2N_P}(y)$, making them independent of the dynamics of (3.4.15) and only representing geometric constraints on configurations of the generalized phase-state space.

An interesting relation between symplectic and Poisson manifolds is offered by the Lie-Weinstein splitting theorem, stating that locally, in the neighborhood \mathcal{U}_{u^*} of any point $u^* \in \mathcal{M}_P$, a Poisson manifold can be split into a $2N$ -dimensional symplectic manifold \mathcal{M} and an r -dimensional Poisson manifold M . Following on this result, Darboux' theorem guarantees the existence of local coordinates $(q_1, \dots, q_N, p_1, \dots, p_N, c_1, \dots, c_r)$, where $\{q_i, p_i\}_{i=1}^N$ corresponds to canonical symplectic coordinates and $\{c_i\}_{i=1}^r$ are the Casimirs, such that the Poisson tensor $J_{2N_P}(u)$ is recast, via Darboux' map, in the canonical form $J_{2N_P}^C$, i.e.,

$$J_{2N_P}^C = \begin{bmatrix} & 2N & r \\ J_{2N} & 0 & \\ 0 & 0 & \end{bmatrix} \begin{matrix} 2N \\ r \end{matrix},$$

with $J_{2N} \in \mathbb{R}^{2N \times 2N}$ being the canonical Poisson tensor defined in (3.1.4).

In [127], a quasi-structure-preserving algorithm for problems of the form (3.4.15) has been proposed, leveraging the Lie-Weinstein splitting, an approximation of the Darboux' map and

traditional symplectic RB techniques. Let

$$\begin{cases} u^{j+1} = u^j + \Delta t J_{2N_p}(\tilde{u}^j) \nabla_u \mathcal{H}(\tilde{u}^j), \\ u^0 = u_0, \end{cases} \quad (3.4.16)$$

be the fully-discrete formulation of (3.4.15), where j is the integration index, and \tilde{y}^j represents intermediate state/states dictated by the temporal integrator of choice. Given $\mathcal{M}_{P,j}$, an open subset of \mathcal{M}_P comprising the discrete states u^j , \tilde{u}^j , and u^{j+1} , the authors of [127] introduce an approximation $\varphi_{j+\frac{1}{2}} : \mathcal{M}_{P,j} \mapsto \mathcal{M}_s \times \mathcal{N}_j$ of the Darboux' map at \tilde{u}^j , with \mathcal{M}_s being a $2N$ -dimensional canonical symplectic manifold and \mathcal{N}_j approximating the null space of the Poisson structure. The proposed approximation exploits a Cholesky-like decomposition (see [127, Proposition 2.11, page 1708]) of the noncanonical rank-deficient $J_{2N_p}(\tilde{u}^j)$ and exactly preserves the dimension of \mathcal{N}_j , hence the number of independent Casimirs. By introducing the natural transition map $T_j := \varphi_{j+\frac{1}{2}} \cdot \varphi_{j-\frac{1}{2}}^{-1}$ between the neighboring and overlapping subsets \mathcal{M}_{j-1} and \mathcal{M}_j , problem (3.4.16) is locally recast in the canonical form

$$\begin{cases} \bar{u}^{j+1} = T_j \bar{u}^j + \Delta t J_{2N_p}^C \nabla_{\bar{u}} \mathcal{H}^j(\bar{u}^j), \\ \bar{u}^0 = u_0, \end{cases} \quad (3.4.17)$$

where $\bar{u}^{j+1} := \varphi_{j+\frac{1}{2}} u^{j+1}$, $\bar{u}^j := \varphi_{j+\frac{1}{2}} u^j$, $\tilde{\bar{u}}^{j+1} := \varphi_{j+\frac{1}{2}} \tilde{u}^j$, and $\mathcal{H}^j(\bar{u}^j) := \mathcal{H}(\varphi_{j+\frac{1}{2}}^{-1}(\bar{u}^j))$. Even though the flow of (3.4.17) is not a *global* $J_{2N_p}^C$ -Poisson map because the splitting is not exact, the approximation is *locally* structure-preserving for each neighborhood $\mathcal{M}_{P,j}$. By exploiting a similar splitting principle, the canonical Poisson manifold $\mathcal{M}_s \times \mathcal{N}_j$ is projected on a reduced Poisson manifold $\mathcal{A} \times \mathcal{N}_j$, with the reduction acting only on the symplectic component of the splitting and $\dim(\mathcal{A}) = 2n \ll 2N$. The corresponding reduced model is obtained via Galerkin projection of (3.4.17) using an orthogonal J_{2n}^C -symplectic basis of dimension $2n$, generated via a greedy iterative process inspired by the symplectic greedy method described in Section 3.3.3. Different theoretical estimates and numerical investigations show the proposed technique's accuracy, robustness, and conservation properties, up to errors in the Poisson tensor approximation.

4 Symplectic dynamical reduced basis

Hamiltonian systems describe conservative dynamics and non-dissipative phenomena in, for example, classical mechanics, transport problems, fluids, and kinetic models. We consider finite-dimensional Hamiltonian systems, in canonical symplectic form, that depends on a set of parameters associated with the geometric configuration of the problem or which represent physical properties of the problem. The development of numerical methods for the solution of parametric Hamiltonian systems in many-query and long-time simulations is challenged by two major factors: the high computational cost required to achieve sufficiently accurate approximations and the possible onset of numerical instabilities resulting from failing to satisfy the conservation laws underlying non-dissipative dynamics. MOR and RB methods provide an effective procedure to reduce the computational cost of such simulations by replacing the original high-dimensional problem with models of reduced dimensionality without compromising the accuracy of the approximation. The success of RB methods relies on the assumption that the problem possesses a low-rank nature, i.e., that the set of solutions, obtained as time and parameters vary, can be approximated by low dimensional space. However, non-dissipative phenomena do not generally exhibit such global low-rank structure and are characterized by slowly decaying Kolmogorov n -widths. In Chapter 2, we have seen, for some examples characterized by sharp discontinuities or dominated by advection processes, that it is possible to define stable reduced models but to obtain accuracy in the representation it is necessary to increase the dimension of the approximating space significantly. This implies that traditional reduced models derived via linear approximations, such as the gROMs discussed so far, are generally ineffective in reducing the computational cost required to solve advection dominated problems .

In recent years, there has been a growing interest in developing of model order reduction techniques for transport-dominated problems to overcome the limitations of linear global approximations. A large class of methods consists in constructing nonlinear transformations of the solution manifold and to recast it in a coordinate framework where it admits a low-rank structure, e.g. [45; 83; 137; 158; 187; 212; 248; 264; 39]. A second family of MOR techniques focuses on online adaptive methods that update local reduced spaces depending on parameter and time, e.g. [47; 196; 216]. None of the aforementioned methods provides any guarantee on the preservation of the physical properties and the geometric structure of the problem considered, and they might therefore be unsuitable to treat non-dissipative phenomena.

In parametric dynamical systems with finite numbers of parameter realizations, the state can

be represented, at each time, as a matrix whose columns are the solution vectors associated with the different parameter values. In this perspective, finding a low-dimensional space in which the solution state can be well approximated is strictly related to the problem of low-rank matrix approximations. In a time-dependent setting, dynamical low-rank approximation [146] provides a low-rank factorization updating technique to efficiently compute approximations of time-dependent large data matrices. This approach can be equivalently seen as a reduced basis method based on a modal decomposition of the approximate solution with dynamically evolving modes. A geometric perspective on the relation between dynamical low-rank approximation and model order reduction in the context of time-dependent matrices has been proposed in [89]. To the best of the author knowledge the only dynamical low-rank approximation methods able to preserve the geometric structure of Hamiltonian dynamics were proposed in [183] to deal with the spatial approximation of the stochastic wave equation and in [189] to deal with finite-dimensional Hamiltonian systems. The gist of these methods is to approximate the full model solution in a low-dimensional manifold that evolves in time and possesses the symplectic structure of the full phase-space. The reduced dynamics is then derived via a symplectic projection of the Hamiltonian vector field onto the tangent space of the reduced symplectic manifold at each reduced state.

Their success notwithstanding, traditional dynamical low-rank approximation techniques are based on a reduced (low-rank) space whose dimension is fixed at the beginning of the evolution. This is a major limitation since it frequently happens that the rank of the initial condition does not correctly reflect the effective rank of the solution at all times. Consider, as an example, a linear advection problem in 1D, where the parameter represents the transport velocity. It is clear that if the initial condition does not depend on the parameter, its rank is equals one. However, its rank rapidly increases as the initial condition is advected in time with different velocities. Approximating such dynamics with a time-dependent sequence of reduced manifolds of rank-1 matrices yields poor approximations.

Conversely, an overapproximation of the initial condition, and possibly of the solution at other times, could improve the accuracy but will inevitably yield situations of rank-deficiency, as observed in [146]. This example demonstrates that, in a dynamical reduced basis approach, it is crucial to accurately capture the rank of the full model solution at each time. However, this issue has received little attention so far [59; 225]. In this Chapter, we propose a novel dynamical low-rank approximation scheme for the solution of parametric Hamiltonian systems that combines adaptivity in the rank of the solution with preservation of the Hamiltonian structure of the dynamics.

The proposed rank-adaptive algorithm can be summarized as follows.

- Given a fixed partition of the temporal domain, in each temporal subinterval, the discretized reduced dynamical system obtained with the structure-preserving approach of [189] is considered. While in [189] the rank of the approximate solution is fixed a priori, here *the rank adaptively changes* from one temporal interval to the next one.
- To this aim, a surrogate error based on a linearization of the problem residual is computed at chosen times and for all tested parameters. If the error indicator reveals, according to a specific criterion, that the current reduced space is too small to approximate the state, we augment it in the direction that is worst approximated by the current reduced basis. The reduced dynamical system is then evolved, in the subsequent temporal interval, in the augmented manifold. Approximations for new instances of the parameter can be added by

interpolation in the coefficient space. In case of overapproximation, the size of the reduced space is, instead, decreased.

- Two major difficulties are associated with this approach: (i) to maintain the *global* Hamiltonian structure of the dynamics while modifying the reduced phase space; and (ii) to evolve the system on the updated space starting from a rank-deficient initial condition. To address these problems, we devise a regularization of the velocity field of the reduced flow so that the resulting vector belongs to the tangent space of the updated reduced manifold, and the Hamiltonian structure is then preserved. Although this introduces a small error in the Hamiltonian function, the Hamiltonian structure is exactly conserved resulting in a stable reduced model.

The remainder of the Chapter is organized as follows. In Section 4.1, we extend the definition of Hamiltonian systems made in Chapter 3 by considering the parametric case. The dynamical reduced basis method proposed in [189], which we adopt here, is summarized in Section 4.2. Section 4.3 deals with the numerical temporal integration of the reduced dynamics: first, we summarize the structure-preserving integration method for the evolution of the reduced basis and expansion coefficients, and then we design partitioned RK schemes that are accurate with order 2 and 3 and preserve the geometric structure of each evolution problem. The problem of overapproximation and rank-deficiency is discussed in Section 4.4, where the regularization algorithm is introduced. Section 4.5 pertains to the rank-adaptive algorithm. We describe the major steps: computation of the error indicator, criterion for the rank update, and update of the reduced state. The computational complexity of the adaptive dynamical reduced basis algorithm is thoroughly analyzed in Section 4.6. Section 4.7 is devoted to extensive numerical simulations of the proposed algorithm and its numerical comparisons with global reduced basis methods.¹

4.1 Problem formulation

Let $\mathcal{T} := (t_0, T]$ be a temporal interval and let $\Gamma \subset \mathbb{R}^d$, with $d \geq 1$, be a compact set of parameters. For each $\eta \in \Gamma$, we consider the Hamiltonian system, introduced in Chapter 3 for the non-parametric case, described by the initial value problem: For $u_0(\eta) \in \mathcal{V}_{2N}$, find $u(\cdot, \eta) \in C^1(\mathcal{T}, \mathcal{V}_{2N})$ such that

$$\begin{cases} \frac{d}{dt}u(t; \eta) = J_{2N} \nabla_u \mathcal{H}(u(t; \eta); \eta), & \text{for } t \in \mathcal{T}, \\ u(t_0; \eta) = u_0(\eta), \end{cases} \quad (4.1.1)$$

where the dot denotes the derivative with respect to time t , \mathcal{V}_{2N} is a $2N$ -dimensional vector space, and $C^1(\mathcal{T}, \mathcal{V}_{2N})$ denotes continuous differentiable functions in time taking values in \mathcal{V}_{2N} . Moreover, the function $\mathcal{H} : \mathcal{V}_{2N} \times \Gamma \rightarrow \mathbb{R}$ is the Hamiltonian of the system, ∇_u is the gradient with respect to the state variable u , and J_{2N} is the so-called canonical symplectic tensor defined as

$$J_{2N} := \begin{pmatrix} \mathbb{0}_N & \mathbb{I}_N \\ -\mathbb{I}_N & \mathbb{0}_N \end{pmatrix} \in \mathbb{R}^{2N \times 2N}, \quad (4.1.2)$$

¹Professor Pagliantini was responsible for defining the approximating space of reduced dimension and reduced dynamics by structure-preserving projection onto tangent spaces, reported in Sections 4.1 and 4.2, respectively. The author participated in defining the numerical integrator described in Section 4.3 and was the main contributor to the material described in Sections 4.4, 4.5, 4.6, and 4.7., and designed and carried out the numerical experiments reported at the end of the Chapter.

with $\mathbb{I}_N, \mathbb{0}_N \in \mathbb{R}^{N \times N}$ denoting the identity and zero matrices, respectively. The operator J_{2N} identifies a symplectic structure on the phase-space of the Hamiltonian system (4.1.1), as seen in Section 3.1. Equivalently, the vector space \mathcal{V}_{2N} admits a global basis that is symplectic and orthonormal according to the following definition, which is another way of formulating the matrix constraint given in (3.2.3).

Definition 4.1.1 (Orthosymplectic basis). The set of vectors $\{e_i\}_{i=1}^{2N}$ is said to be *orthosymplectic* in the $2N$ -dimensional vector space \mathcal{V}_{2N} if

$$e_i^\top J_{2N} e_j = (J_{2N})_{i,j}, \quad \text{and} \quad (e_i, e_j) = \delta_{i,j}, \quad \forall i, j = 1 \dots, 2N,$$

where (\cdot, \cdot) is the Euclidean inner product and J_{2N} is the canonical symplectic tensor (4.1.2) on \mathcal{V}_{2N} .

4.2 Dynamical reduced basis method for Hamiltonian systems

We are interested in solving the Hamiltonian system (4.1.1) for a given set of p vector-valued parameters $\{\eta_j\}_{j=1}^p \subset \Gamma$, that, with a small abuse of notation, we denote $\eta_h \in \Gamma_h$. Then, the state variable u in (4.1.1) can be thought of as a matrix-valued application $\mathcal{T} \ni t \rightarrow u(t, \cdot) \in \mathcal{V}_{2N}^p$ where $\mathcal{V}_{2N}^p := \mathcal{V}_{2N} \times \dots \times \mathcal{V}_{2N}$. Throughout this Chapter, for a given matrix $\mathcal{R} \in \mathbb{R}^{2N \times p}$, we denote with $\mathcal{R}_j \in \mathbb{R}^{2N}$ the vector corresponding to the j -th column of \mathcal{R} , for any $j = 1, \dots, p$. Let $[a_1|a_2|\dots|a_r]$ denote the matrix of size $2N \times (m_1 + \dots + m_r)$ resulting from the horizontal concatenation of the matrices $a_j \in \mathbb{R}^{2N \times m_j}$ for $j = 1, \dots, r$. The Hamiltonian system (4.1.1), evaluated at η_h , can be recast as a set of ordinary differential equations in a $2N \times p$ matrix unknown in \mathcal{V}_{2N}^p as follows. For $\mathcal{R}_0(\eta_h) := [u_0(\eta_1)|\dots|u_0(\eta_p)] \in \mathcal{V}_{2N}^p$, find $\mathcal{R} \in C^1(\mathcal{T}, \mathcal{V}_{2N}^p)$ such that

$$\begin{cases} \frac{d}{dt} \mathcal{R}(t) = \mathcal{X}_{\mathcal{H}}(\mathcal{R}(t), \eta_h) := J_{2N} \nabla \mathcal{H}(\mathcal{R}(t); \eta_h), & \text{for } t \in \mathcal{T}, \\ \mathcal{R}(t_0) = \mathcal{R}_0(\eta_h), \end{cases} \quad (4.2.1)$$

where $\mathcal{H} : \mathcal{V}_{2N}^p \rightarrow \mathbb{R}^p$ and, for any $\mathcal{R} \in \mathcal{V}_{2N}^p$, its gradient $\nabla \mathcal{H}(\mathcal{R}; \eta_h) \in \mathcal{V}_{2N}^p$ is defined as $(\nabla \mathcal{H}(\mathcal{R}; \eta_h))_{i,j} = \frac{\partial \mathcal{H}_j}{\partial \mathcal{R}_{i,j}}$, for any $i = 1, \dots, 2N$, $j = 1, \dots, p$. The function \mathcal{H}_j is the Hamiltonian of the dynamical system (4.1.1) corresponding to the parameter η_j , for $j = 1, \dots, p$. We assume that, for a fixed sample of parameters $\eta_h \in \Gamma_h$, the vector field $\mathcal{X}_{\mathcal{H}}(\cdot; \eta_h) \in \mathcal{V}_{2N}^p$ is Lipschitz continuous in the Frobenius norm $\|\cdot\|_F$ uniformly with respect to time, so that (4.2.1) is well-posed. We stress how this is a different setting than the one presented in Chapter 3. In this case, the set of parameters for which we want to obtain the solution is fixed a priori.

Let us consider the splitting of the time domain \mathcal{T} into the union of intervals $\mathcal{T}_\tau := (t^{\tau-1}, t^\tau]$, $\tau = 1, \dots, N_\tau$, with $t^0 := t_0$ and $t^{N_\tau} := T$, and let the local time step be defined as $\Delta t_\tau = t^\tau - t^{\tau-1}$ for every τ . For the model order reduction of (4.2.1) we propose an adaptive dynamical scheme based on approximating the full model solution in a lower-dimensional space that is evolving, and whose dimension may also change over time. To this aim, we adopt a local perspective by considering, in each temporal interval, an approximation of the solution of (4.2.1) of the form

$$\mathcal{R}(t) \approx R(t) = \sum_{i=1}^{2n_\tau} U_i(t) Z_i(t, \eta_h) = U(t) Z(t), \quad \forall t \in \mathcal{T}_\tau, \quad (4.2.2)$$

where $U(t) = [U_1 | \dots | U_{2n_\tau}] \in \mathbb{R}^{2N \times 2n_\tau}$, and $Z \in \mathbb{R}^{2n_\tau \times p}$ is such that $Z_{i,j}(t) = Z_i(t, \eta_j)$ for $i = 1, \dots, n_\tau$, $j = 1, \dots, p$, and any $t \in \mathcal{T}_\tau$. Here $n_\tau \in \mathbb{N}$ satisfies $2n_\tau \leq p$ and $n_\tau \ll N$, and is updated over time according to Algorithm 4 that will be discussed in Section 4.5. With this notation, we introduce the collection of reduced spaces of $2N \times p$ matrices having rank at most $2n_\tau$, and characterized as

$$\mathcal{M}_{2n_\tau} := \{R \in \mathbb{R}^{2N \times p} : R = UZ \text{ with } U \in \mathcal{U}_\tau, Z \in \mathcal{Z}_\tau\}, \quad \forall \tau = 1, \dots, N_\tau,$$

where U represents the reduced basis and it is taken to be orthogonal and symplectic, while Z are the expansion coefficients in the reduced basis, i.e.

$$\mathcal{U}_\tau := \{U \in \mathbb{R}^{2N \times p} : U^\top U = \mathbb{I}_{2n_\tau}, U^\top J_{2N} U = J_{2n_\tau}\}, \quad (4.2.3)$$

$$\mathcal{Z}_\tau := \{Z \in \mathbb{R}^{2n_\tau \times p} : \text{rank}(ZZ^\top + J_{2n_\tau}^\top Z Z^\top J_{2n_\tau}) = 2n_\tau\}. \quad (4.2.4)$$

To approximate the Hamiltonian system (4.2.1) in \mathcal{T}_τ with an evolution problem on the reduced space \mathcal{M}_{2n_τ} we need to prescribe evolution equations for the reduced basis $U(t) \in \mathcal{U}_\tau$ and the expansion coefficients $Z(t) \in \mathcal{Z}_\tau$. For this, we follow the approach proposed in [183] and [189], and derive the reduced flow describing the dynamics of the reduced state R in (4.2.2) by applying to the Hamiltonian vector field $\mathcal{X}_\mathcal{H}$ the symplectic projection $\Pi_{T_{R(t)}\mathcal{M}_{2n_\tau}}$ onto the tangent space of the reduced manifold at the current state. The resulting local evolution problem reads: Find $R \in C^1(\mathcal{T}_\tau, \mathcal{M}_{2n_\tau})$ such that

$$\frac{d}{dt}R(t) = \Pi_{T_R\mathcal{M}_{2n_\tau}} \mathcal{X}_\mathcal{H}(R(t), \eta_h), \quad \text{for } t \in \mathcal{T}_\tau, \quad (4.2.5)$$

where we assume, for the time being, that the initial condition of (4.2.5) at time $t^{\tau-1}$, $\tau \geq 1$, is given, and we refer to Section 4.5.3 for a complete description of how such an initial condition is prescribed.

By exploiting the characterization of the projection operator $\Pi_{T_{R(t)}\mathcal{M}_{2n_\tau}}$ in [189, Proposition 4.2], we obtain the local evolution equations for the factors U and Z in the modal decomposition of the reduced solution (4.2.2), as in [183, Proposition 6.9] and [189, Equation (4.10)]. In more details, for any $\tau \geq 1$, given $(U(t^{\tau-1}), Z(t^{\tau-1})) \in \mathcal{U}_\tau \times \mathcal{Z}_\tau$ we seek $(U, Z) \in C^1(\mathcal{T}_\tau, \mathcal{U}_\tau) \times C^1(\mathcal{T}_\tau, \mathcal{Z}_\tau)$ such that

$$\begin{cases} \frac{d}{dt}Z(t) = J_{2n} \nabla_Z \mathcal{H}_U(Z, \eta_h), \\ \frac{d}{dt}U(t) = (\mathbb{I}_{2N} - UU^\top) (J_{2N} Y Z^\top - Y Z^\top J_{2n_\tau}^\top) (Z Z^\top + J_{2n_\tau}^\top Z Z^\top J_{2n_\tau})^{-1}, \end{cases} \quad (4.2.6a)$$

$$\quad (4.2.6b)$$

where $Y(t) := \nabla \mathcal{H}(R(t); \eta_h) \in \mathcal{V}_{2N}^p$, and $R(t) = U(t)Z(t)$ for all $t \in \mathcal{T}_\tau$. Observe that the local expansion coefficients $Z \in \mathcal{Z}_\tau$ satisfy a Hamiltonian system (4.2.6a) of reduced dimension $2n_\tau$, where the reduced Hamiltonian is defined as $\mathcal{H}_U(Z; \eta_h) := \mathcal{H}(UZ; \eta_h)$, similarly to (3.2.9) for the case of a single parameter.

To compute the initial condition of the reduced problem at time t_0 we perform the complex SVD [198, Section 4.2] of $\mathcal{R}_0(\eta_h) \in \mathbb{R}^{2N \times p}$ in (4.2.1), truncated at the n_1 -th mode. Then, the initial reduced basis $U_0 \in \mathcal{U}_1$ can be derived from the unitary matrix of left singular vectors of $\mathcal{R}_0(\eta_h)$, via the isomorphism between \mathcal{U}_1 and the Stiefel manifold of unitary $N \times n_1$ complex matrices. The expansion coefficients matrix is initialized as $Z_0 = U_0^\top \mathcal{R}_0(\eta_h)$. In Figure 4.1, we sketch how the computational costs break down into the offline and online phases for the method considered in comparison with the classical techniques seen in Chapters 1 and 3.

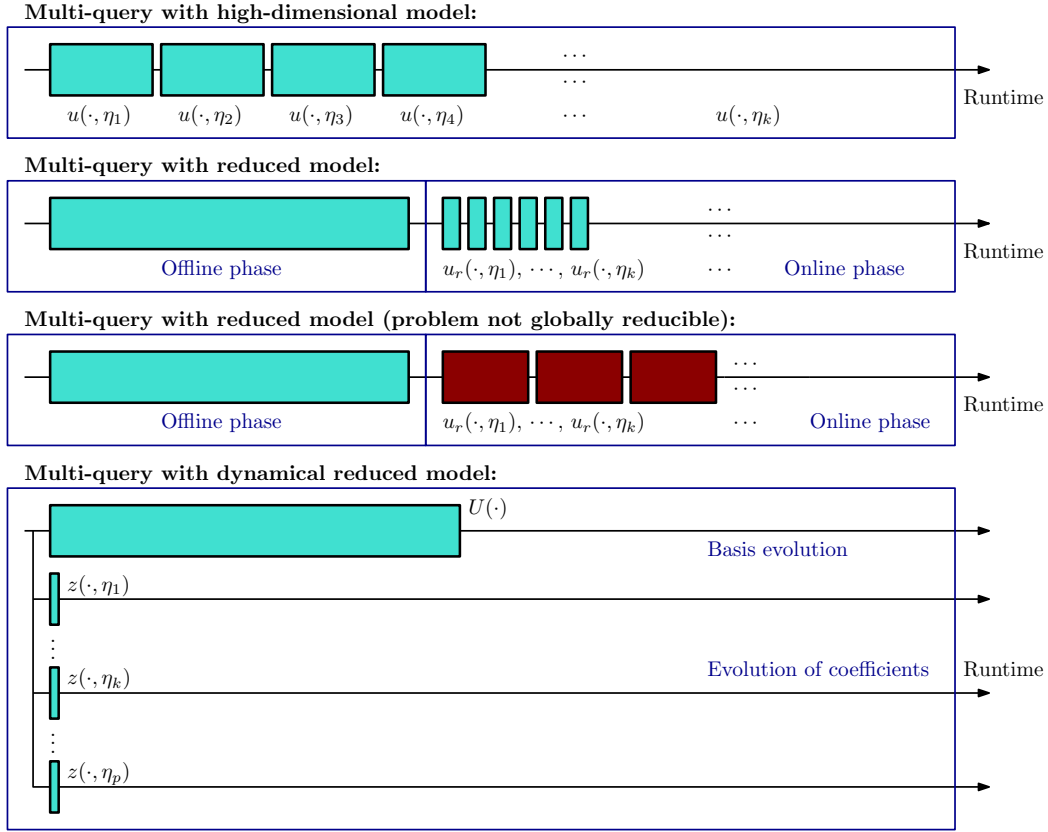


Figure 4.1: Comparison of the distribution of computational costs required to solve parameter problems in a multi-query context. In particular, we report the cases of FOM, gROM, and dynamical low-rank model. We underline how in the case in which the problem is not globally reducible, as we have seen in Chapter 2, the efficiency of the classical reduction methods is compromised.

4.3 Partitioned Runge-Kutta method

For the numerical time integration of the reduced dynamical system (4.2.5) we rely on partitioned Runge–Kutta (RK) methods. Partitioned RK methods were originally introduced to deal with stiff evolution problems by splitting the dynamics into a stiff and a nonstiff part so that the two subsystems could be treated with different temporal integrators. There are many other situations where a dynamical system possesses a natural partitioning, for example Hamiltonian or singularly perturbed problems, or nonlinear systems with a linear part. In our setting, the factorization of the reduced solution (4.2.2) into the basis U and the coefficients Z provides the natural splitting expressed in (4.2.6).

In this section we first consider structure-preserving numerical approximations of the evolution problems (4.2.6b) and (4.2.6a), treated separately. Then, for the numerical integration of the coupled system (4.2.6), we design partitioned RK schemes that are accurate with order 2 and 3 and preserve the geometric structure of each evolution problem.

Since the evolution equation (4.2.6a) is a Hamiltonian system (of reduced dimension) we can rely on symplectic methods for its temporal approximation, so that the symplectic properties of the flow are preserved at the discrete level, as seen in Chapter 3. The evolution equation (4.2.6b) for the reduced basis is approximated using tangent methods that we briefly summarize here. The idea of tangent methods for the solution of differential equations on manifolds, as introduced in [53], is to recast the local dynamics on the tangent space of the manifold, which is a linear space. The temporal approximation of (4.2.6b) by tangent methods allows to obtain, at a computational cost linear in N , a discrete reduced basis that is orthogonal and symplectic. Let $\mathcal{F}(\cdot, \cdot; \eta_h) : \mathbb{R}^{2N \times 2n_\tau} \times \mathcal{Z}_\tau \rightarrow \mathbb{R}^{2N \times 2n_\tau}$ denote the velocity field of the evolution (4.2.6b) of the reduced basis, namely

$$\mathcal{F}(U, Z; \eta_h) := (\mathbb{I}_{2N} - UU^\top) (J_{2N} Y Z^\top - Y Z^\top J_{2n_\tau}^\top) S^{-1}, \quad \forall U \in \mathbb{R}^{2N n_\tau}, Z \in \mathcal{Z}_\tau, \quad (4.3.1)$$

It can be easily shown that, for any $Q \in \mathcal{U}_\tau$, $\mathcal{F}(Q, Z; \eta_h)$ belongs to the space

$$H_Q := \{X \in \mathbb{R}^{2N \times 2n_\tau} : X^\top Q = 0, X J_{2n_\tau} = J_{2n_\tau} X\}. \quad (4.3.2)$$

This is a subspace of the tangent space of the manifold \mathcal{U}_τ of orthosymplectic $2N \times 2n_\tau$ matrices at the point $Q \in \mathcal{U}_\tau$. Let us assume to know, in each temporal interval \mathcal{T}_τ , the approximate solution $Q := U_{\tau-1} \in \mathcal{U}_\tau$ of $U(t^{\tau-1})$. Then, any element of \mathcal{U}_τ , in a neighborhood of Q , can be expressed as the image of a vector $G \in H_Q$ via the retraction

$$\begin{aligned} \mathcal{R}_Q : H_Q &\longrightarrow \mathcal{U}_\tau, \\ G &\longrightarrow \text{cay} (GQ^\top - QG^\top) Q \end{aligned} \quad (4.3.3)$$

where cay is the Cayley transform, defined as $\text{cay}(M) = (\mathbb{I}_N - M/2)^{-1}(\mathbb{I}_N + M/2)$ for any skew-symmetric and Hamiltonian square matrix $M \in \mathbb{R}^{2N \times 2N}$. Since \mathcal{R}_Q is a retraction, rather than solving (4.2.6b) for U , one can derive the local behavior of U in a neighborhood of Q by evolving $G(t)$, with $U(t) = \mathcal{R}_Q(G(t))$, in the space H_Q . By computing the local inverse of the tangent map of the retraction \mathcal{R}_Q , the evolution problem for the vector V reads: for any $t \in \mathcal{T}_\tau$,

$$\frac{d}{dt} G(t) = f_\tau(G(t), Z(t); \eta_h) := -Q (\mathcal{R}_Q(G)^\top Q + \mathbb{I}_{2n_\tau})^{-1} (\mathcal{R}_Q(G) + Q)^\top \Phi + \Phi - Q \Phi^\top Q \quad (4.3.4)$$

where $\Phi := \left(2\mathcal{F}(\mathcal{R}_Q(G), Z; \eta_h) - (GQ^\top - QG^\top)\mathcal{F}(\mathcal{R}_Q(G), Z; \eta_h)\right)(Q^\top \mathcal{R}_Q(G) + \mathbb{I}_{2n_\tau})^{-1}$. We refer to [189, Section 5.3.1] for further details on the derivation of the function f_τ .

The resulting set of evolution equations describes the reduced dynamics in each temporal interval \mathcal{T}_τ as: given $(U_{\tau-1}, Z_{\tau-1}) \in \mathcal{U}_\tau \times \mathcal{Z}_\tau$, find $Z(t) \in \mathcal{Z}_\tau$ and $G(t) \in H_{U_{\tau-1}}$ such that $U(t) = \mathcal{R}_{U_{\tau-1}}(G(t))$ for all $t \in \mathcal{T}_\tau$ and

$$\begin{cases} \frac{d}{dt}Z(t) = \mathcal{G}(\mathcal{R}_{U_{\tau-1}}(G(t)), Z(t); \eta_h), \\ \frac{d}{dt}G(t) = f_\tau(G(t), Z(t); \eta_h), \\ G(t^{\tau-1}) = 0 \in H_{U_{\tau-1}}, \\ Z(t^{\tau-1}) = Z_{\tau-1} \in \mathcal{Z}_\tau, \end{cases} \quad (4.3.5)$$

where $\mathcal{G} := J_{2n} \nabla \mathcal{H}_U(Z, \eta_h)$ from (4.2.6a) and f_τ is defined in (4.3.4).

For the numerical approximation of (4.3.5), we derive partitioned Runge–Kutta methods. Let $P_Z = (\{b_i\}_{i=1}^s, \{a_{ij}\}_{i,j=1}^s)$ be the collection of coefficients of the Butcher tableau describing an s -stage *symplectic* RK method, and let $\hat{P}_U = (\{\hat{b}_i\}_{i=1}^s, \{\hat{a}_{ij}\}_{1 \leq j < i \leq s})$ be the set of coefficients of an s -stage *explicit* RK method. Then, the numerical approximation of (4.3.5) via partitioned RK integrators reads

$$\begin{aligned} Z_\tau &= Z_{\tau-1} + \Delta t \sum_{i=1}^s b_i k_i, & G_\tau &= \Delta t \sum_{i=1}^s \hat{b}_i \hat{k}_i, \\ k_1 &= \mathcal{G}\left(U_{\tau-1}, Z_{\tau-1} + \Delta t \sum_{j=1}^s a_{1,j} k_j; \eta_h\right), & \hat{k}_1 &= f_\tau\left(U_{\tau-1}, Z_{\tau-1} + \Delta t \sum_{j=1}^s a_{1,j} k_j; \eta_h\right) \\ k_i &= \mathcal{G}\left(\mathcal{R}_{U_{\tau-1}}\left(\Delta t \sum_{j=1}^{i-1} \hat{a}_{i,j} \hat{k}_j\right), Z_{\tau-1} + \Delta t \sum_{j=1}^s a_{i,j} k_j; \eta_h\right), & i &= 2, \dots, s, \\ \hat{k}_i &= f_\tau\left(\Delta t \sum_{j=1}^{i-1} \hat{a}_{i,j} \hat{k}_j, Z_{\tau-1} + \Delta t \sum_{j=1}^s a_{i,j} k_j; \eta_h\right), & i &= 2, \dots, s, \\ U_\tau &= \mathcal{R}_{U_{\tau-1}}(G_\tau). \end{aligned} \quad (4.3.6)$$

Runge–Kutta methods of order 2 and 3 with the aforementioned properties can be characterized in terms of the coefficients P_Z and \hat{P}_U as in the following result.

Lemma 4.3.1. *Consider the numerical approximation of (4.3.5) with the s -stage partitioned Runge–Kutta method (4.3.6) obtained by coupling the Runge–Kutta methods $P_Z = (\{b_i\}_{i=1}^s, \{a_{ij}\}_{i,j=1}^s)$ and $\hat{P}_U = (\{\hat{b}_i\}_{i=1}^s, \{\hat{a}_{ij}\}_{1 \leq j < i \leq s})$. Then, the following statements hold.*

- Symplectic condition [120, Theorem VI.4.3]. The Runge–Kutta method P_Z is symplectic if

$$b_i a_{ij} + b_j a_{ji} = b_i b_j, \quad \forall i, j = 1, \dots, s. \quad (4.3.7)$$

- Order condition [119, Theorem II.2.13]. The Runge–Kutta method P_Z has order k , with

$$k = 2 \quad \text{iff} \quad \sum_{i=1}^s b_i = 1, \quad \sum_{i,j=1}^s b_i a_{ij} = \frac{1}{2}; \quad (4.3.8)$$

$$k = 3 \quad \text{iff} \quad \sum_{i=1}^s b_i = 1, \quad \sum_{i,j=1}^s b_i a_{ij} = \frac{1}{2}, \quad \sum_{i=1}^s b_i \left(\sum_{j=1}^s a_{ij} \right)^2 = \frac{1}{3}, \quad \sum_{i,j,l=1}^s b_i a_{ij} a_{jl} = \frac{1}{6}. \quad (4.3.9)$$

- Coupling condition [120, Section III.2.2]. The partitioned Runge–Kutta method (P_Z, \hat{P}_U)

has order p , if P_Z and \hat{P}_U are both of order k and

$$k = 2 \quad \text{if} \quad \sum_{i=1}^s \sum_{j=1}^{i-1} b_i \hat{a}_{ij} = \frac{1}{2}, \quad \sum_{i=1}^s \sum_{j=1}^s \hat{b}_i a_{ij} = \frac{1}{2}; \quad (4.3.10)$$

$$k = 3 \quad \text{if} \quad \sum_{i=1}^s a_{ij} = \sum_{i=1}^{j-1} \hat{a}_{ij}, \quad \sum_{i,l=1}^s \sum_{j=1}^{i-1} b_i \hat{a}_{ij} a_{jl} = \frac{1}{6}, \quad \sum_{i,j,l=1}^s \hat{b}_i a_{ij} a_{jl} = \frac{1}{6}. \quad (4.3.11)$$

4.4 Reduced dynamics under rank-deficiency

In Section 4.2 we have proposed to approximate the phase space of the full Hamiltonian system (4.2.1) by an evolving low-rank matrix manifold. Particular attention needs to be devoted to the case of *overapproximation* in which a FOM solution with effective rank $r < n$ is approximated by a rank- n matrix, as pointed out first in [146, Section 5.3]. In this case, a rank-deficient reduced dynamical system needs to be solved and it is not clear how the effective rank of the reduced solution will evolve over time. Indeed, in each temporal interval \mathcal{T}_τ , the dynamics may not remain on the reduced manifold \mathcal{M}_{2n_τ} and the matrix $S(Z) := ZZ^\top + J_{2n_\tau}^\top ZZ^\top J_{2n_\tau}$ may become singular or severely ill conditioned. This happens, for example, when the full model state at time t_0 is approximated with a rank deficient matrix, or, as we will see in the rank-adaptive algorithm in Section 4.5, when the reduced solution at a fixed time is used as initial condition to evolve the reduced system on a manifold of states with increased rank.

In this section, we propose an algorithm to deal with the overapproximation while maintaining the geometric structure of the Hamiltonian dynamics and of the factors U and Z in (4.2.2).

Lemma 4.4.1 (Characterization of the matrix S). *Let $S := ZZ^\top + J_{2n}^\top ZZ^\top J_{2n} \in \mathbb{R}^{2n \times 2n}$ with $Z \in \mathbb{R}^{2n \times p}$ and $p \geq 2n$. S is symmetric positive semi-definite and it is skew-Hamiltonian, namely $SJ_{2n} - J_{2n}S^\top = 0$. Moreover, if S has rank $2n$ then S is non-singular and S^{-1} is also skew-Hamiltonian.*

In particular, the null space of S is even dimensional and contains all pairs of vectors $(v, J_{2n}v) \in \mathbb{R}^{2n} \times \mathbb{R}^{2n}$ such that both v and $J_{2n}v$ belong to the null space of Z^\top .

Proof. It can be easily verified that S is symmetric positive semi-definite and skew-Hamiltonian. Any eigenvalue of a skew-Hamiltonian matrix has even multiplicity, hence the null space of S has even dimension. Since S is positive semi-definite, $v \in \ker S$ if and only if $ZZ^\top v = 0$ and $ZZ^\top J_{2n}v = 0$, that is $\ker S = \ker Z^\top \cap \ker Z^\top J_{2n}$. Observe that all the elements v of the kernel of Z^\top are such that $J_{2n}^\top v \in \ker Z^\top J_{2n}$. \square

In addition to the algebraic limitations associated with the solution of a rank-deficient system, the fact that the matrix S might be singular or ill conditioned prevents the reduced basis from evolving on the manifold of the orthosymplectic matrices. If $U(t^{\tau-1}) \in \mathcal{U}_\tau$ then $U(t) \in \mathbb{R}^{2N \times 2n_\tau}$ solution of (4.2.6b) in \mathcal{T}_τ satisfies $U(t) \in \mathcal{U}_\tau$ for all $t \in \mathcal{T}_\tau$, owing to the fact that $\mathcal{F}(U, Z; \eta_h)$ belongs to the space H_U in (4.3.2).

Lemma 4.4.2. *The function $\mathcal{F}(\cdot, \cdot; \eta_h) : \mathbb{R}^{2N \times 2n_\tau} \times \mathcal{Z}_\tau \rightarrow \mathbb{R}^{2N \times 2n_\tau}$ defined in (4.3.1) is such that $\mathcal{F}(U, Z; \eta_h) \in H_U$ if and only if $U \in \mathcal{U}_\tau$ and $Z \in \mathcal{Z}_\tau$.*

Proof. Let $X_U := \mathcal{F}(U, Z; \eta_h) = (\mathbb{I}_{2N} - UU^\top)AS^{-1}$, where $A := J_{2N}YZ^\top - YZ^\top J_{2n_\tau}^\top$. The condition $X_U^\top U = 0$ is satisfied for every $U \in \mathbb{R}^{2N \times 2n_\tau}$ orthogonal and $Z \in \mathbb{R}^{2n_\tau \times p}$. Concerning the second condition, it can be easily shown that $J_{2N}AJ_{2n_\tau}^\top = A$ and $J_{2N}(\mathbb{I}_{2N} - UU^\top) = (\mathbb{I}_{2N} - UU^\top)J_{2N}$. Hence, $J_{2N}X_U = (\mathbb{I}_{2N} - UU^\top)AJ_{2n_\tau}S^{-1}$ and this is equal to $X_U J_{2n_\tau}$ if and only if $J_{2n_\tau}S^{-1} = S^{-1}J_{2n_\tau}$. This condition follows from Lemma 4.4.1. \square

Lemma 4.4.2 can be equivalently stated by considering the velocity field \mathcal{F} as a function of the triple $(U, Z, S(Z))$. Then $\mathcal{F}(U, Z, S(Z); \eta_h)$ belongs to H_U if and only if $U \in \mathcal{U}_\tau$, $Z \in \mathbb{R}^{2n_\tau \times p}$ and $S(Z)$ is non-singular, symmetric and skew-Hamiltonian. If the matrix S is not invertible, i.e. $Z \notin \mathcal{Z}_\tau$, its inverse needs to be replaced by some approximation S^\dagger . By Lemma 4.4.2, if S^\dagger is not symmetric skew-Hamiltonian, then $\mathcal{F}^\dagger(U, Z; \eta_h) := (\mathbb{I}_{2N} - UU^\top)AS^\dagger$ does no longer belong to the horizontal space H_U . If, for example, S^\dagger is the pseudo inverse of S , then the above condition is theoretically satisfied, but in numerical computations only up to a small error, because, if S is rank-deficient, then its pseudoinverse corresponds to the pseudoinverse of the truncated SVD of S .

To overcome these issues in the numerical solution of the reduced dynamics (4.2.6), we introduce two approximations: first we replace the rank-deficient matrix S with an ϵ -regularization that preserves the skew-Hamiltonian structure of S and then, in finite precision arithmetic, we set as velocity field for the evolution of the reduced basis U an approximation of \mathcal{F} in the space $H_{U(t)}$, for all $t \in \mathcal{T}_\tau$. The ϵ -regularization consists in diagonalizing S and then replacing, in the resulting diagonal factor, the elements below a certain threshold with a fixed factor $\epsilon \in \mathbb{R}$. This is possible since (real) symmetric matrices are always diagonalizable by orthogonal transformations. However, unitary transformations do not preserve the skew-Hamiltonian structure. We therefore consider the following Paige Van Loan (PVL) decomposition, based on symplectic equivalence transformations.

Lemma 4.4.3 ([254]). *Given a skew-Hamiltonian matrix $S \in \mathbb{R}^{2n \times 2n}$ there exists a symplectic orthogonal matrix $W \in \mathbb{R}^{2n \times 2n}$ such that $W^\top SW$ has the PVL form*

$$W^\top SW = \begin{bmatrix} S_n & R \\ & S_n^\top \end{bmatrix} \quad (4.4.1)$$

where $S_n \in \mathbb{R}^{n \times n}$ is an upper Hessenberg matrix.

In our case, since the matrix S is symmetric, its PVL decomposition (4.4.1) yields tridiagonal matrices with identical blocks $S_{n_\tau} = S_{n_\tau}^\top$. We further diagonalize S_{n_τ} using orthogonal transformations to obtain $S_{n_\tau} = T^\top D_{n_\tau} T$, with $T^\top T = \mathbb{I}_{n_\tau}$ and diagonal $D_{n_\tau} \in \mathbb{R}^{n_\tau \times n_\tau}$. Hence,

$$S = W \begin{bmatrix} T^\top D_{n_\tau} T & \\ & T^\top D_{n_\tau} T \end{bmatrix} W^\top =: Q D Q^\top$$

with

$$Q := W \begin{bmatrix} T^\top & \\ & T^\top w \end{bmatrix}, \quad D := \begin{bmatrix} D_{n_\tau} & \\ & D_{n_\tau} \end{bmatrix}$$

It can be easily verified that $Q \in \mathbb{R}^{2n_\tau \times 2n_\tau}$ is orthogonal and symplectic. The PVL factorization 4.4.3 can be implemented as in, e.g., [25, Algorithms 1 and 2], with arithmetic complexity $O(n_\tau^3)$. The factorization is based on orthogonal symplectic transformations obtained from Givens rotations and symplectic Householder matrices, defined as the direct sum of Householder reflections

[190]. Once the matrix S has been brought in the PVL form, we perform the ϵ -regularization. Introduce the diagonal matrix $D_{n_\tau, \epsilon} \in \mathbb{R}^{n_\tau \times n_\tau}$ defined as,

$$(D_{n_\tau, \epsilon})_i = \begin{cases} (D_{n_\tau})_i & \text{if } (D_{n_\tau})_i > \epsilon, \\ \epsilon & \text{otherwise,} \end{cases} \quad \forall 1 \leq i \leq n_\tau,$$

and let us denote with $D_\epsilon \in \mathbb{R}^{2n_\tau \times 2n_\tau}$ the diagonal matrix composed of two blocks, both equal to $D_{n_\tau, \epsilon}$. The matrix $S_\epsilon := QD_\epsilon Q^\top \in \mathbb{R}^{2n_\tau \times 2n_\tau}$ is symmetric positive definite and skew-Hamiltonian. Its distance to S is bounded, in the Frobenius norm, as $\|S - S_\epsilon\| = \|Q(D - D_\epsilon)Q^\top\| = \|D - D_\epsilon\| \leq \sqrt{m_\epsilon} \epsilon$, where m_ϵ is the number of elements of D_{n_τ} that are smaller than ϵ . Since the ϵ -regularized matrix S_ϵ is invertible, S_ϵ^{-1} exists and is skew-Hamiltonian. This property allows to construct the vector field $\mathcal{F}_\epsilon := (\mathbb{I}_{2N} - UU^\top)(J_{2N}YZ^\top - YZ^\top J_{2n_\tau}^\top)S_\epsilon^{-1} \in \mathbb{R}^{2N \times 2n_\tau}$ with the property that \mathcal{F}_ϵ belongs to the tangent space of the orthosymplectic $2N \times 2n_\tau$ matrix manifold. To gauge the error introduced by approximating the velocity field \mathcal{F} in (4.3.1) with \mathcal{F}_ϵ , let us denote with \mathcal{L} the operator $\mathcal{L} := (\mathbb{I}_{2N} - UU^\top)(J_{2N}YZ^\top - YZ^\top J_{2n_\tau}^\top)$, so that (4.2.6b) reads $\dot{U}S = \mathcal{L}$. Then, the error made in the evolution of the reduced basis (4.2.6b), by the ϵ -regularization, is

$$\begin{aligned} \|\mathcal{F}_\epsilon - \mathcal{L}\| &= \|\mathcal{L}(S_\epsilon^{-1}S - \mathbb{I}_{2n_\tau})\| = \|\mathcal{L}Q(D_\epsilon^{-1}D - \mathbb{I}_{2n_\tau}Q^\top)\| \\ &\leq \|\mathcal{L}\| \|D_\epsilon^{-1}D - \mathbb{I}_{2n_\tau}\| = \frac{\sqrt{2}}{\epsilon} \|\mathcal{L}\| \sqrt{\sum_{j=n_\tau-m_\epsilon+1}^{n_\tau} |D_j - \epsilon|^2}. \end{aligned}$$

Observe that the resulting vector field \mathcal{F}_ϵ belongs to the space H_U by construction. However, in finite precision arithmetic, the distance of the computed \mathcal{F}_ϵ from H_U might be affected by a small error that depends on the norm of the operators \mathcal{L} and S_ϵ . This rounding error can affect the symplecticity of the reduced basis over time, whenever the matrix S is severely ill conditioned. To guarantee that the evolution of the reduced basis computed in finite precision remains on the manifold of orthosymplectic matrices with an error of the order of machine precision, we introduce a correction of the velocity field \mathcal{F}_ϵ . Observe that any $X_U \in H_U$ is of the form $X_U = [F|J_{2N}^\top F]$, with $F \in \mathbb{R}^{2N \times n_\tau}$ satisfying $U^\top F = 0_{2n_\tau \times n_\tau}$. Let us write \mathcal{F}_ϵ as $\mathcal{F}_\epsilon = [F|G]$, with $F^\top = [F_1^\top | F_2^\top] \in \mathbb{R}^{n_\tau \times 2N}$ and $G^\top = [G_1^\top | G_2^\top] \in \mathbb{R}^{n_\tau \times 2N}$. Since $U^\top \mathcal{F}_\epsilon = [U^\top F | U^\top G] = 0_{2n_\tau \times 2n_\tau}$, we can take $\mathcal{F}_{\epsilon, \star} := [F|J_{2N}^\top F]$. Alternatively, we can define $\mathcal{F}_{\epsilon, \star} := [W|J_{2N}^\top W]$ where $W^\top = [X^\top | -Y^\top] \in \mathbb{R}^{n_\tau \times 2N}$ and $2X := F_1 + G_2$, $2Y := G_1 - F_2$. It easily follows that, with either definitions, $\mathcal{F}_{\epsilon, \star}$ belongs to H_U and the error in the Frobenius norm is

$$\|\mathcal{F}_\epsilon - \mathcal{F}_{\epsilon, \star}\| = \frac{1}{4} \|\mathcal{F}_\epsilon J_{2n_\tau} - J_{2N} \mathcal{F}_\epsilon\|^2 = \|G - J_{2N}^\top F\|^2$$

We summarize the regularization scheme in Algorithm 3.

4.5 Rank-adaptivity

The dynamical reduced basis method that we have introduced in Section 4.2 is based on approximating the full model solution, in each temporal interval \mathcal{T}_τ , on a low-dimensional space of size n_τ . The fact that the size of the reduced space can change over time allows to fully exploit the local low-rank nature of the solution. In this Section, we propose an algorithm to detect when the reduced space needs to be enlarged or reduced and how this operation is performed. The

Algorithm 3 ϵ -regularization

```

1: procedure REGULARIZATION( $U \in \mathcal{U}, Z \in \mathbb{R}^{2n_\tau}, \epsilon$ )
2:   Compute  $S \leftarrow ZZ^\top + J_{2n_\tau}^\top ZZ^\top J_{2n_\tau}$ 
3:   if  $\text{rank}(S) < 2n_\tau$  then
4:     Compute the PVL factorization  $QDQ^\top = S$ 
5:     Set  $S_\epsilon \leftarrow QD_\epsilon Q^\top$  where  $D_\epsilon$  is the  $\epsilon$ -regularization of  $D$ 
6:     Compute  $\mathcal{F}_\epsilon \leftarrow (\mathbb{I}_{2N} - UU^\top) (J_{2N}YZ^\top - YZ^\top J_{2n_\tau}^{-1}) S_\epsilon^{-1}$ 
7:     Compute  $\mathcal{F}_{\epsilon,*}$  by enforcing the skew-Hamiltonian constraint
8:     Set  $\mathcal{F} \leftarrow \mathcal{F}_{\epsilon,*}$ 
9:   else
10:    Compute  $\mathcal{F} \leftarrow (\mathbb{I}_{2N} - UU^\top) (J_{2N}YZ^\top - YZ^\top J_{2n_\tau}^{-1}) S^{-1}$ 
11:  return velocity field  $\mathcal{F} \in H_U$ 
    
```

method is summarized in Algorithm 4.

Here we focus on the case where the current rank of the reduced solution is too small to accurately reproduce the full model solution. In cases where the rank is too large, one can perform an ϵ -regularization following Algorithm 3 or decrease the rank by looking at the spectrum of the reduced state and remove the modes associated with the lowest singular values.

4.5.1 Error indicator

As stated in Chapter 1, error bounds for parabolic problems are long-established and have been widely used to certify global reduced basis methods. However, their extension to noncoercive problems often results in pessimistic bounds that cannot be used to properly assess the quality of the reduced approximation. Few works have focused on the development of error estimates (not bounds) for reduced solutions of advection-dominated problems. In this work, we propose an error indicator based on the linearized residual of the full model. A related approach, known as Dual-Weighted Residual method (DWR) [175], consists in deriving an estimate of the approximation error via the dual full model and the linearization of the error of a certain functional of interest (e.g. surface integral of the solution, stress, displacement, ...). Despite the promising results of this approach, the arbitrariness in the choice of the functional clashes with the goal of having a procedure as general as possible.

We begin with the continuous full model (4.2.1) and, for its time integration, we consider the implicit RK scheme used in the temporal discretization of the dynamical system for the expansion coefficients Z in (4.3.6), and having coefficients $(\{b_i\}_{i=1}^s, \{a_{ij}\}_{i,j=1}^s)$. Then, assuming that $\mathcal{R}_{\tau-1} \in \mathbb{R}^{2N \times p}$ is known,

$$\begin{aligned}
 \mathcal{R}_\tau &= \mathcal{R}_{\tau-1} + \Delta t \sum_{i=1}^s b_i k_i, \\
 k_1 &= J_{2N} \nabla_{\mathcal{R}} \mathcal{H}(\mathcal{R}_{\tau-1}), \\
 k_i &= J_{2N} \nabla_{\mathcal{R}} \mathcal{H} \left(\mathcal{R}_{\tau-1} + \sum_{j=1}^s a_{i,j} k_j; \eta_h \right), \quad i = 2, \dots, s
 \end{aligned} \tag{4.5.1}$$

The discrete residual operator, in the temporal interval \mathcal{T}_τ , is

$$\rho_\tau(\mathcal{R}_\tau, \mathcal{R}_{\tau-1}; \eta_h) = \mathcal{R}_\tau - \mathcal{R}_{\tau-1} - \Delta t \sum_{i=1}^s b_i k_i = 0. \quad (4.5.2)$$

We consider the linearization of the residual operator (4.5.2) at $(R_\tau, R_{\tau-1})$, where R_τ is the approximate reduced solution at time t^τ , obtained from (4.3.6) as $R_\tau = U_\tau Z_\tau$; thereby

$$\begin{aligned} \rho_\tau(\mathcal{R}_\tau, \mathcal{R}_{\tau-1}; \eta_h) &= \rho_\tau(R_\tau, R_{\tau-1}; \eta_h) + \left. \frac{\partial \rho_\tau}{\partial \mathcal{R}_\tau} \right|_{(R_\tau, R_{\tau-1})} (\mathcal{R}_\tau - R_\tau) \\ &\quad + \left. \frac{\partial \rho_\tau}{\partial \mathcal{R}_{\tau-1}} \right|_{(R_\tau, R_{\tau-1})} (\mathcal{R}_{\tau-1} - R_{\tau-1}) + \mathcal{O}(\|\mathcal{R}_\tau - R_\tau\|^2 + \|\mathcal{R}_{\tau-1} - R_{\tau-1}\|^2). \end{aligned} \quad (4.5.3)$$

Similar procedures have been adopted in the formulation of the piecewise linear methods for the approximation of nonlinear operators, providing accurate approximations in case of low-order nonlinearities. From the residual operator, an approximation of the local error $\mathcal{R}_\tau - R_\tau$ is given by the matrix-valued quantity \mathbf{E}_τ defined as

$$\mathbf{E}_\tau := - \left(\left. \frac{\partial \rho_\tau}{\partial \mathcal{R}_\tau} \right|_{(R_\tau, R_{\tau-1})} \right)^{-1} \left(\rho_\tau(R_\tau, R_{\tau-1}; \eta_h) + \left. \frac{\partial \rho_\tau}{\partial \mathcal{R}_{\tau-1}} \right|_{(R_\tau, R_{\tau-1})} \mathbf{E}_{\tau-1} \right), \quad (4.5.4)$$

with $\mathbf{E}_0 := \mathcal{R}(t_0) - U_0 Z_0$. The quantity defined by (4.5.4) is the first order approximation of the error between the reduced and the full model solution. In particular, it quantifies the discrepancy due to the local approximation (4.2.2). Even if the linearization error is negligible, the computational cost related to the assembly of the entire full-order residual ρ and its Jacobian, together with the solution of a linear system for any instance of the p parameters η_h , makes the indicator unappealing if used in the context of highly efficient reduced approximations. In [175], a hierarchical approach has been proposed to alleviate the aforementioned computational bottleneck but it relies on the offline phase to capture the dominant modes of the exact error. Instead, in this work, we solve (4.5.4) on a subset $\widetilde{\eta}_h$ of the p vector-valued parameters η_h of cardinality $\widetilde{p} \ll p$, and only each $N_{\mathbf{E}}$ time steps during the simulation. To further reduce the computational cost, we compute (4.5.4) on a coarse mesh in the parameter domain, whenever possible, and then \mathbf{E}_τ is recovered on the original mesh via spline interpolation. Although the assembly and solution of the sparse linear system in (4.5.4) has, for example, arithmetic complexity $\mathcal{O}(N^{\frac{3}{2}})$ [96] for problems originating from the discretization of two-dimensional PDEs, this sampling strategy allows to reduce the computational cost required by the error estimator as compared to the evolution of the reduced basis and the coefficients, as discussed in Section 4.7.

4.5.2 Criterion for rank update

Let $\mathbf{E}_\tau \in \mathbb{R}^{2N \times p}$ be the error indicator matrix obtained in (4.5.4). To decide when to activate the rank update algorithm, we take into account that, for advection-dominated and hyperbolic problems discretized using spectral methods, the error accumulates, and the effect of unresolved modes on the resolved dynamic contributes to this accumulation [70]. Moreover, it has been noticed [236] that, for many problems of practical interest, the modes associated with initially negligible singular values might become relevant over time, potentially causing a loss of accuracy

if a reduced manifold of fixed dimension is employed.

Let us define t^τ as the current time, t^* as the last time at which the dimension of the reduced basis U was updated and let λ_τ be the number of past updates at time t^τ . At the beginning of the simulation $t^* = t^0$ and $\lambda_0 = 0$. The rank update is performed if the ratio between the norms of error indicators at t^τ and t^* satisfies the criterion

$$\frac{\|\mathbf{E}_\tau\|}{\|\mathbf{E}_*\|} > rc^{\lambda_\tau}, \quad (4.5.5)$$

where $r, c \in \mathbb{R}$ are control parameters larger than 1. The ratio of the norms of the error indicator gives a qualitative indication of how the error is increasing in time and (4.5.5) fixes a maximum acceptable growing slope. Deciding what represents an acceptable slope is a problem-dependent task but the numerical results in Section 4.7 show little sensitivity of the algorithm with respect to r and c . Moreover, the variable λ_τ induces a frequent rank-update when n_τ is small and vice versa when n_τ is large, hence controlling both the efficiency and the accuracy of the updating algorithm. Note that other (combinations of) criteria are possible: one alternative is to check that the norm of the error indicator remains below a fixed threshold; another possibility is to control the norm of some approximate gradient of the error indicator, etc. By numerically testing these various criteria, it has been observed that, at least in the numerical simulations performed, the criterion (4.5.5) based on the ratio of error indicators is reliable and robust and gives the largest flexibility.

4.5.3 Update of the reduced state

If criterion (4.5.5) is satisfied, the rank adaptive algorithm updates the current reduced solution to a new state having a different rank. Specifically, assume that, in the time interval $\mathcal{T}_{\tau-1}$, we have solved the discrete reduced problem (4.3.6) to obtain the reduced solution $R_{\tau-1} = U_{\tau-1}Z_{\tau-1}$ in $\mathcal{M}_{n_{\tau-1}}$.

As a first step, we derive an updated basis $U \in \mathcal{U}_\tau$ from $U_{\tau-1} \in \mathcal{U}_{\tau-1}$, with $n_\tau = n_{\tau-1} + 1$. To this aim, we enlarge $U_{\tau-1}$ with two extra columns derived from an approximation of the error, analogously to a greedy strategy. In greater detail, with the algorithm described in Section 4.5.1, we derive the error matrix \mathbf{E}_τ associated with the reduced solution at the current time. Via a thin SVD, we extract the left singular vector associated with the principal component of the error matrix, and we normalize it in the 2-norm to obtain the vector $e \in \mathbb{R}^{2N}$. We finally enlarge the basis $U_{\tau-1}$ with the two columns $[e \mid J_{2N}^\top e] \in \mathbb{R}^{2N \times 2}$. The rationale for this choice is that we seek to increase the accuracy of the low-rank approximation by adding to the reduced basis the direction that is worst approximated by the current reduced space.

From the updated matrix $[U_{\tau-1} \mid e \mid J_{2N}^\top e] \in \mathbb{R}^{2N \times 2n_\tau}$, we construct an orthosymplectic basis in the sense of Definition 4.1.1, by performing a QR-like decomposition using symplectic unitary transformations. In particular, we employ a symplectic (modified) Gram-Schmidt algorithm, similar to the one employed in the symplectic greedy method discussed in Section 3.3.3, with the possibility of adding reorthogonalization to enhance the stability and robustness of the algorithm. Once the updated reduced basis $U \in \mathcal{U}_\tau$ is computed, we derive the matrix $Z \in \mathbb{R}^{2n_\tau \times p}$ by expanding the current reduced solution $R_{\tau-1}$ in the updated basis. Therefore, the updated Z satisfies $UZ = R_{\tau-1}$, which results in $Z = U^\top R_{\tau-1}$.

Remark 4.5.1. Since the updated reduced state coincides with the reduced solution $R_{\tau-1}$ at time $t^{\tau-1}$, all invariants of (4.2.1) preserved by the partitioned Runge–Kutta scheme (4.3.6) are

conserved during the rank update.

Observe that, even if the current reduced state $R_{\tau-1}$ is in $\mathcal{M}_{2n_\tau-2}$, it does not belong to the manifold \mathcal{M}_{2n_τ} . Indeed, one easily shows that $Z = U^\top R_{\tau-1} \in \mathbb{R}^{2n_\tau \times p}$ does not satisfy the full-rank condition,

$$\begin{aligned} \text{rank}(S(Z)) &= \text{rank}(U^\top U_{\tau-1} [Z_{\tau-1} Z_{\tau-1}^\top + J_{2n_\tau}^\top Z_{\tau-1} Z_{\tau-1}^\top J_{2n_\tau}] U_{\tau-1}^\top U) \\ &\leq \min \{ \text{rank}(U^\top U_{\tau-1}), \text{rank}(Z_{\tau-1} Z_{\tau-1}^\top + J_{2n_\tau}^\top Z_{\tau-1} Z_{\tau-1}^\top J_{2n_\tau}) \} \\ &\leq 2n_\tau - 2 \end{aligned}$$

As shown in Lemma 4.4.2, the fact that $Z \notin \mathcal{Z}_\tau$ implies that the velocity field \mathcal{F} in (4.3.1), describing the evolution of the reduced basis, is not well-defined. Therefore, we need to introduce an approximate velocity field for the solution of the reduced problem (4.2.6) in the temporal interval \mathcal{T}_τ with initial conditions $(U, Z) \in \mathcal{U}_\tau \times \mathbb{R}^{2n_\tau \times p}$. We refer to Section 4.4 for a discussion about this issue and the description of the algorithm designed to solve the rank-deficient reduced dynamics ensuing from the rank update.

Algorithm 4 Rank update

- 1: **procedure** RANK_UPDATE($U_{\tau-1}, Z_{\tau-1}, \mathbf{E}_*$)
 - 2: Compute the error indicator matrix $\mathbf{E}_{\tau-1} \in \mathbb{R}^{2N \times p}$ (4.5.4)
 - 3: **if** criterion (4.5.5) is satisfied **then**
 - 4: Compute $Q\Sigma V^\top = \mathbf{E}_{\tau-1}$ via thin SVD
 - 5: Set $e \leftarrow Q_1 / \|Q_1\|_2$ where $Q_1 \in \mathbb{R}^{2N}$ is the first column of the matrix Q
 - 6: Construct the enlarged basis $\bar{U} \leftarrow [U_{\tau-1} | e J_{2N}^\top] \in \mathbb{R}^{2N \times (2n_\tau+2)}$
 - 7: Compute U via symplectic orthogonalization of \bar{U} with symplectic Gram-Schmidt
 - 8: Compute the coefficients $Z \leftarrow U^\top U_{\tau-1} Z_{\tau-1}$
 - 9: **else**
 - 10: $U \leftarrow U_{\tau-1}, Z \leftarrow Z_{\tau-1}$ and $n_\tau = n_{\tau-1}$
 - 11: **return** updated factors $(U, Z) \in \mathcal{U} \times \mathbb{R}^{2n_\tau \times p}$
-

4.5.4 Approximation properties of the rank-adaptive scheme

To gauge the local approximation properties of the rank-adaptive scheme for the solution of the reduced dynamical system (4.2.6), we consider the temporal interval \mathcal{T}_τ where the first rank update is performed. In other words, assume that $R_{\tau-1} = U_{\tau-1} Z_{\tau-1}$, with $(U_{\tau-1}, Z_{\tau-1}) \in \mathcal{U}_{\tau-1} \times \mathcal{Z}_{\tau-1}$, is the numerical approximation of the solution $R(t^{\tau-1}) \in \mathcal{M}_{2n_{\tau-1}}$ of the reduced dynamical system (4.2.5) at time $t^{\tau-1}$ with $n_{\tau-1} = n_{\tau-2} = \dots = n_1$. After the rank update at time $t^{\tau-1}$, the reduced state R satisfies the local evolution problem

$$\begin{cases} \frac{d}{dt} R(t) = \mathcal{P}_R^\epsilon \mathcal{X}_\mathcal{H}(R(t), \eta_h), & \text{for } t \in \mathcal{T}_\tau, \\ R(t^{\tau-1}) = R_{\tau-1} = U_{\tau-1}^{n_\tau} Z_{\tau-1}^{n_\tau} \end{cases} \quad (4.5.6)$$

where $(U_{\tau-1}^{n_\tau}, Z_{\tau-1}^{n_\tau}) \in \mathcal{U}_\tau \times \mathbb{R}^{2n_\tau \times p}$ are the rank-updated factors, and

$$\mathcal{P}_R^\epsilon \mathcal{X}_\mathcal{H}(R(t), \eta_h) := (\mathbb{I}_{2N} - UU^\top)(\mathcal{X}_\mathcal{H} Z^\top J_{2N} \mathcal{X}_\mathcal{H} Z^\top J_{2n_\tau}^\top) S_\epsilon(Z)^{-1} Z + UU^\top \mathcal{X}_\mathcal{H},$$

for all $R = UZ \in \mathbb{R}^{2N \times p}$. We make the assumption that the reduced problem (4.2.5) is well-posed. Let $\mathcal{R}(t) \in \mathcal{V}_{2N}^p$ be the full model solution of problem (4.2.1) in the temporal interval \mathcal{T}_τ with given initial condition $\mathcal{R}(t^{\tau-1})$. The error between the approximate reduced solution of (4.5.6) and the full model solution at time $t^\tau \in \mathcal{T}$ is given by

$$R_\tau - \mathcal{R}(t^\tau) = (R_\tau - R(t^\tau)) + (R(t^\tau) - \mathcal{R}(t^\tau))$$

The quantity $e_A^\tau := R_\tau - R(t^\tau)$ is the approximation error associated with the partitioned Runge–Kutta discretization scheme, and can be treated using standard convergence analysis techniques, in light of the fact that the retraction map is Lipschitz continuous in the Frobenius norm, as shown in [189, Proposition 5.7]. The term $e_{RA}(t) := R(t) - \mathcal{R}(t)$, for any $t \in \mathcal{T}_\tau$, is associated with the rank update and can be bounded as

$$\begin{aligned} \frac{d}{dt} \|e_{RA}\| &\leq \|\mathcal{P}_R^\epsilon \mathcal{X}_H(R) - \mathcal{X}_H(\mathcal{R})\| \leq \|\mathcal{P}_R^\epsilon \mathcal{X}_H(R) - \mathcal{X}_H(R)\| + \|\mathcal{X}_H(R) - \mathcal{X}_H(\mathcal{R})\| \\ &\leq L_{\mathcal{X}_H} \|e_{RA}\| + \|(\mathbb{I}_{2N} - \mathcal{P}_R^\epsilon) \mathcal{X}_H(R)\|, \end{aligned}$$

where $L_{\mathcal{X}_H}$ is the Lipschitz continuity constant of \mathcal{X}_H . Gronwall's inequality [113] gives, for all $t \in \mathcal{T}_\tau$,

$$\frac{d}{dt} \|e_{RA}(t)\| \leq \|e_{RA}(t_0)\| e^{L_{\mathcal{X}_H} t} + \int_{t^{\tau-1}}^{t^\tau} e^{L_{\mathcal{X}_H}(t-s)} \|(\mathbb{I}_{2N} - \mathcal{P}_R^\epsilon) \mathcal{X}_H(R)\| ds \quad (4.5.7)$$

Observe that the estimate (4.5.7) depends on the distance between the Hamiltonian vector field at the reduced state and its image under the map \mathcal{P}_R^ϵ that approximates the orthogonal projection operator on the tangent space of \mathcal{M}_{2n_τ} . Although a rigorous bound for this term is not available, we expect that it can be controlled arbitrary well by increasing the size of the reduced basis, as will also be demonstrated in Section 4.7. Moreover, the estimate (4.5.7) on the whole temporal interval \mathcal{T} depends exponentially on the final time T . A linear dependence on T can be obtained only in special cases, for example when $\nabla_{\mathcal{R}} \mathcal{H}$ is uniformly negative monotone.

4.6 Computational complexity of the rank-adaptive algorithm

In this Section we discuss the computational cost required for the numerical solution of the reduced problem (4.2.6) with the rank-adaptive algorithm introduced in Section 4.5.

In each temporal interval \mathcal{T}_τ , the algorithm consists of two main steps: the evolution step, which entails the repeated evaluation of the velocity fields \mathcal{F} and \mathcal{G} in (4.3.6) at each stage of the Runge–Kutta temporal integrator, and the rank update step, which requires the evaluation of the error indicator and the update of the approximate reduced solution at the current time step. The rank update strategy introduced in Section 4.5, and summarized in Algorithm 4, has an arithmetic complexity of $O(Np^2) + O(Nn_\tau^2) + O(Npn_\tau)$, and the computational bottleneck is the computation of the error indicator. As suggested in Section 4.5.1, sub-sampling techniques and mesh coarsening can be employed to overcome this limitation. The evolution step consists in solving the discrete reduced system (4.3.6) in each temporal interval. To understand the computational complexity of this step, we neglect the number of nonlinear iterations required by the implicit temporal integrators for the evolution of the coefficients Z . The solution of (4.3.6) requires the evaluation of four operators: the velocity fields \mathcal{G} and \mathcal{F} , the retraction \mathcal{R} and its

inverse tangent map f_τ . The algorithms proposed in [189, Section 5.3.1] for the computation of \mathcal{R} and f_τ have arithmetic complexity $O(Nn_\tau^2)$. We denote with $C_{\mathcal{H}} = C_{\mathcal{H}}(N, n_\tau, p)$ the computational cost to evaluate the gradient of the reduced Hamiltonian at the reduced solution. Finally, the velocity field \mathcal{F} is computed via Algorithm 3 with a computational complexity of $O(Nn_\tau p) + O(Nn_\tau^2) + O(pn_\tau^2) + O(n_\tau^3)$, while $C_{\mathcal{H}}$ is the cost to evaluate Y . It follows that the rank-adaptive algorithm for the solution of the reduced system (4.3.5) with a partitioned Runge–Kutta scheme has a computational complexity being at most linear in the dimension of the full model N , provided the computational cost $C_{\mathcal{H}}$ to evaluate the Hamiltonian vector field at the reduced solution has a comparable cost. Concerning the latter, observe that the assembly of the reduced state R from the factors U and Z and the matrix-vector multiplication $U^\top \nabla_R \mathcal{H}(R; \eta_h)$ require $O(Npn_\tau)$ operations. Therefore, the computational bottleneck of the algorithm is associated with the evaluation of the Hamiltonian gradient at the reduced state R . This problem is well-known in model order reduction and emerges whenever reduced models involve non-affine and nonlinear operators, as discussed in Section 1.5.

As mentioned in Chapter 1, several hyper-reduction techniques have been proposed to mitigate or overcome this limitation, resulting in approximations of nonlinear operators that can be evaluated at a cost independent of the size of the full model. However, we are not aware of any hyper-reduction method able to *exactly* preserve the Hamiltonian phase space structure during model reduction for the general case represented by (4.1.1). Furthermore, hyper-reduction methods entail an offline phase to learn the low-rank structure of the nonlinear operators by means of snapshots of the full model solution. Compared to traditional *global* model order reduction, in a dynamical reduced basis approach the constraints on the computational complexity of the reduced operators is less severe since we allow the dimension of the full model to enter, albeit at most linearly, the computational cost of the operations involved. This means that the dynamical model order reduction can accommodate Hamiltonian gradients where each vector entry depends only on a few, say $k \ll N$, components of the reduced solution, with a resulting computational cost of $C_{\mathcal{H}} = O(Npn_\tau) + O(kNp)$. This is the case when, for example, the dynamical system (4.1.1) ensues from a *local* discretization of a partial differential equation in Hamiltonian form. Note that this assumption is also required for the effective application of DEIM. When dealing with low-order polynomial nonlinearities of the Hamiltonian vector field, we can use tensorial techniques to perform the most expensive operations only once and not at each instance of the parameter, as discussed in the following.

4.6.1 Efficient treatment of the polynomial nonlinearity

Let us consider the explicit expression of the cost $C_{\mathcal{H}}$ for different Hamiltonian functions \mathcal{H} . If the Hamiltonian vector field $\mathcal{X}_{\mathcal{H}}$ in (4.2.1) is linear, then

$$\mathcal{G}(U, Z; \eta_h) = J_{2n} U^\top \nabla_{\mathcal{R}} \mathcal{H}(\mathcal{R}; \eta_h) = J_{2n} U^\top A U Z, \quad \forall \mathcal{R} = UZ \in \mathcal{M}_{2n_\tau},$$

where $A \in \mathbb{R}^{2N \times 2N}$ corresponds to a given linear map, associated with the spatial discretization of the Hamiltonian function \mathcal{H} . Standard matrix-matrix multiplication to compute \mathcal{G} has arithmetic complexity $O(Nn_\tau^2) + O(pn_\tau^2) + O(n_\tau k)$, where k is the number of nonzero entries of the matrix A . The computational complexity of the algorithm is therefore still linear in N provided the matrix A is sparse. This is the case in applications we are interested in where the Hamiltonian system (4.2.1) ensues from a local spatial approximation of a partial differential equation.

In case of low-order polynomial nonlinearities, we use the tensorial representation [240] of the nonlinear function and rearrange the order of computing. The gist of this approach is to exploit the structure of the polynomial nonlinearities to separate the quantities that depend on the dimension of the full model from the reduced variables, by manipulating the order of computation of the various factors. Consider the evolution equations for the coefficients Z in (4.2.6a) for a single value η_j of the parameter $\eta_h \in \Gamma_h$. The corresponding reduced Hamiltonian vector can be expressed in the form

$$J_{2n} \nabla_{Z_j} \mathcal{H}_U(Z_j; \eta_j) = U^\top J_{2N} G^{\{q\}} \underbrace{\left(\bigotimes_{i=1}^q A_i U Z_j \right)}_{\mathcal{G}_U} = \underbrace{U^\top J_{2N} G^{\{q\}} \left(\bigotimes_{i=1}^q A_i U \right)}_{\mathcal{G}_U} \underbrace{\left(\bigotimes_{i=1}^q Z_j \right)}_{\mathcal{Z}} \quad (4.6.1)$$

where $Z_j \in \mathcal{Z}_\tau$ with $p = 1$, $q \in \mathbb{N}$ is the polynomial degree of the nonlinearity, $A_i \in \mathbb{R}^{2N \times 2N}$ are sparse discrete differential operators, $G^{\{q\}}$ represents the matricized q -order tensor and \otimes denotes the Kronecker product. The last expression in (4.6.1) allows to separate the computations involving factors of size N from the reduced coefficients Z , so that the matrix $\mathcal{G}_U \in \mathbb{R}^{2n_\tau \times (2n_\tau)^q}$ can be precomputed during the offline phase.

In the case of the proposed dynamical reduced basis method, we employ the tensorial POD approach to reduce the computational complexity of the evaluation of \mathcal{G} , the RHS of (4.2.6a), and its Jacobian needed in the implicit symplectic integrator at each time step of the numerical integrator. We start by noticing that a straightforward calculation of the second expression in (4.6.1) suggests $O(cNpn_\tau) + O(cpqk) + O(cNpq)$ operations, where the first term is due to the reduced basis ansatz and the Galerkin projection, the second term to the multiplication by the sparse matrices A_i and the third term to the evaluation of a polynomial of degree q for each entry of a $2N \times p$ matrix. The constant c represents the number of iterations of the Newton solver and $k := \max_i k_i$, where k_i is the number of nonzero entries of A_i . Moreover, in each iteration we evaluate not only the nonlinear term but also its Jacobian, with an additional cost of $O(cNp(q-1)) + O(cp k_{\mathcal{G}} n_\tau) + O(cNp n_\tau^2)$ operations, with $k_{\mathcal{G}}$ being the number of nonzero entries of the full-order Jacobian. These terms represent, respectively, the operations required to evaluate the polynomial functions in the Jacobian, the assembly of the Jacobian matrix and its Galerkin projection onto the reduced basis. This high computational cost can again be mitigated by resorting to the second formula in (4.6.1), where the term \mathcal{G}_U is precomputed at each iteration, for each stage of the partitioned RK integrator (4.3.6). To estimate the computational cost of the procedure we resort to the multi-index notation by introducing $\mathbf{n} := (n_\tau, \dots, n_\tau) \in \mathbb{R}^n$ and hence $\mathcal{G}_U \mathcal{Z}$ in (4.6.1) can be recast as

$$\underbrace{\mathcal{G}_U}_{\text{(III)}} \mathcal{Z} = U^\top J_{2n} \sum_{l \leq 2\mathbf{n}} \prod_{1 \leq i \leq \mathbf{q}} \overbrace{\text{diag} \left(\underbrace{A_i U}_{\text{(I)}} \right)}^{\text{(II)}} \underbrace{A_1 U_l}_{\text{(1)}} Z^l. \quad (4.6.2)$$

The arithmetic complexity of this step is $O(qkn_\tau) + O((q-1)Nn_\tau^q) + O(Nn_\tau^{q+1})$, where the first term is due to the matrix multiplication of the q matrices $A_i U$ in (I), the second term to the pointwise and diagonal matrices multiplications involved in the computations of (II) and the third term to the multiplications by $U^\top J_{2N}$ in (III). We stress that the cost required to assemble \mathcal{G}_U is independent of the number of parameters p and the number of iterations of the nonlinear solver. Once \mathcal{G}_U has been precomputed, the evaluation of the reduced RHS has a computational cost of $O(cpn_\tau^{q+1})$ [240]. The same splitting technique is exploited for each evaluation of the

reduced Jacobian and most of the precomputed terms in (4.6.2) can be reused. The proposed treatment of polynomial nonlinearities results in an effective reduction of the computational cost in case of low-order polynomial nonlinearity ($q = 2, 3$), a large set of vector-valued parameters ($p \gg 10$) and a moderate number n_τ of basis vectors.

4.7 Numerical experiments

To assess the performance of the proposed adaptive dynamical structure preserving reduced basis method, we consider finite-dimensional parametrized Hamiltonian dynamical systems arising from the spatial approximation of PDEs. Let $\Omega \subset \mathbb{R}^d$ be a continuous domain and let $u : \mathcal{T} \times \Omega \times \Gamma \rightarrow \mathbb{R}^m$ belong to a Sobolev space \mathcal{V} endowed with the inner product $\langle \cdot, \cdot \rangle$. A parametric evolutionary PDE in Hamiltonian form can be written as

$$\begin{cases} \frac{\partial}{\partial t} u(t, x; \eta) = \mathcal{J} \frac{\delta}{\delta u} \mathcal{H}(u; \eta), & \text{in } \Omega \times \mathcal{T}, \\ u(0, x; \eta) = u^0(x; \eta), & \text{in } \Omega, \end{cases} \quad (4.7.1)$$

with suitable boundary conditions prescribed at the boundary $\partial\Omega$. Here, the dot denotes the derivative with respect to time, and δ denotes the variational derivative of the Hamiltonian \mathcal{H} defined as

$$\frac{d}{d\varepsilon} \mathcal{H}(u + \varepsilon v; \eta)|_{\varepsilon=0} = \left\langle \frac{\delta}{\delta u} \mathcal{H}, v \right\rangle, \quad \forall u, v \in \mathcal{V},$$

so that, for $l = 1, \dots, m$ and $u_{l,k} := \partial_{x_k} u_l$, it holds

$$\frac{\delta}{\delta u_l} \mathcal{H} = \frac{\partial H}{\partial u_l} - \sum_{k=1}^d \frac{\partial}{\partial x_k} \left(\frac{\partial H}{\partial u_{l,k}} \right) + \dots, \quad \text{with} \quad \mathcal{H}(u; \eta) = \int_{\Omega} H(x, u, \partial_x u, \partial_{xx} u, \dots; \eta) dx.$$

In the numerical tests, we consider, for any fixed value of the parameter $\eta_j \in \Gamma_h$, numerical spatial approximations of (4.7.1) that yield a $2N$ -dimensional Hamiltonian system in canonical form

$$\begin{cases} \frac{d}{dt} u_h(t; \eta_h) = J_{2N} \nabla_u \mathcal{H}_h(u_h; \eta_j), & \text{in } \mathcal{T}, \\ u_h(0; \eta_j) = u_h^0(\eta_j), \end{cases} \quad (4.7.2)$$

where u_h belongs to a finite $2N$ -dimensional subspace of \mathcal{V} , ∇_u is the gradient with respect to the state variable u_h and $\mathcal{H}_h : \mathbb{R}^{2N} \rightarrow \mathbb{R}$ is such that $\Delta x_1 \dots \Delta x_d \mathcal{H}_h$ is a suitable approximation of \mathcal{H} . Testing (4.7.2) for p values $\Gamma_h = \{\eta_j\}_{j=1}^p$ of the parameter, yields a matrix-valued ODE of the form 3.2.1, where the j -th column of the unknown matrix $\mathcal{R}(t) \in \mathbb{R}^{2N \times p}$ is equal to $u_h(t, \eta_j)$ for all $j = 1, \dots, p$.

We validate the proposed adaptive dynamical reduced basis method on several representative Hamiltonian systems of the form (4.7.2), of increasing complexity, and compare the quality of the adaptive dynamical approach with a reduced model with a global basis. The proposed approach, including all the steps introduced in the previous sections, is summarized in Algorithm 5.

For the global model, we consider Complex SVD method discussed in Section 3.3.1, where a reduced basis is built via a complex SVD of a suitable matrix of snapshots and the reduced model is derived via symplectic Galerkin projection onto the space spanned by the global basis. The accuracy, conservation properties and efficiency of the reduced models are analyzed and compared by monitoring various quantities. To assess the approximation properties of the reduced model,

Algorithm 5 Rank-adaptive reduced basis method

- 1: **procedure** RANK__ADAPTIVE__REDUCED__BASIS__METHOD($\mathcal{R}_0, \eta_h, \widetilde{\eta}_h, N_{\mathbf{E}}, n_1, \epsilon, r, c$)
 - 2: Compute $U_0 \in \mathcal{U}_1$ via complex SVD of $\mathcal{R}_0(\eta_h)$ truncated at the n_1 -th mode, and $Z_0 \leftarrow U_0^\top \mathcal{R}_0(\eta_h)$
 - 3: Initialize the error indicator matrix $\mathbf{E}_0 \leftarrow \mathcal{R}_0(\widetilde{\eta}_h) - U_0 U_0^\top \mathcal{R}_0(\widetilde{\eta}_h) \in \mathbb{R}^{2N \times \widetilde{p}}$ and $\mathbf{E}_* \leftarrow \mathbf{E}_0$
 - 4: **for** $\tau = 1, \dots, N_\tau$ **do**
 - 5: Calculate $(U_\tau, Z_\tau) \in \mathcal{U}_\tau \times \mathbb{R}^{2n_\tau \times p}$ using partitioned RK integrator (4.3.6), starting from $(U_{\tau-1}, Z_{\tau-1}) \in \mathcal{U}_{\tau-1} \times \mathbb{R}^{2n_{\tau-1} \times p}$:
 - Use the tensorial POD approach (4.6.1) to assemble the operator \mathcal{G}
 - Use the retraction map given in (4.3.3) to compute $\mathcal{R}_{U_{\tau-1}}$
 - Compute f_τ according to (4.3.4), using REGULARIZATION (Algorithm 3), with parameter ϵ as input, to assemble \mathcal{F}
 - 6: **if** $\text{mod}(\tau, N_{\mathbf{E}}) = 0$ **then**
 - 7: Compute the error indicator matrix and check the rank update criterion using RANK__UPDATE (Algorithm 4) as
 $(U_\tau, Z_\tau, \mathbf{E}_*, \mathbf{E}_\tau, \lambda_\tau) = \text{RANK_UPDATE}(U_\tau, Z_\tau, \mathbf{E}_*, \mathbf{E}_{\tau-1}, \lambda_{\tau-1}, r, c)$
-

we track the error, in the Frobenius norm, between the full model solution \mathcal{R} and the reduced solution R at any time $t \in \mathcal{T}$, namely

$$E(t) = \|\mathcal{R}(t) - R(t)\|_2. \quad (4.7.3)$$

Moreover, we study the conservation of the Hamiltonian via the relative error in l_1 -norm in the parameter space Γ_h , that is

$$E_{\mathcal{H}_h}(t) = \sum_{i=1}^p \left| \frac{\mathcal{H}(U_\tau Z_\tau^i; \eta_i) - \mathcal{H}(U_0 Z_0^i; \eta_i)}{\mathcal{H}(U_0 Z_0^i; \eta_i)} \right|. \quad (4.7.4)$$

The reason for the use of the l_1 -norm is the following. For any fixed parameter η_j , let $\mathcal{H}_j := \mathcal{H}(\cdot; \eta_j) : \mathbb{R}^{2N} \rightarrow \mathbb{R}$, and let $\omega_j(a, b) := \omega(a_j, b_j) = a_j^\top J_{2N} b_j$ where $a_j \in \mathbb{R}^{2N}$ denotes the j -th column of the matrix $a \in \mathbb{R}^{2N \times p}$. As explained in Section 4.2, the velocity field of the reduced flow is the symplectic projection of the full model velocity onto the tangent space of the reduced manifold, see (4.2.5). This entails that the reduced solution $R \in C^1(\mathcal{T}, \mathcal{M}_{2n})$ satisfies the symplectic variational principle

$$\sum_{j=1}^p \omega_j \left(\Pi_{T_R \mathcal{M}_{2n}} \mathcal{X}_{\mathcal{H}}(R, \eta_j) - \mathcal{X}_{\mathcal{H}}(R, \eta_j), y \right) = \sum_{j=1}^p \omega_j \left(\dot{R} - J_{2N} \nabla \mathcal{H}_j(R), y \right) = 0, \quad \forall y \in T_R \mathcal{M}_{2n},$$

where $\mathcal{X}_{\mathcal{H}}(R, \eta_j) = J_{2N} \nabla_u \mathcal{H}(R, \eta_j)$. This implies that

$$\sum_{j=1}^p \frac{d}{dt} \mathcal{H}_j(R(t)) = \sum_{j=1}^p (\nabla_{R_j} \mathcal{H}_j(R), \dot{R}_j) = \sum_{j=1}^p \omega(J_{2N} \nabla_{R_j} \mathcal{H}_j(R), \dot{R}_j) = \sum_{j=1}^p \omega_j(\dot{R}, \dot{R}) = 0.$$

Finally, we monitor the computational cost of different reduction strategies. Throughout, the runtime is defined as the sum of the lengths of the offline and online phases in the case of complex SVD (global method); while, for the dynamical approaches it is the time required to evolve basis and coefficients (4.3.5) plus the time required to compute the error indicator and update the dimension of the approximating manifold, in the adaptive case.

The adaptive dynamical RB method is numerically tested on two nonlinear problems, the shallow water and Schrödinger equations in one and two dimensions. Finally, we consider a preliminary application to particle simulations of plasma physics problem with the reduction of the Vlasov equation with a forced external electric field, modeling the evolution of charged particle beams. All numerical simulations are performed using MATLAB computing environment on computer nodes with Intel Xeon E5-2643 (3.40 GHz).

4.7.1 Shallow water equations

The shallow water equations (SWE) describe the kinematic behaviour of a thin inviscid single fluid layer flowing over a variable topography. In the setting of irrotational flows and flat bottom topography, the fluid is described by a scalar potential ϕ and the canonical Hamiltonian formulation is recovered [243]. The resulting time-dependent nonlinear system of PDEs is defined as

$$\begin{cases} \frac{\partial}{\partial t} h + \nabla \cdot (h \nabla \phi) = 0, & \text{in } \Omega \times \mathcal{T}, \\ \frac{\partial}{\partial t} \phi + \frac{1}{2} |\nabla \phi|^2 + h = 0, & \text{in } \Omega \times \mathcal{T}, \\ h(0, x; \eta_h) = h^0(x; \eta_h), & \text{in } \Omega, \\ \phi(0, x; \eta_h) = \phi^0(x; \eta_h), & \text{in } \Omega, \end{cases} \quad (4.7.5)$$

with spatial coordinates $x \in \Omega$, time $t \in \mathcal{T}$, state variables $h, \phi : \Omega \times \mathcal{T} \rightarrow \mathbb{R}$, $\nabla \cdot$ and ∇ divergence and gradient differential operators in x , respectively. The variable ϕ is the scalar potential of the fluid and h represents the height of the free-surface, normalized by its mean value. The system is coupled with periodic boundary conditions for both the state variables. The evolution problem (4.7.5) admits a canonical symplectic Hamiltonian form (4.7.1) with the Hamiltonian

$$\mathcal{H}(h, \phi; \eta) = \frac{1}{2} \int_{\Omega} (h |\nabla \phi|^2 + h^2) dx.$$

We consider numerical simulations in $d = 1$ and $d = 2$ dimensions on rectangular spatial domains. The domain Ω is partitioned using a Cartesian mesh in $M - 1$ equispaced intervals in each dimension, having mesh width Δx and Δy , when $d = 2$. As degrees of freedom of the problem we consider the nodal values of the height and potential, i.e., $u_h(t; \eta_h) := (h, \phi) = (h_1, \dots, h_N, \phi_1, \dots, \phi_N)$, for all $t \in \mathcal{T}$ and $\eta_h \in \Gamma_h$, where $N := M^d$, $h_m = h_{i,j}$ with $m := (j - 1)M + i$, and $i, j = 1, \dots, M$. In 1D, $N = M$, and the index j is dropped.

We consider second order accurate central finite difference schemes to discretize the differential operators in (4.7.5), and denote with D_x and D_y the discrete differential operators acting in the x - and y -direction, respectively. The semi-discrete formulation of (4.7.5) represents a canonical Hamiltonian system with the gradient of the Hamiltonian function with respect to u_h given by

$$\nabla \mathcal{H}_h(u_h; \eta_h) = \begin{pmatrix} \frac{1}{2} [(D_x \phi_h)^2 + (D_y \phi_h)^2] + h_h \\ -D_x(h_h \odot D_x \phi_h - D_y(h_h \odot D_y \phi_h)) \end{pmatrix} \quad (4.7.6)$$

where \odot is the Hadamard product between two vectors. The discrete Hamiltonian is

$$\mathcal{H}_h(u_h; \eta_h) = \frac{1}{2} \sum_{i,j=1}^M \left(h_{i,j} \left[\left(\frac{\phi_{i+1,j} - \phi_{i-1,j}}{2\Delta x} \right)^2 + \left(\frac{\phi_{i,j+1} - \phi_{i,j-1}}{2\Delta y} \right)^2 \right] + h_{i,j}^2 \right). \quad (4.7.7)$$

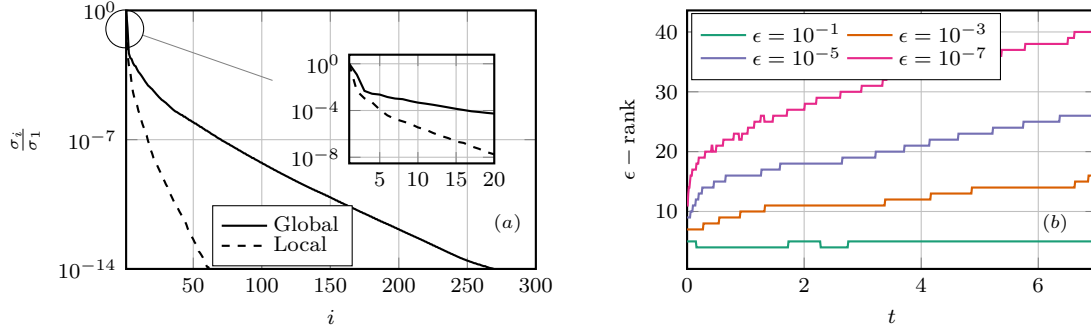


Figure 4.2: SWE-1D: (a) Singular values of the global snapshots matrix S^{u_h} and time average of the singular values of the local trajectories matrix $S_{\tau}^{u_h}$. The singular values are normalized using the largest singular values for each case. (b) ϵ -rank of the local trajectories matrix $S_{\tau}^{u_h}$ for different values of ϵ .

In the one-dimensional case, the operator D_y vanishes.

One-dimensional shallow water equations (SWE-1D)

For this example, we set $\Omega = [-10, 10]$ and we consider the parameter domain $\Gamma = [\frac{1}{10}, \frac{1}{7}] \times [\frac{2}{10}, \frac{15}{10}]$. The discrete set of parameters Γ_h is obtained by uniformly sampling Γ with 10 samples per dimension, for a total of $p = 100$ different configurations. Problem (4.7.5) is completed with the initial condition

$$\begin{cases} h^0(x; \eta_h) = 1 + \alpha e^{-\beta x^2}, \\ \phi^0(x; \eta_h) = 0, \end{cases} \quad (4.7.8)$$

with $\eta_h = (\alpha, \beta)$, where α controls the amplitude of the initial hump in the depth h and β describes its width. We consider a partition of the spatial domain Ω into $N - 1$ equispaced intervals with $N = 1001$. The full model solution $u_h(t; \eta_h)$ is computed using a uniform step size $\Delta t = 10^{-3}$ in the time interval $\mathcal{T} = (0, T := 7]$. We use the implicit midpoint rule as time integrator because, being symplectic, it preserves geometrical properties of the flow of the semi-discrete equation associated to (4.7.6). To study the reducibility properties of the problem, we explore the solution manifold and collect the solutions to the high-fidelity model in different matrices. The global snapshot matrix $S^{u_h} \in \mathbb{R}^{2N \times (N_t p)}$ contains the snapshots associated with all sampled parameters η_h and time steps, while, for any $\tau = 1, \dots, N_t$, the matrix $S_{\tau}^{u_h} \in \mathbb{R}^{2N \times p}$ collects the full model solutions at fixed time t^{τ} .

In Figure 4.2(a), we compare the normalized singular values of S^{u_h} and $S_{\tau}^{u_h}$, averaged over time for the latter. Although, in both cases, the exponential decay of the spectrum suggests the existence of reduced approximation spaces, the decay of the singular values of the averaged $S_{\tau}^{u_h}$ is roughly 5 times faster than that of S^{u_h} . This difference suggests that a low-rank dynamical approach may be beneficial to reduce the computational cost and increase the accuracy of the solution of the reduced model compared to a method with a global basis. Furthermore, the evolution of the numerical rank of $S_{\tau}^{u_h}$ over time, reported in Figure 4.2(b), shows a rapid growth during the first steps, followed by a mild increase in the remaining part of the simulation. This is compatible with the observations, made in Section 4.5.2, about the behavior of the singular value spectrum for advection dominated problems.

In order to compare the performances of local and global model order reduction, we con-

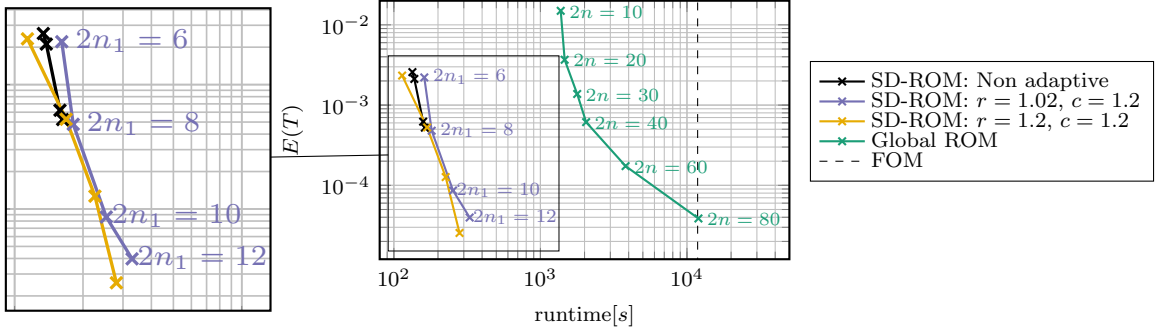


Figure 4.3: SWE-1D: Relative error at time $T = 7$, as a function of the runtime for the complex SVD method (Global ROM), the dynamical RB method (Non adaptive), and the adaptive dynamical RB method for different values of the control parameters r and c . For the sake of comparison, we report the runtime required by the high-fidelity solver to compute the numerical solutions for all values of the parameter $\eta_h \in \Gamma_h$.

sider, as global reduced method, the complex SVD approach with reduced dimension $2n \in \{10, 20, 30, 40, 60, 80\}$. This is used to generate a symplectic reduced basis from the solution of the high-fidelity model (4.7.5) obtained every 10 time steps and by uniformly sampling Γ with 4 samples per dimension. The reduced system is solved using the implicit midpoint rule with the same time step Δt used for the full order model. The quadratic operator, describing the evolution of (4.7.5), is reduced by using the approach described in Section 4.6.1 and the reduced operators are computed once during the offline stage.

Concerning the adaptive dynamical reduced model, we evaluate the initial condition (4.7.8) at all values $\eta_h = \Gamma_h$ and compute the matrix $S_1^{u_h} \in \mathbb{R}^{2N \times p}$ having as columns each of the evaluations. As initial condition for the reduced system, we use

$$\begin{cases} U(0) = U_0, \\ Z(0) = U_0^\top S_1^{u_h}, \end{cases} \quad (4.7.9)$$

where $U_0 \in \mathbb{R}^{2N \times 2n_1}$ is obtained using the complex SVD applied to the snapshot matrix $S_1^{u_h}$. System (4.2.6) is then evolved using the 2-stage partitioned Runge-Kutta method described in Algorithm 5. For the following numerical experiments, we consider $2n_1 \in \{6, 8, 10, 12\}$ as initial dimensions of the approximating reduced manifolds. As control parameters for the rank update criterion of Algorithm 5, we fix the value $c = 1.2$ and study examples with $r \in \{1.02, 1.05, 1.1, 1.2\}$. Moreover, we examine the case in which the rank-updating algorithm is never triggered, i.e., the basis $U(t)$ evolves in time but its dimension is fixed ($n_\tau = n_1$ for all τ). In the adaptive case, the error indicator \mathbf{E}_τ is computed every 100 iterations using a coarse mesh with 500 equispaced intervals on the subset $\eta_h^{\mathbf{E}}$ obtained by sampling 5 parameters per dimension from Γ_h .

In Figure 4.3, we compare the global reduced model, the dynamical models for different values of r , and the high-fidelity model in terms of total runtime and accuracy at the final time T by monitoring the error (4.7.3). The results show that, as we increase the dimension of the global reduced basis, the global reduced model provides accurate approximations but the runtime becomes larger than the one required to solve the high-fidelity problem. Hence, the global method loses the efficiency. The adaptive dynamical reduced approach outperforms the global reduced method by reaching comparable levels of accuracy at a computational time which is one order of magnitude smaller than the one required by the global reduction. Compared to the high-fidelity

solver, the adaptive dynamical reduced method achieves an accuracy of $E(T) = 2.55 \cdot 10^{-5}$ with a speedup up of 42, in the best-case scenario. For this numerical experiment, the effectiveness of the rank update algorithm is limited by the error introduced in the approximation of the initial condition via a reduced basis. While the error is reduced from a factor of 4 in the case of $2n_1 = 8$ to a factor of 20 in the case of $2n_1 = 12$, compared to the non adaptive method, the accuracy is not significantly improved when $2n_1 = 6$. We note that, when the adaptive algorithm is effective, the additional computational cost associated with the evaluation of the error indicator and the evolution of a larger basis is balanced by a considerable error reduction.

To better gauge the accuracy properties of the adaptive dynamical reduced basis method, we compare its error with the error given by the high-fidelity solver for the same initial condition. The solution to the full model, with the projection of (4.7.8) onto the column space of U_0 as the initial condition, is the target of the adaptive reduced procedure, which aims at correctly representing the high-fidelity solution space at every time step. The importance of having a reduced space that accurately reproduces the initial condition can be inferred from Figure 4.4(a): the error associated with a poorly resolved initial condition dominates over the remaining sources of error, and adapting the dimension of the reduced basis is not beneficial in terms of accuracy. As noted above, increasing $2n_1$ not only improves the performance of the non adaptive reduced dynamical procedure but also boosts the potential gain, in terms of relative error reduction, of the adaptive method, as can be seen in Figure 4.4(e).

Moreover, in Figures 4.4 we report the growth of the dimension of the reduced basis for different initial dimension $2n_1$. For the evolution of the error, we do not notice any significant difference as the parameter r for the adaptive criterion varies. Ideally, within each temporal interval, the reduced solution is close, in the Frobenius norm, to the best rank $2n_\tau$ approximation of the full model solution. To verify this property for the adaptive dynamical reduced basis method, we monitor the evolution of the error E_\perp between the full model solution \mathcal{R} , at the current time and for all $\eta_h \in \Gamma_h$, and its projection onto the space spanned by the current reduced basis, namely

$$E_\perp(t) = \|\mathcal{R}(t) - U(t)U(t)^\top \mathcal{R}(t)\|_2.$$

In Figure 4.5, the projection error is shown for different values of the control parameters (Figures 4.5(a) and 4.5(b)) and the corresponding evolution of the reduced basis dimension is reported (Figures 4.5(c) and 4.5(d)). We notice that, when the dimension of the basis U is not adapted, the projection error tends to increase in time. This can be ascribed to the fact that the effective rank of the high-fidelity solution is growing and the reduced basis is no longer large enough to capture the rank-increasing solution. Adapting $2n_\tau$ during the simulation results in a zero-growth scenario, with local negative peaks when the basis is enlarged. This indicates that the strategy of enlarging the reduced manifold in the direction of the larger error (see Section 4.5) yields a considerable improvement of the approximation.

In Figure 4.6 we show the relative error in the conservation of the Hamiltonian for different dimensions of the reduced manifold, and values of the control parameters r and c . As the Hamiltonian (4.7.7) is a cubic quantity, we do not expect exact conservation associated with the proposed partitioned Runge–Kutta temporal integrators. However, the preservation of the symplectic structure both in the reduction and in the discretization yields a good control on the Hamiltonian error, as it can be observed in Figure 4.6.

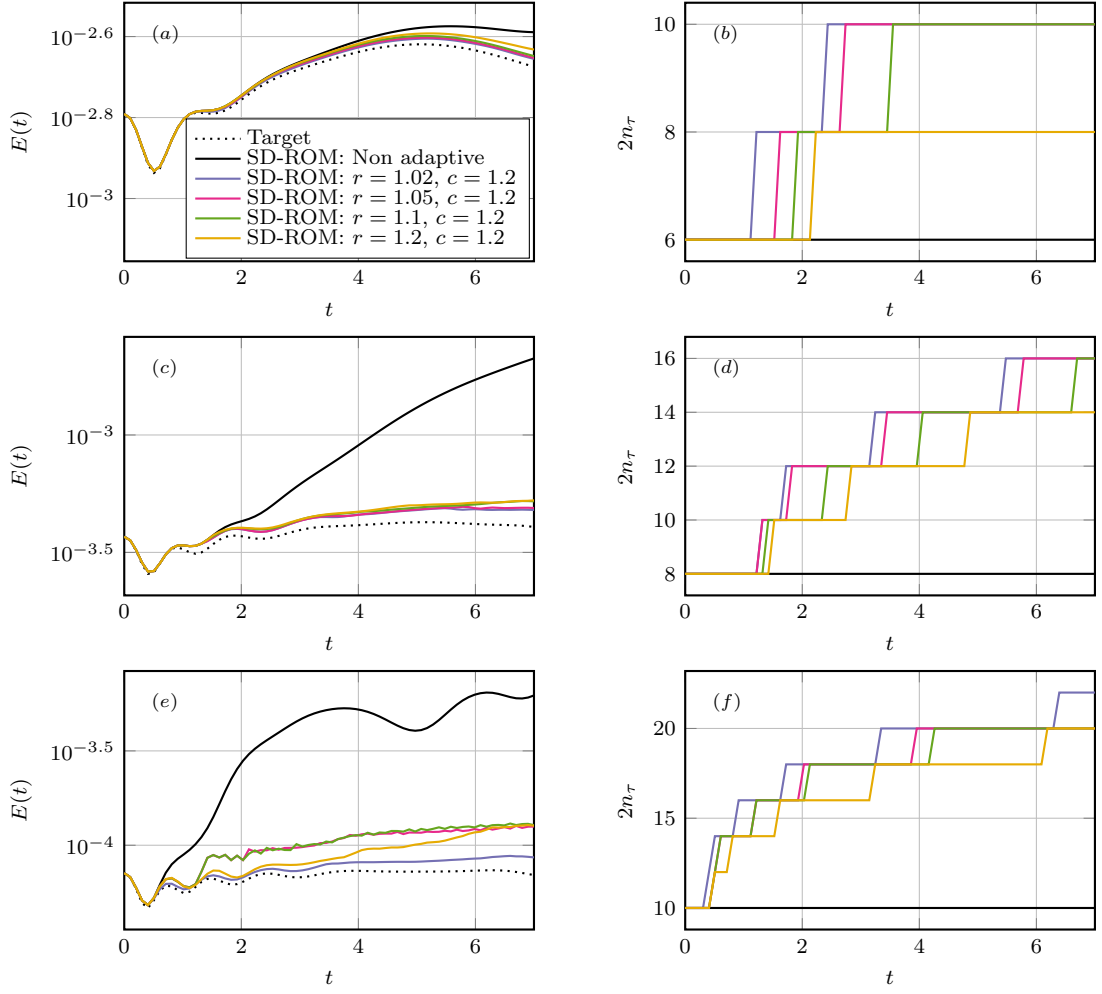


Figure 4.4: SWE-1D: On the left column, we report the evolution of the error $E(t)$ for the adaptive and non adaptive dynamical RB methods for different values of the control parameter r and different dimensions $2n_1$ of the approximating manifold of the initial condition. The target error is obtained by solving the full model with initial condition obtained by projecting (4.7.8) onto a symplectic manifold of dimension $2n_1$. On the right column, we report the evolution of the dimension of the dynamical reduced basis over time. The adaptive algorithm is driven by the error indicator (4.5.4), while in the non adaptive setting, the dimension does not change with time. We consider the cases $2n_1 = 6$ (Figures (a)-(b)), $2n_1 = 8$ (Figures (c)-(d)), and $2n_1 = 10$ (Figures (e)-(f)).

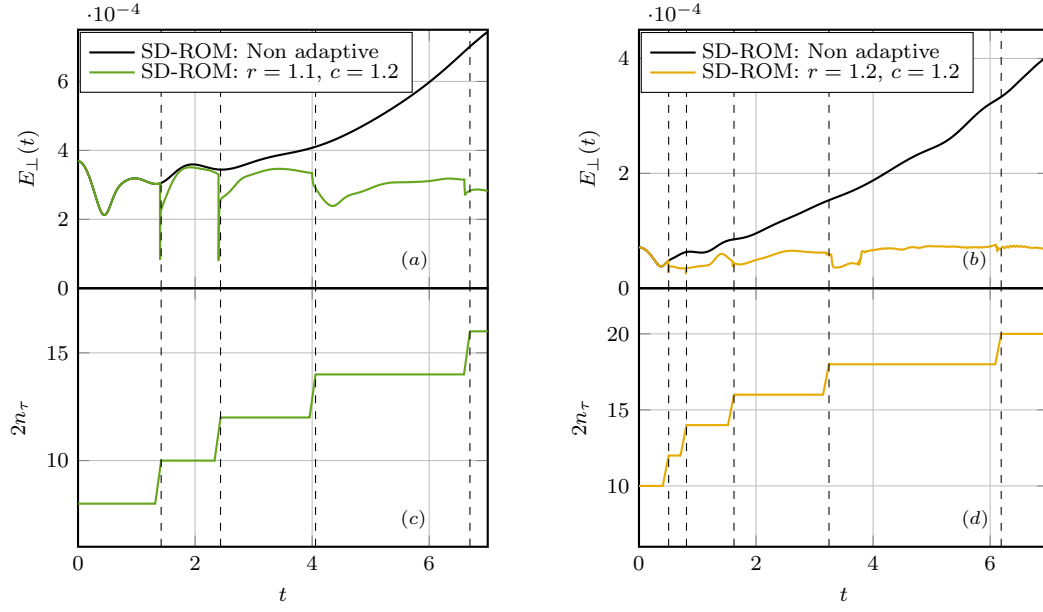


Figure 4.5: SWE-1D: In Figures (a) and (b), we report the evolution of the projection error E_{\perp} for different values of the initial dimension $2n_1$ of the reduced manifold. In Figures (c) and (d), we report the corresponding evolution of the dimension of the reduced manifolds.

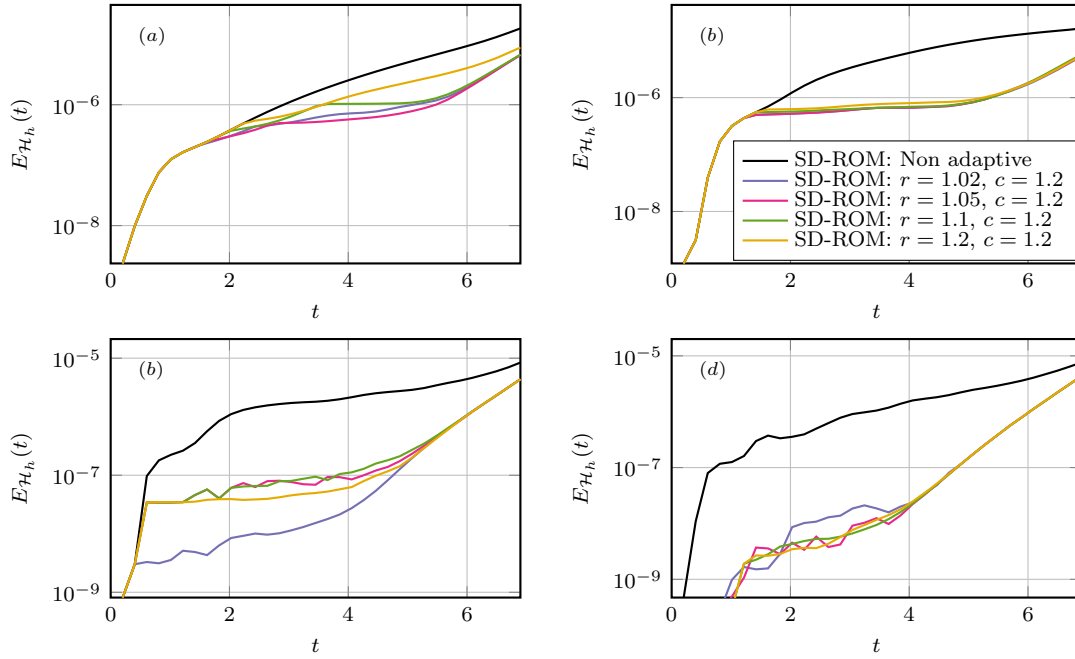


Figure 4.6: SWE-1D: Relative error (4.7.4) in the conservation of the discrete Hamiltonian (4.7.7) for the dynamical reduced basis method with initial reduced dimensions $2n_1 = 6$ (a), $2n_1 = 8$ (b), $2n_1 = 10$ (c), and $2n_1 = 12$ (d).

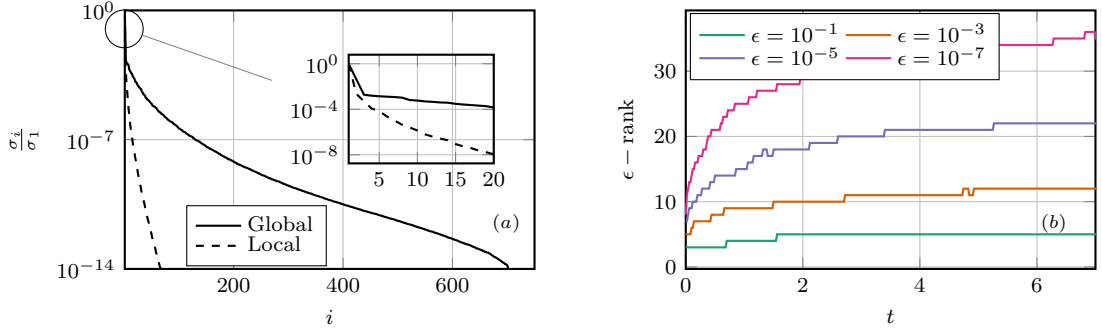


Figure 4.7: SWE-2D: (a) Singular values of the global snapshots matrix S^{u_h} and time average of the singular values of the local trajectories matrix $S_{\tau}^{u_h}$. The singular values are normalized using the largest singular values for each case. (b) ϵ -rank of the local trajectories matrix $S_{\tau}^{u_h}$ for different values of ϵ .

Two-dimensional shallow water equations (SWE-2D)

We set $\Omega = [-4, 4]^2$ as the spatial domain and $\Gamma = [\frac{1}{5}, \frac{1}{2}] \times [\frac{11}{10}, \frac{17}{10}]$ as the domain of parameters. We consider 10 uniformly spaced values of the parameter for each dimension of Ω to define the discrete subset Γ_h . As initial condition, we consider

$$\begin{cases} h^0(x, y; \eta_h) = 1 + \alpha e^{-\beta(x^2+y^2)}, \\ \phi^0(x, y; \eta_h) = 0, \end{cases} \quad (4.7.10)$$

where $\eta_h = (\alpha, \beta)$ represents the natural extension to the two-dimensional setting of the parameter used in the previous example. The domain Ω is partitioned using $M = 51$ points per dimension, so that the resulting mesh width is $\Delta x = \Delta y = 16 \cdot 10^{-2}$. The time domain $\mathcal{T} = [0, T := 20]$ is split into $N_t = 10000$ uniform intervals of length $\Delta t = 2 \cdot 10^{-3}$. The symplectic implicit midpoint is employed as time integrator in the high-fidelity solver, while the reduced dynamics is integrated using the 2-stage partitioned RK method. The spatial and temporal domains considered for this numerical experiment are taken so that the solution of the high-fidelity model is characterized by circular waves that interact and overlap because of the periodic boundary conditions, as shown in Figure 4.8.

The increased complexity of the two-dimensional dynamics is reflected in the behaviour of the spectrum of the matrix snapshots. In Figure (4.7)(a), we show the normalized singular values of the global snapshot matrix $S^{u_h} \in \mathbb{R}^{2N \times (N_t p)}$ and the average of the N_t local-in-time snapshot matrices $S_{\tau}^{u_h} \in \mathbb{R}^{2N \times p}$. The decay of the singular values of the local trajectories is one order of magnitude faster than of the global (in time) snapshots, suggesting that there exists an underlying *local* low-rank structure that can be exploited to improve the efficiency of the reduced model. The evolution of the numerical rank of $S_{\tau}^{u_h}$, reported in 4.7(b), indicates that, while the matrix-valued initial condition is exactly represented using an extremely small basis, the full model solution at times $t \geq 2$ requires a relatively large basis to be properly approximated, and hence adapting the dimension of the reduced manifold becomes crucial. We employ the complex SVD method to build a global reduced order model, using the same sampling rates in time and parameter space as in the 1D test case. With none of the dimensions considered, i.e., $2n \in \{10, 20, 40, 60, 80, 120\}$, we obtain results that are both accurate (error smaller than 10^{-1}) and computationally less expensive than solving the high-fidelity model. Hence, for this two-dimensional test, we only compare the

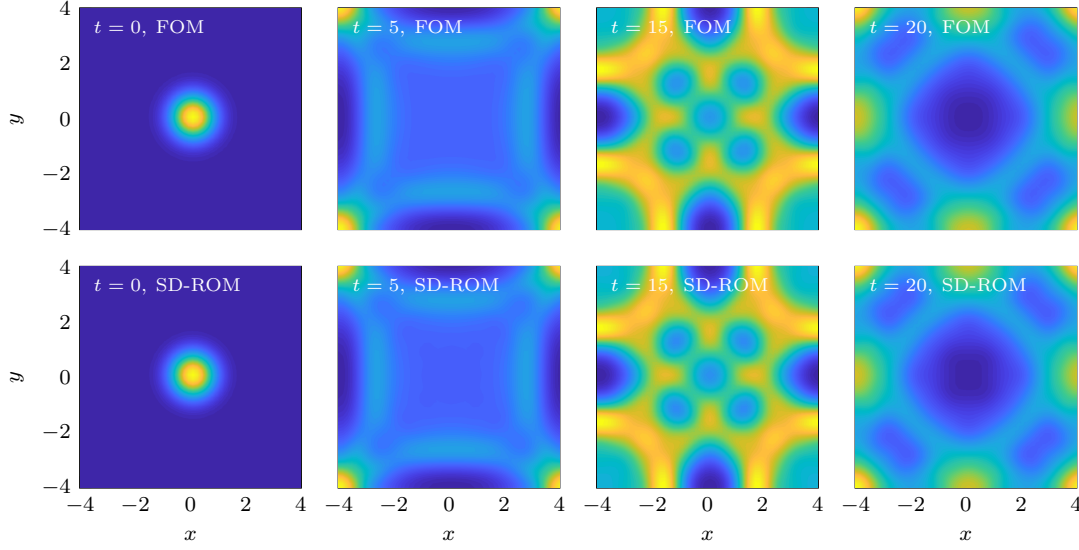


Figure 4.8: SWE-2D: High fidelity solution (Figures (a)-(d)) and adaptive dynamical reduced solution (Figures (e)-(h)) for the parameter $(\alpha, \beta) = (\frac{1}{3}, \frac{17}{10})$ and $t = 0, 5, 15$, and 20 s. In the adaptive reduced approach, we set $r = 1.1$, $c = 1.3$, and $2n_1 = 6$.

performances of the adaptive and the non-adaptive dynamical reduced basis method in terms of accuracy and computational time. As initial condition for the reduced dynamics we consider the initialization (4.7.9) where $S_1^{u_h}$ is given by (4.7.10). Moreover, for the adaptive method, we compute the error indicator every 10 iterations and on a subset η_h^E of 25 uniformly sampled parameters. Different combinations of the initial reduced manifold dimension $2n_1 = \{4, 6, 8\}$ and control parameters $r = 1.1, 1.2, 1.3$ and $c = 1.1, 1.2, 1.3$, are considered to study their impact on the accuracy of the method.

Figure 4.8 shows the high-fidelity solution for $(\alpha, \beta) = (\frac{1}{3}, \frac{17}{10})$ with its adaptive reduced approximation at different times. The results are qualitatively equivalent.

Figure 4.9 reports the error $E(t)$ versus the runtime required to compute the solution for all $\eta_h \in \Gamma_h$ by means of the adaptive and non-adaptive dynamical reduced methods, for different values of $2n_1$, r , and c . Observe that the runtime of the high-fidelity solver is $3.29 \cdot 10^5$ s. The results show that both reduction methods are able to accurately approximate the high-fidelity solution, with speed-ups of 261 for the non-adaptive approach and 113 for the adaptive approach. The exceptional efficiency of the dynamical reduced approach in this context is a result of the combination of three main factors: the low degree polynomial nonlinearity, the large number of degrees of freedom needed to represent the high-fidelity solution, and the compact dimension of the local reduced manifold. Despite the small computational overhead for the adaptive method due to the error estimation, the basis update and the larger approximating spaces used, the adaptive algorithm leads to approximations that are one ($2n_1 = 4$) to two ($2n_1 = 10$) orders of magnitude more accurate than the approximations obtained by the non adaptive method.

The results presented in Figure 4.10 on the evolution of the error for $2n_1 = \{4, 6, 8\}$, corroborate the conclusions, already drawn from the 1D test case, regarding the effect of a poorly approximated initial condition on the performances of the adapting procedure. The evolution of the basis dimension is reported in Figures 4.10(b), 4.10(d) and 4.10(f) for different values of r , c , and $2n_1$.

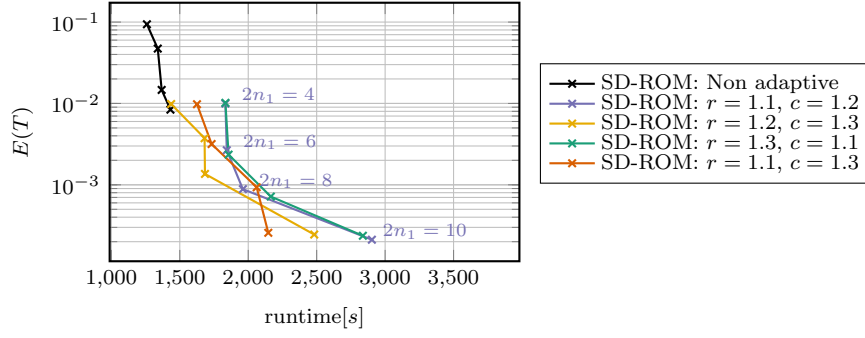


Figure 4.9: SWE-2D: Error (4.7.3), at time $T = 20$, as a function of the runtime for the dynamical RB method and the adaptive dynamical RB method for different values of the control parameters r and c for the simulation of all the sampled parameters in Γ_h . For comparison, the high-fidelity model runtime is $3.3 \cdot 10^5$ s.

4.7.2 Nonlinear Schrödinger equations

The nonlinear Schrödinger equation (NLS) is used to model, among others, the propagation of light in nonlinear optical fibers and planar waveguides and to describe the Bose–Einstein condensates in a macroscopic gaseous superfluid wave-matter state at ultra-cold temperature. In our setting, we test the adaptive strategy in the case of a Fourier mode cascade, where, starting from an initial condition represented by few low Fourier modes, the energy exchange to higher modes quickly complicates the dynamics of the problem [46]. More specifically, in the spatial domain Ω , we consider the cubic Schrödinger equation

$$\begin{cases} i \frac{\partial}{\partial t} u + \nabla u + |u|^2 u = 0, & \text{in } \Omega \times \mathcal{T}, \\ u(t_0, x; \eta) = u^0(x; \eta), & \text{in } \Omega, \end{cases} \quad (4.7.11)$$

with periodic boundary conditions, and vector-valued parameter η . By writing the complex-valued solution u in terms of its real and imaginary parts as $u = q + iv$, (4.7.11) can be written as a Hamiltonian system in canonical symplectic form with Hamiltonian

$$\mathcal{H}(q, v; \eta) = \frac{1}{2} \int_{\Omega} \left[\sum_{i=1}^d \left(\left(\frac{\partial}{\partial x_i} q \right)^2 + \left(\frac{\partial}{\partial x_i} v \right)^2 \right) - \frac{1}{2} (q^2 + v^2)^2 \right] dx.$$

Two-dimensional nonlinear Schrödinger equations

Let us consider the spatial domain $\Omega = [-2\pi, 2\pi]^2$ and the set of parameters $\Gamma = [0.97, 1.03]^2$. We seek the numerical solution to (4.7.11), for $p = 64$ uniformly sampled parameters $\eta_h := (\alpha, \beta) \in \Gamma_h$ entering the initial condition

$$u^0(x, y; \eta_h) = (1 + \alpha \sin x)(2 + \beta \sin y). \quad (4.7.12)$$

This problem is characterized by an energy exchange between Fourier modes. Although this process is local, it is not well understood how the energy exchange mechanism is influenced by the problem dimension and parameters. In particular, although the values of α and β have a limited impact on the low-rank structure of the initial condition (4.7.12), the explicit effect of

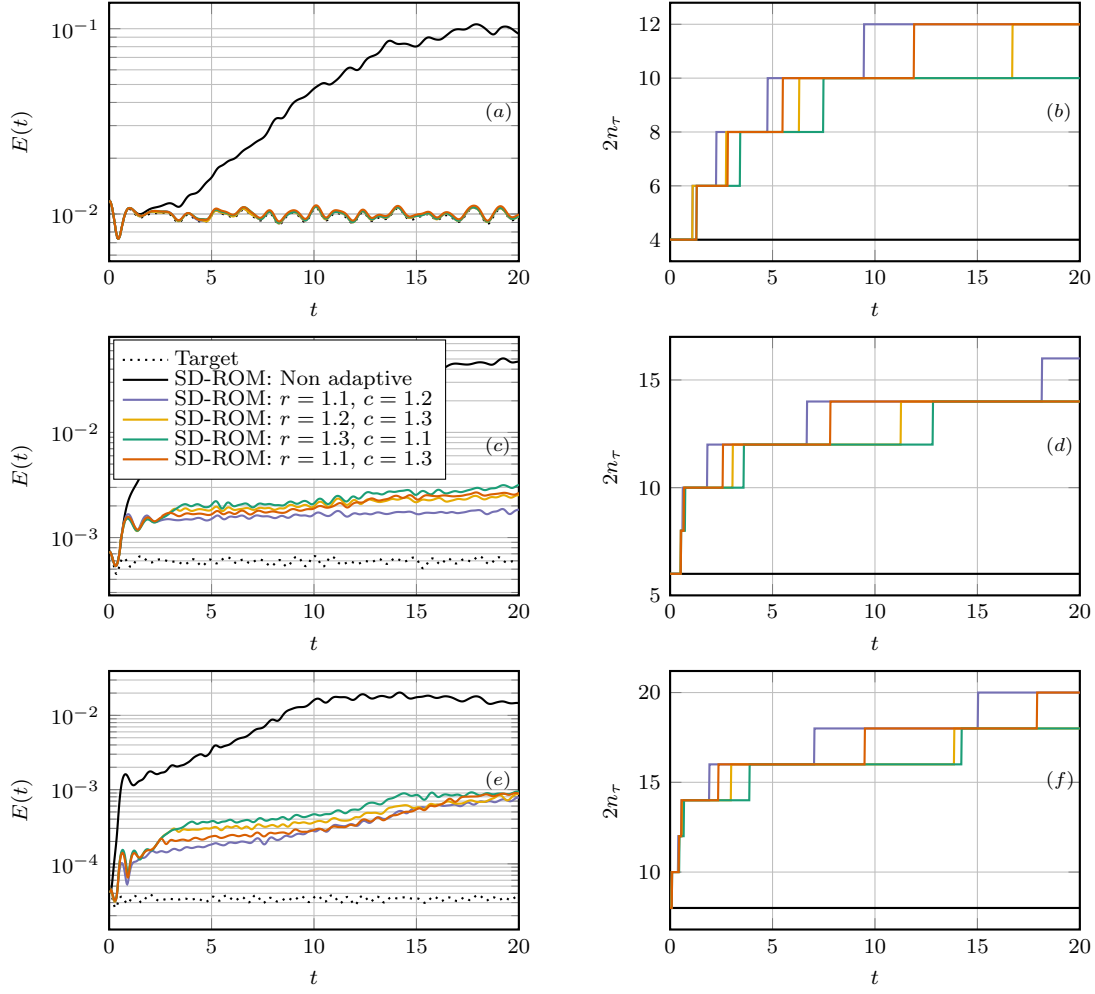


Figure 4.10: SWE-2D: On the left column, we report the evolution of the error $E(t)$ (4.7.3) for the adaptive and non adaptive dynamical RB methods for different values of the control parameters r and c , and for different dimensions $2n_1$ of the initial reduced manifold. The target error is obtained by solving the full model with initial condition obtained by projecting (4.7.10) onto a symplectic manifold of dimension $2n_1$. On the right column, we report the evolution of the dimension of the dynamical reduced basis over time. The adaptive algorithm is driven by the error indicator (4.5.4), while in the non adaptive setting, the dimension does not change with time. We consider the cases $2n_1 = 4$ (Figures (a)-(b)), $2n_1 = 6$ (Figures (c)-(d)), and $2n_1 = 8$ (Figures (e)-(f)).

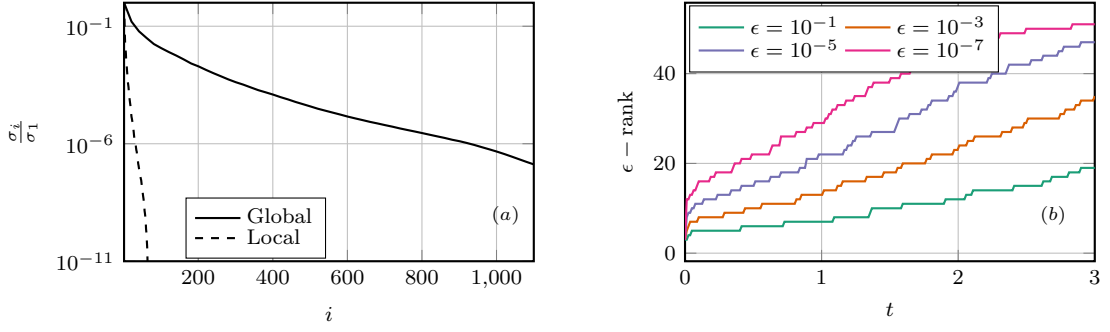


Figure 4.11: NLS-2D: (a) Singular values of the global snapshots matrix S^{u_h} and of the time average of the local trajectories matrix $S_{\tau}^{u_h}$. The singular values are normalized using the largest singular value for each case. (b) ϵ -rank of the local trajectories matrix $S_{\tau}^{u_h}$ for different values of ϵ .

their variation on the energy exchange process is not known. We use a centered finite difference scheme to discretize the Laplacian operator. The domain Ω is partitioned using $M = 101$ nodes per dimension, for a total of $N = 10000$ intervals of width $\Delta x = \Delta y = 4\pi \cdot 10^{-2}$. Let $u_h(t; \eta_h)$, for all $t \in \mathcal{T}$ and $\eta_h \in \Gamma_h$, be the vector collecting the degrees of freedom associated with the nodal approximation of u . The semi-discrete problem is canonically Hamiltonian with the discrete Hamiltonian function

$$\mathcal{H}_h(u_h; \eta_h) = \frac{1}{2} \sum_{i=1}^N \left[\left(\frac{q_{i+1,j} - q_{i,j}}{\Delta x} \right)^2 + \left(\frac{v_{i+1,j} - v_{i,j}}{\Delta x} \right)^2 + \left(\frac{q_{i,j+1} - q_{i,j}}{\Delta y} \right)^2 + \left(\frac{v_{i,j+1} - v_{i,j}}{\Delta y} \right)^2 - \frac{1}{2} (q_{i,j}^2 + v_{i,j}^2)^2 \right],$$

with periodic boundary conditions for $q_{i,j}$ and $v_{i,j}$. We consider $N_t = 12000$ time steps in the interval $\mathcal{T} = (0, T := 3]$ so that $\Delta t = 2.5 \cdot 10^{-4}$. As in the previous examples, the implicit midpoint rule is used as the numerical integrator in the high-fidelity solver. The reduced dynamics is integrated using the 2-stage partitioned RK method.

To assess the reducibility of the problem, we collect in $S^{u_h} \in \mathbb{R}^{2N \times (N_t p)}$ the snapshots associated with all parameters η_h and times t^τ , and in $S_{\tau}^{u_h} \in \mathbb{R}^{2N \times p}$ the snapshots associated with all parameters η_h at fixed time t^τ , with $\tau = 1, \dots, N_t$. The slow decay of the singular values of S^{u_h} , reported in Figure 4.11(a), suggests that a global reduced basis approach is not viable for model order reduction. The growing complexity of the high-fidelity solution, associated with different values of α and β , is reflected by the growth of the numerical rank shown in Figure 4.11(b). Hence, despite the exponential decay of the singular values of $S_{\tau}^{u_h}$, Figure 4.11(b) indicates that this test represents a challenging problem even for the adaptive algorithm and a balance between accuracy and computational cost is necessary while adapting the dimension of the reduced manifold. We consider several combinations of $r \in \{1.1, 1.2\}$ and $c \in \{1.05, 1.1, 1.2\}$ and different initial dimensions of the reduced manifold $2n_1 \in \{6, 8\}$. The error indicator is computed every 10 time steps on a subset $\Gamma_I \subset \Gamma_h$ of 16 uniformly sampled parameters. Both adaptive and non-adaptive reduced models are initialized using (4.7.9), with U_0 obtained via a complex SVD of the snapshots matrix $S_1^{u_h}$ of the initial condition (4.7.12). Figure 4.12 confirms that the evolving basis U generated by the dynamical reduced basis method satisfies the orthogonality and symplecticity constraints to machine precision. In line with the fact that the full model solution

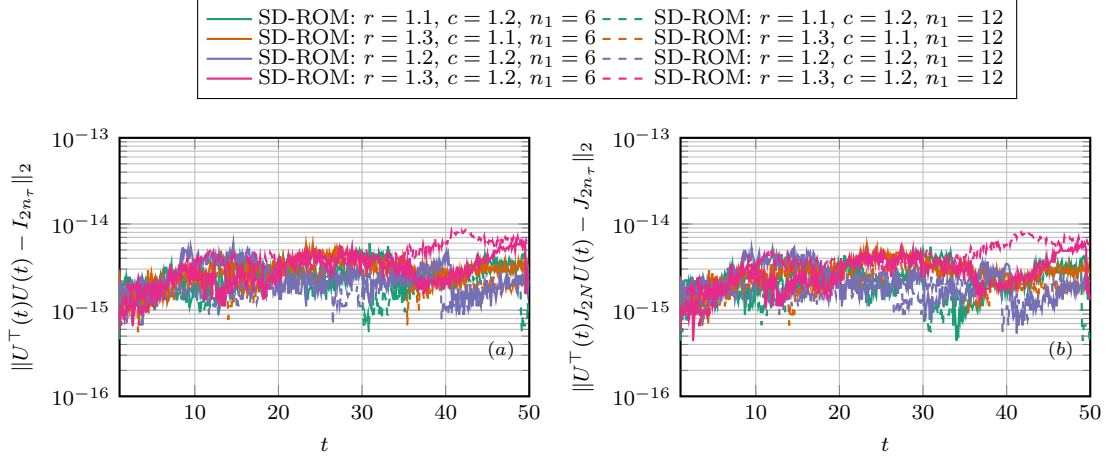


Figure 4.12: NLS-2D: Evolution of the error in the orthogonality (a) and symplecticity (b) of the reduced basis obtained with the adaptive dynamical RB method for different choices of the control parameters r and r and initial dimension of the reduced manifold $2n_1$.

has a gradually increasing rank (see Figure 4.11 (b)), adapting the dimension of the basis improves the accuracy of the approximation, as shown in Figure 4.13. In terms of the computational cost of the adaptive dynamical model, we record a speedup of at least 58 times with respect to the high-fidelity model, whose runtime is $6.2 \cdot 10^5$ s. These results can be explained as for the 2D shallow water test: in the presence of polynomial nonlinearities the strategy proposed in Section 4.6.1 allows computational costs that scale only linearly with N .

In Figure 4.13, we observe that, although increasing in time, the error associated with the adaptive reduced dynamical model has a smaller slope than the error of the non-adaptive method.

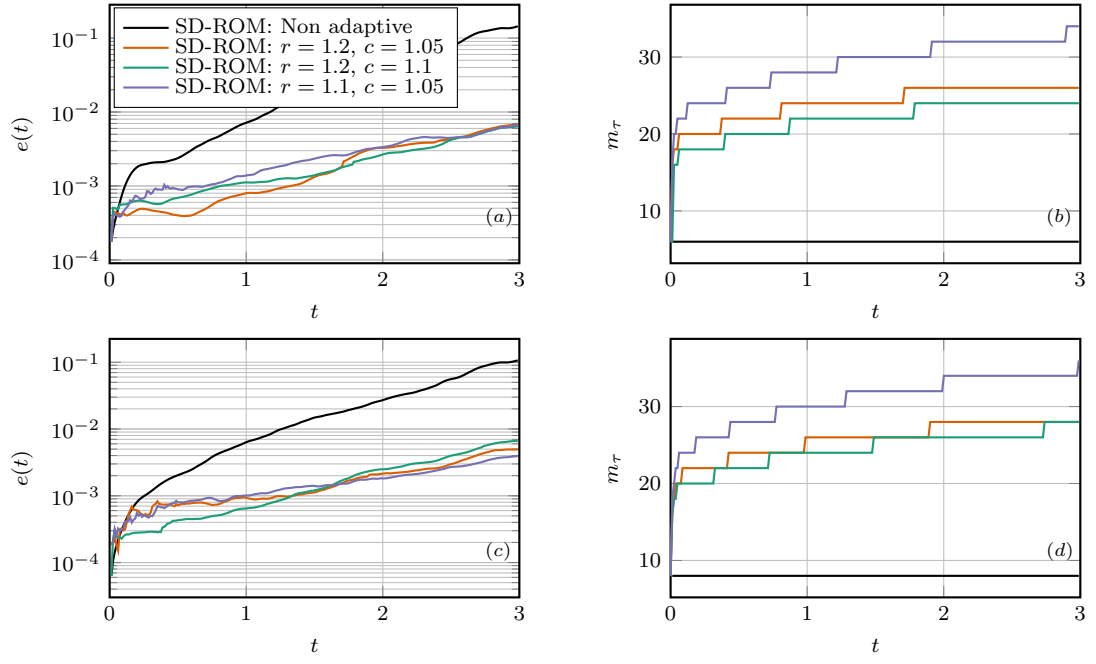


Figure 4.13: NLS-2D: On the left column, we report the evolution of the error $E(t)$ (4.7.3) for the adaptive and non adaptive dynamical RB methods for different values of the control parameters r and c , and for different dimensions $2n_1$ of the initial reduced manifold. On the right column, we report the evolution of the dimension of the dynamical RB over time. We consider the cases $2n_1 = 6$ (Figures (a)-(b)) and $2n_1 = 8$ (Figures (c)-(d)).

5 Model order reduction of the Vlasov equation

5.1 Introduction

The kinetic modeling of collisionless magnetized plasmas is based on the Vlasov–Maxwell equations, which describe the evolution of the distribution function of a collection of charged particles under the action of self-consistent electromagnetic fields. Because of the high dimensionality of the phase space, the large separation of scales, the inherent nonlinearity, and the infinitely many conserved quantities, the numerical treatment of the Vlasov–Maxwell equations, and of its electrostatic limit of Vlasov–Poisson equations is a challenging task.

Arguably, the most widely used family of numerical methods for the solution of kinetic plasma models are Particle-In-Cell (PIC) methods [32]. The idea of PIC schemes is to sample the distribution function in velocity space using a finite number of macro-particles that are evolved along their characteristics. The electromagnetic fields are discretized on a grid in the computational domain, and the macro-particles move through the grid according to the Lorentz force. To preserve key physical properties of the problem, such as conservation of total energy, PIC schemes have evolved into variational algorithms based on least action principles [159; 163; 237] or on the Hamiltonian formulation of the Vlasov–Maxwell and Vlasov–Poisson equations [172; 86]. In parallel, several numerical methods [237; 86] for kinetic plasma models have leveraged discrete differential forms and de Rham complexes for the geometric approximation of the electromagnetic fields through Maxwell’s equations. Combining these ideas of structure-preserving approximations has led to finite element PIC methods able to exactly satisfy physical constraints, like the Gauss laws, and guarantee the preservation of the Hamiltonian structure of the problem. Examples include the canonical [207] and non-canonical [266] symplectic particle-in-cell algorithms, the Hamiltonian particle-in-cell method of [124], and its generalization, the Geometric Electromagnetic PIC (GEMPIC) method [152].

The multiscale nature of plasmas implies that PIC codes require a significant amount of computational resources to resolve the shortest length scale and the fastest plasma frequency and, thus, to yield stable and accurate numerical approximations. Moreover, the slow convergence rate of particle-based methods necessitates the use of many particles to achieve sufficient accuracy, to capture, for example, time-dependent physical phenomena such as plasma instabilities. As a result, PIC methods can have prohibitively expensive computational cost. Furthermore, the computational burden can become intractable in the parametric case when simulations for many input parameters are of interest. This problem has been tackled from an algorithmic standpoint by improving algorithms’ structure and using suitable computational hardware. In this work,

we propose to address this computational issue through model order reduction. Starting from the high-resolution geometric PIC approximation of the Vlasov–Poisson problem, the idea is to derive a low-dimensional surrogate model that can be solved at a reduced computational cost and still provides accurate approximate solutions.

In recent years, some effort has been devoted to the development of numerical methods for the infinite-dimensional Vlasov–Poisson problem based on model order reduction techniques and low-rank approximations, with the intent to optimize the number of degrees of freedom needed for a sufficiently accurate and stable approximation of the solution. In [82], the solution of the full-Eulerian time-dependent Vlasov–Poisson system is approximated using a tensor decomposition whose rank is adapted at each time step. In [84], the continuous distribution function of the Vlasov–Poisson problem is expanded into a finite sum of low-rank factors, for which a new dynamical system is derived. A conservative discretization in space and velocity of the resulting problem yields a low-rank approximation of the original dynamics. To address the problem of the fluid closure for the collisionless linear Vlasov system, an interpolatory order reduction is proposed in [101]. In the context of a particle-based discretization of kinetic plasma models, a dynamic mode decomposition (DMD) strategy has been proposed in [184] to reconstruct the electric field within an Electromagnetic particle-in-cell (EMPIC) algorithm. Although the proposed approach can effectively capture and extrapolate the electric field behavior around equilibria, the computational burden associated with the high number of particles is not overcome. Model order reduction — in the number of particles — of *parametric* plasma models is, to the best of our knowledge, an open problem. This work aims at addressing this issue.

We focus on the parametric 1D-1V Vlasov–Poisson problem and consider its discretization in space and velocity via a geometric PIC method. The resulting dynamical system has a Hamiltonian form and, hence, corresponds to a symplectic flow. Model order reduction of the Vlasov–Poisson problem poses some major challenges, and standard reduced basis techniques are prone to fail in terms of numerical stability, computational efficiency, and accuracy of the simulations. The application [252] of the symplectic reduced basis methods proposed in [198; 3] to the Vlasov equation, with a fixed external electrostatic field, shows the importance of retaining the symplectic structure of Vlasov’s equation in the reduced model. However, the multi-scale nature of the problem makes it difficult for a reduced-order model to characterize, with sufficient accuracy, the plasma behavior using a small number of degrees of freedom. This implies that accurate reduced representations of the solution may require large approximation spaces that eliminate model order reduction benefits. To overcome this limitation, we adapt the method described in Chapter 4. This approach accurately describes the plasma evolution with a considerably reduced number of particles without compromising the simulation quality. Furthermore, the bulk of the computational effort to solve the reduced dynamics is due to the nonlinearity of the particles-to-grid mapping, and thus the Hamiltonian, whose evaluation needs to be performed in the original high-dimensional space. To alleviate these computational inefficiencies, we propose a strategy that approximates the reduced Hamiltonian gradient via a combination of hyper-reduction techniques and parameter sampling procedures. A reduction in the computational runtimes of the algorithm is achieved by decoupling the operations that depend on the number of particles from those that depend on the number of parameters while retaining an accurate representation of the plasma dynamics. The resulting discrete dynamical system preserves the symplectic structure of the problem, ensures the stability of the approximation, and exploits the local-in-time low-rank nature of the solution by using the dynamical low-rank method described in Chapter 4.

The remainder of this Chapter is organized as follows. In Section 5.2, the parametric Vlasov–Poisson problem is introduced both in its classical Eulerian formulation and in the Hamiltonian formulation. Moreover, the semi-discrete approximation of the problem via a particle method

coupled with a finite element discretization of the Poisson problem is described. In Section 5.3, the model order reduction of the parametric dynamical system originating from the semi-discretization of the Vlasov–Poisson equation is considered. We apply the nonlinear structure-preserving approach given in Chapter 4 and we set the notation for the following chapters. After discussing the computational complexity of the dynamical reduced basis algorithm, Section 5.4 is devoted to the DMD-DEIM structure-preserving approximation of the nonlinear Hamiltonian gradient and the particles-to-grid mapping, which is the author’s novel contribution in the thesis. Numerical experiments in Section 5.5 on benchmark tests show that the proposed method can accurately reproduce the dynamics of particle-based kinetic plasma models with significant speedups compared to solving the original system.

5.2 The physical model

We consider the parametric Vlasov–Poisson problem with parameters that describe physical properties of the system. In particular, we focus on the study of the effect of parametrized initial distributions on the plasma dynamics. Let us assume that the parameters range in a compact set $\Gamma \subset \mathbb{R}^q$ with $q \geq 1$. The plasma, at any time $t \in \mathcal{T}$, is described in terms of the distribution function $f^s(t, x, v; \eta)$ in the Cartesian phase space domain $(x, v) \in \Omega := \Omega_x \times \Omega_v \subset \mathbb{R}^2$. Here s denotes the particle species, i.e., ion and electron in our case. Assume that $\Omega_x := \mathbb{T} = \mathbb{R}/(2\pi\mathbb{Z})$ is the one dimensional torus and $\Omega_v := \mathbb{R}$. For $\eta \in \Gamma$ fixed, we introduce the space

$$\mathcal{V}_\eta := \left\{ f(t, \cdot, \cdot; \eta) \in L^2(\Omega) : f(t, x, v; \eta) > 0 \text{ for all } (x, v) \in \Omega, f(t, \cdot, v; \eta) \sim e^{-v^2} \text{ as } |v| \rightarrow \infty \right\}.$$

The 1D-1V Vlasov-Poisson problem reads: For each $\eta \in \Gamma$ and $f_0^s(\eta) \in \mathcal{V}_\eta|_{t=0}$, find $f^s(\cdot, \cdot, \cdot; \eta) \in C^1(\mathcal{T}; L^2(\Omega)) \cap C^0(\mathcal{T}; \mathcal{V}_\eta)$, and the electric field $E(\cdot, \cdot; \eta) \in C^0(\mathcal{T}; L^2(\Omega))$ such that

$$\begin{cases} \frac{\partial}{\partial t} f^s(t, x, v; \eta) + v \frac{\partial}{\partial x} f^s(t, x, v; \eta) + \frac{q^s}{m^s} E(t, x; \eta) \frac{\partial}{\partial v} f^s(t, x, v; \eta) = 0, & \text{in } \Omega \times \mathcal{T}, \forall s, \\ \frac{\partial}{\partial x} E(t, x; \eta) = \sum_s q^s \int_{\Omega_v} f^s(t, x, v; \eta) dv, & \text{in } \Omega_x \times \mathcal{T}, \\ f^s(0, x, v; \eta) = f_0^s(x, v; \eta), & \text{in } \Omega. \end{cases}$$

Here q^s is the charge and m^s is the particle mass. The boundary conditions for f^s are periodic in space and prescribed via the space \mathcal{V}_η in velocity.

Since the electric field can be written as the spatial derivative of the electric potential ϕ , namely $E(t, x; \eta) = -\frac{\partial}{\partial x} \phi(t, x; \eta)$, the Vlasov-Poisson problem can be recast, for each $\eta \in \Gamma$, as

$$\begin{cases} \frac{\partial}{\partial t} f^s(t, x, v; \eta) + v \frac{\partial}{\partial x} f^s(t, x, v; \eta) - \frac{q^s}{m^s} \frac{\partial}{\partial x} \phi(t, x; \eta) \frac{\partial}{\partial v} f^s(t, x, v; \eta) = 0, & \text{in } \Omega \times \mathcal{T}, \forall s, \\ -\frac{\partial^2}{\partial x^2} \phi(t, x; \eta) = \rho(t, x; \eta) := \sum_s q^s \int_{\Omega_v} f^s(t, x, v; \eta) dv & \text{in } \Omega_x \times \mathcal{T}. \end{cases} \quad (5.2.1)$$

The Lagrangian and Hamiltonian formulation of the Vlasov–Poisson and Vlasov–Maxwell equations reveals a set of mathematical and geometric features that encode the physical properties of these systems. The Vlasov–Poisson problem admits a Hamiltonian formulation [52], with a

Lie-Poisson bracket, and Hamiltonian given by the sum of the kinetic and electric energy as

$$\mathcal{H}(f, \eta) = \sum_s \frac{m_s}{2} \int_{\Omega} v^2 f^s(t, x, v; \eta) dx dv + \frac{1}{2} \int_{\Omega_x} |E(t, x; \eta)|^2 dx. \quad (5.2.2)$$

Eulerian-based discretizations of kinetic plasma models in Hamiltonian form, with general noncanonical Poisson brackets, do not appear to inherit the phase space structure of the continuous problem, as has been observed in [182; 229]. On the contrary, particle-in-cell methods have led to the geometric approximation of these models when coupled to the discretization of the electromagnetic fields via discrete differential forms [266; 124; 152; 82]. For the structure-preserving approximation and reduction of the Vlasov–Poisson system (5.2.1) we rely on the Hamiltonian structure of its semi-discrete formulation obtained via particle-based methods as derived in the following.

5.2.1 Geometric particle-based discretization

We consider a particle method for the approximation of the Vlasov equation, coupled with a H^1 -conforming discretization of the Poisson problem for the electric potential. In detail, the distribution function f^s is approximated by the superposition of N computational macro-particles as

$$f^s(t, x, v; \eta) \approx f_h^s(t, x, v; \eta) = \sum_{l=1}^N \omega_l^s S(x - X_l^s(t, \eta)) \delta(v - V_l^s(t, \eta)),$$

where $\omega_l^s \in \mathbb{R}$ is the weight of the l -th particle, δ is the Dirac delta, S is a compactly supported shape function, and, for each $\eta \in \Gamma$ and $t \in \mathcal{T}$, $X^s(t, \eta) \in \mathbb{R}^N$ and $V^s(t, \eta) \in \mathbb{R}^N$ denote the vectors of the position and velocity of the macro-particles, respectively. The idea of particle methods is to derive the time evolution of the approximate distribution function f_h^s by advancing the macro-particles along the characteristics of the Vlasov equation, i.e. the particles' positions and velocities satisfy the following set of ordinary differential equations

$$\begin{cases} \frac{d}{dt} X_l^s(t, \eta) = V_l^s(t, \eta), & \text{in } \mathcal{T}, \forall s, \\ \frac{d}{dt} V_l^s(t, \eta) = \frac{q^s}{m^s} E(t, X_l^s(t, \eta); \eta), & \text{in } \mathcal{T}, \forall s, \end{cases} \quad (5.2.3)$$

under suitable initial conditions. The macro-particles move through a computational grid under the influence of electromagnetic fields. The latter are self-consistently calculated from the positions of the particles on the grid via the Poisson equation (5.2.1). On a partition of the spatial domain Ω_x , we consider a finite element discretization of the Poisson equation in the space $\mathcal{P}_k \Lambda^0(\Omega_x) \subset H^1(\Omega_x)$ of continuous piecewise polynomial functions of degree at most $k \geq 1$. The semi-discrete variational problem reads: For each $\eta \in \Gamma$, find $\phi_h(\cdot; \eta) \in C^1(\mathcal{T}; \mathcal{P}_k \Lambda^0(\Omega_x))$ such that

$$a_h(\phi_h(t, \cdot; \eta), \psi) = g(\psi), \quad \forall \psi \in \mathcal{P}_k \Lambda^0(\Omega_x), \quad (5.2.4)$$

where the bilinear form a_h corresponds to the Laplace operator, while the linear function g is associated with the density ρ ; thereby

$$a_h(\varphi, \psi) := \int_{\Omega_x} \frac{d}{dx} \varphi(x) \frac{d}{dx} \psi(x) dx, \quad g(\psi) := \sum_s q^s \int_{\Omega_v} f_h^s(t, x, v; \eta) \psi(x) dv dx,$$

5.3 Model order reduction of the Vlasov-Poisson problem

for all $\varphi, \psi \in \mathcal{P}_k \Lambda^0(\Omega_x)$. Then the time-dependent algebraic system ensuing from (5.2.4) reads

$$L\phi(t, \eta) = \sum_s \Lambda^0(X^s(t, \eta))^\top M_q^s =: \rho_h(X(t, \eta); \eta), \quad (5.2.5)$$

where $M_q^s \in \mathbb{R}^N$ is the vector of entries $(M_q^s)_l = q^s \omega_l^s$, for $l = 1, \dots, N$. The proposed discretization of the electromagnetic field allows to recast the characteristic equation (5.2.3) as a Hamiltonian system. As discussed in Chapter 3, the phase space of Hamiltonian systems is characterized by a symplectic geometric structure. We denote with $\mathcal{V}_{2N} \subset \mathbb{R}^{2N}$ the phase space of (5.2.3) and we assume it is a $2N$ -dimensional symplectic vector space.

For a species s , let $W^s(t, \eta) = [X^s(t, \eta); V^s(t, \eta)] \in \mathcal{V}_{2N}$ denote the vector of all particle positions and velocities at a given time $t \in \mathcal{T}$ and parameter value $\eta \in \Gamma$, obtained by concatenating the vectors $X^s(t, \eta)$ and $V^s(t, \eta)$. The latter are also known as generalized position and momentum in the symplectic formalism. For N_s number of species, we denote with $W \in \mathcal{V}_{2N}^{N_s} \subset \mathbb{R}^{2N N_s}$ the state vector collecting the positions and velocities of all particles of all species. The Hamiltonian (5.2.2), resulting from the proposed discretization, is given by

$$\begin{aligned} \mathcal{H}(W(t, \eta)) &= \sum_s \sum_{l=1}^N \frac{m_l^s}{2} \omega_l^s V_l^s(t, \eta) + \frac{1}{2} \int_{\Omega_x} \left| \frac{\partial}{\partial x} \phi_h(t, x; \eta) \right|^2 dx \\ &= \frac{1}{2} \sum_s V^s(t, \eta)^\top M_p^s V^s(t, \eta) + \frac{1}{2} \phi_h(t, \eta)^\top L \phi_h(t, \eta)^\top \\ &= \frac{1}{2} \sum_s V^s(t, \eta)^\top M_p^s V^s(t, \eta) + \sum_{s, s'} \frac{1}{2} (M_q^s)^\top \Lambda^0(X^s(t, \eta)) L^{-1} \Lambda^0(X^{s'}(t, \eta))^\top M_q^{s'}, \end{aligned} \quad (5.2.6)$$

where $M_p^s = \text{diag}(m_1^s \omega_1^s, \dots, m_N^s \omega_N^s) \in \mathbb{R}^{N \times N}$, with $\text{diag}(d)$ denoting a diagonal matrix with entries given by the vector d . Differentiating the discrete Hamiltonian in (5.2.6) with respect to the vector X^s of particles' positions and the vector V^s of particles' velocities, results in the semi-discrete system in Hamiltonian form

$$\begin{pmatrix} \frac{d}{dt} X(t, \eta) \\ \frac{d}{dt} V(t, \eta) \end{pmatrix} = J_{2N} \begin{pmatrix} (M_p^s)^{-1} & 0 \\ 0 & (M_p^s)^{-1} \end{pmatrix} \begin{pmatrix} \text{diag}(M_q^s) \nabla \Lambda^0(X^s(t, \eta)) L^{-1} \sum_{s'} \Lambda^0(X^{s'}(t, \eta))^\top M_q^{s'} \\ M_p^s V^s(t, \eta) \end{pmatrix}, \quad \forall s. \quad (5.2.7)$$

Here $\nabla \Lambda^0(X^s) \in \mathbb{R}^{N \times N_x}$ is defined as $(\nabla \Lambda^0(X^s))_{l,i} := (d_x \lambda_i^0)(X_l^s)$, for $i = 1, \dots, N_x$ and $l = 1, \dots, N$. A generalization of this discretization to the case of Vlasov-Maxwell's equations leads to the GEMPIC method introduced in [152]. Note that the proposed semi-discretization preserves the Hamiltonian, which corresponds to the discrete energy of the system, but not the momentum. In PIC it does not appear possible to simultaneously conserve momentum and energy [32].

5.3 Model order reduction of the Vlasov-Poisson problem

For better readability we restrict ourselves to the case of a single-species plasma model and, thus, drop the superscript s . Moreover, we assume homogeneous macro-particles' weights so that $m_l = m$ and $\omega_l = \omega$, for all $l = 1, \dots, N$. The proposed method can be extended to the general case *mutatis mutandis*.

To simplify the notation, we re-define the Hamiltonian \mathcal{H} from (5.2.6) up to the constant

$m_p := (m\omega)^{-1}$, such that, for any $W(t, \eta) = [X(t, \eta); V(t, \eta)] \in \mathcal{V}_{2N}$,

$$\mathcal{H}(W(t, \eta)) = \frac{1}{2} V(t, \eta)^\top V(t, \eta) + \frac{m_p^{-1}}{2} M_q^\top \Lambda^0(X(t, \eta)) L^{-1} \Lambda^0(X(t, \eta))^\top M_q, \quad (5.3.1)$$

We also introduce the (nonlinear) electric energy $\mathcal{E} : \mathbb{R}^{2N} \times \Gamma \rightarrow \mathbb{R}$ defined as

$$\mathcal{E}(X(t, \eta); \eta) := \frac{m_p^{-1}}{2} M_q^\top \Lambda^0(X(t, \eta)) L^{-1} \Lambda^0(X(t, \eta))^\top M_q. \quad (5.3.2)$$

The parametric Hamiltonian system (5.2.7) reads: For each $\eta \in \Gamma$ and for $W_0(\eta) = [X(0, \eta); V(0, \eta)] \in \mathcal{V}_{2N}$, find $W(\cdot, \eta) \in C^1(\mathcal{T}, \mathcal{V}_{2N})$ such that

$$\begin{cases} \frac{d}{dt} W(t, \eta) = J_{2N} \nabla_W \mathcal{H}(W(t, \eta)), & \text{in } \mathcal{T}, \\ W(0, \eta) = W_0(\eta), \end{cases}$$

where the initial condition $W_0(\eta) \in \mathcal{V}_{2N}$ is prescribed by the initial distribution $f_0(\eta)$.

5.3.1 Dynamical structure-preserving MOR

The non-dissipative nature of the Hamiltonian problem (5.3.1) is associated with poor reducibility and a slowly decaying Kolmogorov n -width. If traditional symplectic reduction approaches are employed, such as the ones described in Chapter 3, the approximation of the solution to (5.3.1) may require large approximation spaces that jeopardize the benefits of model order reduction. Hence, we adopt the structure-preserving dynamic RB method for Hamiltonian system described in Chapter 4.

Assume we seek to solve the semi-discrete Vlasov-Poisson problem (5.3.1) for p parameters $\Gamma_h := \{\eta_j\}_{j=1}^p \subset \Gamma$. Similarly to Section 4.2, we recast (5.3.1) as an evolution equation in the matrix-valued unknown $\mathcal{R}(t) := [W(t, \eta_1) | \dots | W(t, \eta_p)] \in \mathcal{V}_{2N}^p \subset \mathbb{R}^{2N \times p}$, with $\mathcal{V}_{2N}^p := \mathcal{V}_{2N} \times \dots \times \mathcal{V}_{2N}$. Given $\mathcal{R}_0 = [W_0(\eta_1) | \dots | W_0(\eta_p)]$, we look for $\mathcal{R} \in C^1(\mathcal{T}, \mathcal{V}_{2N}^p)$, such that

$$\begin{cases} \frac{d}{dt} \mathcal{R}(t) = J_{2N} \nabla_{\mathcal{R}} \mathcal{H}^p(\mathcal{R}(t)), & \text{in } \mathcal{T}, \\ \mathcal{R}(0) = \mathcal{R}_0. \end{cases} \quad (5.3.3)$$

The Hamiltonian of (5.3.3) is a vector-valued quantity $\mathcal{H}^p : \mathcal{V}_{2N}^p \rightarrow \mathbb{R}^p$ that collects the values of the Hamiltonian of (5.3.1) for each parameter in Γ_h , namely $(\mathcal{H}^p(\mathcal{R}(\cdot)))_j = \mathcal{H}(W(\cdot, \eta_j))$, for $j = 1, \dots, p$. Moreover, the gradient $\nabla_{\mathcal{R}} \mathcal{H}^p(\mathcal{R}(t)) \in \mathcal{V}_{2N}^p$ is defined as

$$(\nabla_{\mathcal{R}} \mathcal{H}^p(\mathcal{R}(t)))_{l,j} = \frac{\partial}{\partial W_l(\cdot, \eta_j)} \mathcal{H}(W(\cdot, \eta_j)), \quad l = 1, \dots, 2N, j = 1, \dots, p.$$

The idea of dynamical low-rank approximations is to expand the full-order solution in a truncated modal decomposition where both the basis and the expansion coefficients are time-dependent. For all $t \in \mathcal{T}$, we approximate $W(t)$, in (5.3.3), as $R(t) = U(t)Z(t)$, where $U(t) \in \mathbb{R}^{2N \times 2n}$ is the time-dependent orthosymplectic basis and $Z(t) \in \mathbb{R}^{2N \times n}$ are the associated expansion coefficient,

5.3 Model order reduction of the Vlasov-Poisson problem

with $Z(t) := [Z_1(t) | \dots | Z_p(t)]$, and $Z_i(t) := Z(t, \eta_i)$. The reduced space is then defined as

$$\mathcal{M}_{2n}^p := \{R \in \mathbb{R}^{2N \times p} : R = UZ \text{ with } U \in \mathbb{S}(2n, \mathbb{R}^{2N}), Z \in \mathcal{Z}\},$$

where $\mathbb{S}(2n, \mathbb{R}^{2N})$ is the set of ortho-symplectic matrices in $\mathbb{R}^{2N \times 2n}$ defined in (3.3.2) and

$$\mathcal{Z} := \{Z \in \mathbb{R}^{2n \times p} : \text{rank}(ZZ^\top + J_{2n}ZZ^\top J_{2n}) = 2n\},$$

with $n \ll N$ and $2n < p$. Let $X_r^i(t) := U_X(t)Z_i(t) \in \mathbb{R}^N$ and $V_r^i(t) := U_V(t)Z_i(t) \in \mathbb{R}^N$ denote the reduced position and velocity vectors, respectively, associated with the parameter η_i , for $i = 1, \dots, N$ and $j = 1, \dots, 2n$. For fixed U and for each parameter $\eta_i \in \Gamma_h$, the flow map of the coefficient equation is a canonical symplectic map with a Hamiltonian $\mathcal{H}_U(Z(t)) := \mathcal{H}(U(t)Z(t))$ having the i -th entry equal to

$$\mathcal{H}_{U,i}(Z_i(t)) := \mathcal{H}(U(t)Z_i(t)) = \frac{1}{2}V_r^i(t, \eta)^\top V_r^i(t, \eta) + \mathcal{E}_{U,i}(Z_i(t)), \quad (5.3.4)$$

where the first part is quadratic in the coefficients Z_i , and $\mathcal{E}_{U,i} : \mathbb{R}^N \rightarrow \mathbb{R}$ is the nonlinear electric energy component (5.3.2) of the Hamiltonian function, i.e., for all $i \in p$,

$$\mathcal{E}_{U,i}(Z_i(t)) = \mathcal{E}(X_r^i; \eta_i) = \frac{m_p^{-1}}{2} M_q^\top \Lambda^0 (X_r^i) L^{-1} \Lambda^0 (X_r^i)^\top M_q. \quad (5.3.5)$$

We introduce the matrices

$$G_{\mathcal{H}}^p(U, Z) := [G_{\mathcal{H}}^p(U, Z_1) | \dots | G_{\mathcal{H}}^p(U, Z_p)] \in \mathbb{R}^{2N \times p}, \quad (5.3.6)$$

and

$$g_{\mathcal{H}}^p(U, Z) := [g_{\mathcal{H}}^p(U, Z_1) | \dots | g_{\mathcal{H}}^p(U, Z_p)] \in \mathbb{R}^{2n \times p},$$

having as columns the p instances of the gradient of the Hamiltonian and of the reduced Hamiltonian, respectively,

$$G_{\mathcal{H}}^p(U, Z_i) := \nabla_{U Z_i} \mathcal{H}_{U,i}(Z_i) \in \mathbb{R}^{2N},$$

and

$$g_{\mathcal{H}}^p(U, Z_i) := U^\top G_{\mathcal{H}}^p(U, Z_i) = \nabla_{Z_i} \mathcal{H}_{U,i}(Z_i) \in \mathbb{R}^{2n},$$

where

$$\nabla_{U Z_i} \mathcal{H}_{U,i}(Z_i(t)) = \begin{pmatrix} m_p^{-1} \text{diag}(M_q) \nabla \Lambda^0 (X_r^i(t)) L^{-1} \Lambda^0 (X_r^i(t)) M_q \\ V_r^i(t) \end{pmatrix}. \quad (5.3.7)$$

Under these assumptions, a dynamical system for the reduced solution is characterized via the symplectic projection of the velocity field of the full dynamical system (5.3.3) onto the tangent space of \mathcal{M}_{2n}^p at each state [189; 183]. The resulting reduced dynamics is given in terms of evolution equations for the reduced basis and for the expansion coefficients as

$$\begin{cases} \frac{d}{dt} Z(t) = J_{2n} g_{\mathcal{H}}^p(U, Z) = J_{2n} \nabla_Z \mathcal{H}_U(Z), & (5.3.8a) \end{cases}$$

$$\begin{cases} \frac{d}{dt} U = (\mathbb{I}_{2N} - U U^\top) (J_{2N} G_{\mathcal{H}}^p(U, Z) Z^\top - G_{\mathcal{H}}^p(U, Z) Z^\top J_{2n}^\top) S(Z)^{-1} & (5.3.8b) \end{cases}$$

$$\begin{cases} U(t_0) Z(t_0) = U^0 Z^0, & (5.3.8c) \end{cases}$$

where $S(Z) = ZZ^\top + J_{2n}^\top ZZ^\top J_{2n}$, and the initial condition $U^0 Z^0 \in \mathcal{M}_{2n}^p$ is computed via a truncated complex SVD of $\mathcal{R}_0 \in \mathbb{R}^{2N \times p}$. Equation (5.4.2) describes the evolution of the coefficients $Z(t)$ and is a system of p independent equations, each in n unknowns. It corresponds to the Galerkin projection of the full-order Hamiltonian systems onto the space spanned by the columns of $U(t)$, as obtained with the global symplectic reduced basis method in (3.2.9). Here, however, the basis U changes in time, and the evolution problem (5.3.8b), for the basis U , is a matrix-valued problem in $N \times n$ unknowns on the manifold of ortho-symplectic rectangular matrices. Observe that the reduced basis depends on the parameters, but it is the same for all parameters in the set Γ_h .

5.4 Efficient treatment of nonlinear terms

In this Section, we discuss the computational cost of the numerical solution of the reduced problem (5.3.8), and propose a novel algorithm for the efficient and structure-preserving treatment of the nonlinear operators.

The proposed reduction of the nonlinear terms is independent of the numerical time integrators used to solve the reduced dynamical system (5.3.8). However, the algorithm can be optimized depending on the time integrator of choice. We consider the structure-preserving partitioned Runge-Kutta temporal integrators described in Chapter 4. In particular, the evolution of the basis U is approximated with an explicit method, while a symplectic temporal integrator is employed for the evolution of the coefficients Z , and this latter will generally be an implicit scheme. Observe that we do not require that the stages of the RK integrators for the basis U and coefficients Z coincide. We will discuss the details and implementation of such schemes in Section 5.5. Although not strictly necessary, here we also assume that the first step of the partitioned RK method involves the evolution of the reduced basis; this assumption implies that we have the information on the Hamiltonian gradient at the beginning of the temporal interval (at least for some parameter values).

Let us split the temporal domain \mathcal{T} into sub-intervals $\mathcal{T}_\tau := (t_{\tau-1}, t_\tau]$, for any $\tau = 1, \dots, N_t$, where $t_0 = 0$ and $\Delta t = t_\tau - t_{\tau-1}$ is the uniform time step. For each temporal interval \mathcal{T}_τ , the dynamical reduced basis method involves the following operations.

- The evolution of the basis U requires $O(Nnp) + O(N_x^2 p) + O(N_x p c) + O(Nn^2) + O(n^2 p)$ flops, where $c \in \mathbb{N}$ is the number of finite element basis functions whose support is contained in a given mesh element, and we recall that N is the number of particles, $2n$ the size of the reduced basis and p denotes the number of parameter values. Note that this is a mild constant, and it is equal to 2 for piecewise polynomial functions in 1D, as in the discretization discussed in Section 5.2.1. The computational costs of this step are distributed as follows

Arithmetic complexity	Operation
$O(Nnp)$	computation of $V_r = U_V Z$ and $X_r = U_X Z$
$O(Npc)$	assembly of $\nabla \Lambda^0(X_r^i)$ and of $\Lambda^0(X_r^i)$, for all $i \in S_p$
$O(Npc) + O(N_x^2 p)$	computation of $\text{diag}(M_q) \nabla \Lambda^0(X_r^i(t)) L^{-1} \Lambda^0(X_r^i(t)) M_q$
$O(pn^2) + O(n^3)$	construction and inversion of the matrix $S(Z)$
$O(Nnp) + O(Nn^2)$	matrix-matrix multiplications in the r.h.s. of (5.3.8b)

The first three rows of the table correspond to the assembly and evaluation of $G_{\mathcal{H}}^p$ for all p parameters in Γ_h .

- The integration, using an implicit time scheme, of the evolution equation (5.4.2) for the p vector-valued coefficients requires $O(Nnp) + O(N_x^2 p) + O(Npc)$ flops:

Arithmetic complexity	Operation
$O(Nn)$	computation of $U_V^\top V_r^i = U_V^\top U_V Z_i$
$O(Nn) + O(N_x^2) + O(Nc)$	assembly and evaluation of $G_{\mathcal{H}}^p(U, Z_i)$
$O(Nn)$	computation of $U_X^\top G_{\mathcal{H}}^p(U, Z_i)$

Each of the operations listed in the table needs to be performed for each parameter $\eta_i \in \Gamma_h$, at each stage of the RK scheme, and at each iteration of the nonlinear solver.

The leading computational cost in both steps depends on the product of the number of particles N and of the number of parameter p , both potentially large in multi-query simulations of high-dimensional problems. This cost is associated with the remapping of the particles to the full dimensional space, in each temporal interval and for each parameter, and with the evaluation of the velocity field of the reduced flow. Indeed, the sole knowledge of the expansion coefficients with respect to the reduced basis is not enough to compute the particles-to-grid mapping needed to evaluate the electric field and, hence, the Hamiltonian. Even in the reduced model (5.3.8), these operations require the reconstruction of the approximate particle positions, at a cost proportional to the size of the full model. This lifting to the high-dimensional space needs to be performed for each instance of the parameter, at each stage of the RK time integrator, and at every iteration of the nonlinear solver. Analogous computational problems are common in model order reduction and emerge whenever non-affine and nonlinear operators are involved, as seen in Section 1.5. In the numerical experiments in Chapter 4, tensorial techniques [240] have been used to separate terms that depend on full spatial variables and on reduced coefficients to allow efficient computations of the nonlinear terms, because of the low-order polynomial nature of nonlinearities. The non-polynomial nature of the nonlinearity in the gradient of (5.3.4) prevents us from using the aforementioned tensorial approach to accelerate the computation. The discrete empirical interpolation method (DEIM), described in Chapter 1, is an interpolatory technique used to approximate the nonlinearity in the projection-based ROM, requiring the computations of only a few components of the original nonlinearity. While effective in the case where each component of the nonlinearity depends only on a few components of the input, it is not suited for the treatment of not component-wise nonlinear terms. Using a sparsity argument [57] or the introduction of auxiliary variables [74], DEIM has been adapted to deal with the approximation of the nonlinear terms at interpolation points that require the evaluation of the reduced solution on a limited number of neighboring mesh points, as it happens for high-order spatial discretization schemes with large stencils. However, the same strategy does not work for the treatment of the gradient of (5.3.4), as the inverse of the discrete Laplacian operator is generally dense, and hence each of its entries requires the computation of X_r^i for p sampled parameters, making traditional approaches computationally impractical. Moreover, traditional hyper-reduction techniques applied to a gradient vector field do not result in a gradient field, which means that the geometric structure of the Hamiltonian dynamics is compromised in the hyper-reduction process.

To achieve computational efficiency in the simulation of (5.3.8) without compromising its geometric structure, we propose a strategy that approximates the reduced Hamiltonian gradient via a combined hyper-reduction technique and sampling procedure. A reduction in the computational runtimes of the algorithm is achieved by decoupling the operations that depend on N from those that depend on p , while retaining an accurate representation of the plasma dynamics. There are several challenges that we need to face in the development of such techniques:

- The preservation of the Hamiltonian structure of the dynamics.
- The lack of information on the full model solution and nonlinear operators, traditionally collected in an offline phase via snapshots.
- The lack of a sparsity pattern in the nonlinear Hamiltonian gradient, i.e., the fact that each entry of the electric energy vector (5.3.5) depends on all N computational particles.

5.4.1 Parameter sampling

The reduced dynamics (5.3.8) involves p evolution equations for the expansion coefficients, one per parameter value, and one evolution problem for the matrix-valued reduced basis. Since, at each time, the reduced basis is the same for all parameters, one can reduce the computational cost required for its evolution by sampling over the parameter space and constructing a reduced basis for only a subsample of parameters, but which remains accurate for all other parameters in Γ_h . This corresponds to a reduction in parameter space. Let us denote by p the cardinality of the set Γ_h and assume that the parameters in Γ_h are indexed from the set $S_p := \{1, \dots, p\}$. Let us consider a subset Γ_h^* of Γ_h of size $p^* \ll p$. Define $S_{p^*} \subset S_p$ to be the set of indices corresponding to the parameters in the selected subset so that $\Gamma_h^* = \{\eta_i \in \Gamma_h \mid i \in S_{p^*}\} \subset \Gamma_h$. The idea of the proposed sampling approach is to replace, in the evolution of the basis (5.3.8), the matrix of the expansion coefficients $Z(t) \in \mathbb{R}^{2n \times p}$ by the matrix obtained via the concatenation of the columns of Z with indices in S_{p^*} . Following the discussion at the beginning of the section, this approximation leads to a computational complexity for the basis evolution of the order of $O(Nnp^*) + O(N_x^2 p) + O(Np^*c) + O(Nn^2) + O(n^2 p^*)$. To preserve the accuracy of the method, we must ensure that the chosen subset Γ_h^* is representative of the entire parameter set Γ_h . For the sake of simplicity, in this work, we set it at $t = 0$, and we keep it fixed over time. Starting from $p^* = \emptyset$, the set p^* is constructed using a greedy algorithm that, at each iteration, adds to the index subset S_{p^*} the index i that satisfies

$$\max_{i \in \Gamma_h \setminus \Gamma_h^*} \min_{j \in \Gamma_h^*} \|Z_i^0 - Z_j^0\|_2, \quad (5.4.1)$$

until a user-defined threshold value of the cost function or a maximum number of iterations. Possible research directions to improve the selection strategy would be to adapt in time the set p^* to capture significant changes in the behavior of the solution relative to the parameters or to modify the cost function to incorporate errors in the evaluation of physical quantities, such as the electric field. However, as (5.4.1) is an NP-complete problem [231], it is not currently known if it is possible to find an optimal parameter selection strategy with a polynomial cost in p . Thus, in the current form, while affordable if performed only once at $t = 0$, the selection strategy may become computationally expensive for large p and could compromise the efficiency of the proposed dynamical RB method if repeated in each temporal sub-interval. Suboptimal, yet more efficient, algorithms could be adopted by framing the problem into the more general column subset selection problem (CSSP) [35], that consists in finding an optimal subset of columns of a given matrix that minimizes the residual of the projection of the given matrix onto the selected column subset. Other parameter reduction strategies, like active subspaces [67], might also be envisioned.

Concerning the expansion coefficients, for which one differential equation per parameter needs to be solved, subsampling is not an option.

5.4.2 DMD-DEIM approximation of the Hamiltonian gradient

In this section, we develop a reduction algorithm where, in each temporal interval \mathcal{T}_τ , DMD is used for the hyper-reduction of the electric potential $\phi(X_r^i(t)) = L^{-1}\Lambda^0(X_r^i(t))^\top M_q$ in (5.3.5), while a DEIM strategy is developed to approximate the component $\Lambda^0(X_r^i(t))$ of the particles-to-grid mapping. Note that in (5.3.4), the quadratic term involving the particles' velocity represents a linear contribution in the gradient of the reduced Hamiltonian and, hence, does not require any hyper-reduction.

Dynamic Mode Decomposition of the electric potential

Dynamic mode decomposition is an equation-free data-driven approach, proposed in [227; 228], that uses only data measurements of a given dynamical system to approximate the dynamics and predict future states. The idea is to decompose the problem into a set of coherent spatial structures, known as DMD modes, and associate correlated data to specific Fourier modes that capture temporal variations. DMD was initially employed as a spectral decomposition method for complex fluid flows [218]. More recently, it has proved successful in a wide range of settings such as background/foreground separation in real-time video [114], characterization of dynamic stall [170], and analysis of the propagation of infectious diseases [204]. DMD hinges on the theory of Koopman operators [147], which allows representing the flow of a nonlinear dynamical system via an infinite-dimensional linear operator on the space of measurement functions. DMD computes a least-squares regression of data measurements to an optimal finite-dimensional linear dynamical system that approximates the infinite-dimensional Koopman operator without explicit knowledge of the operator describing the dynamics. This subsection first describes the classical DMD algorithm following [155]. Next, we introduce a sliding-window-based DMD formulation for the hyper-reduction of the electric potential in the dynamical reduced model (5.3.8) of the Vlasov–Poisson problem.

Consider a general nonlinear dynamical system: Find $\mathbf{y} : \mathcal{T} \rightarrow \mathbb{R}^l$, for $l \in \mathbb{N}$ such that

$$\begin{cases} \frac{d}{dt}\mathbf{y}(t) = F(t, \mathbf{y}(t)), & t \in \mathcal{T}, \\ \mathbf{y}(t_0) = \mathbf{y}_0. \end{cases} \quad (5.4.2)$$

Assume that we have as data measurements exact values or approximations of the state at different time instants, namely

$$\mathbf{Y} = [\mathbf{y}_0 \quad \mathbf{y}_1 \quad \dots \quad \mathbf{y}_{\tau-1}] \in \mathbb{R}^{l \times \tau}, \quad \mathbf{Y}' = [\mathbf{y}_1 \quad \mathbf{y}_2 \quad \dots \quad \mathbf{y}_\tau] \in \mathbb{R}^{l \times \tau}, \quad (5.4.3)$$

where $\mathbf{y}_k = \mathbf{y}(k\Delta t)$ and Δt is the uniform time step. In the DMD method, data measurements are used to approximate the nonlinear dynamics (5.4.2) by a locally linear system $\frac{d}{dt}\mathbf{y} = A\mathbf{y}$, where $A \in \mathbb{R}^{l \times l}$ is the matrix that best fits the measurements in a least-square sense, i.e., $A = \arg \min_{B \in \mathbb{R}^{l \times l}} \|\mathbf{Y}' - B\mathbf{Y}\|_F$. Then, A is given by $A = \mathbf{Y}'\mathbf{Y}^\dagger$, where \dagger denotes the Moore–Penrose pseudoinverse.

From the linear approximation of the dynamics, the DMD algorithm computes a low-rank eigendecomposition of the matrix A by extracting its r_τ largest eigenvalues Λ^A and corresponding eigenvectors $\Theta^A = [\theta_1^A \dots \theta_{r_\tau}^A] \in \mathbb{R}^{l \times r_\tau}$. The resulting DMD approximation of the state $\mathbf{y}(t)$, for

Algorithm 6 DMD algorithm

- 1: **procedure** DMD(\mathbf{Y}, tol)
 - 2: Compute the truncated SVD of \mathbf{Y} , $\mathbf{Y} = U\Sigma V^\top$, using tol as tolerance for singular values selection.
 - 3: Define $A_{tol} = U^\top \mathbf{Y}' V \Sigma^{-1}$.
 - 4: Compute the eigendecomposition of A_{tol} : $A_{tol} W = W \Lambda$.
 - 5: Reconstruct the eigendecomposition of A by defining its eigenvectors as $\Theta = \mathbf{Y}' V \Sigma^{-1} W$.
-

$t > \tau \Delta t$, reads

$$\mathbf{y}(t) \approx \mathbf{y}_{\text{DMD}}(t) = \Theta^A \left(\Pi \odot e^{\Omega(t-\tau\Delta t)} \right) = \sum_{j=1}^{r_\tau} \theta_j^A \pi_j e^{\omega_j(t-\tau\Delta t)}, \quad (5.4.4)$$

where, for any $j = 1, \dots, r_\tau$, $\omega_j := \ln(\Lambda_j^A)/\Delta t$ is the j -th entry of the vector $\Omega \in \mathbb{R}^{r_\tau}$, while π_j is the j -th entry of the vector $\Pi = (\Theta^A)^\dagger \mathbf{y}_0 \in \mathbb{R}^{r_\tau}$ containing the coordinates of the initial condition \mathbf{y}_0 with respect to the DMD modes.

If the size of the matrix A is large, A might be severely ill-conditioned and not directly tractable. In this situation, a different version of the DMD algorithm, proposed in [251], projects the data into a low-rank subspace instead of deriving A directly from the data, as described in Algorithm 6. Moreover, given the sensitivity of the DMD algorithm to the duration and sampling of the series \mathbf{Y} and \mathbf{Y}' , [80] proposes a sliding-window approach where the measurement data are not taken in the whole temporal interval but only in the sampling window $[t_{\tau-T}, t_\tau]$ of length $T \in \mathbb{N}$. The rationale is that if the system is time-varying and the incoming data is harvested in a streaming fashion, it may be beneficial to accuracy and memory storage to consider only the most recent data. The only computational overhead is the computation of the DMD modes and weights in the DMD approximation (5.4.4) as new data are collected. This cost may be mitigated by efficient online updates of the eigenvalues and eigenvectors of A [125] or by means of incremental SVD algorithms [173].

In the context of kinetic plasma PIC simulations, a DMD strategy has been used in [184] to detect and track equilibrium states. The aforementioned method relies on snapshots of the high-fidelity simulation until an equilibrium is detected and, after this time, the solution is extrapolated via the DMD modes. Here, we propose to employ a DMD strategy in a different way, namely to hyper-reduce the self-consistent electric potential $\phi(X_r^i(t)) = L^{-1} \Lambda^0 (X_r^i(t))^\top M_q \in \mathbb{R}^{N_x}$ that enters the reduced Hamiltonian (5.3.4) for each parameter $\eta_i \in \Gamma_h$. The idea is to extract low-dimensional dynamical features from a time-series of the electric potential and use them, as part of the DMD algorithm in (5.4.4), to extrapolate the value of $\phi(X_r^i(t))$ needed for the computation of the reduced Hamiltonian (5.3.4) in each temporal interval \mathcal{T}_τ . In details, let ϕ_τ^i , for a fixed parameter with index $i \in S_{p^*}$ and $\tau = 1, \dots, N_\tau$, be the approximation of $\phi(X_r^i(t))$ at $t = t_\tau$ for each parameter in the subset Γ_h^* . Since the first step of the temporal integrator involves the evolution of the reduced basis, these quantities are computed while assembling the right hand side of the basis evolution equation (5.3.8). For each \mathcal{T}_τ , we collect the time-discrete approximations of the electric potential obtained in a time window of length $T + 1$;

$$\mathbf{Y}_i = [\phi_{\tau-T-1}^i \quad \phi_{\tau-T}^i \quad \cdots \quad \phi_{\tau-2}^i], \quad \mathbf{Y}'_i = [\phi_{\tau-T}^i \quad \phi_{\tau-T+1}^i \quad \cdots \quad \phi_{\tau-1}^i], \quad (5.4.5)$$

where $\mathbf{Y}_i, \mathbf{Y}'_i \in \mathbb{R}^{N_x \times T}$ for $i = 1, \dots, p^*$. Extracting the dominant modes from each realization of the electric potential associated with a fixed parameter is a cumbersome task. To the best of our

knowledge, DMD-based methods for the model order reduction of parametric problems have not been developed. In our setting, the dependence on the parameter comes from the state, and it is propagated via the parametric initial distribution f_0 . This suggests that, instead of extracting the DMD modes for each fixed parameter $\eta_i \in p^*$, we can incorporate the parameter in the DMD procedure to approximate the dynamics of the electric potential for all parameters. A similar approach can be found in [226] when dealing with bifurcation parameters in thermo-acoustic systems. For each parameter index $i \in S_p$, the datasets \mathbf{Y}_i and \mathbf{Y}'_i are concatenated column-wise to form two global datasets \mathbf{Y} and \mathbf{Y}' , i.e.

$$\mathbf{Y} = [\mathbf{Y}_1 \quad \mathbf{Y}_2 \quad \cdots \quad \mathbf{Y}_{p^*}], \quad \mathbf{Y}' = [\mathbf{Y}'_1 \quad \mathbf{Y}'_2 \quad \cdots \quad \mathbf{Y}'_{p^*}], \quad (5.4.6)$$

with $\mathbf{Y}, \mathbf{Y}' \in \mathbb{R}^{N_x \times p^* T}$. This procedure is justified by the absence of an explicit dependence of the electric potential on the parameter. Following Algorithm 6, we generate the DMD eigenvectors $\Theta \in \mathbb{R}^{N_x \times r_\tau}$ and eigenvalues $\Lambda \in \mathbb{R}^{r_\tau \times r_\tau}$ of the linear approximation of the dynamics for the problem of interest. The resulting DMD approximation of the self-consistent electric potential reads

$$\phi(X_r^i(t)) \approx \phi_{\text{DMD}}^i(t) = \Theta \left(\Pi_i \odot e^{W(t-(\tau-1)\Delta t)} \right), \quad \forall i \in S_{p^*}, \forall t \in \mathcal{T}_\tau, \quad (5.4.7)$$

with $W \in \mathbb{R}^{r_\tau}$ is the vector of entries $\omega_j = \ln(\Lambda_j)/\Delta t$, for any $j = 1, \dots, r_\tau$, and $\Pi_i := \Theta^\dagger \Phi_{\tau-1}^i \in \mathbb{R}^{r_\tau}$ for any $i \in S_{p^*}$.

Assuming a smooth dependence of the DMD coordinates Π_i on the parameter, interpolation techniques can be used to recover the DMD coordinates for parameters not included in Γ_h^* , similarly to the POD with interpolation (PODI) [43]. In this work, we adopt the radial basis interpolation [37], with a Gaussian kernel, as interpolation algorithm to reconstruct the DMD coordinates Π_i for $i \in \Gamma_h \setminus \Gamma_h^*$. The computational cost of the interpolation step is negligible as compared to the cost of Algorithm 6, as we comment on at the end of the section. For ease of the notation, we use the same symbol Π_i to represent the interpolated DMD coefficients for all $i \in S_p$. Knowing Π_i for all $i \in S_p$, the electric potential is reconstructed using (5.4.7). The resulting sampling error can be controlled by enriching the subset of parameters Γ_h^* and by optimal placement of the location of the parameters with indices in S_{p^*} in the parameter space. Using the DMD estimate of the potential ϕ , the Hamiltonian function (5.3.4) is approximated as

$$\mathcal{H}_{U,i}^{\text{DMD}}(Z_i) = \frac{1}{2} V_r^i(t)^\top V_r^i(t) + \frac{m_p^{-1}}{2} M_q^\top \Lambda^0(X_r^i(t)) \phi_{\text{DMD}}^i(t), \quad \forall i \in S_p, \forall t \in \mathcal{T}_\tau. \quad (5.4.8)$$

Remark 5.4.1. If the Hamiltonian function depends explicitly on the parameter, the approach outlined above is not legitimized because a non-parametric operator would be used to approximate the parametric potential. An alternative strategy would require an approximation of the form (5.4.7) for each parameter realization η_i , with $i \in S_{p^*}$, using different W_i and Θ_i for each i . The resulting DMD approximations of the potential $\phi_{\text{DMD}}^i(t)$ could then be directly interpolated on S_p or, as suggested in [79], interpolated based on physical concepts as in PODI.

The DMD approximation based on the matrix A , instead of its projection A_{tol} defined in Algorithm 6, would result in a computational complexity $O(N_x^3)$. Although this cost might still be tractable in one dimension, it becomes prohibitive when considering the Vlasov–Poisson problem in a higher dimension. The method described in Algorithm 6 is, therefore, the preferred choice.

The computational cost of the proposed DMD strategy, applied to the electric potential, reduces to the cost needed to perform Algorithm 6 from the datasets $\mathbf{Y}, \mathbf{Y}' \in \mathbb{R}^{N_x \times p^* T}$ in (5.4.6). The

truncated SVD decomposition of \mathbf{Y} has arithmetic complexity $O(N_x p^* T r_\tau)$, where r_τ is the number of retained modes [122]. Observe that, if the number r_τ of truncated modes is chosen based on a tolerance to control the magnitude of the neglected singular values, Algorithm 6 computes the full SVD of \mathbf{Y} and then performs the truncation. This variant of the truncated SVD has computational complexity $O(N_x (p^* T)^2)$, under the assumption that the chosen DMD window length T and number of parameter subsamples p^* satisfy $N_x > p^* T$. The eigendecomposition of $A_{\text{tol}} \in \mathbb{R}^{r_\tau \times r_\tau}$ in Algorithm 6 costs $O(r_\tau^3)$. Finally, the matrix-matrix multiplications to compute A_{tol} and Θ , respectively, require $O(N_x r_\tau^2) + O(N_x p^* T r_\tau)$ operations. The computational cost to compute ϕ_{DMD}^i for every parameter $\eta_i \in \Gamma_h$ – including sampling parameters and reconstructed parameters – is $O(N_x r_\tau p)$. The leading cost is, therefore, $O(N_x r_\tau p) + O(N_x p^* T r_\tau)$, with the last term replaced by $O(N_x (p^*)^2 T^2)$ for a naive implementation of the truncated SVD. This cost is linear in N_x , does not depend on the number N of particles, and only the computation of the DMD coordinates Π_i depends on the number of parameters p .

Discrete Empirical Interpolation Method for reduction in the number of particles

The DMD approach described in the previous Section allows to derive an approximate electric potential that can be evaluated independently on the number of particles. However, the evaluation of the electric energy component of the approximate reduced Hamiltonian (5.4.8) still requires the particles-to-grid mapping for $\Lambda^0(X_r^i(t))$, for each value of the parameter, at each stage of the temporal solver, and at each iteration of the nonlinear solver. The computational cost of this step is a major bottleneck of the algorithm. We propose hyper-reduction of the approximate reduced Hamiltonian (5.4.8) with a DEIM-based strategy to overcome this computational burden.

The DEIM approach is a discrete variant of the empirical interpolation method (EIM) introduced in [21] to approximate nonlinear functions via a combination of projection and interpolation. DEIM constructs carefully selected interpolation indices to specify an interpolation-based projection so that the complexity of evaluating the nonlinear term becomes proportional to the (small) number of selected spatial indices, as seen in Section 1.5.

The application of the classical DEIM procedure for the hyper-reduction of the nonlinear Hamiltonian gradient (5.3.7) is challenged by several factors. As stated at the beginning of the Section, applying the DEIM interpolation directly to the right-hand side of the coefficients evolution equations arising from the dynamic reduced basis approach, would not result in a structure-preserving approximation. Moreover, the classical DEIM algorithm hinges on the availability of snapshots of the full model nonlinear operator of interest collected in the offline phase. In our dynamical model order reduction approach, there is no offline phase and, therefore, snapshots are not available.

We consider the Hamiltonian splitting in (5.3.4), and the approximation of the reduced electric energy (5.3.5) resulting from DMD, namely

$$\mathcal{E}_{U,i}^{\text{DMD}}(Z_i(t), t) := \frac{m_p^{-1}}{2} M_q^\top \Lambda^0(X_r^i(t)) \Phi_{\text{DMD}}^i(t), \quad \forall i \in S_p, \forall t \in \mathcal{T}_\tau,$$

where Φ_{DMD}^i is defined in (5.4.7). Approximating directly the vector $\nabla \mathcal{E}_{U,i}^{\text{DMD}}$ by a DEIM interpolation would not preserve the geometric structure of the problem because it is not possible to define explicitly Hamiltonian gradient from the interpolated vector field. We propose a DEIM approximation of the reduced electric energy via hyper-reduction of the term $\Lambda^0(X_r^i(t)) \in \mathbb{R}^{N \times N_x}$, which otherwise would require the evaluation of the finite element basis functions at each particle position.

Let us introduce the function $\mathcal{N}_i(X_r^i(t), t) := \Lambda^0(X_r^i(t))\phi_{\text{DMD}}^i(t) \in \mathbb{R}^N$; approximated using a DEIM approach in each temporal interval \mathcal{T}_τ as follows. First, we consider snapshots of the nonlinear term associated with the electric potential ϕ (5.2.5) at p^* instances of the parameter and over a temporal window of length $T + 1$. The snapshot matrix $\mathbf{Y} \in \mathbb{R}^{N \times p^*(T+1)}$ is defined as

$$\mathbf{Y} = [\mathbf{Y}_1 \quad \mathbf{Y}_2 \quad \cdots \quad \mathbf{Y}_{p^*}], \quad \mathbf{Y}_i := [\Lambda^0(X_r^i(t_{\tau-T-1}))\Phi_{\tau-T-1}^i \quad \cdots \quad \Lambda^0(X_r^i(t_{\tau-1}))\phi_{\tau-1}^i]. \quad (5.4.9)$$

Note that the terms $\Lambda^0(X_r^i(t_{\tau-j-1}))$ and $\phi_{\tau-j-1}^i$, for $i \in \Gamma_h^*$ and $j = 0, \dots, T$, are available from the evolution equation (5.3.8) for the reduced basis solved at previous time steps. The DEIM basis matrix $\Psi_\tau \in \mathbb{R}^{N \times n_d}$ is obtained by taking the first n_d left singular vectors of the snapshot matrix \mathbf{Y} , where the value n_d is fixed at the beginning of the simulation and might differ for different problems. We will comment on this in the numerical experiments in Section 5.5. Denoting with $P_\tau \in \mathbb{R}^{N \times n_d}$ the matrix corresponding to the DEIM indices obtained as described above, the nonlinear term $\mathcal{N}_i(X_r^i(t), t)$ is approximated by

$$\Psi_\tau^\top (P_\tau^\top \Psi_\tau)^{-1} P_\tau^\top \mathcal{N}_i(X_r^i(t), t), \quad \forall i \in S_p, \forall t \in \mathcal{T}_\tau.$$

Observe that, although the basis Ψ_τ is constructed from the parameter subsample Γ_h^* , the nonlinear term \mathcal{N}_i is approximated by its DEIM projection onto the DEIM space for all instances of the parameter, i.e., for all $i \in S_p$.

To reduce the computational burden associated with the computation of the DEIM sampling points P_τ in each temporal interval \mathcal{T}_τ , we follow an update strategy similar to the one proposed in [196]. All the interpolation indices in the set I_{DEIM} are computed using the standard DEIM greedy method; not at all time steps but only every $k_{\text{DEIM}} > 1$ time steps. In other temporal intervals, we proceed as follows. Assume we have computed the set of DEIM indices $I_{\text{DEIM}}^{\tau-1}$ in the temporal interval $\mathcal{T}_{\tau-1}$, then, in the following interval \mathcal{T}_τ , we update only the indices in the subset $I^* \subset I_{\text{DEIM}}^{\tau-1}$ of cardinality n_{DEIM} given by

$$I^* = \underset{\substack{I \subset I_{\text{DEIM}}^{\tau-1}, \\ \dim(I) = n_{\text{DEIM}}}}{\text{argmax}} \sum_{k \in I} (\psi_k^\tau)^\top \psi_k^{\tau-1}, \quad (5.4.10)$$

where ψ_k^τ denotes the k -th vector of the DEIM basis Ψ_τ at time t_τ . The remaining $n_d - n_{\text{DEIM}}$ indices in $I_{\text{DEIM}}^{\tau-1} \setminus I^*$ are inherited by I_{DEIM}^τ . The rationale for the choice of I^* is to only update the indices associated with the DEIM basis vectors at $t_{\tau-1}$ that have undergone the largest rotations in the DEIM basis update from $\Psi_{\tau-1}$ to Ψ_τ .

The resulting approximate reduced Hamiltonian, associated with the parameter $\eta_i \in \Gamma_h$, reads

$$\mathcal{H}_{U,i}^{\text{DD}}(Z_i(t), t) = \frac{1}{2} V_r^i(t)^\top V_r^i(t) + \frac{m_p^{-1}}{2} M_q^\top \Psi_\tau (P_\tau^\top \Psi_\tau)^{-1} P_\tau^\top \Lambda^0(X_r^i(t)) \phi_{\text{DMD}}^i(t), \quad \forall i \in S_p, \forall t \in \mathcal{T}_\tau. \quad (5.4.11)$$

Observe that the multiplication of the matrix $\Lambda^0(X_r^i)$, of the finite element basis functions evaluated at the particles' position, by the DEIM sampling matrix P_τ^\top , corresponds to evaluating the finite element basis functions only on a subset of $n_d \ll N$ particles. Hence, this operation represents a substantial reduction in the number of particles.

The computational cost of the DEIM algorithm can be summarized as follows. The computation of the snapshot matrix in (5.4.9) only involves the multiplications of the terms $\Lambda^0(X_r^i(t_{\tau-j-1}))$ and $\phi_{\tau-j-1}^i$, for $i \in \Gamma_h^*$ and $j = 0, \dots, T$. Indeed, since these terms are available from the solution of the reduced basis evolution at previous time steps, there is no cost associated with their

assembly, at least at this stage of the proposed model order reduction algorithm. The matrix-matrix multiplications require $O(Np^*Tc)$, where p^* is the dimension of the subset of sampling parameters, T is the length of the sampling window for the snapshots, and c is a mild constant that depends only on the support of the finite element basis functions. The truncated SVD decomposition of the snapshot matrix $\mathbf{Y} \in \mathbb{R}^{N \times p^*(T+1)}$ has arithmetic complexity $O(NpTn_d)$, where n_d is the number of DEIM modes. The computational cost required to assemble the interpolation matrix $P_\tau \in \mathbb{R}^{N \times n_d}$ using [57] only depends on n_d . This cost is further reduced by updating the indices according to the strategy described above and inspired by the adaptive sampling of [196]. Hence, the leading computational cost of the DEIM algorithm is $O(Np^*Td)$.

5.4.3 DMD-DEIM reduced dynamics and computational complexity

From (5.4.11), the Hamiltonian gradient $G_{\mathcal{H}}^p(U, Z_i)$ in (5.3.6) is approximated as

$$G_{\mathcal{H}}^{\text{DD}}(U, Z_i, t) := \nabla_{U, Z_i} \mathcal{H}_{U, i}^{\text{DD}}(Z_i(t), t) = \begin{pmatrix} m_p^{-1} \text{diag}(\nabla \Lambda^0(X_r^i(t)) \phi_{\text{DMD}}^i(t)) \\ V_r^i(t) \end{pmatrix} P_\tau (P_\tau^\top \Psi_\tau)^{-\top} \Psi_\tau^\top M_q, \quad (5.4.12)$$

for all $i \in S_p$, and $t \in \mathcal{T}_\tau$. Similarly, the approximation of the gradient of the reduced Hamiltonian $g_{\mathcal{H}}^p(U, Z_i)$ reads

$$\begin{aligned} g_{\mathcal{H}}^{\text{DD}}(U, Z_i, t) &= U(t)^\top G_{\mathcal{H}}^{\text{DD}}(U, Z_i, t) = \nabla_{Z_i} \mathcal{H}_{U, i}^{\text{DD}}(Z_i(t), t) \\ &= U_V(t)^\top V_r^i(t) + m_p^{-1} U_X(t)^\top \text{diag}(\nabla \Lambda^0(X_r^i(t)) \phi_{\text{DMD}}^i(t)) P_\tau (P_\tau^\top \Psi_\tau)^{-\top} \Psi_\tau^\top M_q, \end{aligned} \quad (5.4.13)$$

for all $i \in S_p$, and $t \in \mathcal{T}_\tau$.

The reduced dynamical system (5.3.8) is approximated by replacing the gradient of the reduced Hamiltonian $g_{\mathcal{H}}^p(U, Z_i) \in \mathbb{R}^{2N}$ with its DMD-DEIM approximation $g_{\mathcal{H}}^{\text{DD}}(U, Z_i, t)$ from (5.4.13) in the evolution equations of the expansion coefficients. The DMD-DEIM reduced dynamics reads

$$\begin{cases} \dot{Z}(t) = J_{2n} g_{\mathcal{H}}^{\text{DD}}, & \text{in } \mathcal{T}, \end{cases} \quad (5.4.14a)$$

$$\begin{cases} \dot{U}(t) = (\mathbb{I}_{2N} - UU^\top)(J_{2N} G_{\mathcal{H}}^p(U, Z) Z^\top - G_{\mathcal{H}}^p(U, Z) Z^\top J_{2n}^\top) S(Z)^{-1}, & \text{in } \mathcal{T}, \end{cases} \quad (5.4.14b)$$

$$U(t_0) Z(t_0) = U^0 Z^0, \quad (5.4.14c)$$

Note that this approximate reduced model retains the geometric structure of the full model.

We analyze the computational cost to assemble and evaluate the right hand side of the DMD-DEIM reduced model (5.4.14) at each time instance. We then compare the results with the ones at the beginning of Section 5.4.1 corresponding to the reduced model. The evolution of the reduced basis requires $O(Nnp^*) + O(N_x^2 p^*) + O(Np^*c) + O(Nn^2) + O(n^2 p^*)$ flops owing to the parameter sampling discussed in Section 5.4.1. This cost also includes the assembly of the quantities $\Lambda^0(X_r^i)$ and ϕ^i needed in the DMD and DEIM algorithms. The computational cost required to assemble and evaluate the velocity field of the flow characterizing the evolution of the coefficients reduces to the cost of the evaluation of the gradient (5.4.13) of the DMD-DEIM Hamiltonian at each time instant t . This includes:

- (1) The cost to compute the linear part $U_V^\top V_r$ of (5.4.13), for all instances of the parameter in Γ_h , is $O(Nn^2) + O(pn^2)$. Note that the cost $O(Nn^2)$ required to assemble the matrix

$U_V^\top U_V$ is performed once per stage of the RK time integrator, while the matrix-matrix product $(U_V^\top U_V)Z$, at cost $O(pn^2)$, has to be performed, at every RK stage, and for each iteration of the nonlinear solver.

- (2) The cost to compute the DMD approximation of the electric potential $\phi_{\text{DMD}}^i \in \mathbb{R}^{N_x}$, for any $i \in S_p$, is $O(N_x p^* T r_\tau) + O(N_x r_\tau p)$, as shown in Section 5.4.2. This cost is linear in N_x , and does not depend on the number N of particles. The evaluation of ϕ_{DMD}^i , that requires $O(N_x r_\tau p)$ flops, needs to be performed at each stage of the RK scheme and at each iteration of the nonlinear solver. The other cost $O(N_x p^* T r_\tau)$ is accounted for once per time step.
- (3) The cost to run the DEIM algorithm is $O(Np^* T d) + O(Np^* T c)$, as described in Section 5.4.2. The matrix-matrix multiplication $(P_\tau^\top \Psi_\tau)^{-\top} \Psi_\tau^\top M_q$ costs $O(Nd) + O(d^3)$. These operations are performed once per time step.
- (4) The computation of the nonlinear time-dependent part of (5.4.13) for all parameters $\eta_i \in \Gamma_h$, namely $U_X^\top \text{diag}(\nabla \Lambda^0(X_r^i(t)) \phi_{\text{DMD}}^i(t)) P_\tau$, requires $O(pdn) + O(pdc)$ operations. This includes the cost of the matrix-matrix multiplications and the cost $O(pdc)$ to assemble $\nabla \Lambda^0(X_r^i)$ for d particles.

To summarize, the DMD-DEIM approximation allows a complete separation of the costs involving the number N of particles and the number p of parameters, both potentially large. Once for each time interval \mathcal{T}_τ , the algorithm requires $O(Np^* T d) + O(Nn^2)$ operations. These are shared by all parameters, resulting in a computational cost independent of the size p of the parameter set Γ_h , but only dependent on the number of parameter subsamples p^* . The arithmetic complexity of the parameter-dependent computations is $O(pn^2) + O(N_x r_\tau p)$. These need to be performed at each RK stage and nonlinear iteration, but their computational cost is independent of the number of particles N .

5.5 Numerical experiments

For the numerical time integration of the DMD-DEIM reduced dynamics (5.4.14), we adopt the partitioned RK method of order 2 proposed in [128] and described in Section 4.3. The idea is to combine a symplectic temporal integrator for the evolution (5.4.14a) of the expansion coefficient, with a time discretization of the basis evolution (5.4.14b) able to preserve the ortho-symplectic constraint. For the latter, we adopt the tangent method proposed in [189], summarized next.

In each temporal sub-interval $\mathcal{T}_\tau = (t_{\tau-1}, t_\tau]$, given the approximate reduced basis $U_{\tau-1}$, the method constructs a local retraction $\mathcal{R}_{\tau-1}$ from the tangent space $T_{U_{\tau-1}}\mathcal{U}$ into \mathcal{U} so that $U(t) = \mathcal{R}_{\tau-1}(\xi(t))$ for some ξ in the tangent space at $U_{\tau-1}$. For the computation of U_τ , the idea is to evolve $\xi(t)$ in the tangent space and then recover U via the retraction. The reduced problem (5.4.14) in each temporal sub-interval \mathcal{T}_τ is re-written in terms of the variable $Z(t)$ and $\xi(t)$ as: Given $Z_{\tau-1}$ and $\xi_{\tau-1} = 0$, find $Z(t)$ and $\xi(t)$ such that

$$\begin{cases} \dot{Z}(t) = J_{2n} g_{\mathcal{H}}^{\text{DD}}(\mathcal{R}_{\tau-1}(\xi(t)), Z(t), t), & \text{in } \mathcal{T}_\tau, \\ \dot{\xi}(t) = \mathcal{Y}(\xi(t), Z(t)), & \text{in } \mathcal{T}_\tau. \end{cases} \quad (5.5.1a) \quad (5.5.1b)$$

The velocity field $\mathcal{Y} : T_{U_{\tau-1}}\mathcal{U} \times \mathcal{Z} \rightarrow \mathbb{R}^{2N \times 2n}$, describing the local flow on the tangent space at

$U_{\tau-1}$, is

$$\mathcal{Y}(\xi, Z) := -U_{\tau-1}(\mathcal{R}_{\tau-1}^\top(\xi)U_{\tau-1} + \mathbb{I}_{2n})^{-1}(\mathcal{R}_{\tau-1}(\xi) + U_{\tau-1})^\top \Upsilon(\xi, Z) + \Upsilon(\xi, Z) - U_{\tau-1} \Upsilon^\top(\xi, Z) U_{\tau-1},$$

where $\Upsilon(\xi, Z)$ is given by

$$\Upsilon(\xi, Z) := \left(2\mathcal{X}(\mathcal{R}_{\tau-1}(\xi), Z) - (WU_{\tau-1}^\top - U_{\tau-1}W^\top)\mathcal{X}(\mathcal{R}_{\tau-1}(\xi), Z) \right) (U_{\tau-1}^\top \mathcal{R}_{\tau-1}(\xi) + \mathbb{I}_{2n})^{-1},$$

with $2W := (2\mathbb{I}_{2N} - U_{\tau-1}U_{\tau-1}^\top)\xi$ and $\mathcal{X} : \mathbb{R}^{2N \times 2n} \times \mathcal{Z} \rightarrow \mathbb{R}^{2N \times 2n}$ being the velocity field of the approximate basis evolution in (5.4.14b), i.e.

$$\mathcal{X}(U, Z) := (\mathbb{I}_{2N} - UU^\top) \left(J_{2N} G_{\mathcal{H}}^{p*} Z^\top - G_{\mathcal{H}}^{p*} Z^\top J_{2n}^\top \right) (ZZ^\top + J_{2n}^\top Z Z^\top J_{2n})^{-1}.$$

The retraction is defined according to [189] as $\mathcal{R}_{\tau-1}(\xi) = \text{cay}(WU_{\tau-1}^\top - U_{\tau-1}W^\top)U_{\tau-1}$ where cay denotes the Cayley transform. We refer the reader to [189; 128] for further details regarding the formal derivation of (5.5.1). Note that, with the algorithm proposed in [189], the computation of the retraction \mathcal{R} and the assembly of the operator \mathcal{Y} have arithmetic complexity $O(Nn^2)$.

The partitioned Runge-Kutta scheme applied to (5.5.1) reads

$$Z_\tau = Z_{\tau-1} + \Delta t \sum_{l=1}^{n_s} b_l k_l, \quad (5.5.2a)$$

$$\xi_\tau = \Delta t \sum_{l=1}^{n_s} \widehat{b}_l \widehat{k}_l, \quad U_\tau = \mathcal{R}_{U_{\tau-1}}(\xi_\tau), \quad (5.5.2b)$$

$$k_1 = J_{2n} g_{\mathcal{H}}^{\text{DD}}(U_{\tau-1}, Z_{\tau-1} + \Delta t \sum_{j=1}^{n_s} a_{1,j} k_j, t_{\tau-1}), \quad \widehat{k}_1 = \mathcal{X}\left(U_{\tau-1}, Z_{\tau-1} + \Delta t \sum_{j=1}^{n_s} a_{1,j} k_j\right), \quad (5.5.2c)$$

$$k_l = J_{2n} g_{\mathcal{H}}^{\text{DD}}(\mathcal{R}_{U_{\tau-1}}(\Delta t \sum_{j=1}^{l-1} \widehat{a}_{l,j} \widehat{k}_j), Z_{\tau-1} + \Delta t \sum_{j=1}^{n_s} a_{l,j} k_j, t_{\tau-1} + c_l \Delta t) \quad l = 2, \dots, n_s, \quad (5.5.2d)$$

$$\widehat{k}_l = \mathcal{Y}\left(\Delta t \sum_{j=1}^{l-1} \widehat{a}_{l,j} \widehat{k}_j, Z_{\tau-1} + \Delta t \sum_{j=1}^{n_s} a_{l,j} k_j\right), \quad l = 2, \dots, n_s, \quad (5.5.2e)$$

where $\{a_{l,j}, b_j, c_j\}$ and $\{\widehat{a}_{l,j}, \widehat{b}_j\}$ are the set of coefficients corresponding to the implicit midpoint rule and the explicit midpoint method, respectively, cf. [128]. Note that the Hamiltonian (5.3.1) is separable, i.e. the gradient of the electric energy determines the evolution of the state variable and the gradient of the kinetic energy describes the dynamics of the momentum. The separability of the full-order Hamiltonian is, however, not inherited by the reduced model, as seen in (5.3.4). This precludes the explicit integration of (5.8). Even though explicit numerical integrators for non-separable Hamiltonian, based on Hamiltonian extensions, have been recently proposed [249], further investigations are required to assess their accuracy in the framework of partitioned Runge-Kutta schemes.

For comparison purposes, in the numerical experiments, we will solve the full-order model (5.2.7). The Störmer-Verlet scheme [119] is the most popular symplectic integrator for separable Hamiltonian systems and yields the following system of equations

$$X_\tau^i = X_{\tau-1}^i + \Delta \tau \left(V_{\tau-1}^i + \frac{\Delta t}{2} E_h(X_{\tau-1}^i; \eta_i) \right), \quad (5.5.3a)$$

$$V_\tau^i = V_{\tau-1}^i + \frac{\Delta t}{2} (E_h(X_{\tau-1}^i; \eta_i) + E_h(X_\tau^i; \eta_i)), \quad (5.5.3b)$$

to be solved for each of the p parameters $\eta_i \in \Gamma_h$. Here E_h denotes the approximate electric field (up to constants), i.e. $E_h(X_\tau^i; \eta_i) = -m_p^{-1} \text{diag}(M_q) \nabla \Lambda^0(X_\tau^i) L^{-1} \Lambda^0(X_\tau^i) M_q$.

5.5.1 Implementation and numerical study

In this Section, we apply the proposed structure-preserving dynamical model order reduction approach to several periodic electrostatic benchmark problems. In all the examples, computational macro-particles are loaded from a perturbed initial distribution given by

$$f(0, x, v; \eta) = f_v(v; \eta) f_x(x; \eta), \quad (5.5.4)$$

where $f_v(v; \eta)$ is the initial velocity distribution, and $f_x(x; \eta) := 1 + \alpha \cos(kx)$ is the initial perturbation, with k as the wavenumber and α as the amplitude of the perturbation. We have chosen physical units such that the particle mass and particle charge are normalized to one for electrons, i.e., $q = -1$ and $m = 1$, and the weight w of the computational macro-particles is set to N^{-1} . To reduce the statistical noise [133; 153] introduced by the particle discretization (5.2.1) of the initial condition (5.5.4), particles are loaded following a quiet start procedure based on a quasirandom sequence of samples. In detail, particles' positions and velocities are initialized by evaluating the inverse cumulative distribution function of $f(0, x, v; \eta)$ at the points defined by the Hammersley sequence [245] of length N . The distribution is defined over $\Omega := \Omega_x \times \Omega_v$, with $\Omega_v = [-10, 10]$ for all numerical experiments and Ω_x specified for each example. The quasirandom Hammersley sequence is characterized by a discrepancy value proportional to N^{-1} , whereas for a random distribution, the discrepancy is proportional to $N^{-1/2}$. Since the discrepancy measures the highest and lowest densities of points in a sequence, the Hammersley sequence guarantees that the particles are almost evenly distributed, and a significant noise reduction in the electrostatic field is therefore achieved.

The DMD-DEIM reduced-order model (5.4.14) is numerically integrated in time according to the scheme described in Section 5.5, resulting in the system of equations in (5.5.2). The full-order model is solved using the Störmer-Verlet scheme (5.5.3). The same integration step Δt and number of time steps N_τ are considered for the numerical integration of the two models. In the following, we adopt the notation $W_\tau^i := [X_\tau^i; V_\tau^i] \in \mathbb{R}^{2N}$ and $R_\tau^i := [X_{r,\tau}^i; V_{r,\tau}^i] = U_\tau Z_{i,\tau} \in \mathbb{R}^{2N}$ to denote the numerical solutions of the discrete full-order model (5.5.3) and the discrete reduced-order model (5.5.2) for the parameter η_i at time t_τ , respectively.

The reducibility of the considered benchmark tests is studied in terms of the decay of the singular values of the snapshots matrices

$$S_X = [X_0^1 \ \cdots \ X_0^p \ \cdots \ X_{N_\tau}^1 \ \cdots \ X_{N_\tau}^p] \quad \text{and} \quad S_V = [V_0^1 \ \cdots \ V_0^p \ \cdots \ V_{N_\tau}^1 \ \cdots \ V_{N_\tau}^p], \quad (5.5.5)$$

collecting the position and velocity components of W_τ^i . As in traditional reduced basis methods, the space spanned by the selected snapshots is assumed to be representative of the solution set. The behavior of the singular values of S_X and S_V is also compared to the decay of the singular values of the *local* snapshots matrices

$$S_X^\tau = [X_\tau^1 \ \cdots \ X_\tau^p] \quad \text{and} \quad S_V^\tau = [V_\tau^1 \ \cdots \ V_\tau^p], \quad \forall \tau = 1, \dots, N_\tau, \quad (5.5.6)$$

to assess the applicability and the benefits of the dynamical approach over standard global reduction methods. For the local snapshots matrices, we compute the ordered singular values $\{\sigma_{X,j}^\tau\}_j$ of S_X^τ , for each $\tau = 1, \dots, N_\tau$, and normalize them with respect to the maximum singular

value $\sigma_{X,1}^\tau$. Then, for each j , we consider the average over time, i.e., $\sum_\tau \sigma_{X,j}^\tau$, and the maximum over time, i.e., $\max_\tau \sigma_{X,j}^\tau$. The same study is carried out for the matrix S_V^τ . A further indicator of the reducibility properties of the problem is given by the numerical rank of S_X^τ and S_V^τ , defined as the number of singular values larger than a user-defined threshold tolerance. In the following, different tolerances are considered.

The accuracy of the DMD-DEIM-ROM is evaluated by computing, for each $\tau = 1, \dots, N_\tau$, the relative errors

$$\varepsilon_{\text{rel},X}(t_\tau) = \frac{\|S_X^\tau - X_{r,\tau}\|_F}{\|S_X^\tau\|_F}, \quad \text{and} \quad \varepsilon_{\text{rel},V}(t_\tau) = \frac{\|S_V^\tau - V_{r,\tau}\|_F}{\|S_V^\tau\|_F}, \quad (5.5.7)$$

where $X_{r,\tau}, V_{r,\tau} \in \mathbb{R}^{N \times p}$ are the position and velocity components of the discrete reduced-order solution $R_\tau = U_\tau Z_\tau \in \mathbb{R}^{2N \times p}$, respectively. We study the error in the position and velocity of the particles separately because they are characterized by different scales of absolute error. The relative errors (5.5.7) are compared to the target values given by the projection errors

$$\varepsilon_{\text{rel},X}^{\text{Target}}(t_\tau) = \frac{\|S_X^\tau - S_{X,\text{cSVD}}^\tau\|_F}{\|S_X^\tau\|_F}, \quad \text{and} \quad \varepsilon_{\text{rel},V}^{\text{Target}}(t_\tau) = \frac{\|S_V^\tau - S_{V,\text{cSVD}}^\tau\|_F}{\|S_V^\tau\|_F}, \quad (5.5.8)$$

where $S_{X,\text{cSVD}}^\tau, S_{V,\text{cSVD}}^\tau \in \mathbb{R}^{N \times p}$ are the position and velocity components of the projection of the snapshots onto the space spanned by the ortho-symplectic basis of size $2N \times 2n$ obtained from the Complex SVD [198] of the matrix $S_X^\tau + iS_V^\tau$. Since the Complex SVD provides the ortho-symplectic basis that minimizes the projection error in the snapshots [198, Theorem 4.6], comparing (5.5.7) and (5.5.8) allows to test the approximability properties of the reduced basis constructed in the dynamical approach.

Moreover, we analyze the evolution of the electric field energy (5.3.2) for the reduced-order approximation, i.e., $\mathcal{E}(X_{r,\tau}^i; \eta_i)$, and for the full-order solution, i.e., $\mathcal{E}(X_\tau^i; \eta_i)$, for each instance η_i of the parameter. This term gives information of the macroscopic behavior of the plasma, and it is also the one affected by more levels of approximation.

Finally, the efficiency of the proposed approach is investigated by comparing the running times required for the integration over a single time step of the fully-discrete DMD-DEIM reduced model (5.5.2) and of the discrete full-order model (5.5.3). The running time for the full model is obtained by summing the times required for each instance of the parameter $\eta_i \in \Gamma_h$. The comparison focuses on the scalability in the approximation of parametric problems as the size p of the parameter set increases, a typical scenario in a multi-query context. To compare the efficiency of the different methods, we analyze the runtime required for integration over a single time interval for all parameter values considered. The values reported were obtained as the average of the runtimes obtained in the first 25 time intervals. For the dynamical reduced basis method, we also analyze separately the contributions to the computational cost due to the basis evolution (5.8)-(5.5.2e) and to the coefficients evolution (5.8)-(5.5.2c)-(5.5.2d), in line with the theoretical findings of Section 5.4.3.

In the construction of the DMD-DEIM reduced model, we consider a tolerance equal to 10^{-5} in the computation of the DMD modes in Algorithm 6. Moreover, the nonlinear system, (5.5.2c) and (5.5.2d), describing the evolution of the increments k_l , $l = 1, \dots, n_s$, is solved using the fixed point iteration method. As a stopping criterion for the nonlinear solver, we check when the relative norm of the update to k_l is smaller than the threshold value 10^{-9} . All numerical simulations are performed using Matlab on computer nodes with Intel Xeon E5-2643 (3.40GHz).

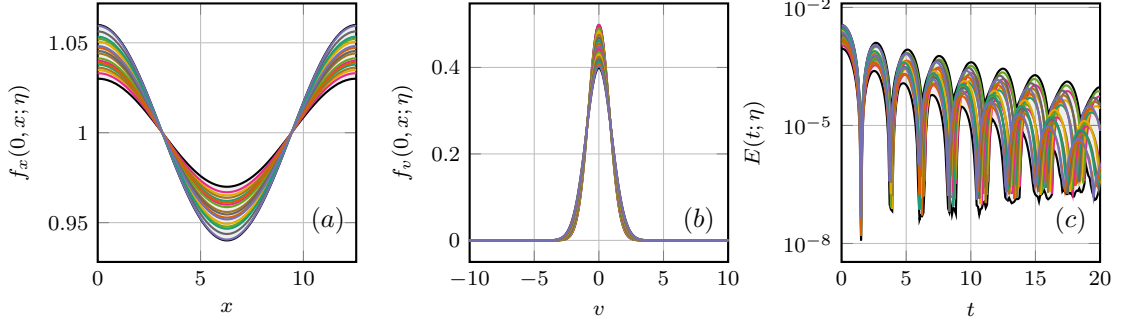


Figure 5.1: LD: Initial positions and velocity distributions for selected values of the parameter in Γ_h (a) – (b). Exponential time decay of the electrostatic energy $\mathcal{E}(X_\tau^i; \eta_i)$ obtained from the full model solution, for selected values of η_i in Γ_h (c). Since not all parameters in Γ_h are reported, the black lines in each subplot are used to mark the region where the plotted quantity is contained, for any value of the parameter in Γ_h .

5.5.2 Weak Landau damping of 1D Langmuir waves

The first application we consider is the study of the damped propagation of small amplitude plasma waves, also known as Landau damping (LD). The resonance between physical particles and the propagating wave generates damping of the electric field energy, without particle collisions. This process is used in particle accelerators to prevent coherent beams oscillations that could cause potential instabilities [126]. The initial condition is given by (5.5.4) with the velocity distribution function

$$f_v(v; \eta) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{v^2}{2\sigma^2}\right), \quad (5.5.9)$$

where the amplitude of the perturbation α and the standard deviation σ of the velocity Maxwellian are the study parameters $\eta = (\alpha, \sigma)$, with $\eta \in \Gamma = [0.03, 0.06] \times [0.8, 1]$, and the perturbation wavenumber k is fixed to 0.5. Following the sampling procedure described in Section 5.5.1, we solve the Landau damping problem for $p = 300$ different realizations of the parameter η . In Figure 5.1(a) and 5.1(b) the initial position and velocity distributions are shown for several of the selected parameter values. We consider periodic boundary conditions on the physical space domain $\Omega_x := (0, \frac{2\pi}{k})$ with a uniform neutralizing background charge. For the numerical solution of the full-order model, we use $N_x = 32$ piecewise linear basis for the Poisson solver, and $N = 5 \times 10^4$ macro-particles for the approximation of the solution density. A uniform time step $\Delta t = 0.0025$ has been adopted for the evolution of particles' positions and velocities over the time interval $\mathcal{T} = (0, 20]$.

In Figure 5.2, the decay of the singular values of the global snapshots matrices S_X and S_V , normalized with respect to the corresponding largest singular value, are compared to the maximum and averages over τ of the normalized singular values of the local counterparts S_X^τ and S_V^τ , computed as described in Section 5.5.1. Concerning the particles position, although a plateau of the singular values can be seen for both global and local matrix, the initial decay is sharper in the local case with singular values that are two orders of magnitude smaller than in the global case, suggesting a more efficient representation using a local low-rank model. This gap increases when considering the particles velocity, suggesting that a *global* reduced basis approach would not be effective in reducing the computational cost of the Landau damping simulation. In Figure 5.3, we report the numerical rank of the matrices S_X^τ and S_V^τ as a function of τ and for different

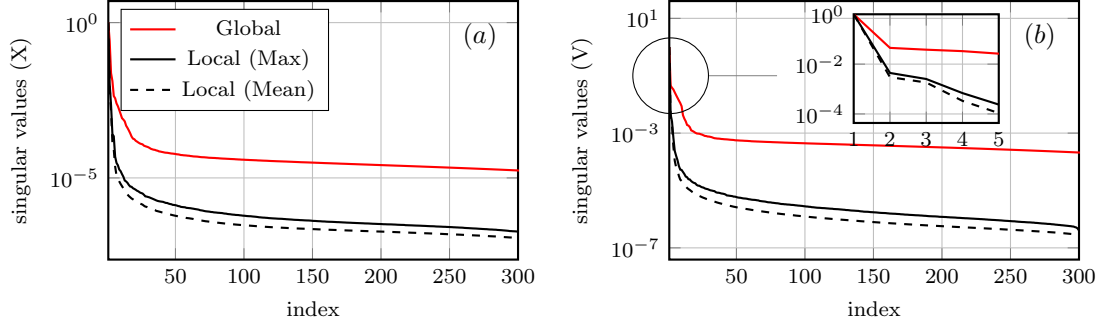


Figure 5.2: LD: Singular values of the global snapshots matrices S_X (a) and S_V (b) compared to the maximum and time average (in τ) of the singular values of the local matrices S_X^τ and S_V^τ .

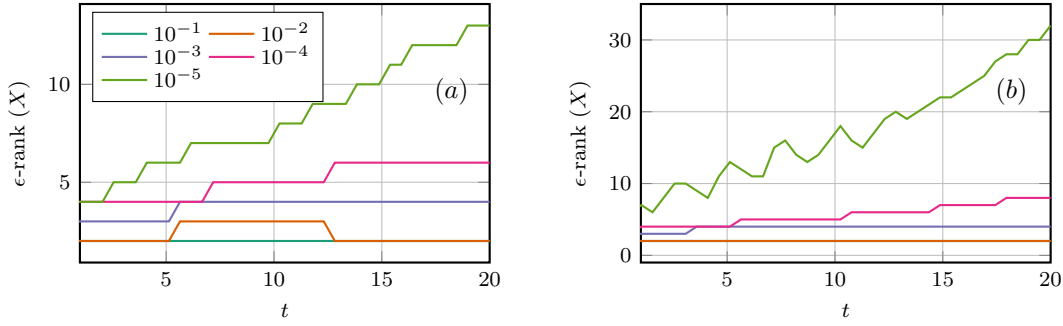


Figure 5.3: LD: Numerical rank of S_X^τ in (a) and S_V^τ in (b), as a function of τ . Different colors are associated with different values of the threshold, according to the legend.

values of the threshold. The numerical rank remains constant and below 4 for tolerances larger than 10^{-3} , grows to 8 for a tolerance of 10^{-4} , and reaches a maximum of 13, for positions and 32 for velocities, when the tolerance is set to 10^{-5} . The increase in the solution complexity over time is partially due to the accumulation of statistical noise associated with the discretization of $f(t, x, v; \eta)$ by macro-particles. In Figure 5.4, in support of this conclusion, we note that as the average number of particles per cell increases during the initial particle loading phase, the numerical ranks of S_X^τ and S_V^τ , at fixed tolerance, decrease. This behavior of the numerical rank suggests that evolving the basis, but keeping the rank of the approximation constant, is sufficient to accurately approximate the solution of the full-order model, at least in this test case.

Concerning the reduced dynamical model, we consider $2n = 4$ as the reduced manifold dimension. For the DEIM reduction described in Section 5.4.2, $d = 32$ interpolation points have been used to reduce the approximation error, and $n_{\text{DEIM}} = 12$ DEIM indices are updated at each time step, for the sake of efficiency, according to (5.4.10). All DEIM indices are recomputed every $k_{\text{DEIM}} = 3$ time steps.

For this test case, we include a numerical study of the evolution of the approximation errors $\varepsilon_{\text{rel},X}$ and $\varepsilon_{\text{rel},V}$ in (5.5.7) under variations of the size p^* of the subset of the parameters used to evolve the basis effectively, according to Section 5.4.2, and the length $T + 1$ of the time window adopted to harvest the self-consistent electric potential $\Phi(X_r^i)$ for the DMD extrapolation step, as described in Section 5.4.2. In particular, we consider $p^* \in \{8, 12, 16\}$ and $T \in \{3, 5\}$ and the results are shown in Figure 5.5. In all tested combinations of T and p^* , the error is proportional to the best approximation error, both in position and velocity. As p^* increases, both errors

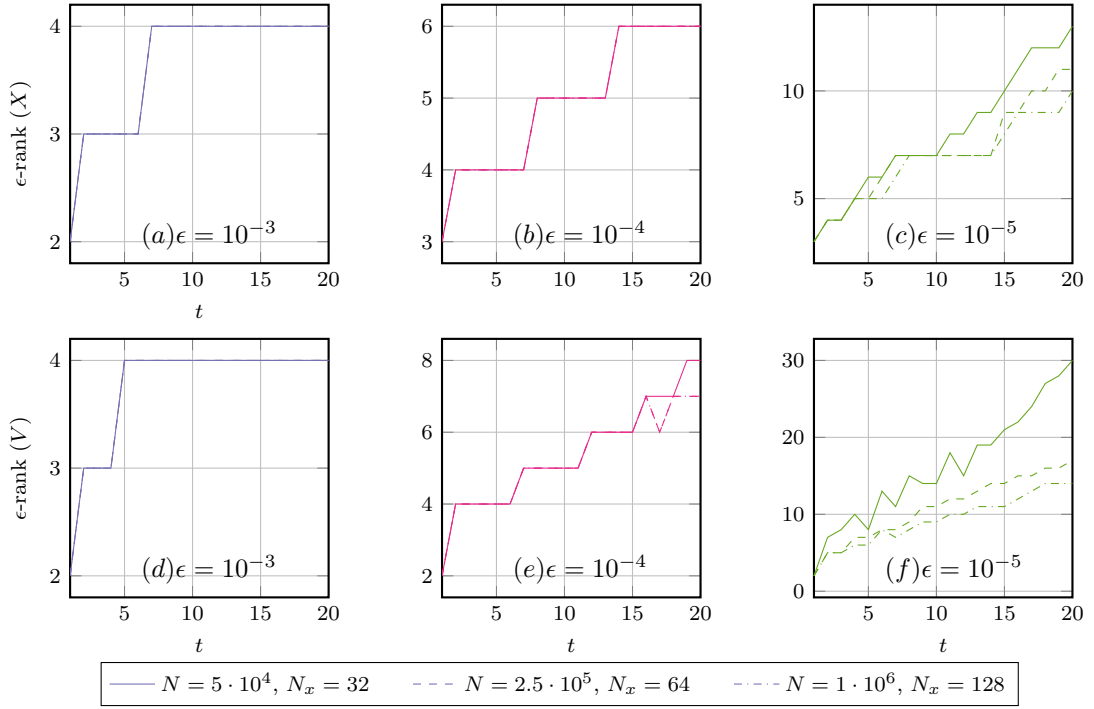


Figure 5.4: LD: Evolution of the numerical ranks of S_X^T (a) – (c) and of S_V^T (d) – (f) for different threshold indicated by ϵ . In each subfigure, the rank behavior for different values of N_x and N , in the discretization of the full-order model, is compared.

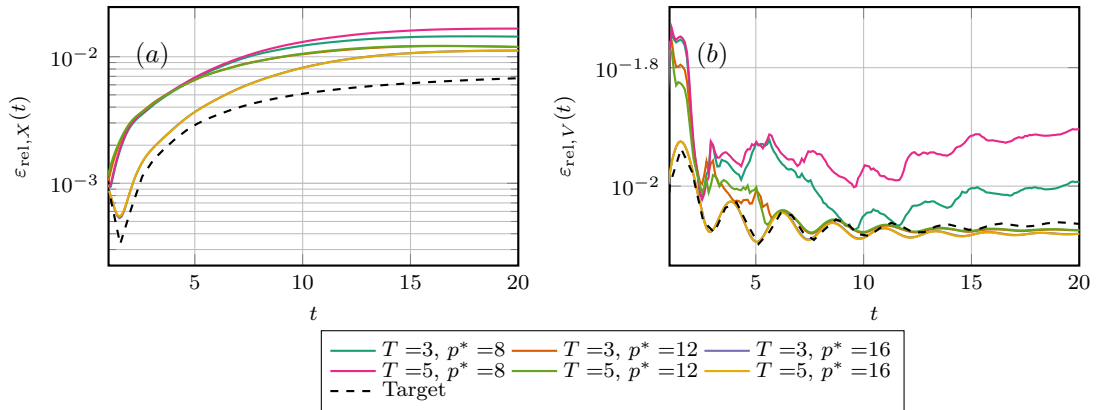


Figure 5.5: LD: Evolution of the position (a) and velocity (b) relative errors, as defined in (5.5.7), for different choices of p^* and T . These errors are compared to the target values given by the position component $\varepsilon_{\text{rel},X}^{\text{Target}}$ and the velocity component $\varepsilon_{\text{rel},V}^{\text{Target}}$ of the relative projection errors defined in (5.5.8). The target reduced basis has dimension 4 and is computed, for each time step, using the Complex SVD algorithm, as described in Section 5.5.1.

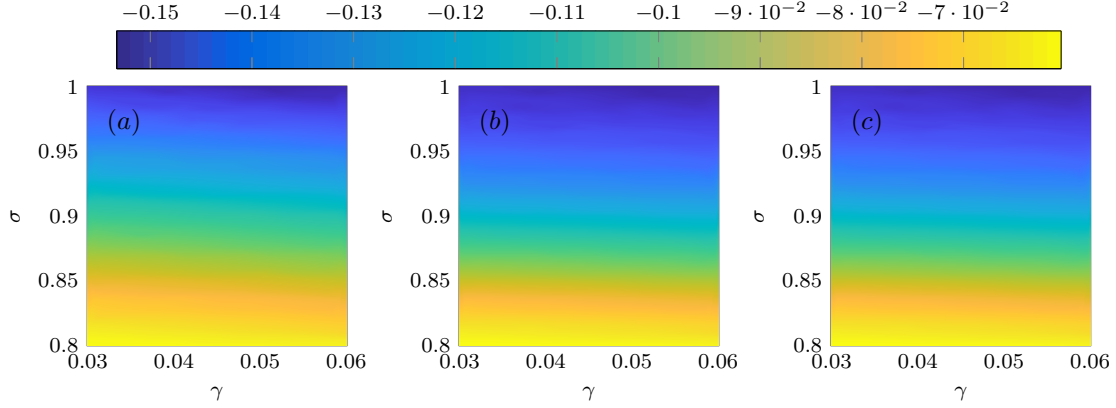


Figure 5.6: LD: Damping rates of the exponential time decay of the electric energy $\mathcal{E}(X^i; \eta)$, defined in (5.3.2), as a function of the two-dimensional parameter $\eta = (\alpha, \sigma)$. The plots refer to (a) the full-order model; (b) the dynamical reduced model with $T = 3$ and $p^* = 16$; and (c) the dynamical reduced model with $T = 5$ and $p^* = 8$.

decrease: this is expected since a more refined sampling of the subset Γ_h^* results in a more accurate representation of the dynamics of evolution of the bases in (5.4.14b). We also note that, for $p^* = 16$, the error $\varepsilon_{\text{rel}, V}$ is, at several time instances, smaller than the target value. This performance can be explained by the fact that the optimality of the Complex SVD algorithm concerns the projection of the entire state $[S_X^T S_V^T]$ and not of its components individually. We observe that the DMD window length $T + 1$ has no impact on the error when p^* is large, and a small accuracy degradation is even registered for $p^* = 8$ when $T = 5$ is chosen over $T = 3$. The optimal choice of T remains an open problem: as pointed out in [80], it should capture slow and fast scales of the local dynamics, but a rigorous optimization strategy would require a study of the multi-scale properties of the solution to the Vlasov–Poisson equation for each of the parameter realization considered. However, we stress that the results are relatively robust concerning this parameter.

Landau theory [27] establishes that, for small perturbations of the initial analytical data of the form (5.5.9), the electric energy $\mathcal{E}(X^i; \eta_i)$ decays (in time) exponentially with a damping factor that depends on the standard deviation σ of the Maxwellian distribution f_v but is independent of the amplitude of the perturbation α . As can be seen from Figure 5.6, this dependence of the damping rate on the considered parameters is captured by the reduced model numerical solution.

In Figure 5.7(a), we report the evolution of the relative error of the Hamiltonian (5.3.1) computed in the reduced and full model solutions, i.e.

$$\frac{\|\mathcal{H}([X_\tau, V_\tau]) - \mathcal{H}(U_\tau Z_\tau)\|_2}{\|\mathcal{H}([X_\tau, V_\tau])\|_2}, \quad \forall \tau = 1, \dots, N_\tau. \quad (5.5.10)$$

It is observed that the error is bounded and grows only slowly over time. The reason why the Hamiltonian is not exactly preserved is twofold: the numerical temporal integrator is symplectic but not Hamiltonian-preserving, and the reduced model possesses a Hamiltonian structure but with an approximate Hamiltonian function. To better understand these two sources of errors, we consider the error in the Hamiltonian at two consecutive time instances of the solution. For the DMD-DEIM reduced model discretized with the partitioned Runge–Kutta method described in

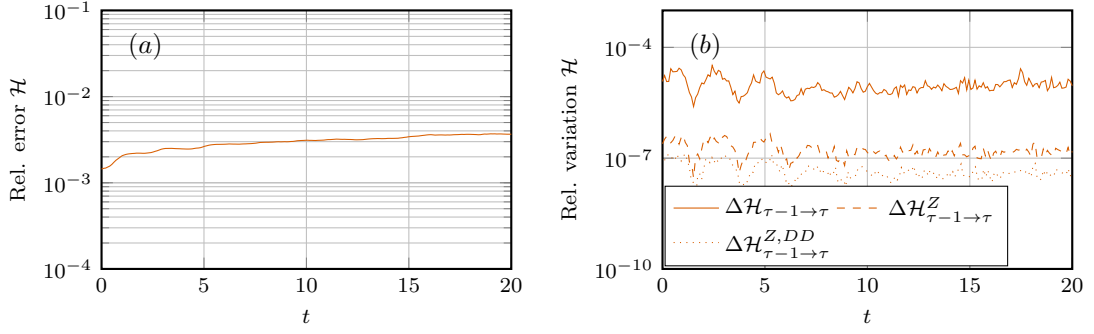


Figure 5.7: LD: Evolution of the relative error (5.5.10) of the Hamiltonian (a). Evolution of the components $\Delta \mathcal{H}_{\tau-1 \rightarrow \tau}$, $\Delta \mathcal{H}_{\tau-1 \rightarrow \tau}^Z$ and $\Delta \mathcal{H}_{\tau-1 \rightarrow \tau}^{Z,DD}$ of the error bound (5.5.11), (5.5.12) in the local conservation of the reduced Hamiltonian (b). The values of the hyper-parameters are set to $p^* = 12$ and $T = 3$, respectively.

Section 5.5, it holds

$$\Delta \mathcal{H}_{\tau-1 \rightarrow \tau} := \|\mathcal{H}(U_\tau Z_\tau) - \mathcal{H}(U_{\tau-1} Z_{\tau-1})\|_2 \leq \left\| \mathcal{H}(U_\tau Z_\tau) - \mathcal{H}(U_{\tau-\frac{1}{2}} Z_\tau) \right\|_2 \quad (5.5.11a)$$

$$+ \left\| \mathcal{H}(U_{\tau-\frac{1}{2}} Z_{\tau-1}) - \mathcal{H}(U_{\tau-1} Z_{\tau-1}) \right\|_2 \quad (5.5.11b)$$

$$+ \left\| \mathcal{H}(U_{\tau-\frac{1}{2}} Z_\tau) - \mathcal{H}(U_{\tau-\frac{1}{2}} Z_{\tau-1}) \right\|_2. \quad (5.5.11c)$$

The first two terms (5.5.11a) and (5.5.11b) depend on the numerical time integration of the basis equation (5.8), while the last term (5.5.11c) also depends on the DMD-DEIM approximation of the Hamiltonian. In particular, it holds

$$\Delta \mathcal{H}_{\tau-1 \rightarrow \tau}^Z := \left\| \mathcal{H}(U_{\tau-\frac{1}{2}} Z_\tau) - \mathcal{H}(U_{\tau-\frac{1}{2}} Z_{\tau-1}) \right\|_2 \leq \left\| \mathcal{H}(U_{\tau-\frac{1}{2}} Z_\tau) - \mathcal{H}_{U_{\tau-\frac{1}{2}}}^{\text{DD}}(Z_\tau, t_\tau) \right\|_2 \quad (5.5.12a)$$

$$+ \left\| \mathcal{H}_{U_{\tau-\frac{1}{2}}}^{\text{DD}}(Z_{\tau-1}, t_{\tau-1}) - \mathcal{H}(U_{\tau-\frac{1}{2}} Z_{\tau-1}) \right\|_2, \quad (5.5.12b)$$

$$+ \left\| \mathcal{H}_{U_{\tau-\frac{1}{2}}}^{\text{DD}}(Z_\tau, t_\tau) - \mathcal{H}_{U_{\tau-\frac{1}{2}}}^{\text{DD}}(Z_{\tau-1}, t_{\tau-1}) \right\|_2, \quad (5.5.12c)$$

where $\mathcal{H}_U^{\text{DD}}$ is defined in (5.4.11). The first two terms (5.5.12a) and (5.5.12b) depend on the approximation of the Hamiltonian introduced by the DMD-DEIM method, while the last term (5.5.12c), that we dub $\Delta \mathcal{H}_{\tau-1 \rightarrow \tau}^{Z,DD}$, is only associated with the numerical time integrator of the coefficient equation. In Figure 5.7(b) we report the time evolution of $\Delta \mathcal{H}_{\tau-1 \rightarrow \tau}$, $\Delta \mathcal{H}_{\tau-1 \rightarrow \tau}^Z$ and $\Delta \mathcal{H}_{\tau-1 \rightarrow \tau}^{Z,DD}$, for the hyper-parameters $T = 3$ and $p^* = 12$. We can observe that the DMD-DEIM method provides a good approximation of the Hamiltonian (dashed line). To study the algorithm efficiency, we investigate the runtime as a function of the number p of tested parameters. The proposed approach outperforms the full-order solver, as shown in Figure 5.8(a), and the gap widens as the value of p increases. Depending on the choice of hyper-parameters of the reduced model, the algorithm speed-up varies between 1.9 and 3.3 when $p = 30$ and between 46 and 71 when $p = 1000$. For $p \geq 2000$, the evolution of the expansion coefficients (5.4.14a) becomes computationally more demanding than the evolution of the reduced basis (5.4.14b), as shown in Figure 5.8(b), and the overall computational cost of the reduced model begins to grow approximately linearly as a function of p . Thus, for values of p larger than 2000, the ratio between the time required to integrate the full model and the time to integrate the reduced model remains constant, with speed-ups ranging between 141 and 183, depending on the values

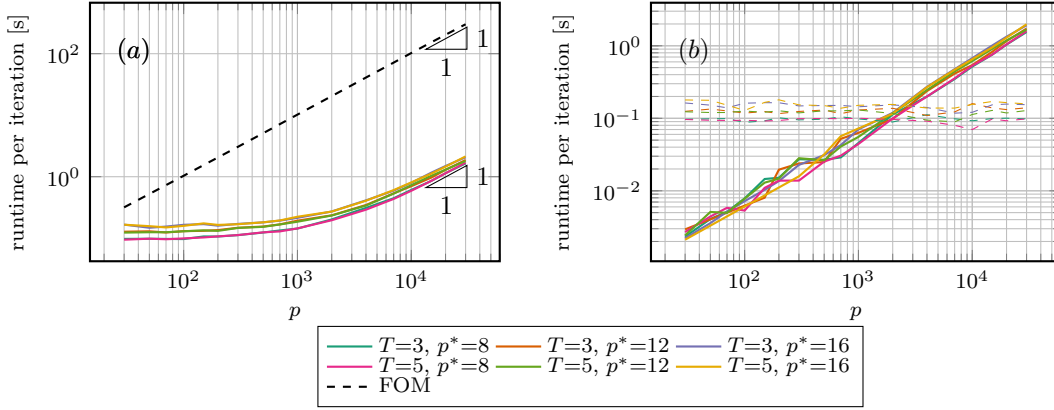


Figure 5.8: LD: Comparison of the runtime (in seconds) between the full-order solver and the dynamical reduced basis approach for different hyper-parameter configurations, as function of the parameter sample size p (a). Separation of contributions to the running time of the reduced model due to basis evolution (dashed lines) and coefficients evolution (continuous line) (b).

of the hyper-parameters. We also remark that the computational cost to evolve the reduced basis (dashed lines in Figure 5.8(b)) is independent of p because it only depends on the number p^* of subsamples.

5.5.3 Nonlinear Landau damping of 1D Langmuir waves

For larger initial perturbation amplitudes, the linear theory does not hold and, after an initial shearing in phase space, leading to Landau damping, the damping is halted, and strong particle-trapping vortices are formed, leading to a growth of the system's potential energy [169]. To simulate this scenario, starting from the same initial condition (5.5.9) and periodic domain $\Omega_x := (0, \frac{2\pi}{k})$ of the previous test, we take the parameter $\eta = [\alpha, \sigma]$ in the domain $\Gamma = [0.46, 0.5] \times [0.96, 1]$ and consider $p = 300$ different realizations. In Figure 5.9, we report the behavior of the initial velocity and position distributions along with the evolution in time of the electric field energy. The full-order simulations are conducted using $N_x = 64$ degrees of freedom for the discretization of the Laplacian operator, and $N = 10^5$ particles for the approximation of the distribution function. We consider the time interval $\mathcal{T} = (0, 40]$, with $\Delta t = 0.002$ and the numerical time integrators described in Section 5.5.

The decay of the singular values of the global and local snapshot matrices, defined in (5.5.6) and (5.5.5), is shown in Figure 5.10. Compared to the linear Landau damping, the nonlinear test case is unsuitable for reduction with a global reduced basis approach both in terms of particles position and velocity. Regarding reducibility via a local basis in time, we note that although the problem is more challenging than the weak Landau damping, the normalized singular values of S_X^τ and S_V^τ reach $3.9 \cdot 10^{-4}$ and 2.2×10^{-3} , respectively, at the sixth singular value, making the problem amenable to local reduction.

Similar conclusions are drawn from the behavior of the numerical rank, shown in Figure 5.11 as a function of time, from which we also note that the problem becomes significantly more complex in the final part of the time interval considered, corresponding to the formation of the particle attractor vortices and as the nonlinear contribution to the dynamics of the particles becomes dominant.

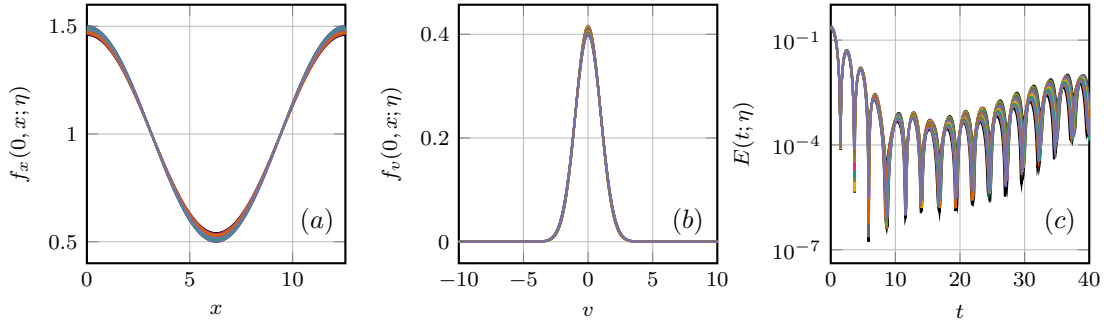


Figure 5.9: NLD: Initial position and velocity distributions for selected values of the parameter in Γ_h (a) – (b). Exponential time decay of the electrostatic energy $\mathcal{E}(X_\tau^i; \eta_i)$ obtained from the full model solution, for selected values of η_i in Γ_h (c). Since not all parameters in Γ_h are reported, the black lines in each subplot are used to mark the region where the plotted quantity is contained, for any value of the parameter in Γ_h .

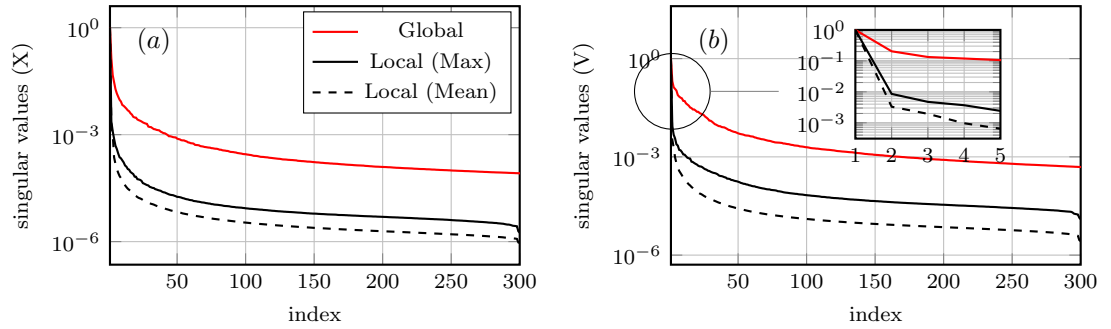


Figure 5.10: NLD: Singular values of the global snapshots matrices S_X and S_V compared to the maximum and time average (in τ) of the singular values of the local matrices S_X^τ and S_V^τ .

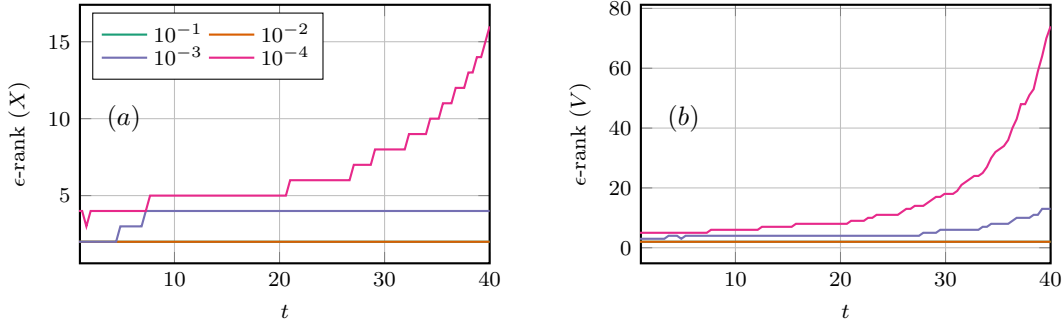


Figure 5.11: NLD: Numerical rank of S_X^τ in (a) and S_V^τ in (b), as a function of τ . Different colors are associated with different values of the threshold, according to the legend.

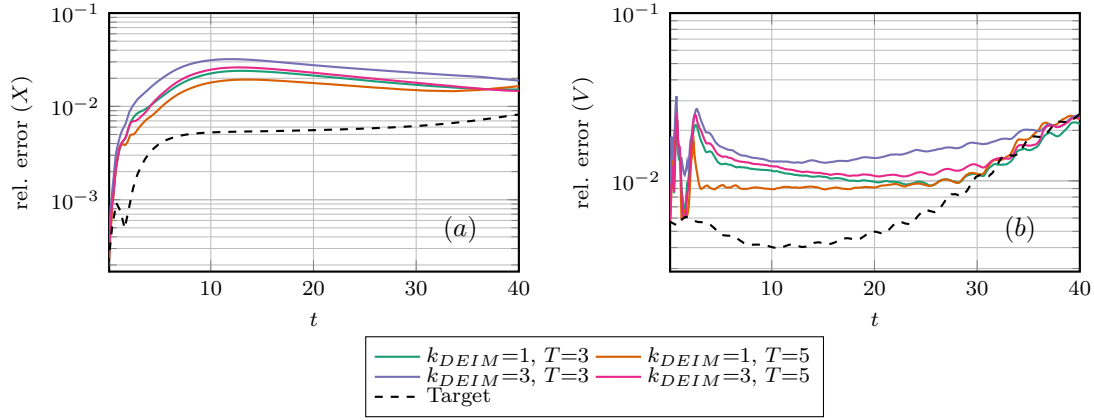


Figure 5.12: NLD: Evolution of the position (a) and velocity (b) relative errors, as defined in (5.5.7), for different choices of p^* and T . These errors are compared to the target values given by the position component $\varepsilon_{\text{rel},X}^{\text{Target}}$ and the velocity component $\varepsilon_{\text{rel},V}^{\text{Target}}$ of the relative projection errors defined in (5.5.8). The target reduced basis has dimension 6 and is computed, for each time step, using the Complex SVD algorithm, as discussed in 5.5.1.

To reduce this test problem, we consider a symplectic dynamical basis of dimension $2n = 6$ and the same number $d = 32$ of DEIM interpolatory indices as used for the weak Landau damping. In addition, a subset of $p^* = 8$ parameters, taken according to Section 5.4.1, is considered for the efficient evolution of the basis. The relative errors for the different choices of the DMD windows length $T + 1$ and frequency k_{DEIM} are shown in Figure 5.12: the error does not deteriorate over time for any of the chosen hyper-parameters, and the increase of k_{DEIM} only marginally impacts the performances of the reduced model.

In Figure 5.13, we plot the distribution function $f_h(t, x, v; \eta)$ reconstructed from the macro-particles for the parameter $\eta = (0.4912, 0.9889)$. The numerical solution of the approximate reduced model is in good agreement with the full model solution, and the various dynamical stages, from the initial shearing to the development of the two particle-trapping vortices, are correctly captured. Furthermore, although tiny artifacts in the vortex structure can be observed in the case of hyper-parameters $T = 3$ and $k_{\text{DEIM}} = 3$ at $t = 40$, this is not the case for the choice $T = 5$ and $k_{\text{DEIM}} = 1$.

To better understand the macroscopic effects of the order reduction on the numerical solution, we consider, in Figure 5.14, the evolution of the electric field energy for different realizations of

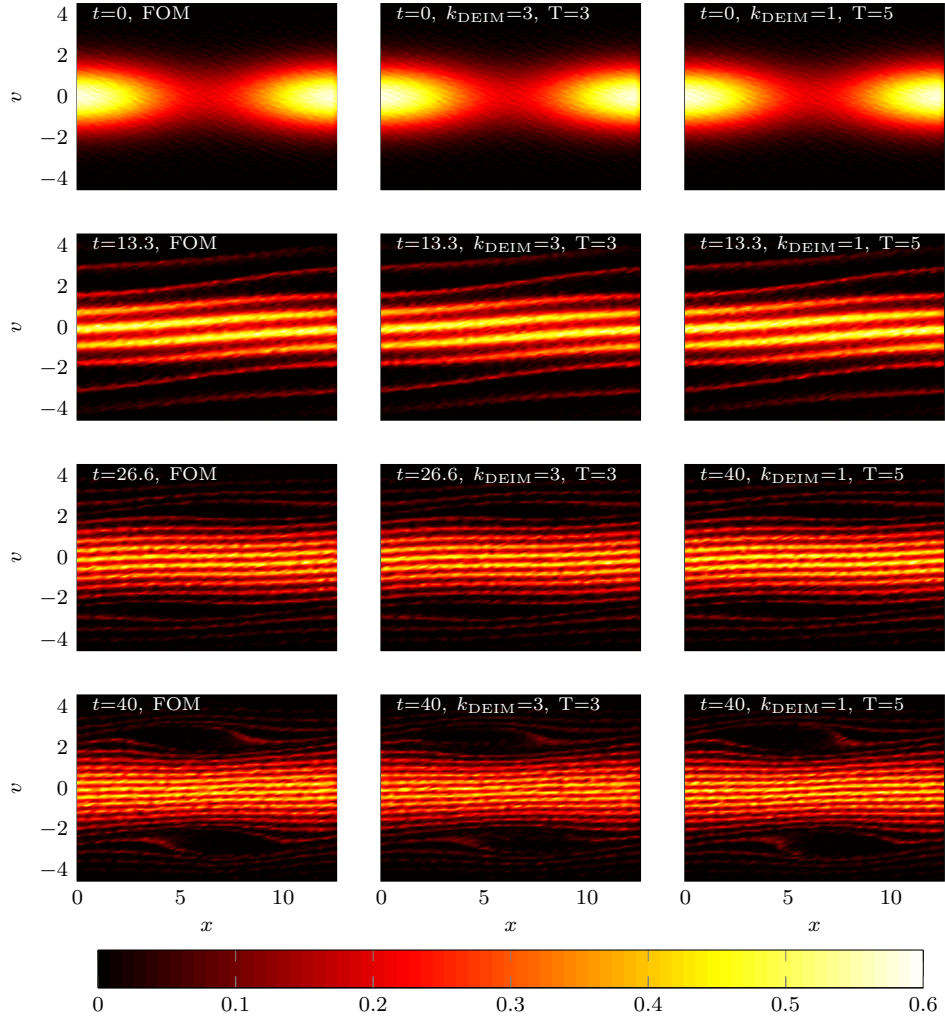


Figure 5.13: NLD: Numerical distribution function for $\eta = (0.4912, 0.9889)$ at different times obtained from (a) the full-order model; (b) the dynamical reduced model with $T = 3$ and $k_{\text{DEIM}} = 3$; and (c) the dynamical reduced model with $T = 5$ and $k_{\text{DEIM}} = 1$. Starting from the perturbed Maxwellian distribution, particles with different energies oscillate with different frequencies leading to the typical filamentation that starts developing at $t = 13.33$. Two trapping vortices, centered at opposite phase velocities, form at $t = 26.66$ and fully develop at $t = 40$.

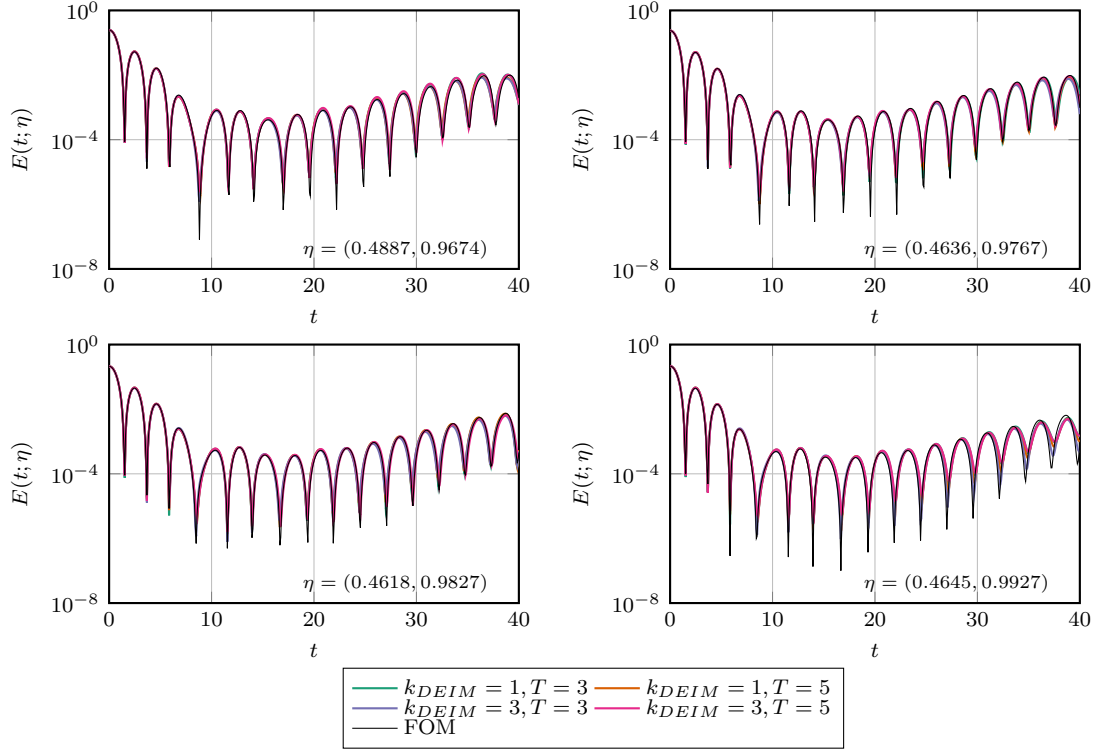


Figure 5.14: NLD: Evolution of the electric field energy $\mathcal{E}(\cdot; \eta_i)$. The energy is evaluated at the positions $X_{r,\tau}^i$ computed using the high-fidelity solver and at the positions $X_{r,\tau}^i$ computed using the reduced model, for different values η_i of the parameter.

the parameter. The reduced model solution gives accurate results, both in terms of the amplitude and frequency of the peaks. As a further analysis, in Figure 5.16, we report the exponential damping rate of $\mathcal{E}(X_{r,\tau}^i; \eta_i)$, which is obtained during the initial phase of Landau damping, and the exponential growth rate of $\mathcal{E}(X_{r,\tau}^i; \eta_i)$ that characterizes the subsequent formation of particle-trapping vortices in phase space. For $\eta_i = (0.5, 1)$, the values obtained are around -0.287 and 0.078 , which is in agreement with the literature [152]. We show in Figure 5.15 the peaks of $\mathcal{E}(X_{r,\tau}^i)$ that have been fitted for the calculations of the damping and growth rates. Finally, in Figure 5.17, we compare the running times of the full-order solver and the reduced-order solver. For this numerical simulation, the choice of the hyper-parameters T and k_{DEIM} has a mild impact on the computational cost required to advance the reduced state of a single time step. Once the cost to integrate the evolution of the coefficients has exceeded the cost to integrate the basis equation, the most computationally expensive choice of hyper-parameters (i.e., $T = 5$ and $k_{\text{DEIM}} = 1$) is only around 1.35 times more demanding than the computationally cheapest choice (i.e., $T = 3$ and $k_{\text{DEIM}} = 3$). This result is in agreement with the analysis of the arithmetic complexity of the reduction algorithm presented in Section 5.4.3: for large p , the dominant cost has order $O(pn^2) + O(N_x r_\tau p) + O(pn_d n) + O(pn_d c)$, which depends on the hyper-parameters only via the number r_τ of retained DMD eigenvalues. Although the value r_τ might be different for different choices of the window length T , this does not significantly affect the computational cost of the algorithm.

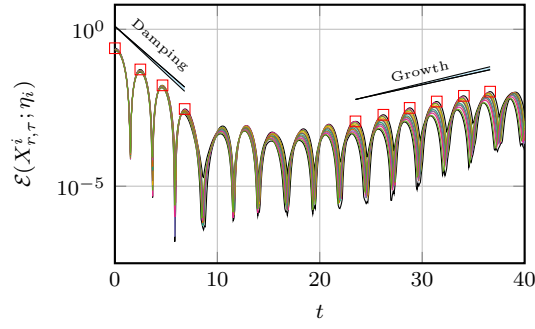


Figure 5.15: NLD: Peaks of the electric field energy $\mathcal{E}(X_{r,\tau}^i; \eta_i)$ selected for the computation of the exponential damping and growth rates.

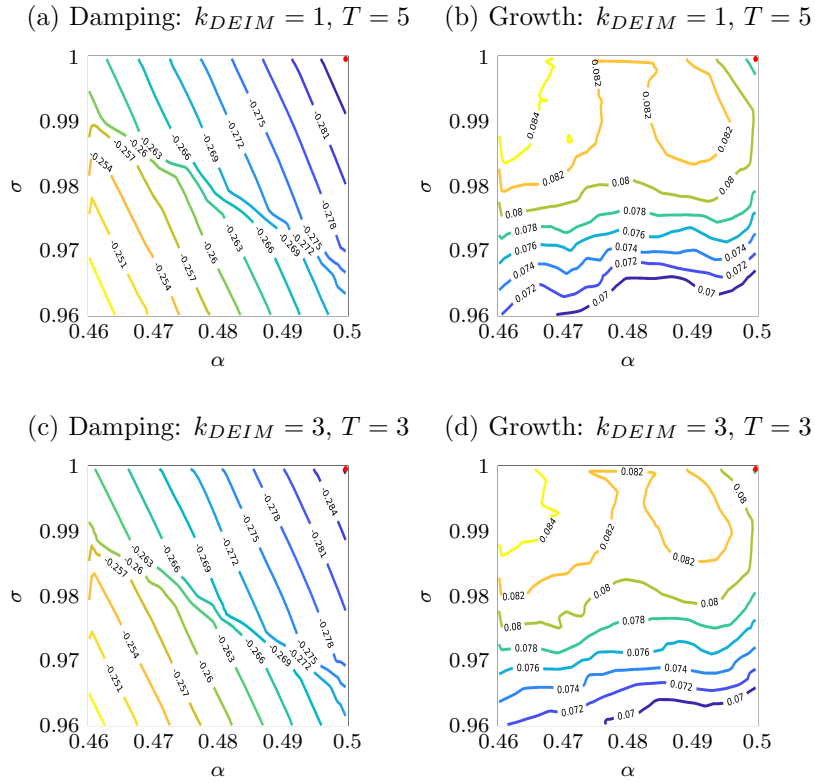


Figure 5.16: NLD: Contour plots of the damping rate ((a), (c)) and growth rate ((b), (d)) of the electric field energy $\mathcal{E}(X_{r,\tau}^i; \eta_i)$ for different values of k_{DEIM} and T .

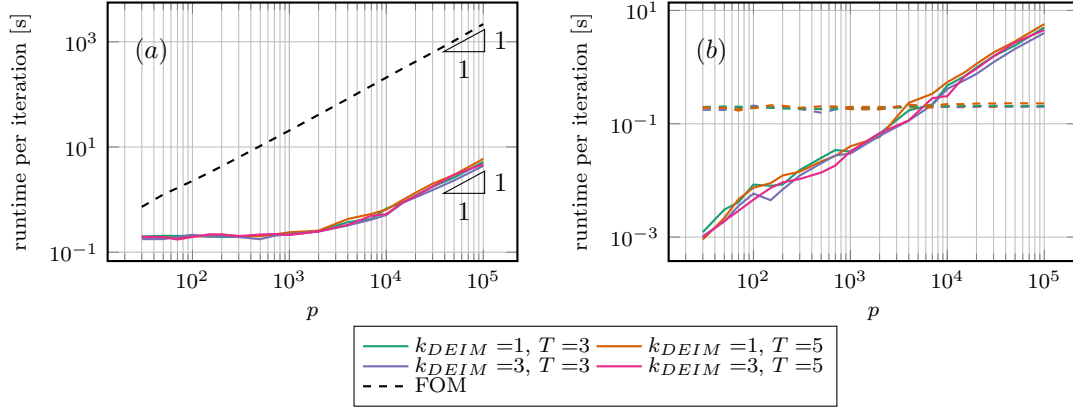


Figure 5.17: NLD: (a) Comparison of the runtime (in seconds) between the full-order solver and the dynamical reduced basis approach for different hyper-parameter configurations, as a function of the parameter sample size p . (b) Separation of contributions to the running time of the reduced model due to basis evolution (5.8) (dashed lines) and coefficient evolution (5.8) (continuous line).

5.5.4 Two-stream instability

The two-stream instability is a well-known instability in plasma physics generated by two counterstreaming beams, where the kinetic energy of particles excites a plasma wave and, consequently, transfers to electric potential energy [10]. In this study, we focus on the temporal interval that includes the first two stages in which the evolution of electric field energy is distinct, namely the initial, short transient stage, and the subsequent growth stage. For the latter stage, the dynamic is defined by the interplay between harmonics characterized by different growth rates. We consider the spatial domain $\Omega_x := (0, \frac{2\pi}{k})$ with periodic boundary conditions. The initial velocity distribution is given by

$$f_v(v; \eta) = \frac{1}{2\sqrt{2\pi}\sigma} \exp\left(-\frac{(v-v_0)^2}{2\sigma^2}\right) + \frac{1}{2\sqrt{2\pi}\sigma} \exp\left(-\frac{(v+v_0)^2}{2\sigma^2}\right), \quad (5.5.13)$$

where $v_0 = 3$ is the initial velocity displacement in phase space. The wavenumber k of the perturbation is set to 0.2, and the parameter $\eta = (\sigma, \alpha)$ varies in the domain $\Gamma = [0.009, 0.011] \times [0.98, 1.02]$ discretized using $p = 300$ samples. Figures 5.18(a)-(b) show the initial parametric distributions of position and velocity. The evolution of electric energy is shown in Figure 5.18(c). We note that the ratio between the maximum and the minimum of $\mathcal{E}(X_{r,\tau}^i; \eta_i)$ under variations of the parameter η_i is slightly larger than 2, indicating a certain variability of the solution in the range of parameters considered.

The distribution function $f(t, x, v; \eta)$ is approximated with $N = 1.5 \cdot 10^5$ computational macro-particles, and $N_x = 64$ piecewise linear functions have been adopted to discretize the Poisson equation. We solve the discrete systems (5.5.2) and (5.5.3) with a time step $\Delta t = 0.0025$ over the temporal domain $\mathcal{T} = (0, 20]$.

Compared to previous tests, there is a dissimilarity between the decays of the singular values of the snapshots matrices for positions and velocities. The decay of the singular values of the S_X and S_X^τ is rather fast, and the singular values of the global snapshot matrix become smaller than 10^{-3} after the fourteenth singular value. On the contrary, the decay of the singular values of the snapshots matrices associated with the velocity of the particles suggests that a local basis might be more effective in approximating the evolution of the particles' velocity.

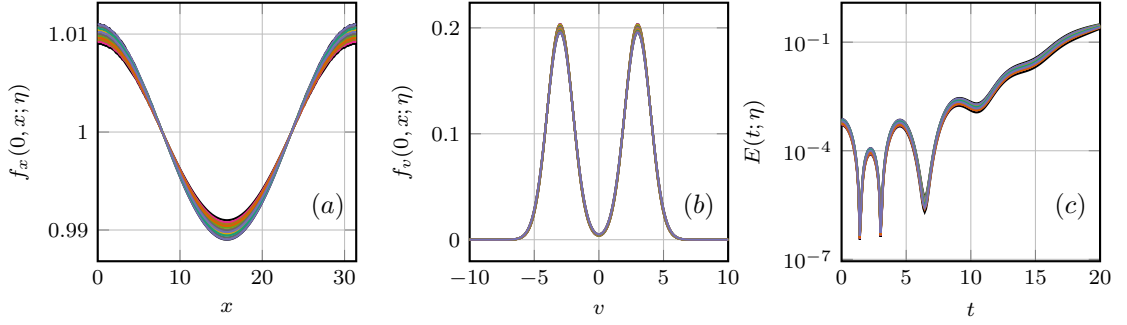


Figure 5.18: TSI: (a) – (b) Initial position and velocity distributions for selected values of the parameter in Γ_h . (c) Exponential time decay of the electrostatic energy $\mathcal{E}(X_\tau^i; \eta_i)$ obtained from the full model solution, for selected values of η_i in Γ_h . Since not all parameters in Γ_h are reported, the black lines in each subplot are used to mark the region where the plotted quantity is contained, for any value of the parameter in Γ_h .

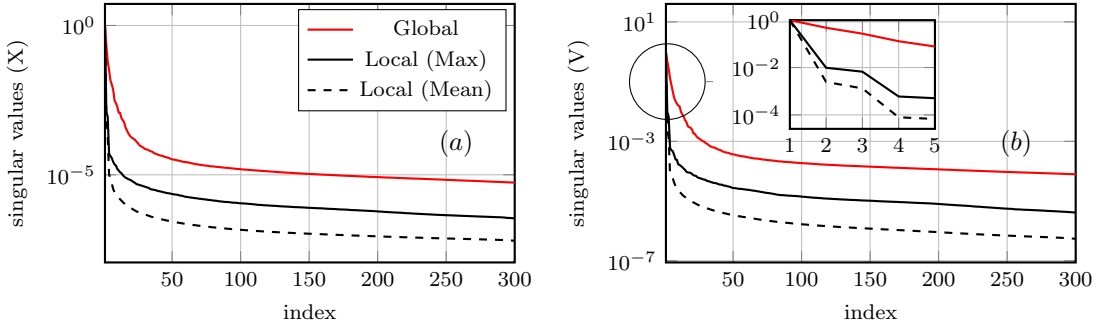


Figure 5.19: TSI: Singular values of the global snapshots matrices S_X and S_V compared to the maximum and time average of the singular values of the local trajectories matrices S_X^τ and S_V^τ . We study position (a) and velocity (b) variables separately. The singular values are normalized using the largest singular value for each case.

A similar conclusion can be drawn from the evolution of the numerical ranks of S_X^τ and S_V^τ , shown in Figures 5.20(a)-(b). In Figure 5.20(c), we also report the numerical rank of the self-consistent electric potential $\phi(X_i^\tau)$ obtained from the full model at different time instants t_τ . It can be observed that the electric potential is low-rank throughout the simulation, which justifies the use of hyper-reduction strategies, in our case provided by the DMD-DEIM approach, to accelerate the computation of the nonlinearity in the Vlasov–Poisson equation. This speedup is ensured on the entire time interval since the numerical rank remains, on average, constant over time. We observe that, in principle, the rank of the hyper-reduced approximation provided by DMD and DEIM can change over time. As a general consideration, although the electric potential depends on the particles’ positions, there seems to be no straightforward connection between the reducibility properties of the sets $\{X_i^\tau\}_\tau$ and $\{\Phi(X_i^\tau)\}_\tau$.

In Figure 5.21, the evolution of the errors in the positions and velocities of the particles are reported: the approximability properties of the dynamical approach are not affected by the choice of the length T of the DMD window and the frequency k_{DEIM} of full updates of the DEIM indices. The dominant component of the error is the projection error. The growth in the error can be explained by the increase in time of the rank of the full model solution. The same growth rate and the small difference between the relative error for the proposed approach and the relative

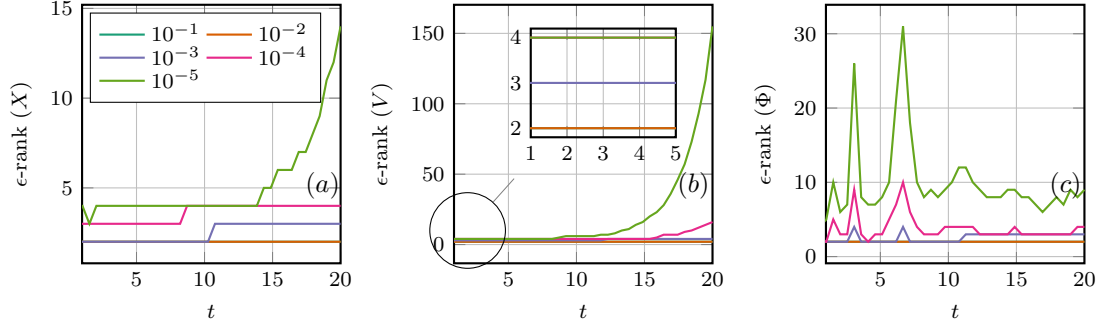


Figure 5.20: TSI: Numerical rank of S_X^τ in (a) and S_V^τ in (b), as a function of τ . Different colors are associated with different values of the threshold, according to the legend. In (c) is reported the evolution of the numerical rank of the electric potential obtained from the full model.

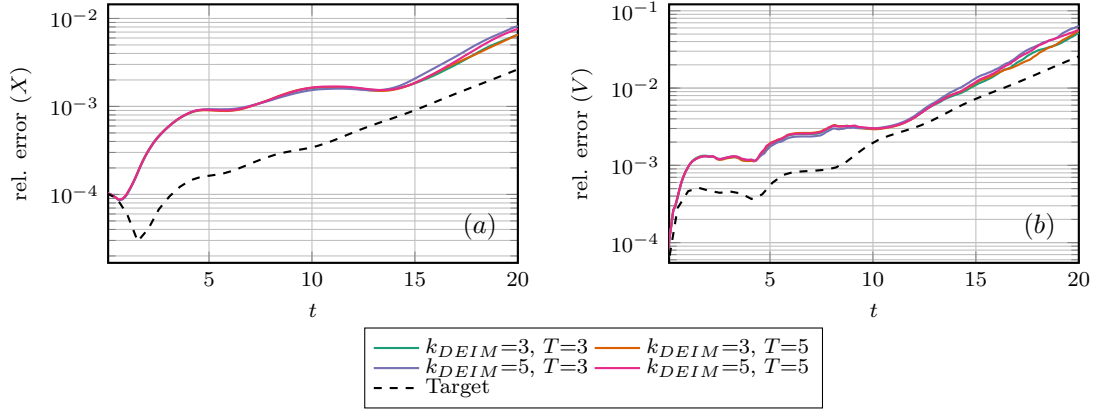


Figure 5.21: TSI: Evolution of the position (a) and velocity (b) relative errors, as defined in (5.5.7), for different choices of k_{DEIM} and T . These errors are compared to the target values given by the position component $\varepsilon_{\text{rel},X}^{\text{Target}}$ and the velocity component $\varepsilon_{\text{rel},V}^{\text{Target}}$ of the relative projection errors defined in (5.5.8). The target reduced basis has dimension 4 and is computed, for each time step, using the Complex SVD algorithm, as described in Section 5.5.1.

projection error committed using an optimal ortho-symplectic basis of dimension $2n$ for both positions and velocities support the same conclusion. We also stress that the error scales for the two components are different, as to be expected from the trend of singular values, and the greater accuracy in approximating the position is not affected by the velocity error. To study the convergence properties of the proposed scheme in terms of the reduced basis size $2n$, we consider the same test case but in the parametric domain $\Gamma = [0.0075, 0.0125]$ and with a larger number $N = 5 \cdot 10^5$ of macro-particles. Increasing the size of the parametric domain produces an increase in the rank of the initial datum, while increasing the number of macro-particles reduces the statistical noise in the numerical rank that plagues particle simulations. As expected, Figure 5.22 shows a decrease of the error between the reduced and the full-order solution as the size n of the dynamical reduced basis is increased. The numerical rank of the full model solution, shown in Figure 5.20, and the error evolution in Figures 5.21 and 5.22 suggest that enlarging the reduced basis U over time to increase the rank of the reduced model solution may improve the accuracy. Although a rank-adaptive algorithm has been proposed in [128], its direct application to the Vlasov–Poisson DMD–DEIM reduced model (5.4.14) would require the solution of a linear

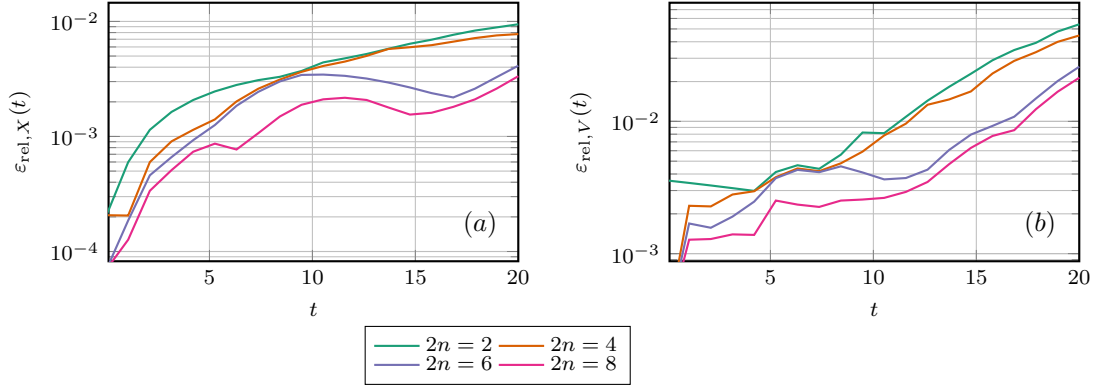


Figure 5.22: TSI: Evolution of the position (a) and velocity (b) relative errors, as defined in (5.5.7), for different choices of the reduced basis dimension $2n$. The values of the hyper-parameters k_{DEIM} and T are both set to 3.

system of dimension proportional to the number of particles to determine a candidate vector for the expansion of U . This cost would limit the computational speed-up obtained in the reduced model when compared to the full-order model. For this case, the exploration of rank-adaptive algorithms provides a possible direction for future investigation.

The evolution of the electric energy (5.3.2) is shown in Figure 5.23: the behavior of the electric energy obtained from the approximate reduced model almost coincides with the full model except for slight mismatches in the amplitude of the oscillations during the transient phase.

Similar to the two numerical tests on the Landau damping, the proposed dynamical model order reduction method outperforms, in terms of efficiency, the full-order solver, with speed-ups reaching 340 for $p = 10^5$ parameters, as shown in Figure 5.24.

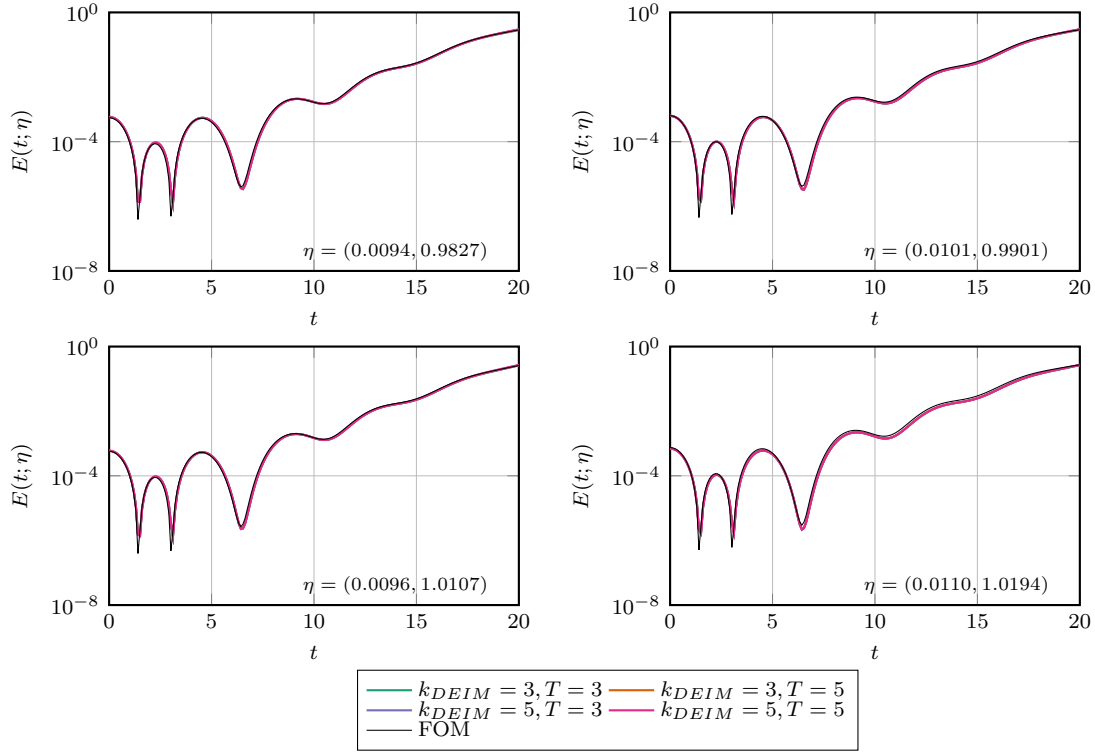


Figure 5.23: TSI: Evolution of the electric field energy $\mathcal{E}(\cdot; \eta_i)$. The energy is evaluated at the positions X_τ^i computed using the high-fidelity solver and at the positions $X_{r,\tau}^i$ computed using the reduced model, for different values η_i of the parameter.

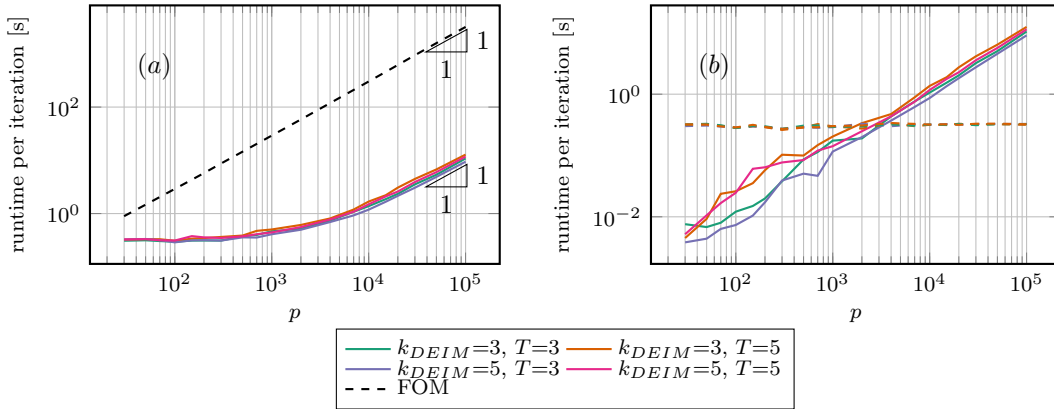


Figure 5.24: TSI: (a) Comparison of the runtime (in seconds) between the full-order solver and the dynamical reduced basis approach for different hyper-parameter configurations, as function of the parameter sample size p . (b) Separation of contributions to the running time of the reduced model due to basis evolution (5.8) (dashed lines) and coefficients evolution (5.8) (continuous line).

6 Conclusion of Part I

In Chapter 2, we have shown that conservation of nonlinear invariants is not, in general, guaranteed with conventional model reduction techniques. The violation of such invariants often results in a qualitatively wrong or unstable reduced system, even when the high-fidelity system is stable. This is particularly important for fluid flow, where the conservation of the energy as a nonlinear invariant of the system is crucial for a correct numerical evaluation. However, we discuss that conservative properties of the skew-symmetric form for fluid flow can naturally be extended to the reduced system. Conventional RB techniques preserve the skew-symmetry of differential operator, resulting in the conservation of quadratic invariants at the level of the reduced system. Furthermore, the reduced system also contains quadratic invariants with respect to the reduced variables that approximate the invariants of the high-fidelity system. This results in the construction of a physically meaningful reduced system rather than a mere coupled system of differential equations.

Chapter 3 provides an overview of model reduction methods targeting Hamiltonian problems, which lays the basis for the notation and concepts of the approaches proposed in Chapters 4 and 5. The symplectic Galerkin projection has been discussed as a tool to generate a reduced Hamiltonian approximation of the original dynamics. PSD algorithms used to compute low-order projection on symplectic spaces have been introduced and compared. Such strategies have been classified in ortho-symplectic and symplectic procedures, depending on the structure of the computed RB. A greedy alternative for the generation of ortho-symplectic basis, characterized by an exponentially fast convergence, has been illustrated as an efficient iterative approach to overcome the computational cost associated with SVD-based techniques that require a fine sampling of the solution manifold of the high-dimensional problem. For problems where the Hamiltonian dynamics is coupled with a dissipative term, structure-preserving reduced models can be constructed with the symplectic reduction process by resorting to an extended non-dissipative Hamiltonian reformulation. Finally, we have described RB strategies to reduce problems having a non-canonical Hamiltonian structure that either enforce properties typical of a symplectic basis or use canonical symplectic reductions as an intermediate step to preserve the structure of the original model.

In Chapter 4, we introduce a model order reduction approach for parametrized non-dissipative problems in their canonical symplectic Hamiltonian formulation. By leveraging the local low-rank structure, we propose a nonlinear structure-preserving reduced basis method approximating the

problem solution with a modal decomposition where both the expansion coefficients and the reduced basis are evolving in time. Moreover, the dimension of the reduced basis is updated in time according to an adaptive strategy based on an error indicator. The resulting reduced models allow to achieve stable and accurate results with small reduced basis even for problems characterized by a slowly decaying Kolmogorov n -width. The strength is the combination of the dynamical adaptivity of the reduced basis and the preservation of the geometric structure underlying key physical properties of the dynamics, illustrated by examples.

In Chapter 5, the nonlinear structure-preserving introduced in Chapter 4 is further adapted for the model order reduction of parametric particle-based kinetic plasma problems. High resolution simulations of such problems may require a high number of particles and thus can become computationally intractable in multi-query scenarios. The proposed method combines dynamical low-rank approximation where the reduced space changes in time with adaptive hyper-reduction techniques to efficiently deal with the nonlinear operators. The resulting DMD-DEIM reduced dynamical system retains the Hamiltonian structure of the full model, provides good approximations of the solution using only few particles, and can be solved at a reduced computational cost. Several benchmark plasma models have been used to numerically assess the performances of the proposed method.

The study of efficient and structure-preserving algorithms for general nonlinear Hamiltonian vector fields and the development of partitioned Runge–Kutta methods that ensure the exact preservation of (at least linear and quadratic) invariants are still open problems and provide interesting directions of investigation. Possible directions of future investigation include the derivation of partitioned Runge–Kutta schemes that can ensure the preservation of the Hamiltonian (at least when this is a lower degree polynomial), the study of parameter sampling and reduction techniques to speedup the computation of the reduced basis, and the development of efficient error estimators to allow to dynamically adapt the rank of the reduced-order solution.

Closure Modeling Framework for **Part II** Reduced Order Models

The second part of the thesis explores possible advancements of the state of the art in the fields of modeling and approximation of ROM closures. Novel data-driven approaches based on the Mori-Zwanzig formalism have been developed and tested on several numerical experiments.

One of the main obstacles to using ROMs in complex applications, apart from the stability problem addressed in Part I, is the inaccuracy of the critical under-resolved regime. This regime is encountered when the ROM size is not large enough to capture the dynamics of the FOM target, resulting in a significant degradation in the accuracy of the approximation. In Part II, we focus on ROM closures, consisting of additional terms introduced in the reduced dynamics to model the effect of discarded ROM modes in the under-resolved regime, emphasizing data-driven closure approaches. Interdependent physical and computational challenges related to the formulation of ROM closures will also be addressed. In Chapter 7, we discuss the optimal prediction framework introduced by Chorin as a reinterpretation of the Mori-Zwanzig (MZ) formalism for statistical mechanics. Starting from the POD modes truncation discussed in Section 1.2, projection operators onto resolved and unresolved subspaces are used to define a low-dimensional non-Markovian system that takes the name of generalized Langevin equation. In this system, the effect of the discarded modes is exactly represented as a convolutional integral known as memory term. Despite being exact, this additional term is computationally intractable for nonlinear problems, and this rigorous framework is therefore used as a starting point for developing approximate closure models. The application of this framework as a practical tool is then tested on several numerical problems.

In Chapter 8, the shortcomings of traditional methods are addressed by means of machine learning techniques. Towards efficient model reduction of general problems, this thesis presents a recurrent neural network (RNN) closure of the parametric POD-Galerkin ROM. Based on the short time history of the reduced-order solutions, the RNN predicts the memory integral, representing the unresolved scales' impact on the resolved scales. A conditioned long short-term memory (LSTM) network is utilized as the regression model of the memory integral, in which the POD coefficients at a number of time steps are fed into the LSTM units, and the physical/geometrical parameters are fed into the initial hidden state of the LSTM. The reduced-order model is integrated in time using an implicit-explicit (IMEX) Runge-Kutta scheme, in which the memory term is integrated explicitly, and the remaining right-hand-side term is integrated implicitly to improve the computational efficiency. Numerical results demonstrate that the RNN closure can significantly improve the accuracy and efficiency of the POD-Galerkin reduced-order model of nonlinear problems. The POD-Galerkin ROM with the RNN closure is also shown to provide accurate predictions, well beyond the time interval of the training data, for several test cases.¹

A summary of the results and possible insights for future research for each of the chapters are provided in Chapter 9.

¹In accordance with the Elsevier publishing agreement, parts of this chapter are adapted from [260].

7 Mori-Zwanzig closure models

In Section 1.3.1, we have seen that the reduced basis is computed, following the POD method, by the singular value decomposition of the snapshots matrix and that the decay of the singular values is a good indicator of the quality of the approximation as a function of the size of the approximating space. The left singular vectors corresponding to the first n singular values can be interpreted as the spatial structures that best represent the FOM and are taken to assemble the reduced basis. In the Chapter 4, we pointed out that the quality of the approximation does not depend only on the size n of the basis, but that this depends on the type of problem and that for some classes of problems, using a small n leads to inaccurate approximations. We refer to this setting, where the number n of POD modes is insufficient to capture the problem's dynamics, as under-resolved simulation. In the same Chapter, we have seen that using reduced models that evolve in time represents a possible solution to overcome this problem, limiting the issue of reducibility of FOM for the entire interval \mathcal{T} .

If the under-resolution is not severe, then the projection error introduced with Galerkin's ansatz is limited, and the most important features of the FOM solution can potentially be captured. Despite this, suppressing the interaction between the resolved and unresolved part of the problem affects the quality of the projection of the solution trajectory, thus worsening the accuracy of the reduced model for large T . In [111], it is rigorously proven, with arguments similar to those used in finite element theory, that instabilities and inaccuracies in the approximation are attributable to Galerkin projections, justifying the use of the Petrov-Galerkin projection. In the context of Galerkin projections, several approaches, under the name of closure modeling, have been introduced to approximate the effect of discarded modes on the dynamics of resolved modes. Traditionally, closure models have been developed in fluid dynamics [5] and are divided into phenomenological and mathematical models. Although there is overlap between these two groups, in the former, knowledge about the physics of the problem is leveraged to make assumptions about the closure term, while filtering and mean-field modeling techniques are used for the latter. Relevant examples of closure models based on physical principles, particularly eddy viscosity and Kolmogorov turbulent scales, are mixing-length [15; 201], Smagorinsky [211], and dynamic SGS [97] ROM model closures. The idea behind these methods, supported theoretically and numerically even in the case of POD-based reduction, is that the discarded modes correspond to dissipative scales, and their truncation leads to energy accumulation in the resolved scales and thus instability. Regarding filtering, we recall the variational multi-scale approach based on mode ordering introduced in [238; 214]. Based on the similarities between LES and ROM, other filtering techniques are described in [267; 145].

This thesis focuses on closure models for type (1.4.2) ROMs based on Mori-Zwanzig formalism and the memory effect. We cite, in this regard, the work of Stinis [242], Parish, and Duraisamy [191] for reduced model closure without scale separation in the LES setting and the relationship between MZ-based closure models and VMS in [193].

This Chapter of the thesis is broken down as follows. In Section 7.1, we analyze how energy transfers between different POD scales, showing that the local transfer principle holds. Although this result is not used in the remainder of the thesis, it is important to show it to the reader as an example of a physical principle used in model closure since it represents the foundational idea of the variational multi-scale method. In Section 7.2, we introduce the reader to the Mori-Zwanzig formalism, starting with the linear FOM case and then considering the nonlinear case. Section 7.3 addresses the problem of how to efficiently approximate the closure term, starting with a study of its kernel. We conclude the Chapter with several numerical experiments showing that the proposed closure models succeed in improving the accuracy of the reduced model solution when compared to simple truncation of unresolved modes.

7.1 Example of interaction between different scales: energy scale identification

The POD method, as a data compression tool, was introduced to identify and study coherent structures of turbulent flows [165]. Rather than relying on one of the several definitions of coherent structure [90], according to the optimization problem (1.3.4), POD is based on the principle of energy maximization. Hence, selecting the first n left singular vectors allows representing the most energetic scales of the simulated flow.

However, even though unresolved POD modes are not relevant in terms of data compression, they are vital for the correct representation of the dynamics of the resolved modes over long simulation intervals. The core of the studies regarding the relation between resolved and unresolved modes has roots in the dynamics of turbulence fluctuations in the Fourier domain, with the notions of forward and backward energy cascades and dissipation associated with high order modes. In the last five decades, following these first theoretical studies, several closure models have been proposed to address the problem, and they have been grouped under the still fertile field of turbulence modeling.

The same interest was not shown for POD analysis until recently. Couplet et al. [15] investigated the transfer of energy between modes in the POD setting numerically, concluding that it was the same characterizing the exchange between Fourier modes. In particular, one of the main findings is that the interaction is local between modes, i.e., POD modes interact with modes of similar scales.

Aubry et. al [15] proposed a modification to the standard POD-Galerkin approximation (1.4.2) for fluid flows by adding an eddy viscosity-based (EV) additional term to the dynamics of the reduced model to approximate the impact of the unresolved part of the simulation. Similar and more sophisticated strategies have been proposed in [202; 201]. Most recent efforts regard the reuse of traditional (and successful) LES closure models to the POD setting. The main barrier to the application of these techniques is the treatment of the nonlinear term. Several solutions have been proposed, either by incorporating the Empirical Interpolation Method (EIM) or the two-level discretization method [261], used to adapt the Smagorinsky closure models to POD. A complete discussion of state-of-the-art for POD closure models for turbulent flows is presented in [262].

7.1 Example of interaction between different scales: energy scale identification

Despite this progress, numerical approximations of the interactions between resolved and unresolved POD modes are limited to the scenario of fluid flows. Moreover, most of them are strongly based on the unidirectionality and locality of the energy transfer, thus preventing its use for problems characterized by energy back-scattering.

Accounting for the effects of the unresolved modes is essential to determine the component of the error parallel to the approximating manifold. Indeed, while the orthogonal component depends on the chosen reduced space, the parallel component is due to unmodeled interactions with unresolved modes. In this Section, we consider a different approach, not based on the extension of techniques developed for spectral methods for fluid flows, to obtain a closed model taking into account the interactions with the unresolved part of the simulation, and hence improving the accuracy of the traditional POD-Galerkin approximation. Before proceeding with the description of the formalism used to compute the closure term, we would like to point out that for certain problems, particularly fluid flows problems, it is possible to have a physical intuition of the dependencies of the closure term. Consider the incompressible Navier-Stokes equations in 2D

$$\begin{cases} \frac{\partial}{\partial t} u + (u \cdot \nabla) u + \sigma \Delta u + \nabla p = 0, \\ \Delta p = -\nabla \cdot ((u \cdot \nabla) u), \end{cases} \quad (7.1.1)$$

with periodic boundary conditions on the domain Ω . The pressure variable is computed as the solution of the Poisson pressure equation [68], obtained by applying the divergence to the momentum equation and using the incompressibility condition. For a POD-Galerkin system, the weak formulation of (7.1.1) using the RB basis $\{u_i\}_{i=1}^n$ as test functions for the momentum equation is given by

$$\left(\frac{\partial}{\partial t} u + (u \cdot \nabla) u, u_i \right) + \sigma \left[\sum_{d=1}^D (\nabla u^d, \nabla u_i^d) \right] + (\nabla p, u_i) = 0, \quad \forall i = 1, \dots, m, \quad (7.1.2)$$

where (\cdot, \cdot) represents the standard inner product on $L_2(\Omega)$.

Consider the RB ansatz (1.4.1) in the continuous form

$$u(x, t) \approx \sum_{k=1}^m z_k(t) u_k(x), \quad (7.1.3)$$

and inserting (7.1.3) in (7.1.2), we have

$$\begin{aligned} \frac{d}{dt} z_i &= - \left(\left(\left(\sum_{k_1=1}^m z_{k_1} u_{k_1} \right) \cdot \nabla \right) \left(\sum_{k_2=1}^m z_{k_2} u_{k_2} \right), u_i \right) + \sigma \left[\sum_{d=1}^D \left(\nabla \left(\sum_{k=1}^m z_k u_k \right)^d, \nabla u_i^d \right) \right] - (\nabla p, u_i) \\ &= \sum_{k_1=1}^m \sum_{k_2=1}^m \underbrace{-((u_{k_1} \cdot \nabla) u_{k_2}, u_i)}_{C_i^{k_1, k_2}} z_{k_1} z_{k_2} + \sum_{k=1}^m \sigma \underbrace{\left[\sum_{d=1}^D (\nabla u_k^d, \nabla u_i^d) \right]}_{D_i^k} z_k \\ &\quad + \sum_{k_1=1}^m \sum_{k_2=1}^m \underbrace{(\nabla \Delta^{-1} (u_{k_1} \cdot \nabla) u_{k_2}, u_i)}_{P_i^{k_1, k_2}} z_{k_1} z_{k_2} \\ &= \sum_{k_1=1}^m \sum_{k_2=1}^m (C_i^{k_1, k_2} + P_i^{k_1, k_2}) z_{k_1} z_{k_2} + \sum_{k=1}^m D_i^k z_k, \end{aligned}$$

where we used the Poisson pressure equations to formally write the pressure in terms of the velocity in the second step. Assuming unitary mass, the total kinetic energy is

$$K(t) = \frac{1}{2} \|u\|^2 = \frac{1}{2} (u, u) = \frac{1}{2} \left(\sum_{i=1}^m z_i u_i, \sum_{j=1}^m z_j u_j \right) = \frac{1}{2} \sum_{j=1}^m \sum_{i=1}^m z_i z_j \delta_{ij} = \sum_{i=1}^m \frac{1}{2} z_i^2 = \sum_{i=1}^m K_i,$$

with K_i as the contribution of the i -th mode to the total energy balance. Each contribution evolves according to

$$\frac{d}{dt} K_i = z_i \frac{d}{dt} z_i = \sum_{k_1}^m \sum_{k_2}^m \left(C_i^{k_1, k_2} + P_i^{k_1, k_2} \right) z_{k_1} z_{k_2} z_i + \sum_{k=1}^m D_i^k z_k z_i. \quad (7.1.4)$$

In (7.1.4), as reported in [70], we can distinguish two different terms: a diadic interaction resulting from the dissipative term and a triadic interaction due to the quadratic term and the pressure effect on the velocity. A similar scenario is obtained if the POD modes are replaced by Fourier modes where, however, the diadic term simplifies into a linear term and only a few triads of the second term are different from zero. If we consider the time average of expression (7.1.4), represented formally by the operator $\langle \cdot \rangle$, we have

$$\begin{aligned} \left\langle \frac{d}{dt} K_i \right\rangle &= \sum_{k_1}^m \sum_{k_2}^m \left(C_i^{k_1, k_2} + P_i^{k_1, k_2} \right) \langle z_{k_1} z_{k_2} z_i \rangle + D_i^i \\ &= \sum_{k_1}^m \sum_{k_2}^{k_1} \frac{1}{1 + \delta_{k_1, k_2}} \left(C_i^{k_1, k_2} + C_i^{k_2, k_1} + P_i^{k_1, k_2} + P_i^{k_2, k_1} \right) \langle z_{k_1} z_{k_2} z_i \rangle + D_i^i \\ &= \sum_{k_1}^m \sum_{k_2}^{k_1} B_i^{k_1, k_2} \langle z_{k_1} z_{k_2} z_i \rangle + D_i^i, \end{aligned}$$

where we have used the orthogonality property [17] of the POD modes

$$\langle z_i z_j \rangle = \sigma_i \delta_{ij},$$

and rearranged the summation related to the triadic term. It is straightforward to observe that the interactions between different modes, if considered in terms of energy transfers, depends uniquely on the triadic term. More precisely, the term $B_i^{k_1, k_2} \langle z_{k_1} z_{k_2} z_i \rangle$ represents the influence of the POD mode of index k_1 on the variation of the kinetic energy associated to the i -th POD mode. Thus, the quantity

$$\Pi(i|j) = \sum_k^j B_i^{j, k} \langle z_k z_j z_i \rangle$$

could give a deeper insight into the interactions between different modes. In Figure 7.1(a), the behavior of $\Pi(i|j)$ is represented as a global map for the first 30 POD modes and three selected values of i . The diagonal structure in Figure 7.1(b) suggests that, as it happens for Fourier modes for homogeneous turbulence, the interactions between POD modes are mainly local in the spectrum. While for Fourier modes, it can be proved that the mode related to the i -th wavenumber transfer most of its energy in the wavenumbers window $[k/2, 2k]$ (see Kraichnan's works [151; 149; 150] on turbulence), no theoretical results are available in the POD scenario. The conclusion that we draw from this qualitative study of the interaction between resolved and unresolved POD modes is that, for certain physical problems, modes interact mainly with

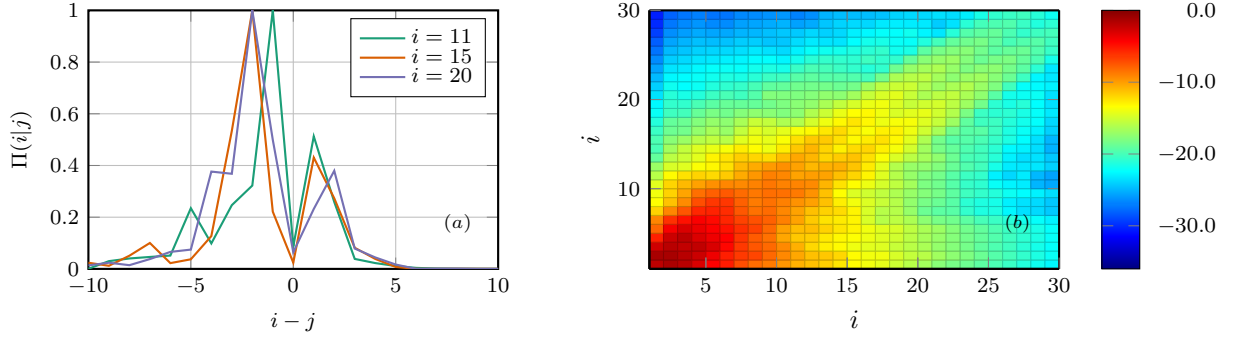


Figure 7.1: Normalized $\Pi(i|j)$ as function of the distance $i-j$ from three different modes i (a). Surface representing the logarithm of $\Pi(i|j)$, for $(i, j) \in [1, 30] \times [1, 30]$ (b).

modes with similar energy content. More generally, we assume that if a proper set of modes has been adopted to discretize the system, this interaction can be modeled with a certain level of approximation with a simple term depending only on the resolved part of the computation. The choice of the best basis to describe the solution of a given problem is still an open question, but for parameter-dependent systems of PDEs, POD approximation represents a legitimate candidate.

7.2 Mori-Zwanzig formalism

7.2.1 Introductory example

We start by considering an illustrative example as an introduction to the Mori Zwanzig formalism [275]. Consider the linear system of equations

$$\begin{cases} \frac{d}{dt}u^1 = A_{11}u^1 + A_{12}u^2, \\ \frac{d}{dt}u^2 = A_{21}u^1 + A_{22}u^2, \end{cases} \quad (7.2.1)$$

with $A_{11} \in \mathbb{R}^{n \times n}$, $A_{12} \in \mathbb{R}^{n \times (N-n)}$, $A_{21} \in \mathbb{R}^{(N-n) \times n}$, and $A_{22} \in \mathbb{R}^{(N-n) \times (N-n)}$. Suppose we are interested in the dynamics described by the variable $u^1 \in \mathbb{R}^n$ in time which could represent the first n POD or Fourier modes, while u^2 contains the remaining $N-n$ discarded modes. Solving the entire system (7.2.1) can be computationally expensive, and n is assumed to be much smaller than N . Hence subselecting entries from the full solution of (7.2.1) is not worth the cost of the procedure.

Our goal is to write a reduced system that describes the evolution of u^1 such that its dynamics depend only on u^1 , that is

$$\frac{d}{dt}u^1 = A_{11}u^1 + w(u^1, t). \quad (7.2.2)$$

To compute w , assume that u^1 is known and integrate the second equation in (7.2.1) to obtain

$$u^2(t) = e^{A_{22}t}u^2(0) + \int_0^t e^{A_{22}(t-s)}A_{21}u^1(s)ds,$$

and then, if we insert (7.2.1) into the first equation, we recover

$$\frac{d}{dt}u^1 = A_{11}u^1 + A_{12} \int_0^t e^{A_{22}(t-s)} A_{21}u^1(s) ds + A_{21}e^{A_{22}t}u^2(0). \quad (7.2.3)$$

Equation (7.2.3) is a generalized Langevin equation (GLE). The dynamics of the resolved variable u^1 is governed by a Markovian contribution, an integral memory term depending only on the resolved u^1 , and a term describing the influence of the unresolved initial condition $u^2(0)$. No approximations have been introduced, and (7.2.3) is exact.

In the linear case, we have shown that it is possible to provide a closed expression for the evolution of the resolved scale. The same approach cannot be generalized to the nonlinear scenario. However, a similar expression is obtained by means of Chorin's formulation of Mori-Zwanzig (MZ) formalism [63], frequently referred to as optimal prediction framework. More in detail, following the introduction of projection operators, the Mori-Zwanzig formalism allows to introduce a GLE describing the evolution of the resolved part. The projection operators separate the phase space of an ordinary differential equation (1.4.2) into resolved and unresolved subspaces. Similarly to the linear case, the Mori-Zwanzig theory reduces a large set of Markovian equations to a lower-dimensional and non-Markovian set of equations for a subset of variables of the original problem. The effect of the unresolved scales on the resolved scales is non-local in time and has the form of a convolution integral, known as the memory term. As expected, this clear separation of contributions to the resolved dynamic comes at a price.

The memory term depends on the so-called projected dynamics, and, even in the linear case, it requires the computation of the exponential of a matrix of dimension $N - n$. Hence, it is not practical to use the resulting GLE equation as direct computational tool given its integrodifferential nature, which requires a number of operations comparable to the one required for the solution of the full model (7.2.1). However, it represents a convenient *exact* starting point for developing closure models. In this respect, although the choice of resolved and unresolved sets is not relevant in the formulation of the GLE, it will be important for its approximation.

Globally hierarchical methods such as Fourier and POD methods are carried out to extract hierarchical sets of structures. If the basis is chosen appropriately, the dominating contribution to the dynamics of the resolved part of the simulation depends mainly on the resolved part itself. In contrast, the additional terms in the dynamics, usually neglected as in (1.4.2), will be amenable of simplification in the MZ description.

After a summary of the Mori-Zwanzig formalism, we apply to the POD-Galerkin setting the method proposed in [108] to estimate the memory integrand *a priori* without computing the projected (or orthogonal) resolved dynamic directly. Even though it requires a considerable computational cost, which does not make it a viable choice for the numerical approximation of the memory term on the fly, it represents a valuable tool to verify the assumptions in the description of possible Markovian models of the integral memory, discussed in Section 7.3.2.

7.2.2 Mathematical foundations

Consider the generic initial value problem

$$\begin{cases} \frac{d}{dt}y = f(t, y(t)), & \text{for } t \in \mathcal{T}, \\ y(0) = y_0, \end{cases} \quad (7.2.4)$$

evolving on a smooth manifold \mathcal{M} . We assume that $f : \mathbb{R}^N \rightarrow \mathbb{R}^N$ is at least uniformly continuous and y_0 resides in an Hilbert space denoted by \mathcal{H} . To provide a context, consider the POD reduced system (1.4.2) in case $n = N$. The general use of the formalism we describe in this Section is to study vector-valued observables of the form $g : \mathcal{M} \rightarrow \mathbb{R}^l$.

The potentially complex dynamics described by the observable g can be formally represented using a semigroup $\mathcal{K}(t, s)$ of operators acting on the Banach space of observables. In particular, we have

$$g(y(t)) = [\mathcal{K}(t, s)g]y(s), \quad (7.2.5)$$

where

$$\mathcal{K}(t, s) = e^{(t-s)\mathcal{L}}, \quad \mathcal{L}g(y) = f(y) \cdot \nabla g(y). \quad (7.2.6)$$

In [63], it is shown that *any* nonlinear ordinary differential equation (7.2.4) is equivalent to the *linear* partial differential equation in the phase space

$$\begin{cases} \frac{\partial}{\partial t} v = \mathcal{L}v, \\ v(y_0, 0) = g(y_0), \end{cases} \quad (7.2.7)$$

known as the Liouville equation, with \mathcal{L} being the Liouville operator and is an exact representation of the original dynamics. The equivalence holds in the sense that the solution to (7.2.7) is given [65] by

$$v(y_0, t) = g(y(y_0, t)), \quad (7.2.8)$$

which means that $v(y_0, t)$ is the solution to the PDE defined by the operator $e^{t\mathcal{L}}$ with initial value $g(y(y_0, 0))$. In particular, if $g(y(y_0, t)) = y_k(t)$, the solution to (7.2.7) is the k -th component of the solution to (7.2.4).

The Liouville operator \mathcal{L} enjoys the following interesting property

$$e^{t\mathcal{L}}g(y(y_0, 0)) = g(e^{t\mathcal{L}}y(y_0, 0)), \quad (7.2.9)$$

which implies that, given the solution $y(t)$ to (7.2.4), also the solution to (7.2.7) is known for any observable function g . This property is used in [108] for the *a priori* approximation of the memory integral.

Consider the solution to (7.2.4), $y(t) = \{y_k(t)\}$, $k \in C$, and two sets of indices R and U , such that $C = R \cup U$, and our set of observables corresponds to the components of the solution y with index in R . Let us divide the vector of initial condition y_0 into a resolved \hat{y}_0 (indices in R) and unresolved \tilde{y}_0 part, such that $y_0 = (\hat{y}_0, \tilde{y}_0)$. In our context, these two sets represent resolved and unresolved POD modes, respectively. The most common use of the Mori-Zwanzig formalism is stochastic modeling of deterministic systems and uncertainty quantification: here, we discard the statistical aspect of the approach and focus on the deterministic part. In the deterministic setting of our approach, we consider the following projection operator for a function l of the resolved and unresolved parts

$$\mathcal{P}l(\hat{y}_0, \tilde{y}_0) = l(\hat{y}_0, 0),$$

and we define the natural complementary orthogonal projector as $\mathcal{Q} = I - \mathcal{P}$.

The first step to derive an exact equation for the dynamic of the observables is to apply Dyson's identity

$$e^{t\mathcal{L}} = e^{t\mathcal{Q}\mathcal{L}} + \int_0^t e^{s\mathcal{L}}\mathcal{P}\mathcal{L}e^{(t-s)\mathcal{Q}\mathcal{L}}ds \quad (7.2.10)$$

to the Koopman operator. In (7.2.10), the term $e^{t\mathcal{Q}\mathcal{L}}$ represents the evolution operator related to the dynamics constrained by the orthogonal projector \mathcal{Q} . Starting from (7.2.10), we obtain the operator equation

$$\frac{\partial}{\partial t} e^{t\mathcal{L}} = e^{t\mathcal{L}} \mathcal{P}\mathcal{L} + e^{t\mathcal{Q}\mathcal{L}} \mathcal{Q}\mathcal{L} + \int_0^t e^{s\mathcal{L}} \mathcal{P}\mathcal{L} e^{(t-s)\mathcal{Q}\mathcal{L}} \mathcal{Q}\mathcal{L} ds. \quad (7.2.11)$$

By applying (7.2.11) to the observable y_0 , we obtain the Mori-Zwanzig identity in phase space

$$\frac{\partial}{\partial t} e^{t\mathcal{L}} y_0 = e^{t\mathcal{L}} \mathcal{P}\mathcal{L} y_0 + e^{t\mathcal{Q}\mathcal{L}} \mathcal{Q}\mathcal{L} y_0 + \int_0^t e^{s\mathcal{L}} \mathcal{P}\mathcal{L} e^{(t-s)\mathcal{Q}\mathcal{L}} \mathcal{Q}\mathcal{L} y_0 ds. \quad (7.2.12)$$

Acting with \mathcal{P} , we obtain the evolution equation for the resolved dynamics

$$\frac{\partial}{\partial t} e^{t\mathcal{L}} \hat{y}_0 = \mathcal{P}f(t, e^{t\mathcal{L}} y_0) + \int_0^t \mathcal{P}K(t, s, e^{t\mathcal{L}} y_0) ds,$$

where

$$F(t, x) = e^{t\mathcal{Q}\mathcal{L}} \mathcal{Q}\mathcal{L}x, \quad K(t, x) = \mathcal{P}\mathcal{L}F(t, x).$$

If we take as projector the truncation $\mathcal{P}g(\hat{y}, \tilde{y}) = g(\hat{y}, 0)$, (7.2.12) can be rewritten, component by component, in terms of $y(t) = (\hat{y}(t), \tilde{y}(t))$ as

$$\frac{d}{dt} y_k(t) = f_k(t, \hat{y}(t)) + F_k(t, y_0) + \int_0^t K_k(t, s, \hat{y}(t)) ds, \quad (7.2.13)$$

with k as index mode. Equation (7.2.13) can be interpreted as a POD-Galerkin discretization of the original FOM problem accounting for the removal of the smaller scales.

It is worth discussing the role of each contribution in (7.2.13). The first term, depending only on the resolved dynamic, represents the resolved variables' self-interaction and is the Markovian contribution to the time derivative of the resolved coefficients. The last term, commonly known as the memory, depends on the resolved part of the simulation at all time s between 0 and t , with the integrand commonly known as the memory kernel. For the second term we have that $F_k(y_0, t)$ satisfies the linear PDE

$$\begin{cases} \frac{\partial}{\partial t} F_k(y_0, t) = \mathcal{Q}\mathcal{L}F_k(y_0, t), \\ F_k(y_0, 0) = \mathcal{Q}\mathcal{L}y_0 = \mathcal{L}y_0 - \mathcal{P}\mathcal{L}y_0 = f_k(y_0, 0) - f_k(\hat{y}_0, 0), \end{cases} \quad (7.2.14)$$

known as the orthogonal dynamics equation. Projecting (7.2.14) gives

$$\begin{cases} \mathcal{P} \frac{\partial}{\partial t} F_k(y_0, t) = \mathcal{P}\mathcal{Q}\mathcal{L}F_k(y_0, t) = 0, \\ \mathcal{P}F_k(y_0, 0) = \mathcal{P}f_k(y_0, 0) - \mathcal{P}f_k(\hat{y}_0, 0) = 0, \end{cases}$$

implying that F_k is orthogonal to the image of the projector \mathcal{P} .

Hence, if the initial condition y_0 belongs to the space of observables, this contribution, known as the noise contribution, is null. A solution of the system (7.2.14) has been proved to exist, for Hamiltonian systems, in a classic sense for finite rank projectors and in a weak sense for projectors in the form of conditional expectations [102].

7.3 Memory approximation

7.3.1 Study of the orthogonal dynamics

Section 7.3.2 is devoted to possible strategies to approximate the memory contribution in (7.2.12). These procedures rely on the introduction of some assumptions regarding the behaviour of the memory kernel to simplify the expression of the memory integrand. Unfortunately, the integrand requires the solution to the orthogonal dynamics equation, making an exact analysis unfeasible for problems with a large number of degrees of freedom. Before proceeding, we seek to clarify how we apply Mori-Zwanzig formalism in the case of POD approximation. Consider the reduced system (1.4.2), introduced in Chapter 1, where we assume that the size of the reduced space is n_1 and is large enough to represent the solution of FOM up to machine precision. To achieve a significant speed-up in the computations, the standard POD procedure truncates the basis to the first n_2 elements, with $n_2 < n_1$, neglecting the effect of the unresolved $n_1 - n_2$ remaining modes on the resolved part.

We introduce the following notation for the set of basis: $U_{n_1} \in \mathbb{R}^{N \times n_1}$ represents the set of n_1 basis vectors, ordered as columns of the matrix and $U_{n_2} \in \mathbb{R}^{N \times n_2}$ is the submatrix obtained by taking the first n_2 basis vectors. In the more general case of application of the Mori-Zwanzig formalism, it is not required that the resolved part of the simulation be represented by the most relevant modes in terms of data compression.

Considering the formalism introduced in Section 7.2.2, which defines an additional memory term to take into account the missing contribution to the evolution dynamics of the first n_2 modes by solving the reduced problem

$$\frac{d}{dt}z = U_{n_1}^\top f(t, U_{n_1}z; \eta) + w(z, t) = \tilde{f}(t, z; \eta) + w(z, t), \quad (7.3.1)$$

where the term w represents a Markovian approximation of the memory term. It can be shown [193] that MZ-based methods belong to the larger class of residual-based methods and w can be written as a function of the residual. The same does not hold for subgrid-scale models based on physical insight of the problem, such as the Smagorinsky closure.

In this work, we assume that the additional noise term due to the unresolved part of the initial condition can be neglected compared to the contribution given by the memory term. In the case of the POD basis, since the basis is computed by solving a constrained optimization problem, with no guarantees regarding the complete resolution of the initial condition, the additional noise term is not null. However, this problem can be easily solved by introducing a bias term in the RB ansatz.

Several attempts have been made to extract the memory kernel and the orthogonal dynamics. For example, Hermite polynomials and Volterra integral equations have been used in [63; 29] to approximate memory terms in the Fourier context for low-dimensional problems. Other methodologies have been considered in [157; 160]: however, they fail to address the case of high-dimensional problems. In this Section, we briefly describe a different procedure proposed by Gouasmi et al. [108] to estimate the memory term by approximating the orthogonal dynamics operator. The advantage of this operation is twofold: on the one hand, we get a picture of the interaction between resolved and unresolved part, and on the other hand, we estimate the relative importance of the unresolved initial condition on the resolved part of the evolution dynamics.

The idea is to grant to the orthogonal dynamic operator $e^{t\mathcal{Q}\mathcal{L}}$ the same composition property (7.2.9) that holds for the Liouville operator \mathcal{L} in case g is a smooth function. While the composition

property can be proved in case of $e^{t\mathcal{L}}$, the same cannot be shown for the orthogonal dynamic operator since it is not a Koopman operator. Results restricted to the linear case are shown in [108]. For the more general nonlinear case, the high-dimensional orthogonal dynamics equation

$$\frac{\partial}{\partial t} F_j(z_0, t) = \mathcal{Q}\mathcal{L}F_j(z_0, t)$$

has to be solved. A more complete presentation of the subject can be found in [195]. First, let us consider

$$\begin{aligned} \frac{\partial}{\partial t} e^{t\mathcal{Q}\mathcal{L}} z_0 &= \mathcal{Q}\mathcal{L}e^{t\mathcal{Q}\mathcal{L}} z_0 \\ &\stackrel{\text{comp}}{\approx} e^{t\mathcal{Q}\mathcal{L}} \mathcal{Q}\mathcal{L}z_0 \\ &= e^{t\mathcal{Q}\mathcal{L}} \mathcal{L}z_0 - e^{t\mathcal{Q}\mathcal{L}} \mathcal{P}\mathcal{L}z_0 \\ &= e^{t\mathcal{Q}\mathcal{L}} f(z_0) - e^{t\mathcal{Q}\mathcal{L}} \mathcal{P}f(z_0) \\ &\stackrel{\text{comp}}{\approx} f(e^{t\mathcal{Q}\mathcal{L}} z_0) - \mathcal{P}f(e^{t\mathcal{Q}\mathcal{L}} z_0), \end{aligned} \quad (7.3.2)$$

which defines a differential equation in $\phi_Q(t, z_0) = e^{t\mathcal{Q}\mathcal{L}} z_0$, with initial condition $\phi_Q(0, z_0) = z_0$, and does not require the explicit knowledge of the orthogonal evolution operator. In the MZ identity in the phase space (7.2.12), the memory term can be written as

$$\begin{aligned} \int_0^t e^{s\mathcal{L}} \mathcal{P}\mathcal{L}e^{(t-s)\mathcal{Q}\mathcal{L}} \mathcal{Q}\mathcal{L}z_0 ds &= \int_0^t e^{(t-s)\mathcal{L}} \mathcal{P}\mathcal{L}e^{s\mathcal{Q}\mathcal{L}} \mathcal{Q}\mathcal{L}z_0 ds \\ &\stackrel{\text{comp}}{=} \int_0^t \mathcal{P}\mathcal{L}e^{s\mathcal{Q}\mathcal{L}} \mathcal{Q}\mathcal{L} \underbrace{e^{(t-s)\mathcal{L}} y_0}_{z(t-s)} ds \\ &= \int_0^t \mathcal{P}\mathcal{L}e^{s\mathcal{Q}\mathcal{L}} \mathcal{Q}\mathcal{L}z(t-s) ds \\ &= \int_0^t \mathcal{P}\mathcal{L}F(s, z(t-s)) ds \\ &= \int_0^t \mathcal{P}\mathcal{L}F(t-s, z(s)) ds, \end{aligned} \quad (7.3.3)$$

where $F(t-s, z(s))$ represents the right hand side of (7.3.2) at time $t-s$, using as initial condition $z(s)$. The Liouville operator applied to $F(s, z(s))$ reads

$$\mathcal{L}F(s, z(s)) = \tilde{f}(z(s)) \cdot \nabla F(s, z(s)). \quad (7.3.4)$$

Gouasmi et al. [108] proposed the following first order approximation of $\mathcal{L}F(s, z(s))$, which does not require all gradient directions but introduces the unitary direction given by $\tilde{f}(z(s)) = f(z(s)) / \|f(z(s))\|$

$$\begin{aligned} \mathcal{L}F(s, z(s)) &= f(z(s)) \cdot \nabla F(s, z(s)) \\ &= \|f(z(s))\| \tilde{f}(z(s)) \cdot \nabla F(s, z(s)) \\ &= \|f(z(s))\| \nabla_{\tilde{f}(z(s))} F(s, z(s)) \\ &\approx \|f(z(s))\| \frac{F(s, z(s) + \epsilon \tilde{f}(z(s))) - F(s, z(s))}{\epsilon}. \end{aligned} \quad (7.3.5)$$

Hence, the memory kernel is approximated as

$$\begin{aligned}\mathcal{P}\mathcal{L}F(t-s, z(s)) &\approx \|f(\mathcal{P}z(s))\| \frac{F(t-s, \mathcal{P}z(s) + \epsilon \bar{f}(\mathcal{P}z(s))) - F(t-s, \mathcal{P}z(s))}{\epsilon}, \\ &= \|f(\mathcal{P}z(s))\| \frac{F(t-s, \mathcal{P}z(s) + \epsilon \bar{f}(\mathcal{P}z(s)))}{\epsilon}, \\ &= \widetilde{\mathcal{P}\mathcal{L}F}(t-s, z(s)), \quad s \in [0, t],\end{aligned}\tag{7.3.6}$$

where the first-order approximation is in the discretization of the gradient. As noted in [108], approximating the memory kernel is equivalent to computing the sensitivity with respect to the initial condition of the solution to the orthogonal dynamics in a direction determined solely by the resolved part of the solution. Once the integrand has been approximated, the memory is computed as

$$\int_0^t \tilde{K}(t, s, z(t)) ds = \int_0^t \widetilde{\mathcal{P}\mathcal{L}F}(t-s, z(s)) ds \approx \sum_{n=0}^{N_{\Delta t}} \widetilde{\mathcal{P}\mathcal{L}F}(t-s_n, z(s_n)), \tag{7.3.7}$$

with $N_{\Delta t}$ being the total number of time steps between 0 and t . The computational cost of this procedure prevents, as previously said, the use of this tool as a memory-approximating procedure for the solution of the reduced system. The computational cost C_{mem} is estimated to be

$$C_{mem} \propto \frac{N_{\Delta t} - 1}{2} C_{full}, \tag{7.3.8}$$

where C_{full} represents the cost of simulating the FOM. For a fixed integration interval $\mathcal{T} := (0, T]$, the estimate (7.3.8) has been computed taking into account that problem (7.3.2) has to be solved around $N_{\Delta t}$ times, each time on a shorter interval $(t_n, T]$ and with $z(t_n)$ as initial condition. The first point of this study concerns the validity in assuming the compatibility property for the operator $e^{t\mathcal{L}}$. Since the exact memory integrand is not available, we compare the approximated memory (7.3.7) with the exact memory term. If we rewrite the FOM in terms of semi-group operators, we recover

$$\begin{aligned}\frac{\partial}{\partial t} e^{t\mathcal{L}} z_0 &= \mathcal{L} e^{t\mathcal{L}} z_0 \\ &= e^{t\mathcal{L}} \mathcal{L} z_0 \\ &= e^{t\mathcal{L}} \mathcal{P}\mathcal{L} z_0 + \underbrace{(e^{t\mathcal{L}} \mathcal{L} z_0 - e^{t\mathcal{L}} \mathcal{P}\mathcal{L} z_0)}_{e^{t\mathcal{L}} \mathcal{Q}\mathcal{L} z_0},\end{aligned}\tag{7.3.9}$$

which, in terms of the subset of resolved variables, becomes

$$\frac{d}{dt} z_k(t) = f_k(t, \hat{z}; \eta) + (f_k(t, z; \eta) - f_k(t, \hat{z}, \eta)). \tag{7.3.10}$$

and hence

$$e^{t\mathcal{L}} \mathcal{Q}\mathcal{L} z_{0k} = f_k(t, z; \omega) - f_k(t, \hat{z}; \omega). \tag{7.3.11}$$

Comparing equations (7.2.12) and (7.3.10), we note that the term $e^{t\mathcal{L}} \mathcal{Q}\mathcal{L} z_0$, computable as the difference between the right-hand side evaluated using all the n_1 modes and the first n_2 modes, encapsulates the two contributions represented by the noise and the memory terms.

To check if our assumption that the noise term, due to unresolved initial condition, is negligible, we compare (7.3.11) to the approximated memory kernel in two settings. In the first case, we

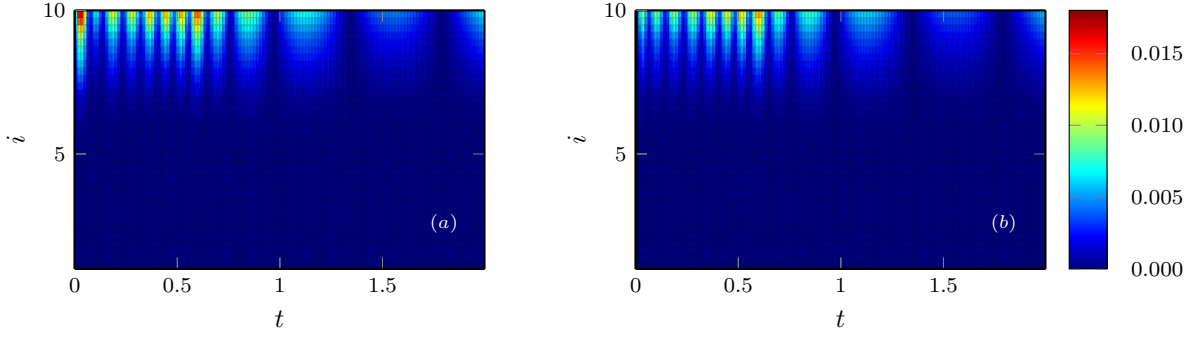


Figure 7.2: BG: Absolute value of the memory contribution for the first $n_2=10$ modes represented as a smooth surface for the exact (a) and approximated (b) memory term as a function of time in case of standard RB ansatz.

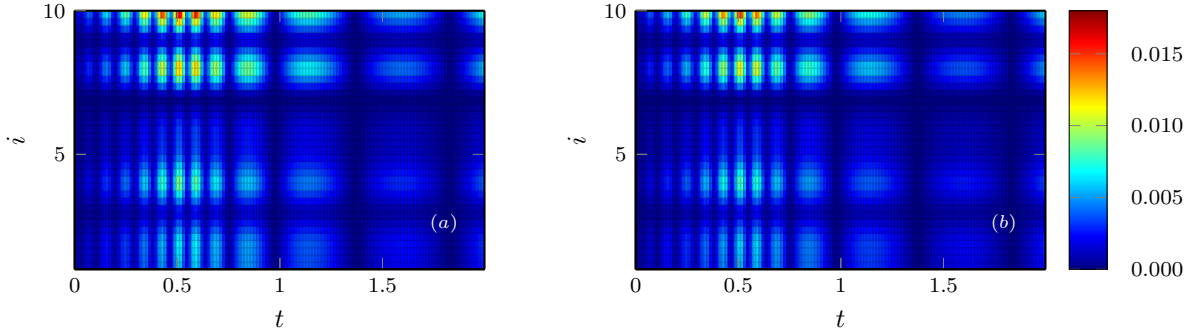


Figure 7.3: BG: Absolute value of the memory contribution for the first $n_2=10$ modes represented as a smooth surface for the exact (left) and approximated (right) memory term as a function of time in case of additional bias term in the RB approximation.

consider the RB ansatz given in (1.4.1), while in the second, we add a bias term b to the ansatz to exactly represent the initial condition, i.e., $u \approx Uz + b$.

As a test case, we consider the viscous Burgers' equation with periodic boundary conditions and a sinusoidal wave as initial condition, which leads to the formation of a standing shock. Time and the viscous coefficient μ are taken as parameters. The POD approximation, if not stated otherwise, is computed using $n_2 = 10$ POD modes and the system is considered completely resolved when $n_1 = 68$ POD modes are used. The memory kernel is approximated using (7.3.6) with $\epsilon = 10^{-8}$ and compared with (7.3.11), in Figure 7.2 for the standard basis RB ansatz and in Figure 7.3 with the additional bias, in terms of the absolute value of the memory contribution. There seems to be a good agreement for the tests considered between the memory term reconstructed from the approximation of the memory integrand and the memory term computed directly from the FOM solution. In the case of a standard RB ansatz, we notice a discrepancy between the two at the start of the simulation, probably due to the unresolved initial condition. This difference becomes neglectable after a few time steps. The same error has not been noticed when the set of basis functions is modified to represent exactly the initial condition. The results allow us to conclude that, despite the approximation of the orthogonal operator, the approximated memory kernel provides an accurate reconstruction of the memory term for the Burgers' problem in the context of POD-Galerkin.

The possibility to study directly the orthogonal dynamics, despite the high but affordable

computational cost required for its approximation, is relevant for a better understanding of the behavior of the memory term. In particular, we focus on the exponential decay of the memory kernel, as shown in the Figures 7.4 and 7.5. This concept is formally defined as *finite memory* and it has been investigated by several authors [64], suggesting that the memory integrand has a finite support,

$$\int_0^t e^{s\mathcal{L}}\mathcal{P}\mathcal{L}e^{(t-s)\mathcal{Q}\mathcal{L}}\mathcal{Q}\mathcal{L}y_0ds \approx \int_{t-\tau(t)}^t e^{s\mathcal{L}}\mathcal{P}\mathcal{L}e^{(t-s)\mathcal{Q}\mathcal{L}}\mathcal{Q}\mathcal{L}y_0ds,$$

with τ length of the support, also known as *memory length*.

We now consider the same Burgers' equation but with a different initial condition and study the behavior of the memory kernel as a function of the size of the resolved part. We consider, as initial condition, the combination of shifted sinusoidal functions with different frequencies and scaled them such that the corresponding individual energy content decays for increasing frequency. In Figure 7.6, the normalized memory kernels profiles, related to different sizes of the resolved part of the simulation are compared. We observe that the decay of the memory kernel sharpens by increasing the size of the approximating space.

The orthogonal dynamics provides insight into the study of the behavior of the memory kernel. Consider the memory integral

$$\int_0^t e^{s\mathcal{L}}\mathcal{P}\mathcal{L}e^{(t-s)\mathcal{Q}\mathcal{L}}\mathcal{Q}\mathcal{L}y_0ds = \int_0^t e^{s\mathcal{L}}\mathcal{P}\mathcal{L}S e^{(t-s)\Lambda}S^{-1}\mathcal{Q}\mathcal{L}y_0ds, \quad (7.3.12)$$

where we have introduced the eigendecomposition of the orthogonal dynamics operator. Expression (7.3.12) is exact for the case of linear FOM, while, for nonlinear problems, a linearization of the operator $e^{\mathcal{Q}\mathcal{L}}$ can be considered as a first approximation. In (7.3.12), Λ and S represent the eigenvalues and eigenvectors of $e^{\mathcal{Q}\mathcal{L}}$. Equation (7.3.12) suggests that the integrand has finite support, and its decay is dictated by the inverse of the operator's eigenvalues. In [108], the dominant decaying factor was approximated by the inverse of the spectral radius ρ of the Jacobian of the resolved operator $\mathcal{P}\mathcal{L}$, assuming the absence of scale separation between resolved and unresolved variables.

In Figure 7.7, it is confirmed that the inversely proportional relation between the memory length and the ρ holds, in the form

$$\tau(t) = C\frac{1}{\rho}, \quad (7.3.13)$$

with $\tau(t)$ being the memory length at time t . Similar dependencies [191] have been found for the same problem in the case of Fourier modes.

7.3.2 Memory Modelling

In the previous Section, we showed that MOR introduces a memory effect based on a projection of the unresolved residual. While in the linear case this is easily demonstrated, for the nonlinear case we used Chorin's formulation of the Mori-Zwanzig formalism. The key observation is that, even though the unresolved part of the simulation remains inaccessible, its contribution to the evolution of the resolved part can be estimated by modeling the memory term. In the context of spectral Fourier-Galerkin methods, MZ-based closure models have been used to improve the

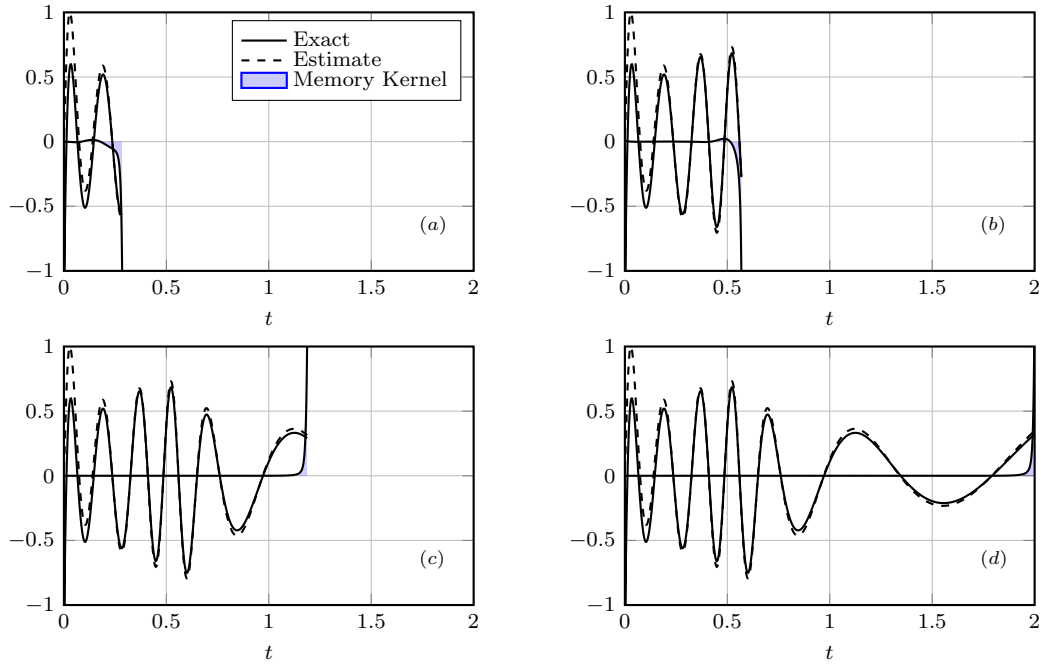


Figure 7.4: BG: Snapshots of the memory integrand and memory at different time frames in case of standard POD basis. The results showed are related to the third element of the basis.

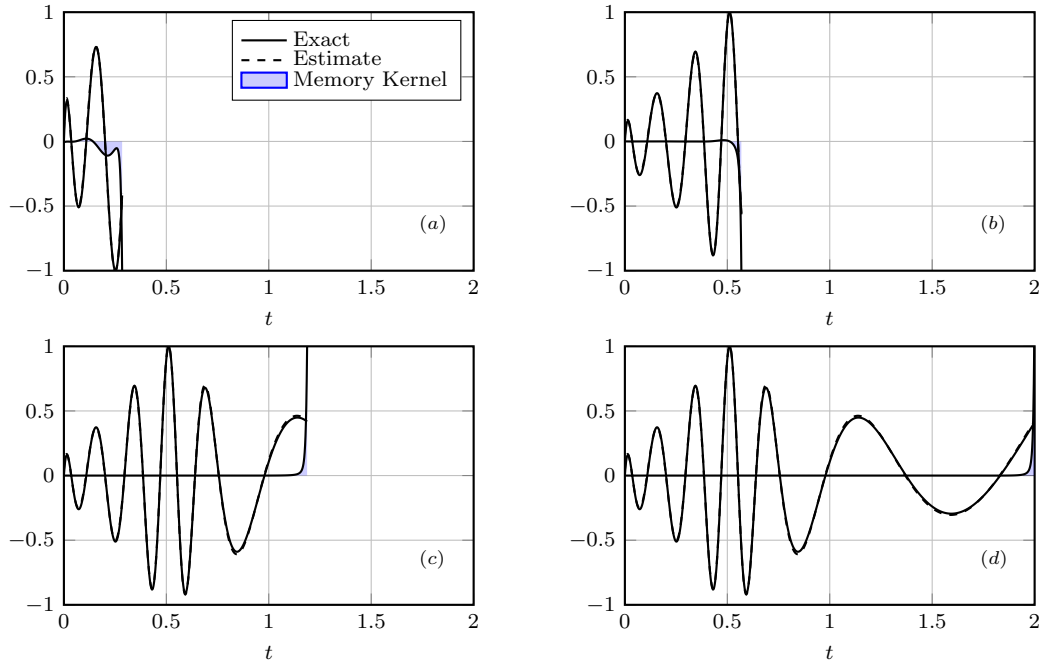


Figure 7.5: BG: Snapshots of the memory integrand and memory at different time frames in case of modified POD basis.

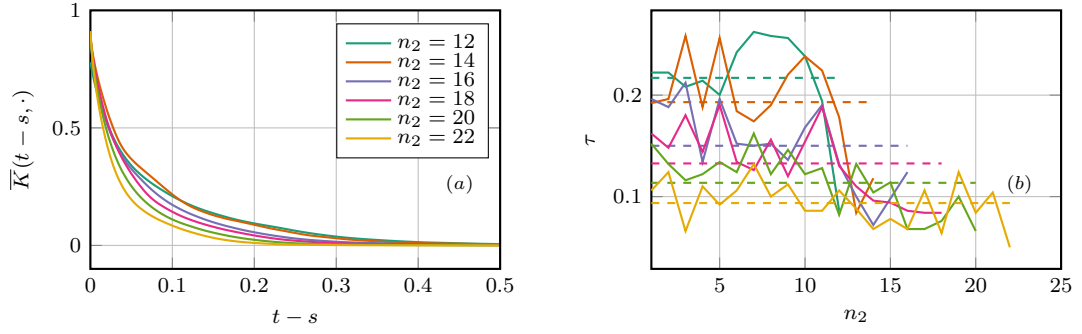


Figure 7.6: BG: Normalized average of the memory kernel over time t and over the entire set of resolved modes for different values of n_2 (a). In (b), the continuous lines represent the average over the resolved set of the memory length, while the dashed lines are the corresponding time averages.

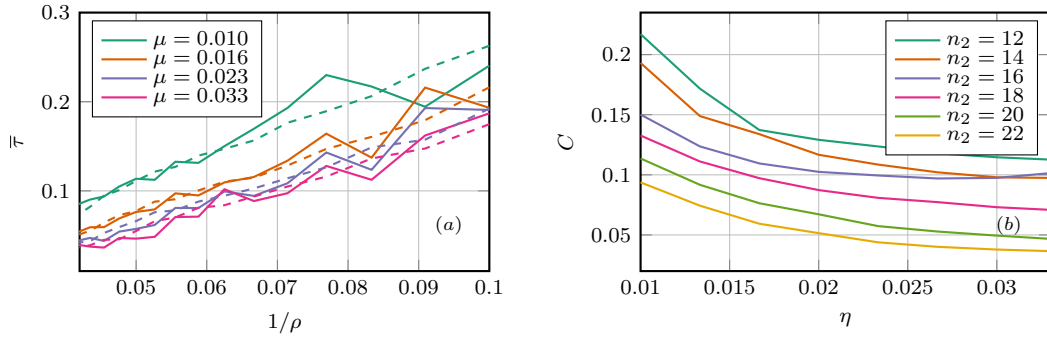


Figure 7.7: BG: Relation between the average memory length over time and the average reciprocal of the spectral radius of the Jacobian over time for the Burgers' equation in (a). The dashed lines represent the least squares approximation of the data. (b) Behavior of the coefficient C in (7.3.13) as a function of the parameter η for different sizes n_2 of the resolved set of coefficients.

accuracy of the solutions of the Navier-Stokes equation in the past [121; 241; 55], using the findings of Section 7.3.1 as starting development point. So far, we have concluded that the memory term takes the form of an integral depending only on the resolved part of the simulation. However, its exact computation is not a computationally viable option, and hence strategies to model the additional memory term are required.

Despite the limited understanding of the orthogonal dynamics, several surrogate models exist, with the t -model [64] being the first of this kind. More recent solutions employ high-order expansions of the memory kernel and have been tested on Burgers' and Euler's equations. In addition, estimates on the accuracy of a large class of closure MZ-based models have been provided by Venturi in [272; 274]. This Section aims to introduce the reader to these MZ-based methods.

t-model

The t -model represents a long memory model [65], which has been applied in fluid flows simulations. It can be derived via numerical integration of the memory term in (7.2.12) over a single interval. Here we present the derivation proposed in [62] that, even though not constructive, clarifies under which assumption it represents a good approximation of the memory contribution. As stressed in Section (7.3.1), the main obstacles in the computation of (7.2.12) is the orthogonal dynamics operator $e^{(t-s)\mathcal{Q}\mathcal{L}}$, which requires the solution of the orthogonal dynamics (7.2.14). Such complication can be avoided by considering

$$e^{t\mathcal{Q}\mathcal{L}} \approx e^{t\mathcal{L}}. \quad (7.3.14)$$

The rationale of this approach is based on the coupling between the resolved and unresolved dynamics. The complementary projector \mathcal{Q} eliminates any dependence on the resolved components and belongs to the unresolved space. If the resolved dynamic does not interfere with the unresolved variables or are weakly coupled, the application of $e^{t\mathcal{L}}$ or $e^{t\mathcal{Q}\mathcal{L}}$ to an element of the unresolved space provides the same results as unresolved variables interact mainly with each other. Starting from the projection of the Mori-Zwanzig model (7.2.12) we have

$$\begin{aligned} \frac{\partial}{\partial t} \mathcal{P}e^{t\mathcal{L}}y_0 &= \mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}y_0 + \int_0^t \mathcal{P}e^{s\mathcal{L}}\mathcal{P}\mathcal{L}e^{(t-s)\mathcal{Q}\mathcal{L}}\mathcal{Q}\mathcal{L}y_0 ds \\ &= \mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}y_0 + \int_0^t \mathcal{P}e^{s\mathcal{L}}\mathcal{L}e^{(t-s)\mathcal{Q}\mathcal{L}}\mathcal{Q}\mathcal{L}y_0 ds - \int_0^t \mathcal{P}e^{s\mathcal{L}}\mathcal{Q}\mathcal{L}e^{(t-s)\mathcal{Q}\mathcal{L}}\mathcal{Q}\mathcal{L}y_0 ds \\ &= \mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}y_0 + \int_0^t \mathcal{P}\mathcal{L}e^{s\mathcal{L}}e^{(t-s)\mathcal{Q}\mathcal{L}}\mathcal{Q}\mathcal{L}y_0 ds - \int_0^t \mathcal{P}e^{s\mathcal{L}}e^{(t-s)\mathcal{Q}\mathcal{L}}\mathcal{Q}\mathcal{L}\mathcal{Q}\mathcal{L}y_0 ds \\ &\approx \mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}y_0 + \int_0^t \mathcal{P}\mathcal{L}e^{t\mathcal{Q}\mathcal{L}}\mathcal{Q}\mathcal{L}y_0 ds - \int_0^t \mathcal{P}e^{t\mathcal{Q}\mathcal{L}}\mathcal{Q}\mathcal{L}\mathcal{Q}\mathcal{L}y_0 ds \\ &= \mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}y_0 + t\mathcal{P}(\mathcal{L}e^{t\mathcal{Q}\mathcal{L}} - e^{t\mathcal{Q}\mathcal{L}}\mathcal{Q}\mathcal{L})\mathcal{Q}\mathcal{L}y_0 \\ &= \mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}y_0 + t\mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}\mathcal{Q}\mathcal{L}y_0, \end{aligned} \quad (7.3.15)$$

where the definition of the evolution operator and (7.3.14) have been used to obtain a Markovian approximation of the memory contribution. In the long, we do not expect approximation (7.3.15) to hold as $e^{t\mathcal{L}}$ may eventually take the initial unresolved $\mathcal{Q}\mathcal{L}y_{0k}$ into the resolved space, impacting the representation of the memory kernel.

Therefore, Stinis [242] proposed a model correction based on a scaled expansion in N_s terms to

improve the accuracy of the t -model, which takes the form

$$\int_0^t \mathcal{P}e^{s\mathcal{L}} \mathcal{P} \mathcal{L} e^{(t-s)\mathcal{Q}\mathcal{L}} \mathcal{Q} \mathcal{L} y_0 ds \approx \sum_{i=1}^{N_s} C_i (-1)^{i+1} \frac{t^i}{i!} \mathcal{P} e^{t\mathcal{L}} (\mathcal{P} \mathcal{L})^i \mathcal{Q} \mathcal{L} y_0$$

where the coefficients C_i are computed *on-the-fly* by enforcing that rates of change of the norms of the solution should be the same when computed from the FOM and the reduced problem.

A mathematically rigorous upper bound, computable in certain settings, has been developed by Venturi et al. [273]

Theorem 7.3.1 ([273]). *Let $e^{t\mathcal{L}}$ and $e^{t\mathcal{Q}\mathcal{L}}$ be strongly continuous semigroups with upper bounds $\|e^{t\mathcal{L}}\| \leq M e^{t\omega}$ and $\|e^{t\mathcal{Q}\mathcal{L}}\| \leq M_{\mathcal{Q}} e^{t\omega_{\mathcal{Q}}}$. Then*

$$\left\| \int_0^t \mathcal{P}e^{s\mathcal{L}} \mathcal{P} \mathcal{L} e^{(t-s)\mathcal{Q}\mathcal{L}} \mathcal{Q} \mathcal{L} y_0 ds - t \mathcal{P} e^{t\mathcal{L}} \mathcal{P} \mathcal{L} \mathcal{Q} \mathcal{L} y_0 \right\| \leq M_1(t) \quad (7.3.16)$$

where

$$M_1(t) = \begin{cases} C_1 \left(\frac{e^{t\omega_{\mathcal{Q}}} - e^{t\omega}}{\omega_{\mathcal{Q}} - \omega} + \frac{t e^{t\omega}}{M_{\mathcal{Q}}} \right), & \omega \neq \omega_{\mathcal{Q}} \\ C_1 \frac{M_{\mathcal{Q}} + 1}{M_{\mathcal{Q}}} t e^{t\omega}, & \omega = \omega_{\mathcal{Q}} \end{cases}$$

and $C_1 = M M_{\mathcal{Q}} \|\mathcal{P}\|^2 \|\mathcal{L} \mathcal{Q} \mathcal{L} y_0\|$.

τ -model

The t -model overestimates the memory contribution because the short memory property of the system is not taken into account in the derivation of the model itself. The first step to improve the model consists in rewriting the memory term as

$$\int_0^t \mathcal{P}e^{s\mathcal{L}} \mathcal{P} \mathcal{L} e^{(t-s)\mathcal{Q}\mathcal{L}} \mathcal{Q} \mathcal{L} y_0 ds \approx \int_{t-\tau(t)}^t \mathcal{P}e^{s\mathcal{L}} \mathcal{P} \mathcal{L} e^{(t-s)\mathcal{Q}\mathcal{L}} \mathcal{Q} \mathcal{L} y_0 ds, \quad (7.3.17)$$

where, depending on the model, $\tau(t)$ represents the finite support of the memory kernel, also known as memory length. Using a trapezoidal quadrature rule and the assumption that $K(t, \tau, \hat{y}(t)) \approx 0$ because of finite memory, we have

$$\int_{t-\tau(t)}^t \mathcal{P}e^{s\mathcal{L}} \mathcal{P} \mathcal{L} e^{(t-s)\mathcal{Q}\mathcal{L}} \mathcal{Q} \mathcal{L} y_0 ds \approx \frac{1}{2} \tau(t) \mathcal{P} e^{t\mathcal{L}} \mathcal{P} \mathcal{L} \mathcal{Q} \mathcal{L} y_0, \quad (7.3.18)$$

known as the τ -model approximation of the memory contribution. In relation to what we observed in Section 7.3.1, this model can also be obtained by forcing the memory kernel to have an exponential behavior. Take $A(t) \in \mathbb{R}^{n_1 \times n_1}$, negative definite and time-dependent, and consider the following approximation

$$\begin{aligned} \int_0^t \mathcal{P}e^{s\mathcal{L}} \mathcal{P} \mathcal{L} e^{(t-s)\mathcal{Q}\mathcal{L}} \mathcal{Q} \mathcal{L} y_0 ds &\approx \int_0^t e^{A(t)s} e^{t\mathcal{L}} \mathcal{P} \mathcal{L} \mathcal{Q} \mathcal{L} y_0 ds \\ &= A(t)^{-1} \left(e^{tA(t)} - \mathbb{I}_{n_1} \right) \mathcal{P} e^{t\mathcal{L}} \mathcal{P} \mathcal{L} \mathcal{Q} \mathcal{L} y_0. \end{aligned} \quad (7.3.19)$$

The complexity of $A(t)$ reflects the complexity of the approximation. A dense matrix could be used to represent strong interactions between different modes, while a block diagonal matrix can be helpful to describe localized effects of the basis. Let us consider the simplest model given by $A(t) = -c(t)\mathbb{I}_{n_1}$, with $c(t) > 0 \forall t \in \mathcal{T}$. From a modeling point of view, this implies that memory kernels, related to different modes, decay independently from each other and at the same rate. Even though this might be an oversimplification of the physics behind the problem, this hypothesis is often satisfied. Consider, for instance, the Burgers' equation analyzed in Section 7.3.1: even though the intensity of the memory contribution depends on the mode considered, as shown in Figure 7.3, the average memory length is not strongly influenced by it, as shown in Figure 7.6.

If we insert the diagonal assumption into (7.3.19), we have

$$\begin{aligned} \int_0^t \mathcal{P}e^{s\mathcal{L}}\mathcal{P}\mathcal{L}e^{(t-s)\mathcal{Q}\mathcal{L}}\mathcal{Q}\mathcal{L}y_0 ds &\approx A(t)^{-1} \left(e^{tA(t)} - I \right) \mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}\mathcal{Q}\mathcal{L}y_0 \\ &= \frac{1}{c(t)} \left(1 - e^{-tc(t)} \right) \mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}\mathcal{Q}\mathcal{L}y_0. \end{aligned} \quad (7.3.20)$$

Equation (7.3.20) is the τ -model, in which the renormalization coefficient and the decay parameter are related by

$$\tau(t) = \frac{2}{c(t)} \left(1 - e^{-tc(t)} \right). \quad (7.3.21)$$

According to our knowledge, we stress that there are no works using models for $A(t)$ more complex than the one just described. Therefore, a good starting point to derive more accurate approximations, tailored to the specific problem considered, could be an energy analysis similar to the one carried out in Section 7.1, with the aim of incorporating the interactions between different modes.

From a practical perspective, the τ -model increases the accuracy of the reduced model by representing the unresolved contribution as a nonlinear dissipative/increasing energy term. In particular, for systems characterized by constant energy, it can be shown that the nature of the additional term depends only on the sign of the τ . To see this consider a system of the form (7.3.1) characterized by the conservation of the norm of the solution

$$E = \frac{1}{2} \|y\|_2^2 = \frac{1}{2} \|(\hat{y}, \tilde{y})\|_2^2 = \frac{1}{2} \left(\|\hat{y}\|_2^2 + \|\tilde{y}\|_2^2 \right). \quad (7.3.22)$$

For fluid flows problems, this is usually interpreted as the kinetic energy of the system, as seen in Chapter 2. If we split the dynamical system to highlight the dynamics of the resolved and unresolved parts we recover

$$\begin{cases} \frac{d}{dt}\hat{y} = \hat{f}(\hat{y}, \tilde{y}), \\ \frac{d}{dt}\tilde{y} = \tilde{f}(\hat{y}, \tilde{y}). \end{cases} \quad (7.3.23)$$

The two relations

$$\begin{aligned} (e^{t\mathcal{L}y_0})^T (\mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}y_0) &= 0, \\ \|\tilde{f}(\hat{y}, 0)\|_2^2 + (e^{t\mathcal{L}y_0})^T (\mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}\mathcal{Q}\mathcal{L}y_0) &= 0, \end{aligned} \quad (7.3.24)$$

have been proved in [121] for (7.3.23) and, considering the τ -model approximation of the dynamics

$$\frac{\partial}{\partial t} \mathcal{P} e^{t\mathcal{L}} y_0 = \mathcal{P} e^{t\mathcal{L}} \mathcal{P} \mathcal{L} y_0 + \tau \mathcal{P} e^{t\mathcal{L}} \mathcal{P} \mathcal{L} \mathcal{Q} \mathcal{L} y_0, \quad (7.3.25)$$

we have

$$\frac{d}{dt} \|\mathcal{P} e^{t\mathcal{L}} y_0\|_2^2 = -\tau \|\tilde{f}(\hat{y}, 0)\|_2^2. \quad (7.3.26)$$

The result provided in (7.3.26) guarantees the energy stability of the τ -model.

Theorem 7.3.2. *Let $e^{t\mathcal{L}}$ and $e^{t\mathcal{Q}\mathcal{L}}$ be strongly continuous semigroups with upper bounds $\|e^{t\mathcal{L}}\| \leq M e^{t\omega}$ and $\|e^{t\mathcal{Q}\mathcal{L}}\| \leq M_{\mathcal{Q}} e^{t\omega_{\mathcal{Q}}}$. Assume that the memory kernel $K(t, s, y_0) = \mathcal{P} e^{s\mathcal{L}} \mathcal{P} \mathcal{L} e^{(t-s)\mathcal{Q}\mathcal{L}} \mathcal{Q} \mathcal{L} y_0$ is twice continuously differentiable and with compact support, i.e. $\text{supp}(K) = \{s \in [0, t] | K(t, s, y_0) \neq 0\} \subset [t - \Delta t(t), t]$. Then*

$$\left\| \int_0^t \mathcal{P} e^{s\mathcal{L}} \mathcal{P} \mathcal{L} e^{(t-s)\mathcal{Q}\mathcal{L}} \mathcal{Q} \mathcal{L} y_0 ds - \tau(t) \mathcal{P} e^{t\mathcal{L}} \mathcal{P} \mathcal{L} \mathcal{Q} \mathcal{L} y_0 \right\| \leq M_2(t)$$

where $M_2(t) = \frac{2}{3} C (\tau(t))^3$, $\tau(t) = \frac{\Delta t(t)}{2}$ and $C \geq \left\| \frac{\partial^2}{\partial s^2} K(t, s, y_0) \right\|, \forall s \in [0, t]$.

Taylor approximation orthogonal dynamics

The first attempt to increase the order of accuracy of the memory approximation by using multiple terms was made in [241]. The idea is to replace the operator $e^{(t-s)\mathcal{Q}\mathcal{L}}$ with its truncated Taylor series. The advantage of this approach is that the evolution of the orthogonal dynamics operator is taken into account, while for the t -model and its derivatives, the approximation does not depend on s . Using a Taylor series expansion, we have

$$\int_0^t \mathcal{P} e^{s\mathcal{L}} \mathcal{P} \mathcal{L} e^{(t-s)\mathcal{Q}\mathcal{L}} \mathcal{Q} \mathcal{L} y_0 ds = \sum_{j=0}^{\infty} \frac{1}{j!} \int_0^t (t-s)^j \mathcal{P} e^{s\mathcal{L}} \mathcal{P} \mathcal{L} (\mathcal{Q}\mathcal{L})^j \mathcal{Q} \mathcal{L} y_0 ds. \quad (7.3.27)$$

In this framework, the t -model can be regarded as a zeroth-order approximation of the memory integrand. Formula (7.3.27) can be approximated by considering a limited number of terms in the expansion. Although tedious to compute by hand in the nonlinear case, the term $\mathcal{P} \mathcal{L} (\mathcal{Q}\mathcal{L})^j \mathcal{Q} \mathcal{L} y_0$ inside the integral can be assembled via a recursive routine from the previous term in the expansion. Each term in the resulting sum of integrals is then approximated by numerical quadrature formulas. In the following, we present a result regarding error convergence for this memory term approximation technique.

Theorem 7.3.3. *Let $e^{t\mathcal{L}}$ and $e^{t\mathcal{Q}\mathcal{L}}$ be strongly continuous semigroups with upper bounds $\|e^{t\mathcal{L}}\| \leq M e^{t\omega}$ and $\|e^{t\mathcal{Q}\mathcal{L}}\| \leq M_{\mathcal{Q}} e^{t\omega_{\mathcal{Q}}}$, $\omega \neq \omega_{\mathcal{Q}}$. Then*

$$\left\| \int_0^t \mathcal{P} e^{s\mathcal{L}} \mathcal{P} \mathcal{L} e^{(t-s)\mathcal{Q}\mathcal{L}} \mathcal{Q} \mathcal{L} y_0 ds - \sum_{j=0}^k \frac{1}{j!} \int_0^t (t-s)^j \mathcal{P} e^{s\mathcal{L}} \mathcal{P} \mathcal{L} (\mathcal{Q}\mathcal{L})^j \mathcal{Q} \mathcal{L} y_0 ds \right\| \leq M_2(t) \quad (7.3.28)$$

where

$$M_2(t) = \frac{1}{k!} M M_{\mathcal{Q}} \|\mathcal{P}\|^2 \|\mathcal{L} (\mathcal{Q}\mathcal{L})^{k+2} y_0\| I(t)$$

and

$$I(t) = -\frac{e^{t\omega_Q}}{\omega_Q^{k+1}(\omega - \omega_Q)}\gamma(k+1, \omega_Q) + \frac{e^{t\omega}}{\omega^{k+1}(\omega - \omega_Q)}\gamma(k+1, \omega)$$

with γ that represents the incomplete Gamma function .

We point out that Taylor expansion is only one of the possible high-order expansions. Stinis himself noted that Padé expansions could provide more stable results, having a larger radius of convergence than standard Taylor expansions. An interesting and complete list of similar approaches is discussed in details in [274].

Hierarchical memory approximation

The truncated Taylor expansion described in Section 7.3.2 can be rearranged in terms of a hierarchical approximation problem, as shown in [274], while taking advantage of the limited memory kernel support. Let us consider the memory term

$$w_0(t) = \int_0^t e^{s\mathcal{L}}\mathcal{P}\mathcal{L}e^{(t-s)\mathcal{Q}\mathcal{L}}\mathcal{Q}\mathcal{L}y_0 ds. \quad (7.3.29)$$

Under the assumption of differentiability of the memory integrand in time, we can differentiate (7.3.29) to have

$$\frac{d}{dt}w_0(t) = \mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}\mathcal{Q}\mathcal{L}y_0 + \underbrace{\int_0^t \mathcal{P}e^{s\mathcal{L}}\mathcal{P}\mathcal{L}e^{(t-s)\mathcal{Q}\mathcal{L}}(\mathcal{Q}\mathcal{L})^2 y_0}_{w_1(t)}. \quad (7.3.30)$$

Equation (7.3.29) comprises a Markovian term that does not include the orthogonal evolution operator and memory-like term w_1 that depends on the orthogonal dynamics equation. The high-order hierarchical memory approximation is realized by iterating the procedure described in the previous paragraph on the term w_1 under the assumption of higher regularity of the memory integrand. This leads to the following exact infinite hierarchy of equations

$$\left\{ \begin{array}{l} \frac{d}{dt}\mathcal{P}e^{t\mathcal{L}}y_{0k} = \mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}y_0 + w_0(t), \\ \frac{d}{dt}w_0 = \mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}\mathcal{Q}\mathcal{L}y_{0k} + w_1(t), \\ \frac{d}{dt}w_1 = \mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}(\mathcal{Q}\mathcal{L})^2 y_{0k} + w_2(t), \\ \vdots \\ \frac{d}{dt}w_{(n-1)} = \mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}(\mathcal{Q}\mathcal{L})^n y_{0k} + w_n(t), \\ \vdots \end{array} \right. \quad (7.3.31)$$

However, (7.3.31) does not provide an efficient closure to the reduced model, since the issue of computing an integral term depending on $e^{(t-s)\mathcal{Q}\mathcal{L}}$ has been shifted to the next element in the hierarchy. The infinite hierarchy must be truncated at a certain index to obtain a closure model. Different solutions have been proposed in [274], but in this work, we focus on the straightforward truncation of the k -th term, i.e. $w_k^H(t) = 0$. The resulting system is known as H -model and

results concerning its accuracy are provided in [274]. Hereafter we mention one of these results.

Theorem 7.3.4. *Let $e^{t\mathcal{L}}$ and $e^{t\mathcal{Q}\mathcal{L}}$ be strongly continuous semigroups with upper bounds $\|e^{t\mathcal{L}}\| \leq M e^{t\omega}$ and $\|e^{t\mathcal{Q}\mathcal{L}}\| \leq M_{\mathcal{Q}} e^{t\omega_{\mathcal{Q}}}$. For some fixed n , we have*

$$\|w_0(t) - w_0^H(t)\| \leq M_H(t),$$

where

$$M_H(t) = M M_{\mathcal{Q}} \left\| (\mathcal{L}\mathcal{Q})^{n+1} \mathcal{L}y_0 \right\| A_1 A_2 \frac{t^{n+1}}{(n+1)!}, \quad (7.3.32)$$

and

$$A_1 = \max_{s \in [0, t]} e^{s(\omega - \omega_{\mathcal{Q}})} = \begin{cases} 1 & \omega \leq \omega_{\mathcal{Q}}, \\ e^{t(\omega - \omega_{\mathcal{Q}})} & \omega \geq \omega_{\mathcal{Q}}, \end{cases}, \quad A_2 = \max_{s \in [0, t]} e^{s\omega_{\mathcal{Q}}} = \begin{cases} 1 & \omega \leq \omega_{\mathcal{Q}}, \\ e^{t\omega_{\mathcal{Q}}} & \omega \geq \omega_{\mathcal{Q}}. \end{cases},$$

An extension to the standard H -model, taking into account the finiteness of the memory integrand support of all the hierarchical integrands, is provided by Parish in [191]. A reasonable assumption is that the support and kernel magnitude of the memory term becomes smaller as we move down the approximation hierarchy. As for the derivation of the τ -model from the t -model, we have

$$\omega_{0k}(t) = \int_0^t e^{s\mathcal{L}} \mathcal{P} \mathcal{L} e^{(t-s)\mathcal{Q}\mathcal{L}} \mathcal{Q} \mathcal{L} y_{0k} ds \approx \int_{t-\tau(t)}^t e^{s\mathcal{L}} \mathcal{P} \mathcal{L} e^{(t-s)\mathcal{Q}\mathcal{L}} \mathcal{Q} \mathcal{L} y_{0k} ds, \quad k \in R. \quad (7.3.33)$$

Differently from (7.3.30), however, if we differentiate the memory contribution we have an additional term

$$\begin{aligned} \frac{d}{dt} w_{0k}(t) &= \mathcal{P} e^{t\mathcal{L}} \mathcal{P} \mathcal{L} \mathcal{Q} \mathcal{L} y_{0k} + \underbrace{\int_{t-\tau(t)}^t \mathcal{P} e^{s\mathcal{L}} \mathcal{P} \mathcal{L} e^{(t-s)\mathcal{Q}\mathcal{L}} (\mathcal{Q}\mathcal{L})^2 y_{0k} ds}_{w_1(t)} \\ &\quad - (1 - \tau'(t)) \mathcal{P} e^{(t-\tau(t))\mathcal{L}} \mathcal{P} \mathcal{L} e^{\tau(t)\mathcal{Q}\mathcal{L}} \mathcal{Q} \mathcal{L} y_0. \end{aligned} \quad (7.3.34)$$

The last term on the right hand side of (7.3.34) depends on the orthogonal dynamics and hence computing it directly in this form is not efficient. By using the trapezoidal rule to approximate (7.3.33) over a single sub-interval we have

$$w_0 \approx \frac{\Delta t(t)}{2} \left(\mathcal{P} e^{t\mathcal{L}} \mathcal{P} \mathcal{L} \mathcal{Q} \mathcal{L} y_0 - \mathcal{P} e^{(t-\Delta t(t))\mathcal{L}} \mathcal{P} \mathcal{L} e^{\Delta t(t)\mathcal{Q}\mathcal{L}} \mathcal{Q} \mathcal{L} y_0 \right), \quad (7.3.35)$$

which, paired with (7.3.34), gives

$$\frac{d}{dt} w_0 = -2 \frac{1 - \tau'(t)}{\Delta t(t)} w_0 + (2 - \tau'(t)) \mathcal{P} e^{t\mathcal{L}} \mathcal{P} \mathcal{L} \mathcal{Q} \mathcal{L} y_0 + w_1(t).$$

To obtain a closed scheme, repeated differentiation of the integral term and quadrature rule to eliminate the dependence on the orthogonal dynamic, followed by a truncation of the last integral,

can be used to have

$$\left\{ \begin{array}{l} \frac{d}{dt} \mathcal{P}e^{t\mathcal{L}} y_0 = \mathcal{P}e^{t\mathcal{L}} \mathcal{P}\mathcal{L} y_0 + w_0(t), \\ \frac{d}{dt} w_0 = -2 \frac{1 - \tau'_0(t)}{\tau_0(t)} w_0(t) + (2 - \tau'_0(t)) \mathcal{P}e^{t\mathcal{L}} \mathcal{P}\mathcal{L}\mathcal{Q}\mathcal{L} y_0 + w_1(t), \\ \frac{d}{dt} w_1 = -2 \frac{1 - \tau'_1(t)}{\tau_1(t)} w_1(t) + (2 - \tau'_1(t)) \mathcal{P}e^{t\mathcal{L}} \mathcal{P}\mathcal{L}(\mathcal{Q}\mathcal{L})^2 y_0 + w_2(t), \\ \vdots \\ \frac{d}{dt} w_{(n-1)} = -2 \frac{1 - \tau'_{n-1}(t)}{\tau_{n-1}(t)} w_{(n-1)}(t) + (2 - \tau'_{(n-1)}(t)) \mathcal{P}e^{t\mathcal{L}} \mathcal{P}\mathcal{L}(\mathcal{Q}\mathcal{L})^n y_0. \end{array} \right. \quad (7.3.36)$$

7.3.3 Estimates of the memory length

To accurately estimate the memory length τ , it is necessary to assemble the memory kernel. Unfortunately, as we saw in Section 7.3.1, this operation requires the solution of the orthogonal dynamics equations (7.2.14), whose computational cost is prohibitive and higher than that required to obtain a solution of the FOM. The procedure described in Section 7.3.1, and proposed by Gouasmi in [108], is useful for asserting the validity of some of the assumptions made during the modeling phase but is not efficient enough to be used in the online phase of ROM. Several approaches have been proposed to estimate hyper-parameters of models generated from the Mori-Zwanzig formalism. We saw in 7.3.2 that Stinis proposed a modification of the t -model based on renormalization coefficients to be estimated in accordance with the rate of change of the p -norms of quantities of interest. Even though this approach shows good results for the Burgers' equation discretized using Fourier basis, we did not obtain a similar accuracy using the POD basis to approximate the same problem.

One popular approach is the dynamic- τ model proposed by Parish in [192]. In this approach, which we will detail below, the value of τ is computed using the assumption of finite memory, the value of the memory kernel for $s = t$, and the Germano identity [174]. We recall that for the τ -model, the memory term is approximated as

$$\int_{t-\tau(t)}^t \mathcal{P}e^{s\mathcal{L}} \mathcal{P}\mathcal{L}e^{(t-s)\mathcal{Q}\mathcal{L}} \mathcal{Q}\mathcal{L} y_{0k} ds \approx \frac{1}{2} \tau(t) \mathcal{P}e^{t\mathcal{L}} \mathcal{P}\mathcal{L}\mathcal{Q}\mathcal{L} y_{0k}, \quad (7.3.37)$$

where $\tau(t)$ has to be specified and is usually considered constant in time. The first step in estimating τ , according to [192], is to adopt Germano's identity to model the behavior of τ as a function of the scale of the resolved part of the system. The idea is to introduce two standard sharp spectral cutoff filters $\hat{\mathcal{G}}$ and $\bar{\mathcal{G}}$, such that, if we decompose the resolved variable as

$$\hat{y}(t) = \{\bar{y}(t), y'(t)\}, \quad (7.3.38)$$

then

$$y(t) = \{\hat{y}(t), \tilde{y}(t)\} = \{\bar{y}(t), y'(t), \tilde{y}(t)\}, \quad (7.3.39)$$

and the filters are chosen in such a way that

$$\hat{\mathcal{G}}(y(t)) = \{\hat{y}(t), 0\} = \{\bar{y}(t), y'(t), 0\}, \quad \bar{\mathcal{G}}(y(t)) = \{\bar{y}(t), 0, 0\}. \quad (7.3.40)$$

From (7.2.12) and (7.3.40), it follows

$$\int_0^t \mathcal{P}K(t, s, e^{t\mathcal{L}}y_0)ds = \hat{\mathcal{G}}(f(y(t))) - f(\hat{\mathcal{G}}(y(t))), \quad (7.3.41)$$

which, after the application of the filter $\bar{\mathcal{G}}$, becomes

$$\bar{\mathcal{G}} \int_0^t \mathcal{P}K(t, s, e^{t\mathcal{L}}y_0)ds = (\bar{\mathcal{G}}(f(y(t))) - f(\bar{y}(t))) + (f(\bar{y}(t)) - \bar{\mathcal{G}}(f(\hat{y}(t))))). \quad (7.3.42)$$

In (7.3.42), the second term can be computed directly from the set (and subset defined via the filter $\bar{\mathcal{G}}$) of the resolved variables, while the first term corresponds to the memory term in the case where the set of solved variables coincides with \bar{y} . Using the τ -model approximation for the r.h.s. and the first term of the l.h.s. in (7.3.42), we obtain

$$\frac{1}{2}\tau(t)\mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}\mathcal{Q}\mathcal{L}y_0 = \frac{1}{2}\bar{\tau}(t)\bar{\mathcal{P}}e^{t\mathcal{L}}\bar{\mathcal{P}}\mathcal{L}\bar{\mathcal{Q}}\mathcal{L}y_0 + (f(\bar{y}(t)) - \bar{\mathcal{G}}(f(\hat{y}(t))))), \quad (7.3.43)$$

where $\bar{\mathcal{P}}$ and $\bar{\mathcal{Q}}$ are the projection operators defined in Section 7.2.2 with respect to the set of resolved variables defined by the filter $\bar{\mathcal{G}}$ and $\bar{\tau}$ is the relative coarse time-scale. Equation (7.3.43) is not sufficient to determine the value of τ , and a constitutive relation between τ and $\bar{\tau}$ must be introduced. In [192], a relation of the form

$$\bar{\tau} = \left(\frac{\hat{\Delta}}{\bar{\Delta}} \right)^p \tau, \quad (7.3.44)$$

is proposed, inspired by classical turbulence results, with $\hat{\Delta}$ and $\bar{\Delta}$ being the scales of the filters $\hat{\mathcal{G}}$ and $\bar{\mathcal{G}}$. The constant p in (7.3.44) has been adjusted based on a priori analysis of the ratio between the memory term and its kernel for the Burgers equation, channel flow, and homogeneous turbulence. For all examples considered, the memory length obeys approximately the form (7.3.44) with $p = 1.5$.

In our work, we exploit the data collected to assemble the snapshot matrix S used to generate the reduced basis for fitting an optimal value from relation (7.3.37). Exploiting (7.3.41) and (7.3.11), an approximation for $\tau(t)$ that is optimal for the training set is given by

$$\tau(t) \approx 2 \frac{\hat{\mathcal{G}}(f(y(t))) - f(\hat{\mathcal{G}}(y(t)))}{\mathcal{P}e^{t\mathcal{L}}\mathcal{P}\mathcal{L}\mathcal{Q}\mathcal{L}y_{0k}}. \quad (7.3.45)$$

To simplify the proposed model, a constant fit for τ over time is computed and, in the case of parametric problems, radial multiquadric basis functions are used to obtain τ values for new parameter evaluations by interpolation.

7.4 Numerical experiments

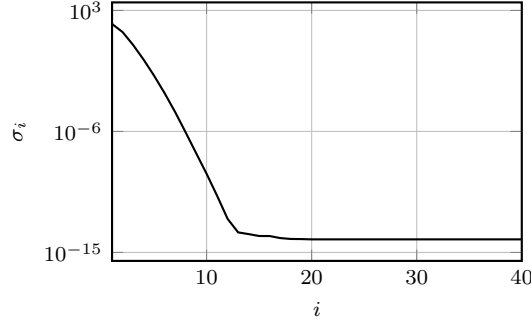


Figure 7.8: CA: Singular values decay of the snapshots matrix computed from the solution of the test Cauchy problem (7.4.1). The exponential decay suggests that a small basis is sufficient to represent the solution.

7.4.1 Randomized Cauchy problem

Consider the following autonomous linear system

$$\begin{cases} \frac{d}{dt}u(t) = Au(t), & t \in \mathcal{T} := (0, T], \\ u(0) = u_0, \end{cases} \quad (7.4.1)$$

where $A \in \mathbb{R}^{N \times N}$, $u \in \mathbb{R}^N$ and $u_0 \in \mathbb{R}^N$ is randomly taken from a standard distribution. For this toy problem, we consider $N = 100$, $T = 4$ and time t is the only parameter. The matrix A is negative definite, i.e.,

$$A = U^T B U, \quad U^T U = \mathbb{I}, \quad B = \text{diag}(\beta),$$

where $\beta \in \mathbb{R}^N$ is vector such that each entry β_i is included in the interval $(0, -3]$. For the time integration of (7.4.1), we use a second-order explicit Runge-Kutta method with time step $\Delta t = 4 \times 10^{-3}$. The resulting $N_t = 1000$ snapshots are collected in the matrix $S \in \mathbb{R}^{N \times N_t}$. Finally, the singular value decomposition is applied to S , and the singular values show the exponential decay reported in Figure 7.8.

Using a POD basis of dimension $n_1 = 16$, we assume that the solution to the problem (7.4.1) could be approximate up to machine precision. To simulate examples of under-resolution regimes, we consider ROMs of sizes $n_2 \in [2, \dots, 9]$. The classical POD-Galerkin-based ROM is compared to the hierarchical closure models described in Section 7.3.2, for different truncation orders, in Figure 7.9, where pointwise convergence to the projected solution on the approximating space of dimension n_2 is tested. We note that increasing the order of approximation in the hierarchical model asymptotically guarantees an improvement in the convergence properties of the closure model. In contrast, this improvement is not consistent when only 1 or 2 terms in the hierarchy are computed before truncation. A similar conclusion can be drawn by directly comparing the entries of the approximate memory term, as seen in Figure 7.10.

This phenomenon is well known for approximation techniques based on truncations similar to those operated for the hierarchical model we are examining, and takes the name of *convergence barrier* (see Corollary 3.4.2 in [272]). One can get an idea of why this problem arises by analyzing the estimate (7.3.32) given in Theorem 7.3.4, despite the fact that this is only an upper bound of the norm of the memory term approximation error. Once the value of n_2 is fixed, the upper

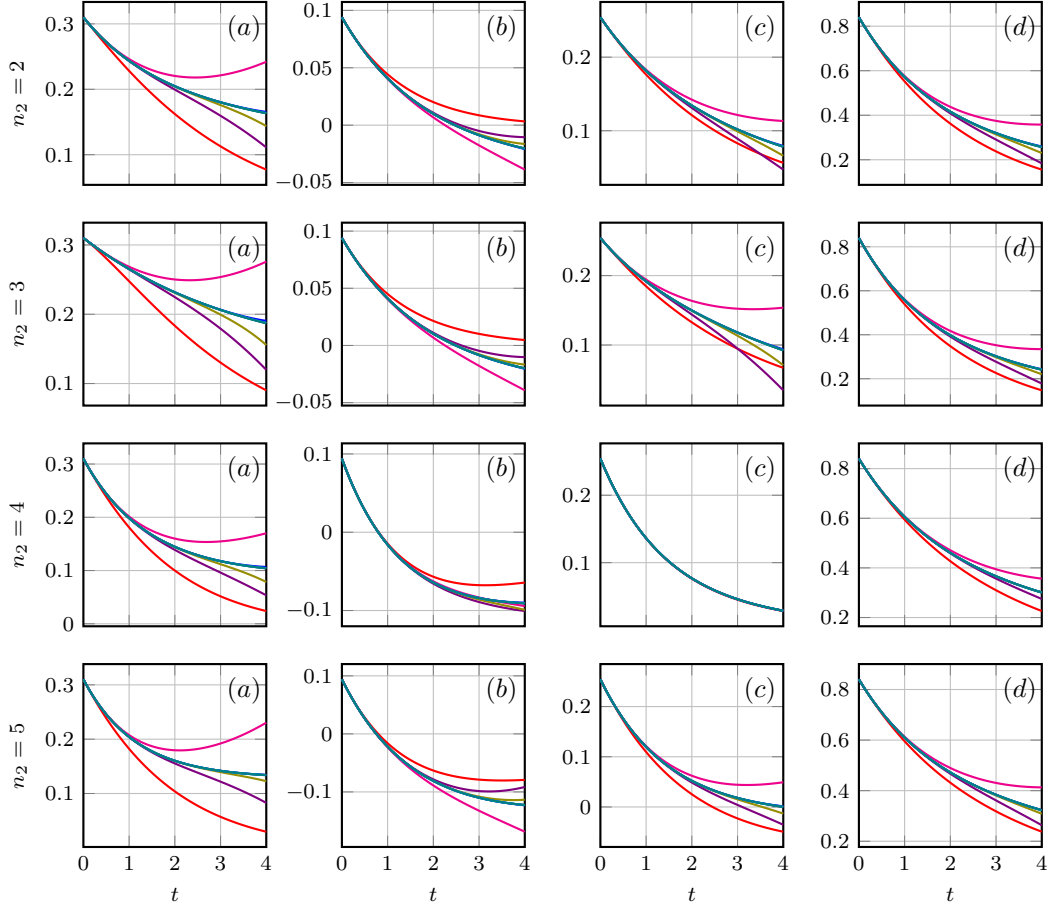


Figure 7.9: CA: Qualitative comparison of the entries 11(a), 34(b), 36(c), and 67(d) of the projection of the exact solution of the problem (7.4.1) onto a space of dimension $n_1 = 16$ (—) and under-resolved ROM solutions for different values of $n_2 \in [2, 3, 4, 5]$. For each value of n_2 , hierarchical models of closure are compared for truncation orders $k = 0$ (—), $k = 1$ (—), $k = 2$ (—), $k = 4$ (—), $k = 7$ (—), $k = 9$ (—), and $k = 11$ (—).

bound in (7.3.32) depends on the term

$$\left\| (\mathcal{LQ})^{n+1} \mathcal{L}y_0 \right\| \frac{t^{n+1}}{(n+1)!},$$

which does not necessarily decrease monotonically with n . A similar conclusion is drawn by comparing directly the approximated memory entries (see Figure 7.10).

7.4.2 1D Burgers' equation

As a second numerical test, we consider the 1D viscous Burgers' equation

$$\begin{cases} \frac{\partial}{\partial t} u(t; \mu) = -\frac{1}{2} \frac{\partial}{\partial x} u(t; \mu)^2 + \mu \frac{\partial^2}{\partial x^2} u(t; \mu), & t \in (0, 5], x \in [0, 2\pi], \\ u(0; \mu) = u_0(x), \end{cases} \quad (7.4.2)$$

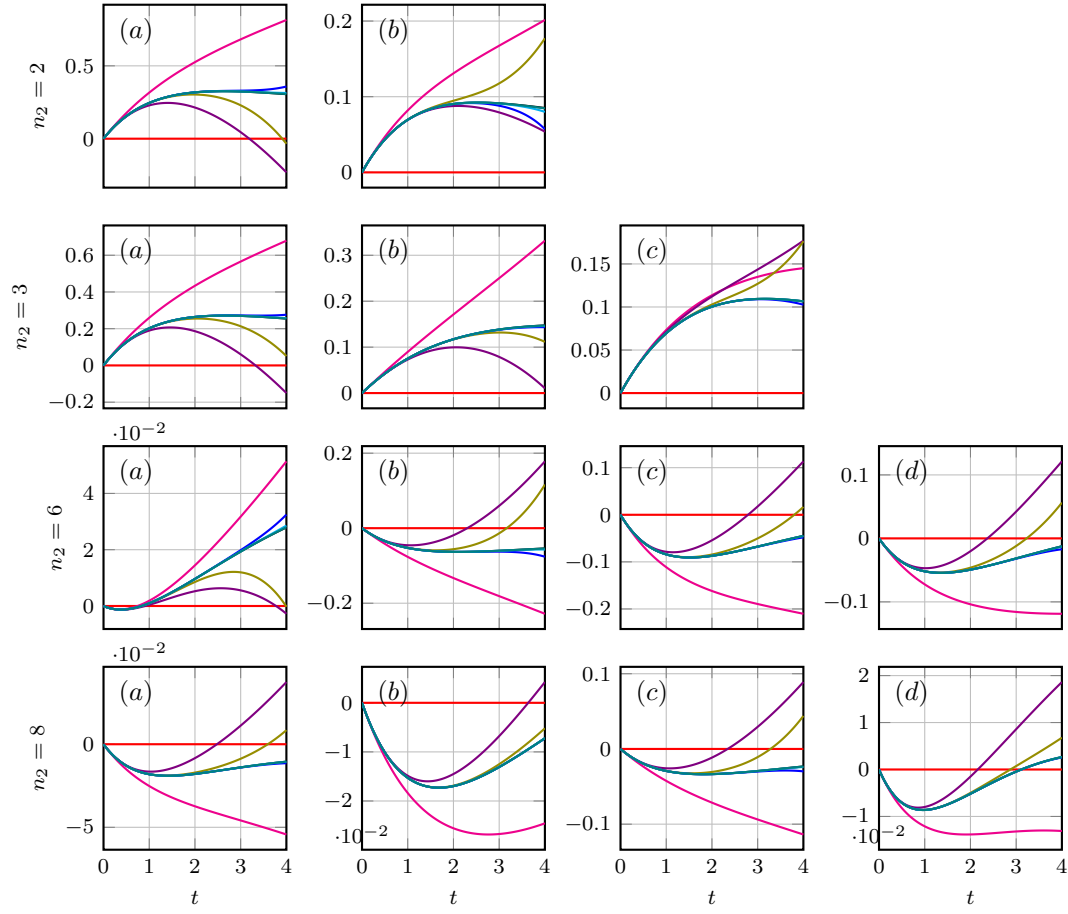


Figure 7.10: CA: Comparison of the entries 1(a), 2(b), 3(c), and 5(d) of the exact memory term and the hierarchical memory approximations for $n_2 \in [2, 3, 6, 8]$ and truncation orders $k = 0$ (—), $k = 1$ (—), $k = 2$ (—), $k = 4$ (—), $k = 7$ (—), $k = 9$ (—), and $k = 11$ (—).

where the initial condition is given as a combination of sinusoidal signals

$$u_0(x) = \sum_i^{k_c} \sqrt{2E(k_i)} \sin(k_i x + \beta_i),$$

where

$$E(k) = \begin{cases} 5^{-\frac{5}{3}}, & \text{if } 1 \leq k \leq 5, \\ k^{-\frac{5}{3}}, & \text{otherwise.} \end{cases}$$

Periodic boundary conditions are considered for this problem. The phase angle β is sampled following a uniform distribution from the interval $[-\pi, \pi]$. Partial derivatives in space are discretized using the method of finite differences on a grid of $N = 1024$ nodes. The FOM is integrated numerically in time using the implicit midpoint method with $N_t = 1000$ time steps, for 8 different values of the viscosity parameter $\mu \in [0.01, 0.015, 0.02, 0.025, 0.03, 0.035, 0.04, 0.045, 0.05]$. Following the POD algorithm, the snapshots for all the values of t and μ considered are collected in the snapshots matrix, and the left singular values are taken as the reduced basis. In this setting, $n_1 = 100$ represents the minimum size of a POD basis to represent the exact solution to the problem (7.4.2) up to machine precision. Hyper-reduction of the nonlinear term is performed by the DEIM algorithm described in Section 1.5 using 80 interpolation points.

With this numerical example, we want to evaluate the ability of the τ model to approximate the memory term. In Figure 7.6, we have seen that using a single value of τ for all the modes of the resolved part of the simulation is a reasonable modeling assumption. This value of τ is then approximated following the strategy defined at the end of Section 7.3.3. We consider the error defined as

$$\varepsilon(t; \mu) = \|U_{n_2} U_{n_2}^\top u(t; \mu) - U_{n_2} z_{n_2}(t; \mu)\|_2, \quad (7.4.3)$$

where $u(t; \mu) \in \mathbb{R}^N$ is the solution to (7.4.2), $U_{n_2} \in \mathbb{R}^{N \times n_2}$ is a POD basis of under-resolved dimension n_2 , and $z_{n_2}(t; \mu) \in \mathbb{R}^{n_2}$ are the POD coefficients obtained by solving the under-resolved ROM, with or without the closure model. The τ model outperforms the standard approach, being between 2 to 10 times more accurate than the counterpart. In Figure 7.12, we compare the exact memory term $w(t)$ with the approximation provided by the τ model for different scales n_2 of the resolved simulation and parameter values. Considering the approximations taken into account, The superiority, in terms of accuracy, obtained through the introduction of the closure term is further evident from the analysis of the energy transfer due to the closure term, shown in Figure 7.13.

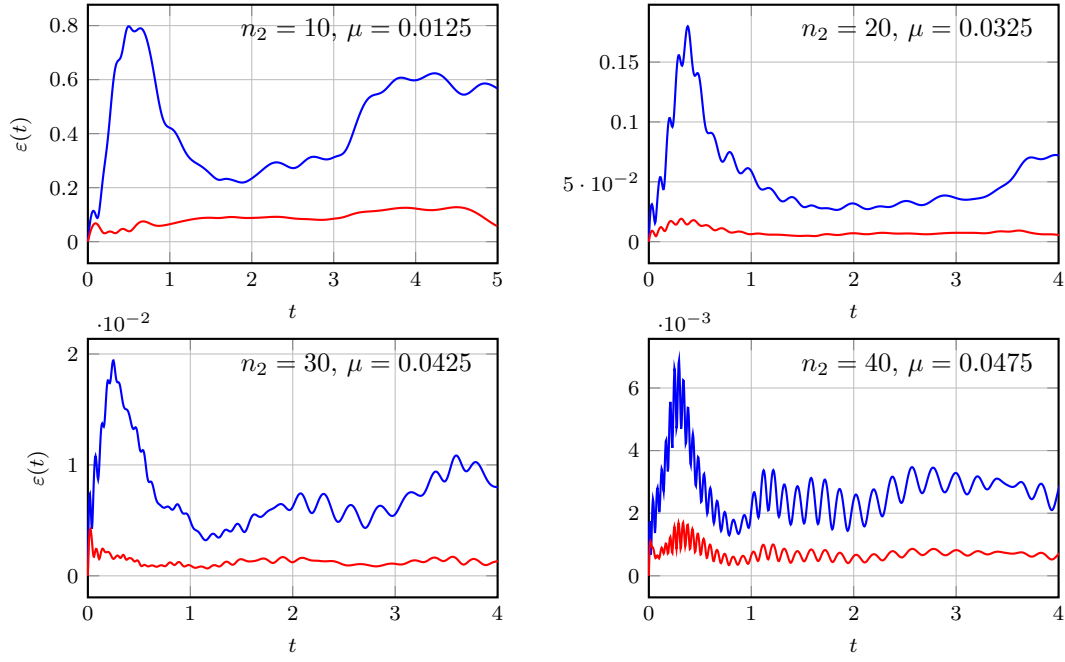


Figure 7.11: BG: Error (7.4.3) for POD-Galerkin model solution with (—) and without closure model (—), evaluated for different reduced model sizes and parameters.

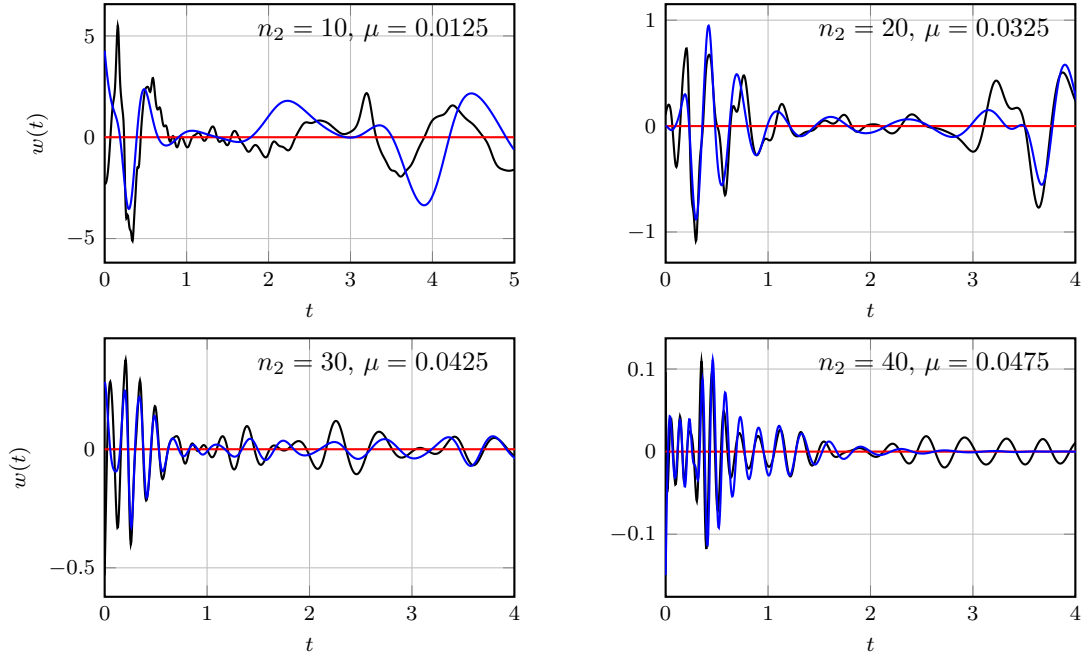


Figure 7.12: BG: Average of the memory contribution over the entire set of n_2 modes in case of exact memory (—), memory approximation with τ -model (—), and truncation (—), evaluated for different reduced model sizes and parameters.

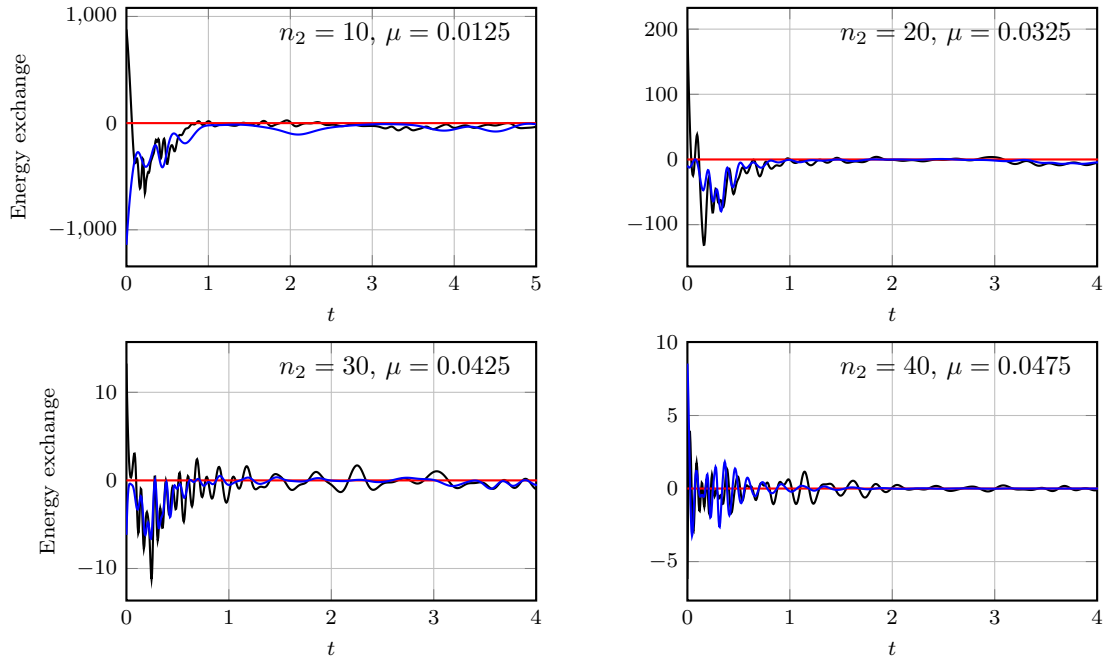


Figure 7.13: BG: Average energy transfer between the resolved and unresolved part of the simulation, in case of exact memory (—), memory approximation with τ -model (—), and truncation (—), evaluated for different reduced model sizes and parameters.

8 Recurrent neural network closure of parametric POD-Galerkin reduced-order models

The availability of a large amount of data and the ease of use of the latest software for machine learning has allowed rapid progress in data-driven modeling. In particular, the fact that projection-based ROMs require a matrix of snapshots to be generated makes it natural to reuse this data to model the closure term in a data-driven manner. Given the nature of the problem, most of the proposed approaches reformulate the model closure problem as a supervised regression task, where the closure term is estimated as a function of the reduced representation coefficients. An initial division between the methods available in the literature is made between *trajectory regression*-based and *model regression*-based methods. For the former, a form of the ROM closure term is established a priori, and the available data are used to estimate the parameters characterizing the closure term, such that the best fit of the reduced coefficients is obtained. They are the first data-driven methods to be developed and, in the case of functional models, have been proposed as a direct improvement over physics-based approaches. We refer the reader to [262; 205; 238; 28] for several examples of data-driven estimates of viscosity coefficients for mixed-length and Smagorinsky models. When the additional closure term is not derived directly from physical principles but seeks to be as general as possible, we refer to these methods as calibration methods. Linear [42; 18], quadratic [69], and cubic [199] models have been proposed and have achieved some success for fluid dynamics problems.

Recently, model regression techniques have received more attention than trajectory-regression methods due to the success of neural networks. They have a greater descriptive capability than the methods described above since they require only mild ansatzes about the structure of the term to be modeled, and regression occurs directly on the memory term and not for the coefficients of the reduced system. In [185], it is conjectured that this type of strategy is more accurate in predicting the solution beyond the time interval of the training set, whereas trajectory regression-based methods would be more faithful to the solution of the FOM in the time interval of the training set. Unfortunately, detailed numerical analyses supporting this conjecture are not available to the author's knowledge.

In this Chapter, we use a particular type of recurrent neural network, called the long short-term memory (LSTM) neural network, to estimate the mismatch between FOM and ROM dynamics in a parametric context, based on the formulation of the memory term provided in Chapter 7. The idea is that the gating mechanism underlying this framework, which is used to select which information should be forgotten or remembered in a sequence of data as time evolves, can be a

good inductive bias for approximating memory terms characterized by exponentially decaying kernels. Some information from the FOM system to be approximated, such as the memory length, can then be enforced in the closure term by selecting the number of cells in the LSTM neural network.

In Section 8.1, the structure and training of an LSTM network are described in detail, emphasizing connections with the Mori-Zwanzig formalism. Next, in Section 8.2, we explain how the reduced model fitted with the closure term is numerically integrated in time. Finally, the Chapter concludes with Section 8.3, where several numerical experiments are conducted to evaluate the approach's effectiveness.¹

8.1 Recurrent neural network memory model

This Section presents the modeling of the memory effect using recurrent neural networks (RNNs). A conditioned long short term memory (LSTM) network predicts the memory integral, given a short history of the reduced basis coefficients. The structure and training of the network will be presented in the remainder of this Section.

8.1.1 Regression of the memory term

Let us consider the discretization $(t_{i-1}, t_i]$ of the interval \mathcal{T} , where each interval has the length $\Delta t = t_i - t_{i-1}$. Taking advantage of the finite memory length approximation introduced in Section 7, the memory integral $w(t_i, z; \eta)$ can be approximated using a short history of the reduced basis solution as

$$\begin{aligned} w(t_i, z; \eta) &= \int_0^{t_i} K(t_i, s, z(s; \eta); \eta) ds \\ &\approx \int_{t_i - \tau}^{t_i} K(t_i, s, z(s; \eta); \eta) ds \\ &\approx \tilde{w}(z_{i-N_{ts}+1}, \dots, z_{i-1}, z_i; \eta), \end{aligned}$$

where N_{ts} is the number of time steps included in the support of the memory kernel, $\tau = N_{ts}\Delta t$ is the memory length, and \tilde{w} is a numerical memory model that serves as the approximation of the map

$$(z_{i-N_{ts}+1}, \dots, z_{i-1}, z_i; \eta) \mapsto w(t_i, z; \eta). \quad (8.1.1)$$

The approximation of the map in (8.1.1) is a regression task. In this Section, we use an artificial neural network (ANN) as the regression model of the memory integral. The ANN seeks to predict the memory integral, given a sequence of the reduced coefficients.

8.1.2 Conditioned long short-term memory network

Recurrent neural networks (RNNs) are a class of neural networks suitable for *sequential modeling* [106]. Hence, RNN is a natural choice for the integral memory regression, a "many to one" sequential modeling task. In our work, the *long short-term memory* (LSTM) [132], one of the

¹The author contributed equally to Dr. Wang in the definition of the general research question, the design of the approach and methodology, and the implementation of the numerical experiments.

most popular gated RNNs that are developed to address the exploding/vanishing gradient issue that can be encountered when training traditional RNNs [106], is selected as the basic RNN structure for memory modeling.

A challenge in the design of the LSTM network for memory modeling is how to incorporate the physical/geometrical parameters encoded in η . We need to predict the memory term at an arbitrary parameter location during the online stage of the model reduction of a parametrized system. Therefore, the LSTM network needs to be conditioned by the non-temporal physical/geometrical parameters. There are several existing works on feeding non-temporal data into the RNN. For example, in the encoder-decoder network for machine translation [60; 244], the final state of the encoder LSTM network is set as the initial state of the decoder LSTM network. In the image caption network [141], the output of the convolutional neural network (CNN) is fed into the hidden state of the first gated recurrent unit (GRU). Inspired by these works, we condition the LSTM network by feeding the physical/geometrical parameters into the initial hidden state. The architecture of the conditioned LSTM network is shown in Figure 8.1.

The conditioned LSTM network consists of one dense layer, one LSTM layer and one another dense layer. If we denote the number of hidden units of the LSTM as n_{hu} , the first dense layer maps the parameter vector $\eta \in \mathbb{R}^d$ to the initial hidden state $h_0 \in \mathbb{R}^{n_{hu}}$, the LSTM layer maps the input sequence $\{x_1, x_2, \dots, x_{n_{ts}}\} \in \mathbb{R}^{n_{ts} \times m}$ to the hidden states $\{h_1, h_2, \dots, h_{n_{ts}}\} \in \mathbb{R}^{n_{ts} \times n_{hu}}$ and the second dense layer maps the final hidden state $h_{n_{ts}} \in \mathbb{R}^{n_{hu}}$ to the output $y \in \mathbb{R}^m$. The initial hidden state h_0 is the output of the first dense layer, which has no bias and uses a linear (identity) activation function, i.e.,

$$h_0 = W_{h_0} \eta. \quad (8.1.2)$$

The initial cell state c_0 is set as zero. The forward propagation of the conditioned LSTM network is achieved by iterating the following recurrent relation for $t = 1$ to N_{ts} :

$$\begin{aligned} f_t &= \sigma(W_f x_t + U_f h_{t-1} + b_f) \\ i_t &= \sigma(W_i x_t + U_i h_{t-1} + b_i) \\ o_t &= \sigma(W_o x_t + U_o h_{t-1} + b_o) \\ \tilde{c}_t &= \tanh(W_c x_t + U_c h_{t-1} + b_c) \\ c_t &= f_t \circ c_{t-1} + i_t \circ \tilde{c}_t \\ h_t &= o_t \circ \tanh(c_t) \end{aligned}$$

where σ is the hard sigmoid activation function

$$\sigma(z) = \begin{cases} 0, & z < -2.5, \\ 0.2z + 0.5, & -2.5 \leq z \leq 2.5, \\ 1, & z > 2.5. \end{cases}$$

The final output of the network is obtained through the second dense layer with a linear (identity) activation function, i.e.,

$$y = W_y h_{n_{ts}} + b_y. \quad (8.1.3)$$

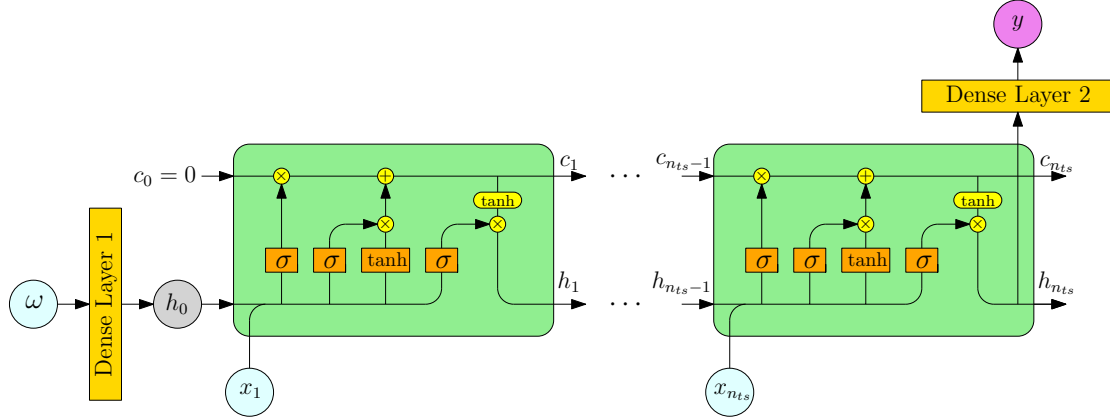


Figure 8.1: Conditioned LSTM network. In this sketch, ε represents the system parameter and x_i are the elements of the input sequence.

The matrices

$$W_{h_0} \in \mathbb{R}^{n_{hu} \times d}, W = \begin{pmatrix} W_f \\ W_i \\ W_o \\ W_c \end{pmatrix} \in \mathbb{R}^{4n_{hu} \times m}, U = \begin{pmatrix} U_f \\ U_i \\ U_o \\ U_c \end{pmatrix} \in \mathbb{R}^{4n_{hu} \times n_{hu}}, W_y \in \mathbb{R}^{m \times n_{hu}},$$

and vectors

$$b = \begin{pmatrix} b_f \\ b_i \\ b_o \\ b_c \end{pmatrix} \in \mathbb{R}^{4n_{hu}}, b_y \in \mathbb{R}^m,$$

are the trainable weights and biases that are adjusted in the training process to achieve an optimal network configuration.

For the conditioned LSTM network for memory modeling, the input is the sequence of the reduced coefficients $x = \{z_{i-N_{ts}+1}, \dots, z_i\}$, the auxiliary input is the physical/geometrical parameter vector η , and the output is the predicted memory integral $w^{\text{LSTM}}(z_{i-N_{ts}+1}, \dots, z_i; \eta)$. The conditioned LSTM network is an approximation of the map in (8.1.1).

8.1.3 Training of the network

The training of the conditioned LSTM network is a *supervised learning* [106] task. In supervised learning, labeled data is used to train the network. The goal of the training is to minimize the difference between the predicted output w^{LSTM} and the desired output w .

A training data set $\mathcal{D}_{tr} = \{((\eta, x), w)_j\}_{1 \leq j \leq N_{tr}}$ and a validation data set $\mathcal{D}_{va} = \{((\eta, x), w)_j\}_{1 \leq j \leq N_{va}}$ are used in the training. Here (η, x) is the *input object*, w is the *desired output*. The training data is collected from high-fidelity simulations with uniformly sampled parameter values, and the validation data is collected from high-fidelity simulations with randomly sampled parameter values.

A component of the input pattern is called a *feature*. The *feature scaling* technique in which

all the features are scaled to the same range can be applied to the data sets to accelerate the training process [136]. In this thesis, a feature χ is scaled by the mean normalization

$$\tilde{\chi} = \frac{\chi - \bar{\chi}}{\sigma_{\chi}},$$

where $\bar{\chi}$ and σ_{χ} are the mean and standard deviation of χ , respectively.

The network training is implemented in Keras [61], with TensorFlow [1] as the backend. The optimal weights and biases of the network are obtained using the *Adam stochastic optimizer* [143], which uses *mini-batches* of size $N_b < N_{tr}$ of the training data to take a single optimization step by minimizing the loss function. The full training data set with N_{tr} data points is shuffled, and N_{tr}/N_b mini-batches are extracted to take N_{tr}/N_b optimization steps. Once the entire training data set is exhausted, the training is said to complete one training epoch. The training is performed for a sufficient number of epochs to obtain a converged network. The *learning rate* η controls the convergence speed of the training.

For the training in this paper, the loss function is the mean absolute error (MAE). To avoid possible overfitting, a *weight regularization* term that is the sum of the L_2 -norm penalties of W_{h_0} , W , U and W_y , is added to the loss function. The weight regularization effect is controlled by a hyper-parameter λ .

At the beginning of the training, the network's weights and biases are randomly initialized using normal distributions [103]. Therefore, the training needs to be performed several times, following a *multiple restarts* approach [135], to prevent the training results from depending on the initialization of the weights. In this paper, ten restarts are performed for the training of each network, and the trained model with the best validation accuracy is selected as the final model. The validation accuracy metric is the mean squared error (MSE).

8.1.4 Model selection

For a certain problem, the size of the conditioned LSTM mainly depends on the number of hidden units n_{hu} and number of time steps N_{ts} . Therefore, we need a strategy to select a trained network with a proper combination of n_{hu} and N_{ts} as the regression model for the memory integral.

Given enough training data, more hidden units imply a larger network, resulting in a higher generalization accuracy. However, this is not the case for the number of time steps. As described in Section 7, every problem has a specific range of memory lengths in the parameter space. Therefore, the network with a too-small number of time steps, corresponding to a too short memory length, does not have enough information to predict the memory integral accurately. However, at the same time, a network with too many time steps, corresponding to a too-long memory length, also can not accurately predict the memory integral since the hidden map between the input and output is too complicated and requires a larger network, invalidating the finite memory assumptions on which we base this approximation. Therefore, if the finite memory assumption is satisfied, a suitable pair of (n_{hu}, N_{ts}) should be found to balance accuracy and cost. In the case of non-autonomous systems, time t can be considered as an additional input parameter to the network.

In this thesis, we train networks with a different number of time steps and hidden units. We select the most accurate model from the models of which the memory lengths lie between the minimum and maximum values estimated, in the parameter range, by the method described in Section 7.

8.2 Parametric POD-Galerkin with the RNN memory model

This section presents the implementation of the RNN memory model in the framework of parametric POD-Galerkin MOR.

8.2.1 POD-Galerkin with memory

For the sake of clarity, we report below some of the equations introduced in Chapters 1 and 7. Consider the POD-Galerkin ROM

$$\frac{d}{dt}z(t; \eta) = U^\top f(t, Uz + \bar{u}; \eta) = \tilde{f}(t, z; \eta), \quad (8.2.1)$$

where z is the reduced coefficient vector, \tilde{f} is the RHS term, and \bar{u} is a bias term introduced in the RB ansatz. A *memory closure* term \mathcal{M} is added to the RHS, which results in the *corrected* reduced-order model

$$\frac{d}{dt}z(t, \eta) = \tilde{f}(t, z; \eta) + w(t, z; \eta). \quad (8.2.2)$$

The motivation for the introduction of the memory term into the reduced-order model is to account for the effect of the unresolved POD modes on the resolved POD modes, which can improve the accuracy and stability of the reduced-order model.

The mechanism of the memory effect for the POD-Galerkin reduced-order model is sketched in Figure 8.2. By multiplying (8.2.2) by $2z(t; \eta)^\top$, we obtain the energy evolution equation

$$\frac{d}{dt}z^\top z(t, \eta) = 2z^\top \tilde{f}(t, z; \eta) + 2z^\top w(t, z; \eta), \quad (8.2.3)$$

in which the second term in the RHS describes the *energy exchange* between the resolved scales and the unresolved scales, which plays the same role as the subgrid-scale stress (SGS) in a large-eddy simulation (LES). The introduction of the memory closure seeks to reduce the difference between the reduced basis solution and the projection of the high-fidelity solution onto the reduced space by providing the missing dynamics caused by omitting the unresolved POD modes in the energy exchange term. With the memory closure, the trajectory of the reduced basis solution can follow the trajectory of the projection of the high-fidelity solution. The projection of the high-fidelity solution is the upper limit of the reduced basis solution in terms of accuracy.

8.2.2 Implicit-explicit Runge-Kutta time integration

Approximation of the memory term results in an improvement, in terms of computational efficiency, over the exact evaluation of the memory term. However, in explicit or implicit time-stepping, such as the Runge-Kutta (RK) method, the memory model needs to be evaluated in each stage or inner iteration step for nonlinear systems, leading to substantial additional computational cost. Therefore, the reduced-order model with the RNN memory closure is usually more expensive than the original reduced-order model.

An efficient implementation of the conditioned LSTM memory model in the POD-Galerkin framework is proposed in this thesis to address this efficiency issue. The *implicit-explicit Runge-Kutta* (IMEX-RK) scheme is used to advance the reduced-order model in time. More specifically, the RHS \tilde{f} is integrated using the diagonally implicit Runge-Kutta (DIRK) scheme, and the

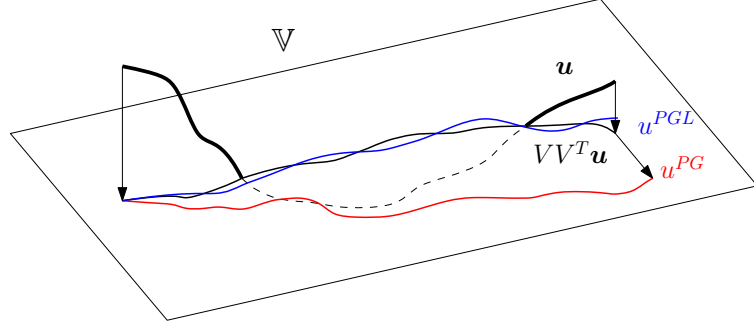


Figure 8.2: Comparison between the solution u to the FOM (—), the projection of u onto the approximating manifold (—), the solution u^{PG} to the POD-Galerkin ROM (—), and the expected solution u^{PGL} to the POD-Galerkin ROM with memory closure based on LSTM network (—).

memory term w is integrated using the explicit Runge-Kutta (ERK) scheme. The s -stage IMEX RK scheme is

$$z_i^{(j)} = z_i + \Delta t \sum_{k=1}^j a_{jk} \tilde{f}(t_i + c_k \Delta t, z_i^{(k)}; \eta) + \Delta t \sum_{k=1}^{j-1} \hat{a}_{jk} w(t_i + \hat{c}_k \Delta t, z^{(k)}; \eta), \quad j = 1, \dots, s, \quad (8.2.4)$$

$$z_{i+1} = z_i + \Delta t \sum_{k=1}^s b_k \tilde{f}(t_i + c_k \Delta t, z_i^{(j)}; \eta) + \Delta t \sum_{k=1}^s \hat{b}_k w(t_i + \hat{c}_k \Delta t, z^{(k)}; \eta), \quad (8.2.5)$$

where $\mathbf{a}, \mathbf{b}, \mathbf{c}$ are the coefficients for the DIRK and $\hat{\mathbf{a}}, \hat{\mathbf{b}}, \hat{\mathbf{c}}$ are the coefficients for the ERK, defined by the following Butcher tableaux

$$\begin{array}{c|cccccc} 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ c_2 & a_{21} & a_{22} & \cdots & 0 & 0 & c_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ c_{s-1} & a_{s-1,1} & a_{s-1,2} & \cdots & a_{s-1,s-1} & 0 & c_{s-1} \\ c_s & a_{s1} & a_{s2} & \cdots & a_{s,s-1} & a_{ss} & c_s \\ \hline & b_1 & b_2 & \cdots & b_{s-1} & b_s & \end{array}, \quad \begin{array}{c|cccccc} 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \hat{c}_2 & \hat{a}_{21} & 0 & \cdots & 0 & 0 & \hat{c}_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \hat{c}_{s-1} & \hat{a}_{s-1,1} & \hat{a}_{s-1,2} & \cdots & 0 & 0 & \hat{c}_{s-1} \\ \hat{c}_s & \hat{a}_{s1} & \hat{a}_{s2} & \cdots & \hat{a}_{s,s-1} & 0 & \hat{c}_s \\ \hline & \hat{b}_1 & \hat{b}_2 & \cdots & \hat{b}_{s-1} & \hat{b}_s & \end{array},$$

with the constraint

$$c_j = \sum_{k=1}^j a_{jk} = \sum_{k=1}^{j-1} \hat{a}_{jk}, \quad j = 1, \dots, s.$$

The memory term $w(t_i + \hat{c}_k \Delta t, z^{(k)})$ is computed as

$$w(t_i + \hat{c}_k \Delta t, z^{(k)}; \eta) = w^{LSTM}(z_{i-N_{ts}+1}^{(k)}, \dots, z_i^{(k)}; \eta). \quad (8.2.6)$$

For linear systems, (8.2.4) is solved directly; while for nonlinear systems, (8.2.4) is solved iteratively, e.g., through a Newton's method.

For nonlinear problems, in each stage of the IMEX-RK, the memory term needs to be computed only once, while the inner iteration needs to be performed until the residual reaches a certain threshold. Therefore, most of the computational time is consumed by this inner iteration.

Recurrent neural network closure of parametric POD-Galerkin reduced-order models

Numerical results for nonlinear problems demonstrate that the POD-Galerkin with memory is much more efficient than the original POD-Galerkin since introducing the memory closure into the IMEX-RK leads to significant accuracy improvement and only a small extra computational cost.

The IMEX-RK schemes used in the numerical experiments in Section 8.3 are:

1. IMEX-Euler

$$z_{i+1} = z_i + \Delta t f(t_{i+1}, z_{i+1}; \eta) + w(t_i, z; \eta) \quad (8.2.7)$$

The corresponding Butcher tableaux are

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 0 & 1 \\ \hline & 0 & 1 \end{array}, \quad \begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1 & 0 \\ \hline & 1 & 0 \end{array}.$$

The IMEX-Euler scheme is first-order accurate.

2. IMEX-Trapezoidal

$$\tilde{z}_{i+1} = z_i + \frac{\Delta t}{2} (f(t_i, z_i; \omega) + f(t_{i+1}, \tilde{z}_{i+1}; \omega)) + \Delta t w(t_i, z; \omega), \quad (8.2.8)$$

$$z_{i+1} = z_i + \frac{\Delta t}{2} (f(t_i, z_i; \omega) + f(t_{i+1}, \tilde{z}_{i+1}; \omega)) + \frac{\Delta t}{2} (w(t_i, z; \omega) + w(t_{i+1}, \tilde{z}; \omega)) \quad (8.2.9)$$

The corresponding Butcher tableaux are

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1/2 & 1/2 \\ \hline & 1/2 & 1/2 \end{array}, \quad \begin{array}{c|cc} 0 & 0 & 0 \\ 1 & 1 & 0 \\ \hline & 1/2 & 1/2 \end{array}.$$

The IMEX-Trapezoidal scheme is second-order accurate.

8.3 Numerical results

This Section presents the numerical results of the POD-Galerkin with the RNN memory closure for model reduction of a 3D Stokes flow, the 1D Kuramoto–Sivashinsky (KS) equation, and the 2D Rayleigh–Bénard convection. The linear 3D Stokes flow problem is used to validate the method. The 1D Kuramoto–Sivashinsky (KS) equation and 2D Rayleigh–Bénard convection problems are used to test the accuracy and efficiency of the POD-Galerkin with the RNN memory closure for nonlinear problems.

The following two metrics are used to measure the generalization accuracy of the trained network:

1. The average relative error of the memory terms on the test data set:

$$\varepsilon_1 = \frac{1}{N_{te}} \sum_{i=1}^{N_{te}} \frac{\|w_i^{LSTM} - w_i\|_2}{\|w_i\|_2}, \quad (8.3.1)$$

2. The relative error of the entire memory data set:

$$\varepsilon_2 = \sqrt{\frac{\sum_{i=1}^{N_{te}} \|w_i^{LSTM} - w_i\|_2^2}{\sum_{i=1}^{N_{te}} \|w_i\|_2^2}}, \quad (8.3.2)$$

where N_{te} is the size of the test data set.

The accuracy of the reduced basis solution is measured by the following L_2 norm error:

$$\|\tilde{u} - u\|_{L_2(0,T;L_2)} = \sqrt{\int_0^T \|\tilde{u} - u\|_2^2 dt},$$

where \tilde{u} is the reduced basis solution and u is the high-fidelity solution.

The following notations are used in some plots to distinguish results for different methods:

1. *Full-Order*: The high-fidelity (full-order) solution;
2. *Projection*: Projection of the high-fidelity solution onto the reduced space;
3. *POD-Galerkin*: Reduced basis solution of the POD-Galerkin method;
4. *LSTM*: Reduced basis solution of the POD-Galerkin with the conditioned LSTM memory closure.

8.3.1 3D Stokes

The POD-Galerkin with the RNN memory closure is applied to the model reduction of the blood flow in the human cardiovascular system [209]. The 3D Stokes equations

$$\begin{cases} \frac{\partial}{\partial t} \mathbf{u} = \nu \Delta \mathbf{u} - \nabla p, & t \in [0, 4] \\ \nabla \cdot \mathbf{u} = 0, \\ \mathbf{u}(\partial\Omega_{\text{inlet}}, t) = \mathbf{f}(t), \\ \mathbf{u}(\partial\Omega_{\text{wall}}, t) = 0, \\ \mathbf{u}(\Omega, 0) = 0, \end{cases} \quad (8.3.3)$$

are used to describe the blood flow. The computational domain Ω is shown in Figure 8.3. The velocity is $\mathbf{u} = [u_x, u_y, u_z]^\top$, where u_x, u_y and u_z are velocity components and p is the pressure. The surface of the domain Ω consists of one velocity inlet, two free outlets and the no-slip wall. The boundary condition at the inlet is $\mathbf{f}(t) = [0, 0, f_z(t)]$, where f_z is the time-dependent boundary value of u_z . The function $f_z(t)$ is shown in Figure 8.4. Following the Mori-Zwanzig formalism, each variable of interest needs to be evaluated using a dynamical equation. Therefore, the continuity equation in (8.3.3)

$$\nabla \cdot \mathbf{u} = 0,$$

is replaced by

$$\beta \frac{\partial}{\partial t} p + \nabla \cdot \mathbf{u} = 0,$$

following the artificial compressibility method [188]. β is the compressibility parameter, and is set to $\beta = 10^{-6}$ in this test case.

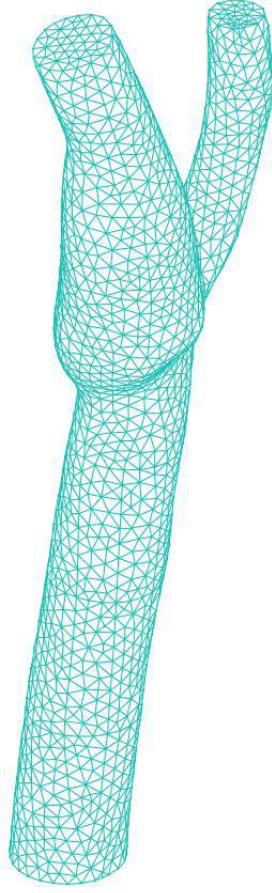


Figure 8.3: ST: 3D carotid bifurcation geometry model and mesh.

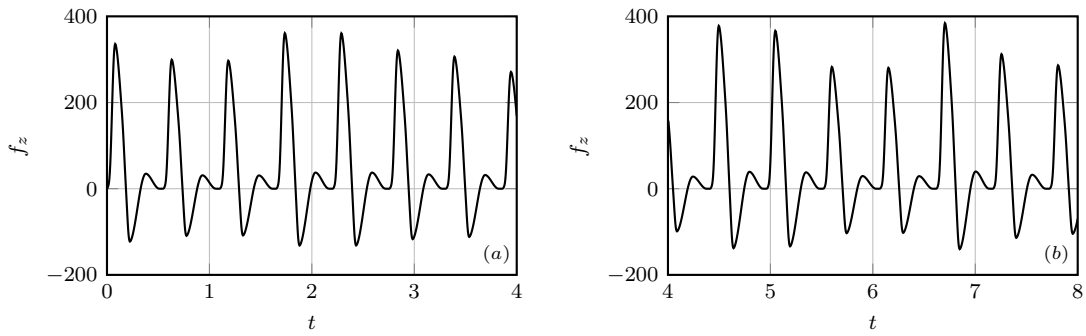


Figure 8.4: ST: Time-dependent velocity in z -direction, at the inlet boundary, for the training (a) and prediction (b) regimes.

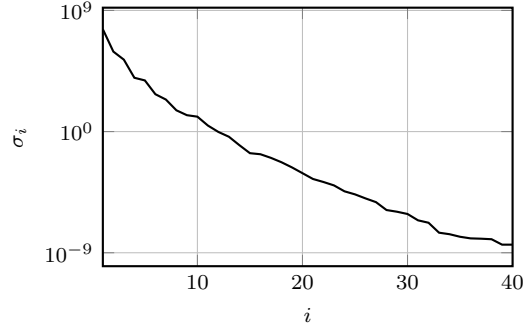


Figure 8.5: ST: Singular value decay of the snapshot matrix S in logarithmic scale. The spectrum shows a very fast decay, which suggests that a reduced basis with less than 20 elements is enough to represent the high-fidelity solution with a reasonable accuracy.

The full-order model is discretized using the finite element method. The Taylor-Hood $P2 - P1$ finite elements are used to satisfy the inf-sup condition, with 20,914 nodes for each velocity component. To obtain a problem in the general dynamical system form in (1.2.1), we condense the mass matrix into a diagonal matrix using mass lumping and multiply the resulting system by the inverse of the mass matrix. The semi-discrete form of (8.3.3) is

$$\begin{cases} \frac{d}{dt} \mathbf{u}_h = A(\mu) \mathbf{u}_h + \mathbf{b}(t), & t \in [0, 4] \\ \mathbf{u}_h(0) = \mathbf{u}_{h,0} \end{cases} \quad (8.3.4)$$

The implicit Euler method is used to integrate (8.3.4) in time, using $N_t = 1000$ time steps. The coefficient ν is the only physical/geometrical parameter of this problem. The range of ν is $[2, 6]$. The snapshots for POD basis generation are collected from the high-fidelity simulations with 9 values of ν that are sampled from a uniform distribution. The leading 100 singular values of the snapshot matrix are plotted in Figure 8.5. The first 16 left singular vectors of the snapshot matrix are selected as the reduced basis functions. The training, validation, and test data sets are generated from the high-fidelity simulation results, with 50 uniformly, 25 uniformly randomly, and 25 uniformly randomly sampled parameter values, respectively.

To obtain an accurate memory model, we train the conditioned LSTMs with 32, 64, and 128 hidden units and 2, 3, \dots , 8 time steps. The optimal length of the input sequence is estimated using the criterion provided in (7.3.45). Ten restarts are performed in each training, with 500 epochs in each restart. In each epoch, the training data is shuffled and divided into mini-batches of size 1000. The learning rate is 0.005. The coefficient of the L_2 regularization is 10^{-9} .

For model selection, we show the test errors of the trained networks with a different number of time steps and hidden units, and the estimated memory length $\bar{\tau}$ in Figure 8.6. The accuracy comparison in Figure 8.6 shows that the models with more hidden units are more accurate but also more costly. The models with 4 time steps are the most accurate among the models of which the memory lengths are inside the estimated range. Therefore, the network with 128 hidden units and 4 time steps is selected as the memory model for further test. The test errors of the trained networks with 128 hidden units and 4 time steps, using data sets generated from high-fidelity simulation results with 10, 20, 40, 50, 100, and 200 uniformly randomly sampled parameter values are shown in Figure 8.7, to study the convergence property of the conditioned RNN with respect to the number of training points. The results show that the test error of the conditioned LSTM network decreases with increasing number of the parameter values in the training data set

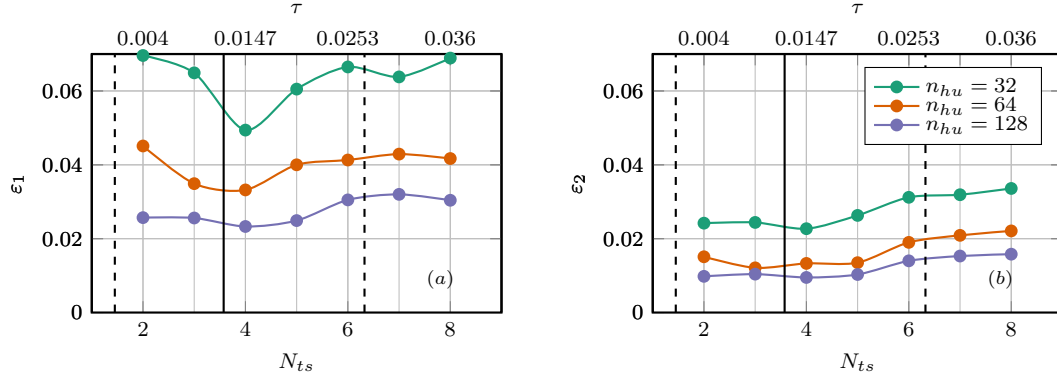


Figure 8.6: ST: Test error of the trained networks for the Stokes problem in the metrics defined in (8.3.1) in (a) and in (8.3.2) in (b). The estimated mean (solid vertical line), minimum and maximum (dashed vertical lines) of the estimated memory length τ over the parameter set are also shown in the plot.

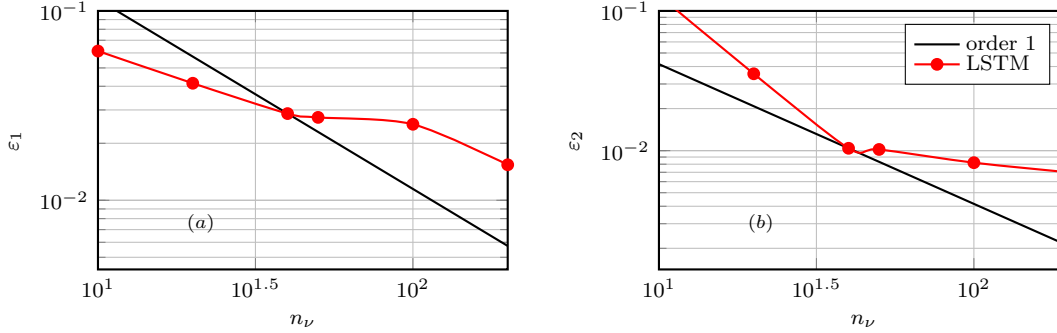


Figure 8.7: ST: Test errors of the memory model in the metrics defined in (8.3.1) in (a) and in (8.3.2) in (b), with respect to different number of parameter values in the training data sets.

without a certain convergence rate.

The POD-Galerkin with the memory model is tested with the 25 random values of ν of the test data set. The simulations are performed using the IMEX-Euler time integration until $t = 8$, which is beyond the range of the training data $[0, 4]$, to test the prediction capability of the reduced-order model.

The energy contribution of the closure to the reduced-order system is shown in Figure 8.8 to provide an intuition of the accuracy of the memory model. It is observed in Figure 8.8 that the conditioned LSTM network accurately models the memory closure.

The error between the reduced solutions of the original POD-Galerkin and the POD-Galerkin with memory for 4 different parameter values is shown in Figure 8.9. The results show that the error of the POD-Galerkin with memory is 2 to 3 orders of magnitude smaller than the original POD-Galerkin, which demonstrates that the LSTM memory model can significantly improve the accuracy of the POD-Galerkin method.

It is also observed in the numerical test that, for some small values of ν , the POD-Galerkin is unstable. No instability was recorded for the reduced model with LSTM closure. The solutions of $\nu = 3$ are shown in Figure 8.10 to give an example of the instability of POD-Galerkin. The results in Figure 8.10 show that the solution of the POD-Galerkin with memory closure is quite close to

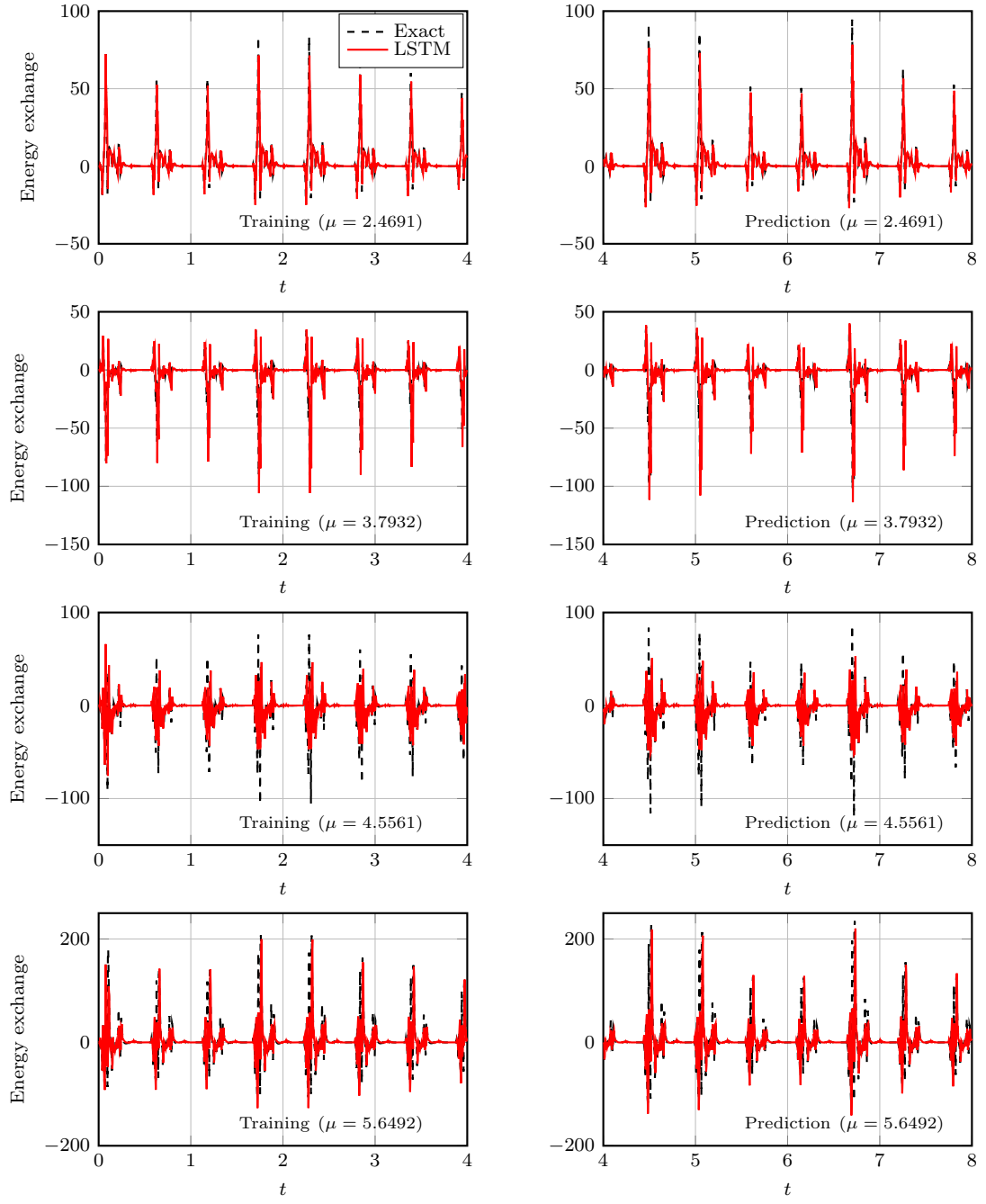


Figure 8.8: ST: Evolution of the energy exchange term $2z^\top w$ of the reduced-order models of the Stokes problem for different parameter values in the training and prediction regimes.

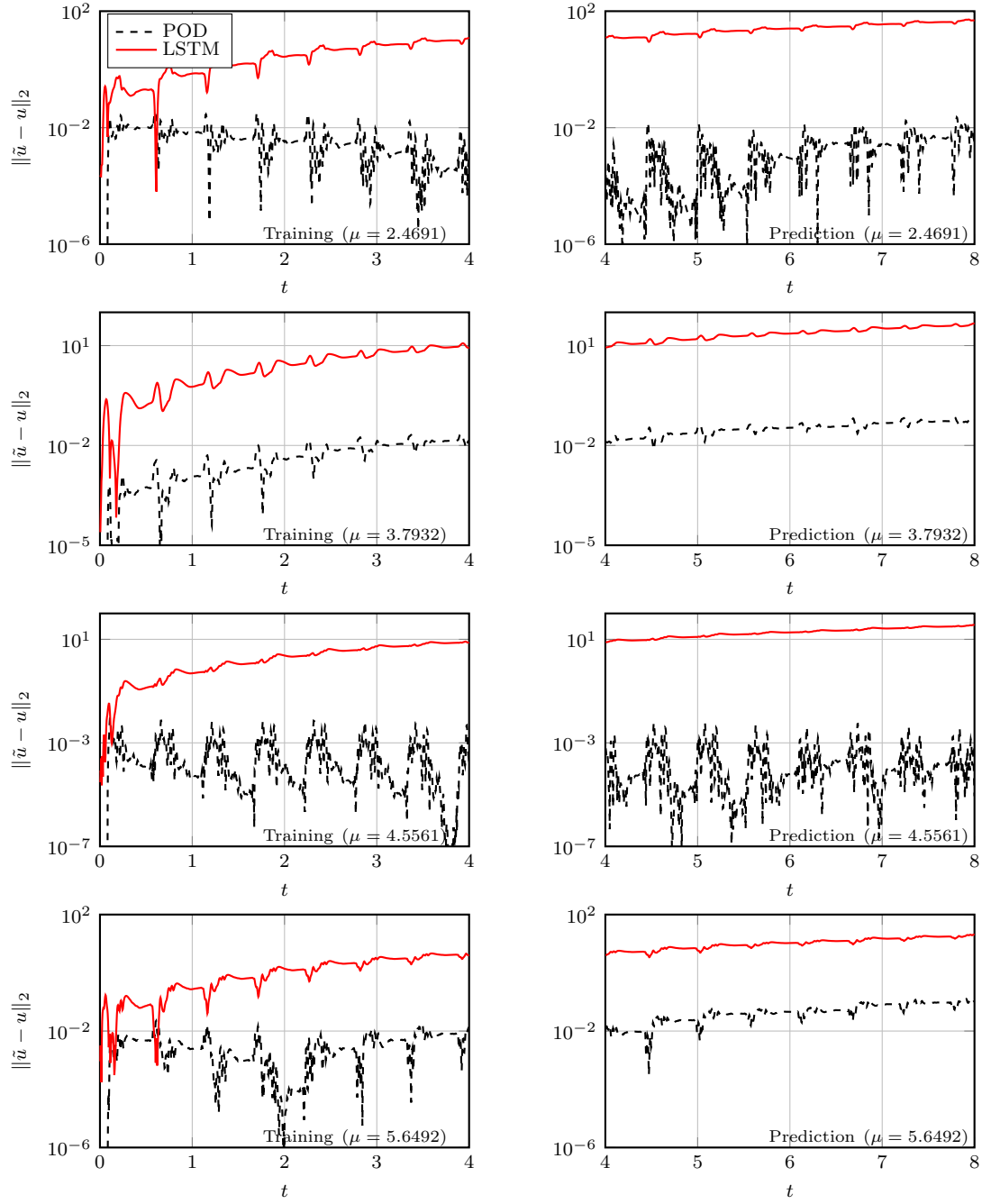


Figure 8.9: ST: Evolution of the errors of the reduced order models for the Stokes problem for different parameter values in the training and prediction regimes.

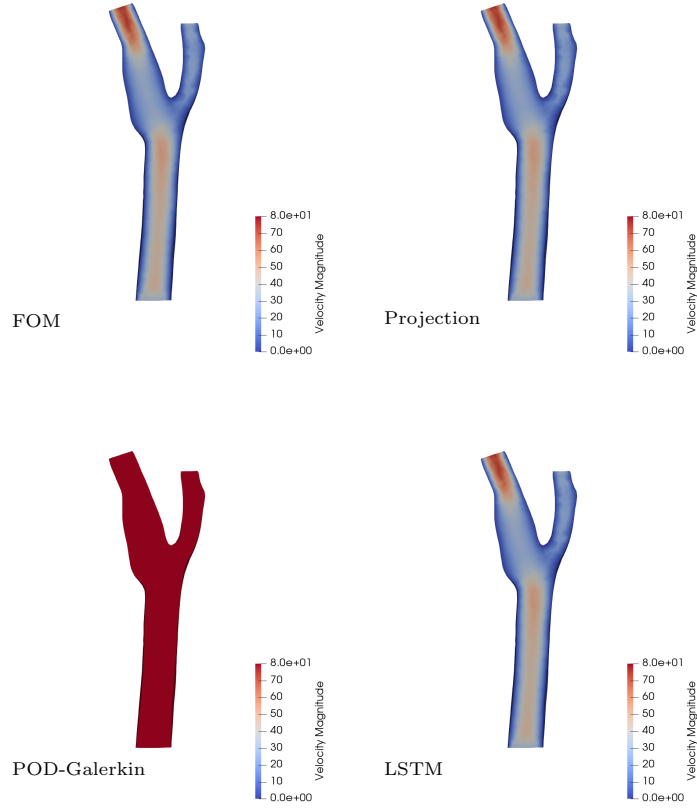


Figure 8.10: ST: A sectional view of the velocity magnitude contours of the reduced-order solutions at $t = 2.4$ for $\nu = 3$ for the FOM solution, the projection of the FOM solution onto the reduced space, the solution to the POD-Galerkin model, and the solution to the POD-Galerkin model with the LSTM correction.

the high-fidelity solution, while the solution of the original POD-Galerkin diverges. Hence, the memory closure improves the accuracy and the stability of the reduced-order model.

The error-cost plot of the reduced-order models is shown in Figure 8.11 for efficiency comparison, where we consider only parameter values for which the POD-Galerkin model without memory is stable. We can observe that the original POD-Galerkin model can reach a certain accuracy level using less computational time and is thus more efficient. As discussed in Section 8.2.2, we do not expect efficiency improvement for linear problems, for the cost of evaluating the approximation of the memory term.

8.3.2 Kuramoto–Sivashinsky equation

The fourth-order one-dimensional Kuramoto–Sivashinsky (KS) equation is used to test the memory modeling capability of the conditioned LSTM network for highly nonlinear problems.

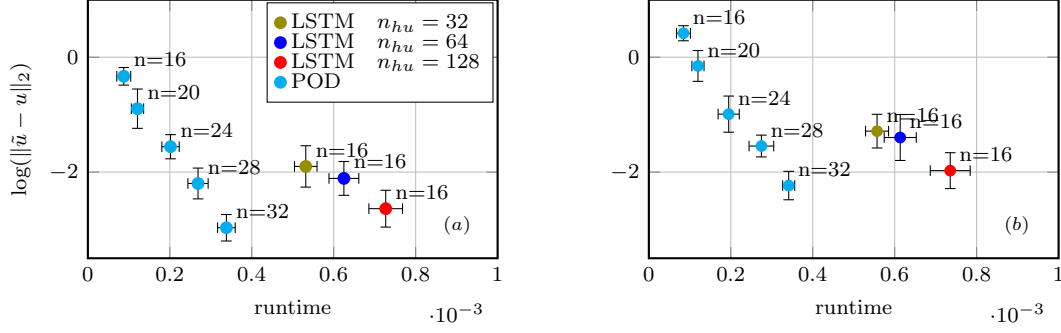


Figure 8.11: ST: Error-cost plot of the reduced-order models for the Stokes problem for the training (a) and prediction (b) intervals.

The parametrized KS equation is

$$\begin{cases} \frac{\partial}{\partial t} u = -\frac{1}{2} \partial_x \cdot u^2 - \partial_{xx} u - \nu \partial_x^4 u, \\ u(x + L, t) = u(x, t), \\ u(x, 0) = g(x), \end{cases} \quad (8.3.5)$$

where L is the spatial period, $g(x)$ is the initial datum, and ν is the parameter. In our test, we take $L = 22$ and $g(x)$ is obtained by the inverse Fourier transform from a series of Fourier modes of which the first 4 mode coefficients are 0.06. The computational domain is partitioned into $N_h = 1024$ elements. The full-order solver utilizes a finite-element discretization and a second-order implicit trapezoidal time integration. The solution is updated until $t = 50$ using a time step size $\Delta t = 0.025$.

The coefficient ν of the fourth-order viscosity term is the only parameter in this problem. Following Lu et al. [164], the dissipative term $\Delta^2 u$ provides damping at small scales. Therefore, the smaller the ν , the less dissipative the system. In our test, we see that very small ν yields a chaotic or quasi-chaotic system, making the model reduction challenging.

We set the range of the parameter ν as $[0.3, 1.5]$. The snapshots for the POD basis generation are collected from the high-fidelity simulations with 25 values of ν that are uniformly distributed in the log space, which means that more data points are sampled for small parameter values. The basis is extracted from the snapshots via POD. We chose 25 left singular vectors of the snapshot matrix as the reduced basis functions.

The training, validation, and test data sets are generated from high-fidelity simulation results, with 124 equidistant, 62 uniformly randomly, and 61 uniformly randomly sampled parameter values in the log space, respectively.

We train the conditioned LSTMs with 32, 64, and 128 hidden units and 2, 3, 4, 5, 6, and 10 time steps to get an accurate memory model. Ten restarts are performed in each training, with 500 epochs in each restart. In each epoch, the training data is shuffled and divided into mini-batches of a size of 1000. The learning rate is 0.005. The coefficient of the L_2 regularization is 10^{-9} .

The relative errors of the trained networks on the test data set and the estimated memory length are shown in Figure 8.12. We observe that the networks with 128 hidden units are the most accurate. Furthermore, the models with 3, 4, and 5 time steps are the most accurate among the models whose memory length is inside the estimated range. However, we see in Figure 8.12 that the errors of the networks are quite large. To understand the cause of this large error, we show the test errors of the networks with 128 hidden units with respect to different parameter

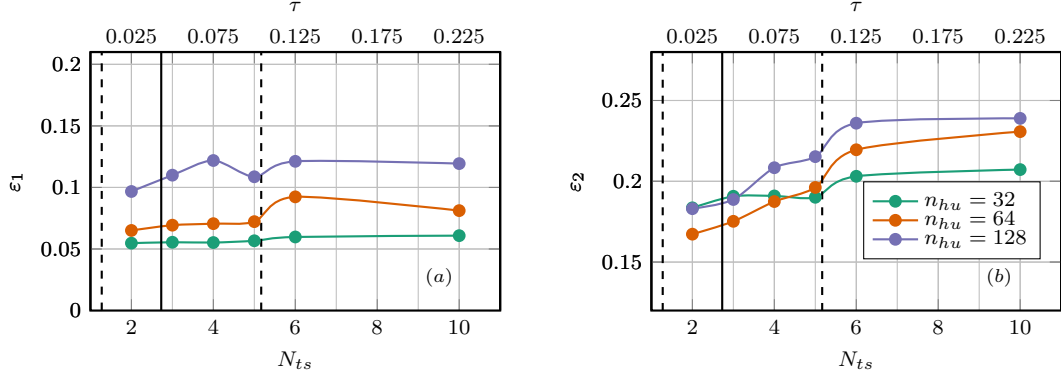


Figure 8.12: KS: Test error of the trained networks for the KS equation in the metrics defined in (8.3.1) (in (a)) and in (8.3.2) (in (b)). The estimated mean (solid vertical line), minimum and maximum (dashed vertical lines) of the estimated memory length τ over the parameter set are also shown in the plot.

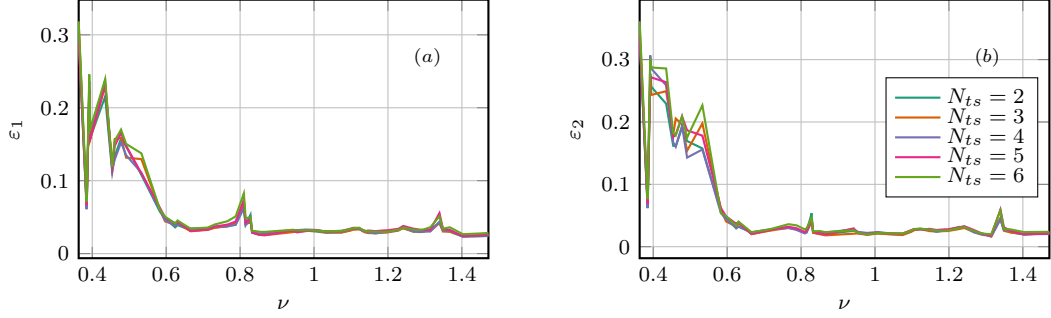


Figure 8.13: KS: Test error plots of the trained networks with 128 hidden units for different parameter values in the metrics defined in (8.3.1) (in (a)) and in (8.3.2) (in (b)).

values in Figure 8.13. The results in Figure 8.13 show that the networks are very accurate for $\nu \in [0.6, 1.5]$, while less accurate for $\nu \in [0.3, 0.6]$, which is reasonable since the small values of ν correspond to quasi-chaotic dynamics that are very difficult to capture accurately. Furthermore, for the well-modeled parameter range, the networks with different number of time steps have very similar accuracy. Therefore, we chose the network with 128 hidden units and 4 time steps as the memory model. The POD-Galerkin with the selected memory model is tested using the physical parameter sampling of the test data set which includes 61 random values of ν . The simulations are performed using the IMEX-Trapezoidal time integration in (8.2.8) until $t = 100$, which is beyond the time range of the training data $[0, 50]$, to test the prediction capability of the reduced-order models.

The energy contribution of the closure to the reduced-order system is shown in Figure 8.14 to provide an intuition of the accuracy of the memory model. It is observed in Figure 8.14 that the conditioned LSTM network accurately models the memory closure.

The reduced solutions of the original POD-Galerkin and the POD-Galerkin with memory with 4 different parameter values are shown in Figure 8.15-8.18. For the small parameter value ($\nu = 0.34756$) case, both the reduced-order models with/without memory models fail to follow the high-fidelity model's fast dynamics. For the cases with large parameter values, the results show that the reduced basis solutions computed by the POD-Galerkin with memory are much closer to

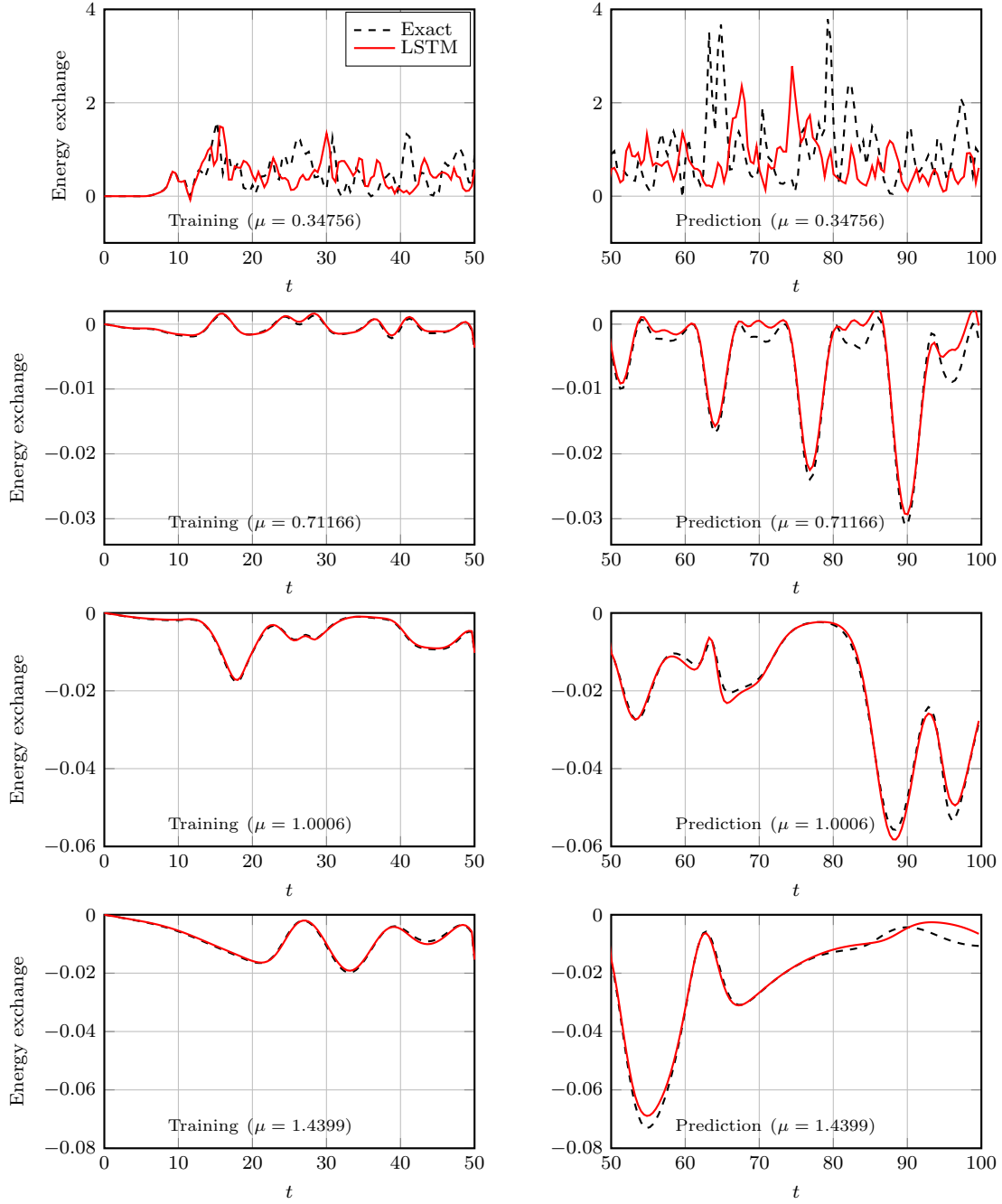


Figure 8.14: KS: Evolution of the energy exchange term $2z^T w$ of the reduced-order models of the KS problem for different parameter values in the training and prediction regimes.

the high-fidelity solutions than the original POD-Galerkin. We note that, for certain parameter values, the POD-Galerkin reduced-order model can follow the high-fidelity solution only for a short time interval, as shown in Figure 8.18. At the same time, the reduced-order model with memory closure remains accurate for a much longer time and makes accurate predictions beyond the training time interval.

For a global view of the dynamics, the contours of solutions for $\nu = 0.64424$ on the $x - t$ plane are shown in Figure 8.19. The results in Figure 8.19 show that the POD-Galerkin with memory closure provides accurate solutions in both the training and prediction intervals, while the original POD-Galerkin model loses the dynamics after a certain time.

We show the error-cost plot of the reduced-order models in Figure 8.20 for efficiency comparison. The POD-Galerkin model with memory has 10 to 100 times smaller errors than the original POD-Galerkin model, with slightly more computational time, and is thus much more efficient.

8.3.3 Rayleigh-Bénard convection

The two-dimensional Rayleigh-Bénard convection problem is used as the last case to test the capability of memory modeling of the conditioned LSTM network for multi-dimensional nonlinear problems. The non-dimensionalized governing equations are

$$\begin{cases} \nabla \cdot \mathbf{u} = 0, \\ \frac{\partial}{\partial t} \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \sqrt{\frac{Pr}{Ra}} \Delta \mathbf{u} + T \mathbf{e}_y, \\ \frac{\partial}{\partial t} T + \mathbf{u} \cdot \nabla T = \frac{1}{\sqrt{Pr Ra}} \Delta T, \end{cases} \quad (8.3.6)$$

where \mathbf{u} , T and p are the dimensionless velocity, temperature and pressure, respectively. The Rayleigh number (Ra) and the Prandtl number (Pr) are the dimensionless quantities that control the flow. The simulation setup, including the computational domain and the boundary conditions, is shown in Figure 8.21. The high-fidelity simulations are performed until $t = 50$ on a triangular mesh with 152,888 nodes, using a finite-element space discretization and the implicit trapezoidal time integration with a time step size $\Delta t = 0.01$. The solution of a low Rayleigh number case $Ra = 33019.2725$, $Pr = 0.85$ at $t = 50$, starting from a stationary flow with linear temperature distribution between the hot and cold plates, is used as the initial condition for the high-fidelity simulations.

We select Ra and Pr as the two physical parameters for model reduction of this problem. The parameter domain of the problem is $(Ra, Pr) \in [5 \times 10^6, 1.5 \times 10^7] \times [0.85, 0.95]$. The snapshots for POD basis generation are collected from the high-fidelity simulations with 36 parameter values generated via the Latin hypercube sampling. The reduced basis functions are extracted from the snapshots using the randomized SVD [122]. The singular value decay is shown in Figure 8.22, and it is observed that the singular values decay slowly, implying that a large number of reduced basis vectors is necessary to recover the main dynamics, making the model reduction of this problem challenging. We select 24 left singular vectors of the snapshot matrix as the reduced basis functions.

The training data is generated using the same high-fidelity simulation results used for reduced basis generation. The validation and test data sets are obtained from high-fidelity simulations with 18 randomly sampled parameter values.

To obtain an accurate memory model, we train the conditioned LSTMs with 128 hidden units and $2, 3, \dots, 8$ time steps. Each network is optimized by 10 restarts, with 1200 epochs in each restart. In each epoch, the training data is shuffled and divided into mini-batches of size 1000.

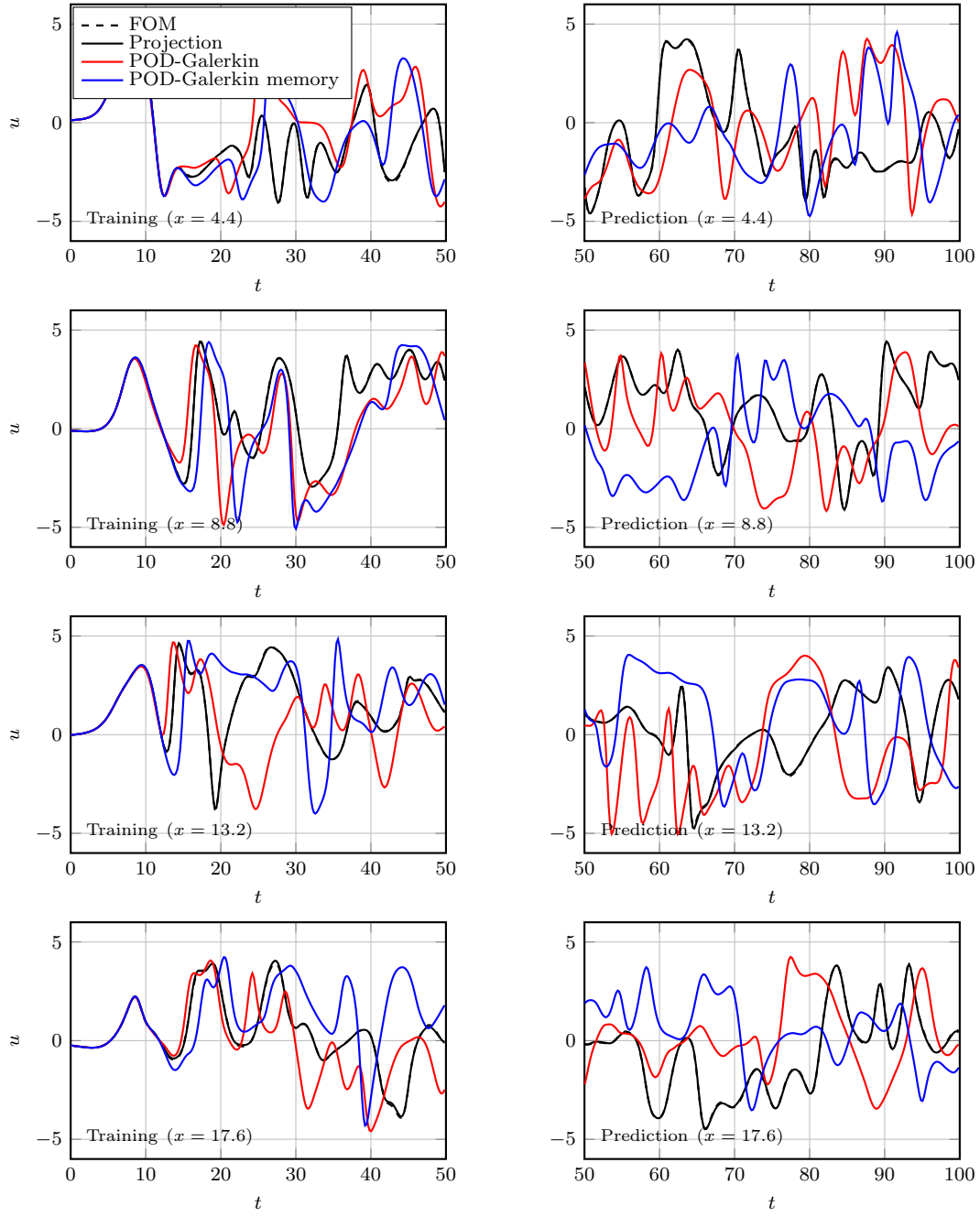


Figure 8.15: KS: Numerical solutions for parameter $\nu = 0.34756$. ----- High-fidelity; — Projection of high-fidelity; — POD-Galerkin; — POD-Galerkin with memory.

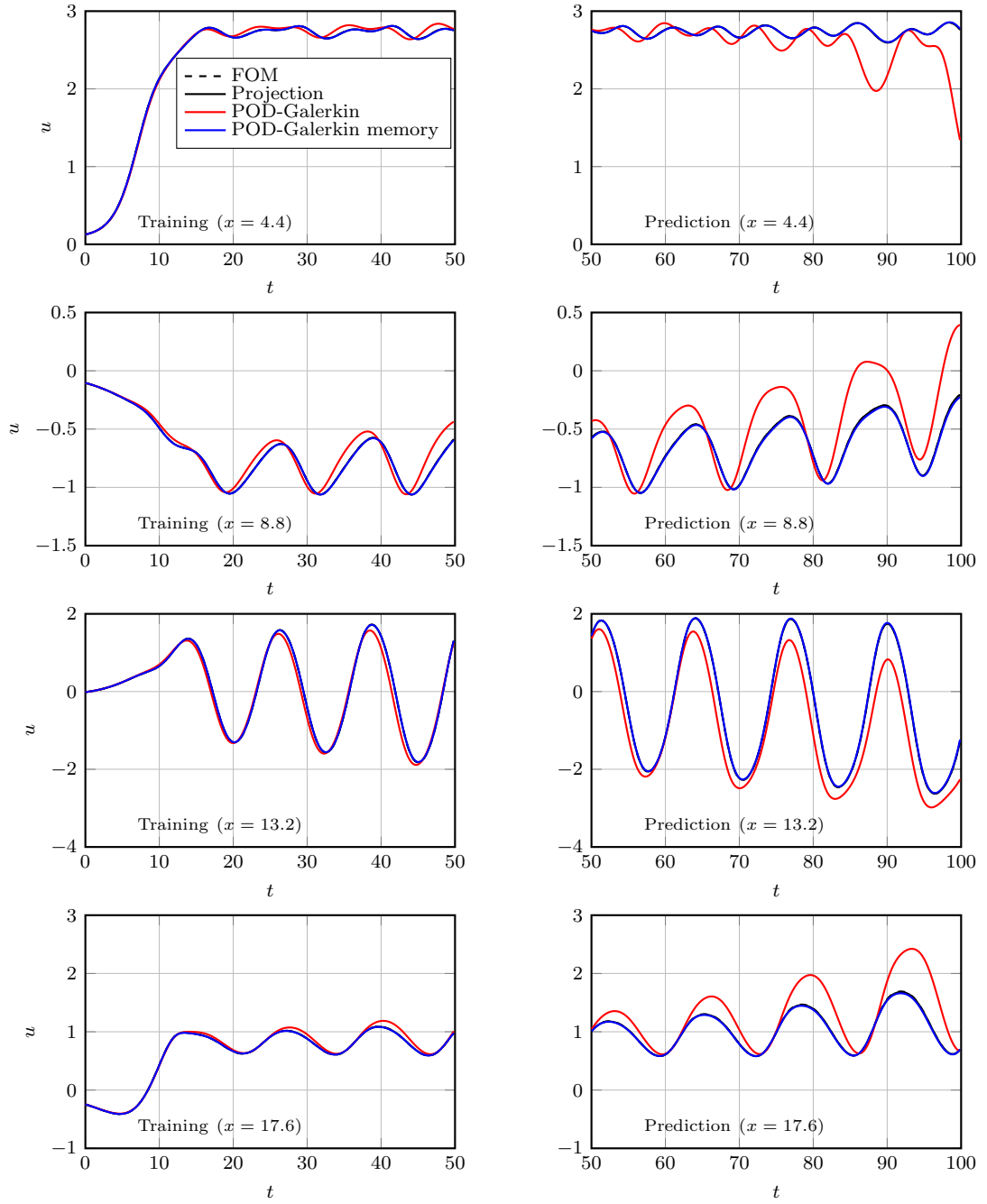


Figure 8.16: KS: Numerical solutions for parameter $\nu = 0.71166$. ----- High-fidelity; — Projection of high-fidelity; — POD-Galerkin; — POD-Galerkin with memory.

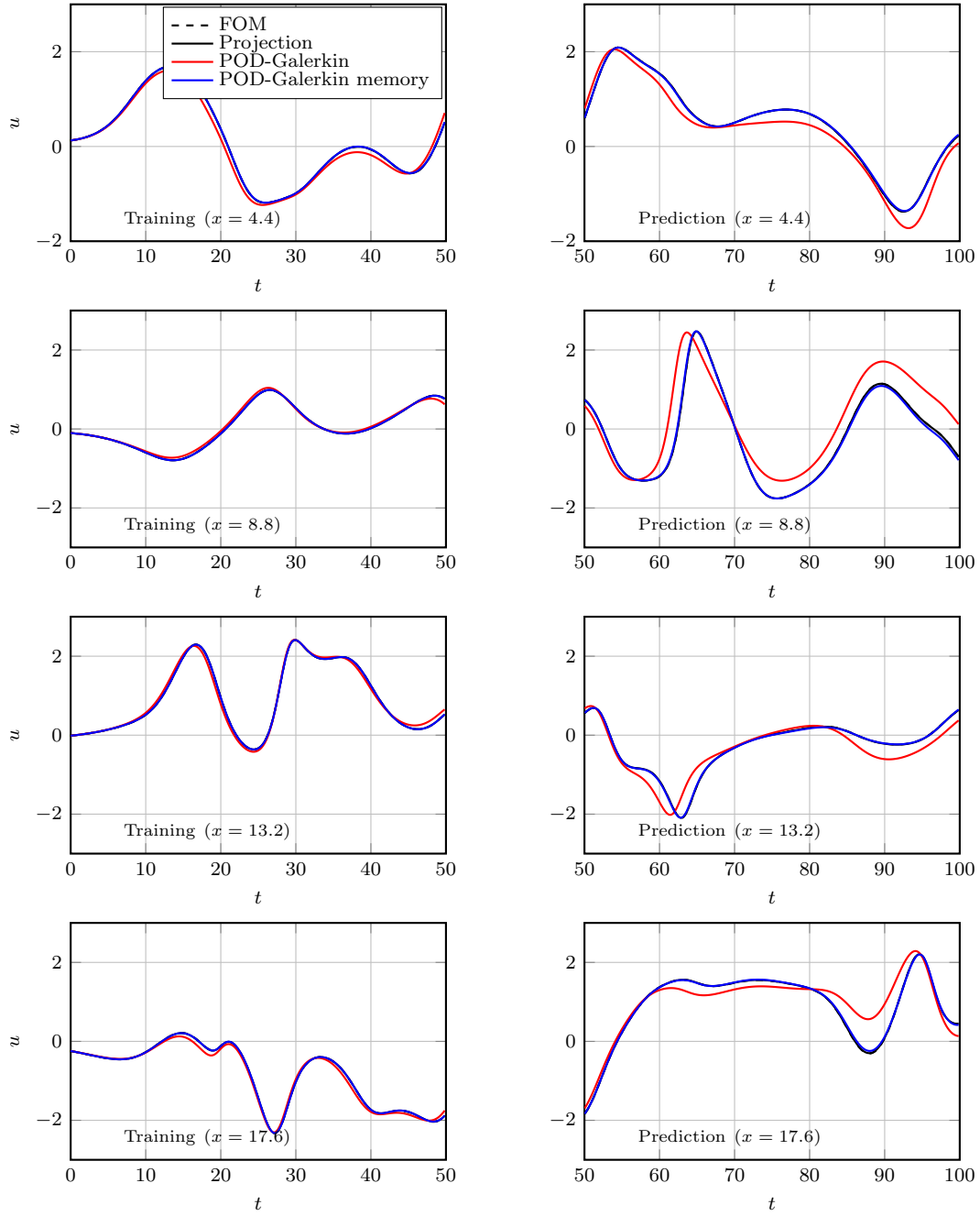


Figure 8.17: KS: Numerical solutions for parameter $\nu = 1.0006$. ----- High-fidelity; — Projection of high-fidelity; — POD-Galerkin; — POD-Galerkin with memory.

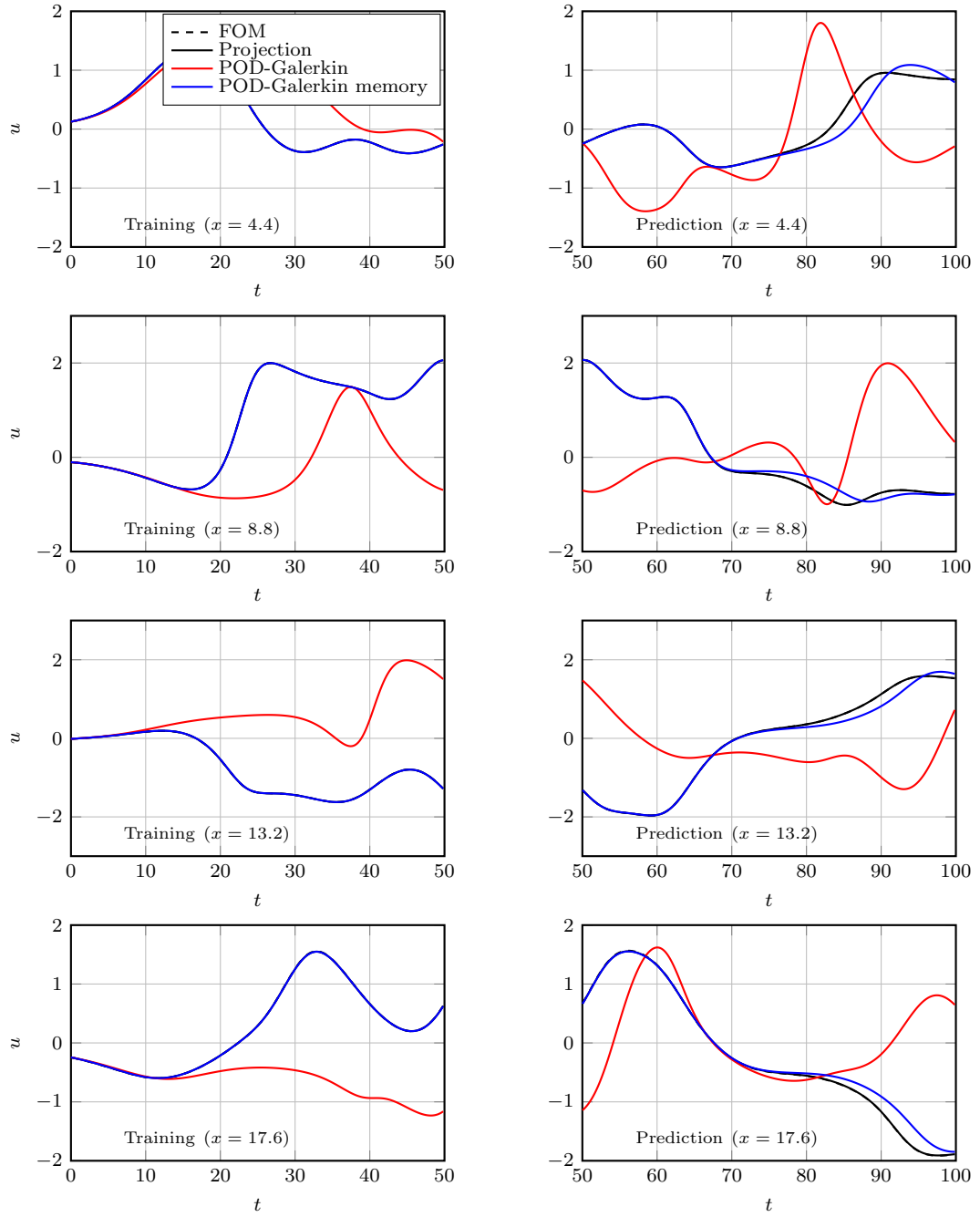


Figure 8.18: KS: Numerical solutions for parameter $\nu = 1.4399$. - - - High-fidelity; — Projection of high-fidelity; — POD-Galerkin; — POD-Galerkin with memory.

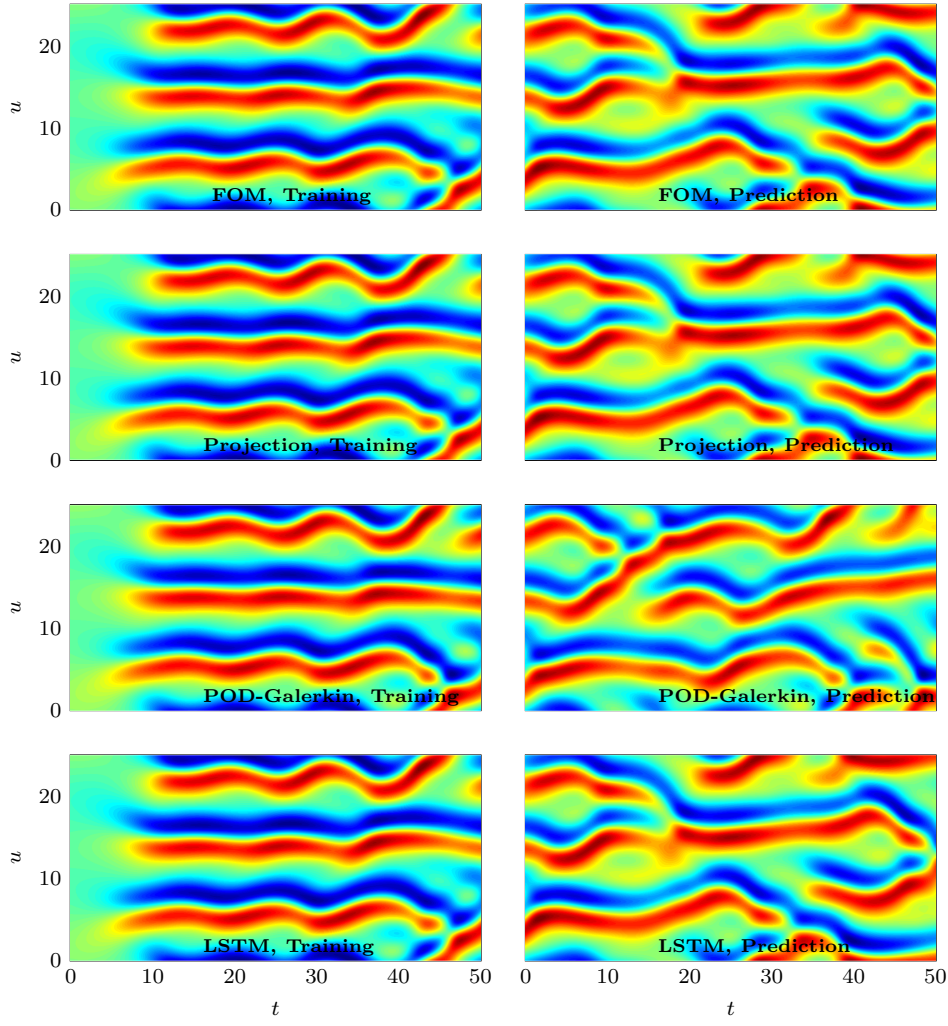


Figure 8.19: KS: Evolution of the numerical solutions of the KS equation with $\eta = 0.64424$.

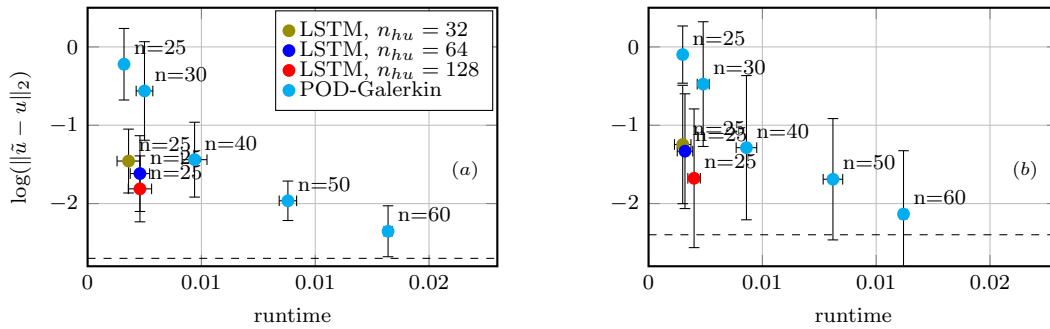


Figure 8.20: KS: Error-cost plot of the reduced-order models for the KS problem for the training (a) and prediction (b) intervals. The horizontal dashed line represents the projection error.

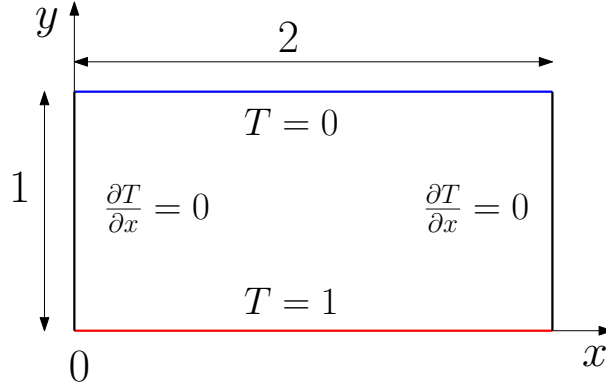


Figure 8.21: RB: Simulation setup of the Rayleigh-Bénard convection.

The learning rate is 0.005. The coefficient λ of the L_2 regularization is 10^{-9} .

The relative errors of the trained networks on the test data set and the estimated memory length are shown in Figure 8.23. The network with 3 time steps is the most accurate among the models of which the memory length is inside the estimated range and is thus selected as the memory model.

The POD-Galerkin with the selected memory model is tested using the physical parameter sampling of the test data set that includes 61 random values of ν . The simulations are performed using the IMEX-Trapezoidal time integration until $t = 100$, which is beyond the time range of the training data $[0, 50]$, to test the prediction capability of the reduced-order model.

The energy contribution of the closure to the reduced-order system is shown in Figure 8.24 to provide an intuition of the accuracy of the memory model. It is observed in Figure 8.24 that the conditioned LSTM network accurately captures the memory closure.

The reduced solutions of the original POD-Galerkin and the POD-Galerkin with memory for $(Ra, Pr) = (14024512.0002, 0.86976)$ are shown in Figure 8.25. The results show that the reduced solutions computed by the POD-Galerkin with memory are much closer to the high-fidelity solutions. Furthermore, the results also show that the POD-Galerkin model with memory closure can make accurate predictions, even in the case that the reduced basis can not accurately represent the dynamics of the high-fidelity solution.

For a global view of the dynamics, the contours of solutions for $(Ra, Pr) = (14024512.0002, 0.86976)$ are shown in Figure 8.26. The results show that the POD-Galerkin with memory closure can provide much more accurate solutions in both the training and prediction intervals than the original POD-Galerkin model.

We show the error-cost plot of the reduced-order models in Figure 8.27 for efficiency comparison. We can see from Figure 8.27 that, the POD-Galerkin model with memory has 5 to 10 times smaller error than the original POD-Galerkin model, at only slightly more computational cost, and is thus much more efficient. We highlight the fact that the error of the POD-Galerkin model with memory is very close to the projection error, which means that the conditioned LSTM memory model is very accurate and the reduced basis solution evolves with almost exact dynamics.

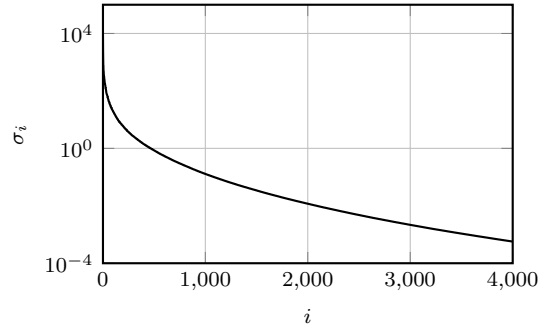


Figure 8.22: RB: Singular values of the Rayleigh-Bénard convection problem.

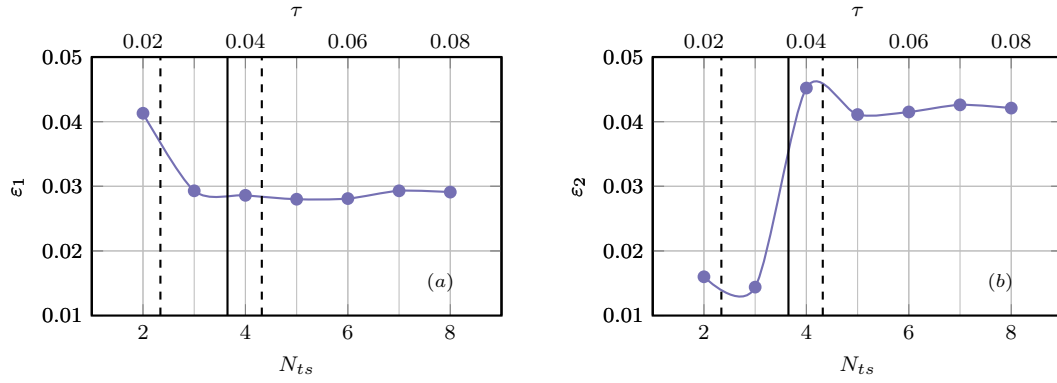


Figure 8.23: RB: Test error of the trained networks for the Rayleigh-Bénard convection in the metrics defined in (8.3.1) (in (a)) and in (8.3.2) (in (b)). The estimated mean (solid vertical line), minimum and maximum (dashed vertical lines) of the estimated memory length τ over the parameter set are also shown in the plot.

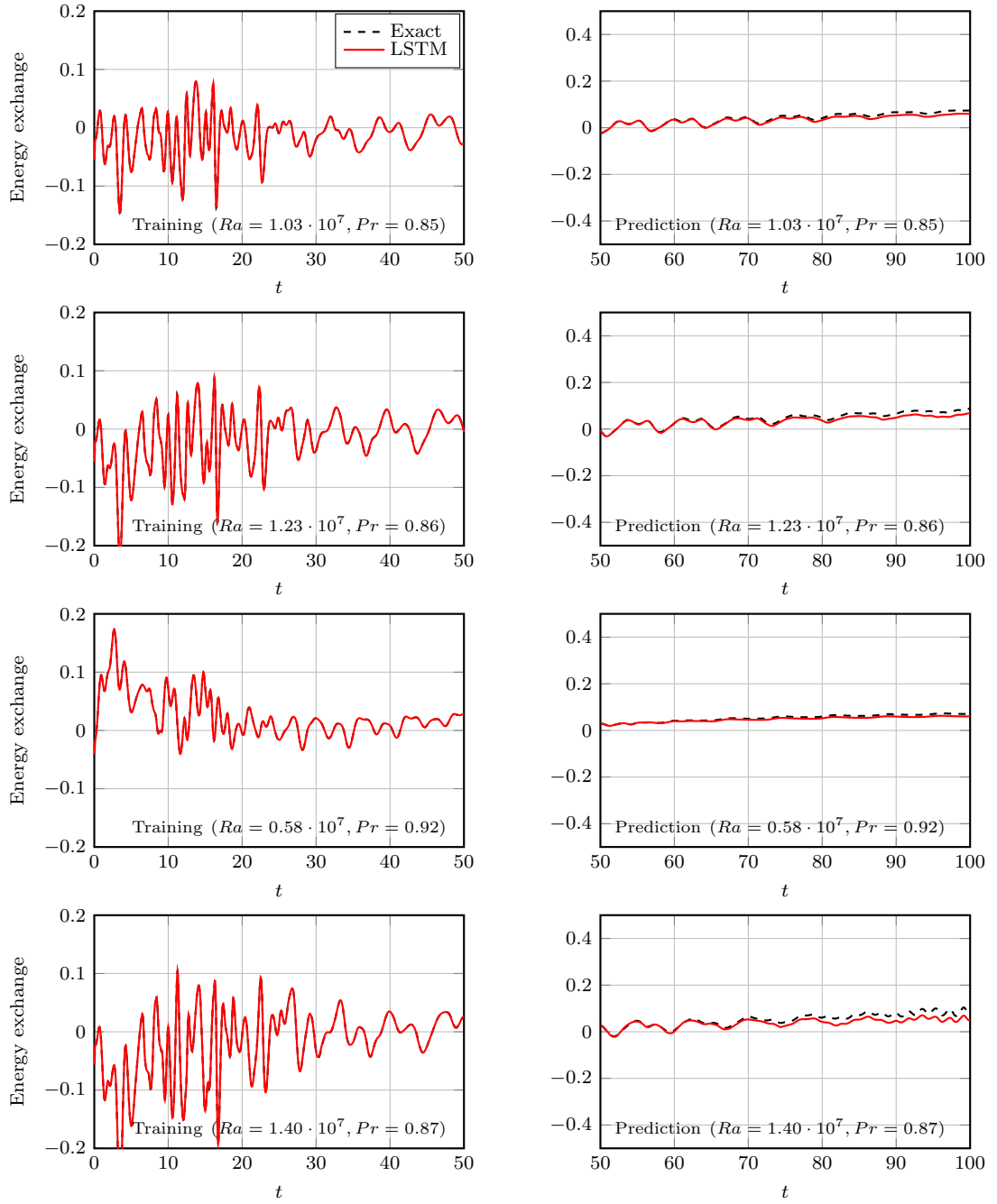


Figure 8.24: RB: Evolution of the energy exchange term $2z^T w$ of the reduced-order models of the Rayleigh-Bénard problem for different parameter values in the training and prediction regimes.

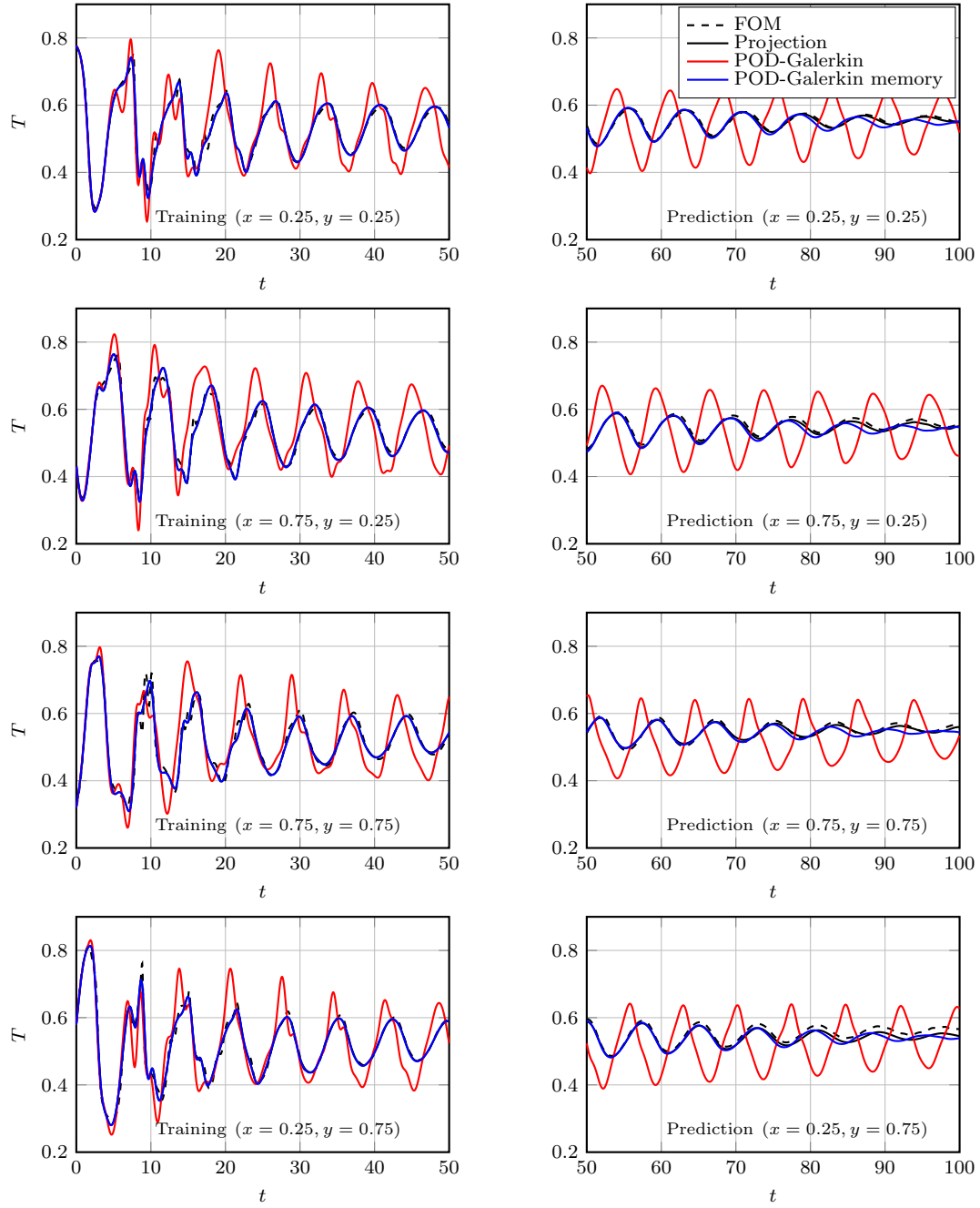


Figure 8.25: RB: Temperature evolution of Rayleigh-Bénard convection problem at four points, for $Ra = 14024512.0002$, $Pr = 0.86976$. ----- High-fidelity; — Projection of high-fidelity; — POD-Galerkin; — POD-Galerkin with memory.

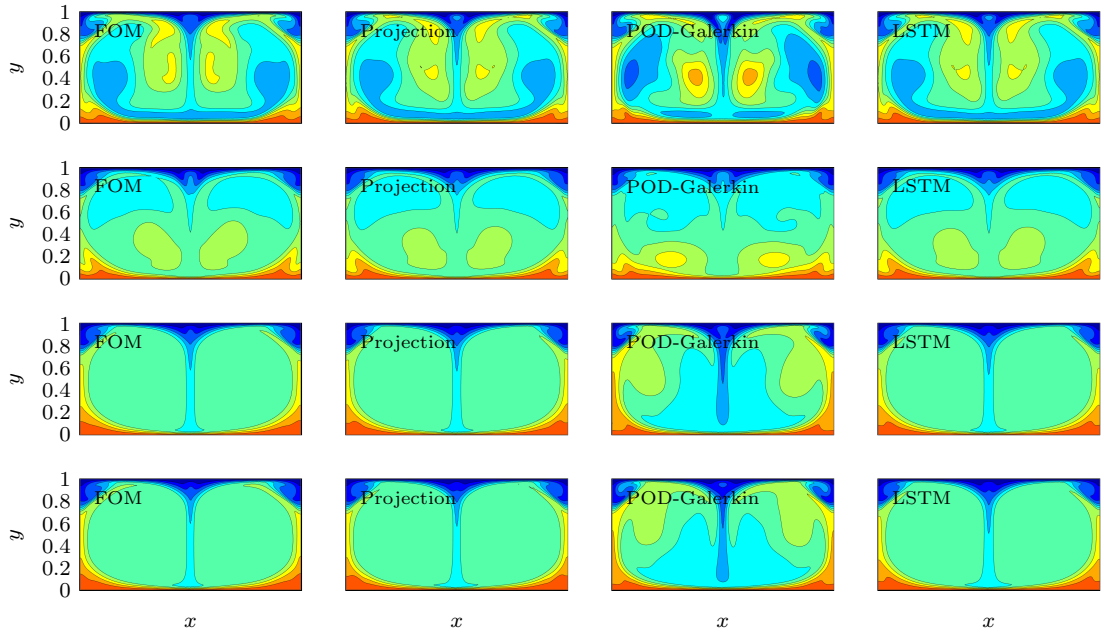


Figure 8.26: RB: Contour of Rayleigh-Bénard convection problem at $t = 10, 25, 70$, and 85 , for $Ra = 14024512.0002$, $Pr = 0.86976$.

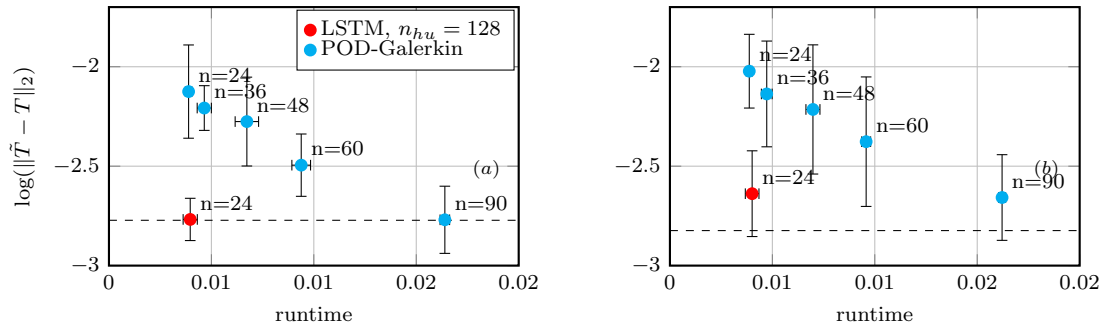


Figure 8.27: RB: Error-cost plot of the reduced-order models of the Rayleigh-Bénard convection for the training (a) and prediction (b) intervals. The horizontal dashed line represents the projection error.

9 Conclusion of Part II

In Chapter 7, we have seen that the Mori-Zwanzig formalism is a consistent framework for closing reduced models. The generalized Langevin equation obtained by rewriting the FOM system allows for greater insight into the effect of separation between resolved and unresolved scales. In particular, by studying the dynamics of the orthogonal dynamics equation, we noticed that there is a class of problems for which this interaction, in the case of approximation via POD-Galerkin, manifests itself as a memory integral term characterized by limited support. The appeal of this approach is that the closure term is derived by exploiting information directly from the FOM, and is not based solely on heuristic reasoning. Numerical examples demonstrate the method's robustness and superior accuracy in the cases considered, while also predicting the energy contribution of the unresolved term.

In Chapter 8, we propose an RNN closure for parametric POD-Galerkin ROM. A conditioned LSTM network is used to predict the memory integral that accounts for the impact of the unresolved scales on the resolved scales, given the physical/geometrical parameter values and the short time history of the resolved scales as inputs. The RNN closure is embedded into the POD-Galerkin model in the framework of implicit-explicit (IMEX) Runge-Kutta time integration, in which the RNN memory term is computed only once in each time step or inner iteration step, resulting in an efficient reduced-order model. Numerical results demonstrate that the POD-Galerkin ROM with the RNN closure is much more efficient than its original scheme for nonlinear problems.

We have seen that MOR is a valuable tool for creating low-dimensional models from data generated from computationally expensive FOM simulations. However, for complex problems, a limited number of modes is not sufficient to capture all the features of the relevant dynamics, resulting in poor accuracy or instability. The Mori-Zwanzig formalism provides a valuable mathematical framework for developing closure models that guarantee a good representation of the dynamics of the solved part of the problem, as an alternative to traditional methods that only exploit physical insights of the FOM dynamics. Although the use of data-driven approximations in this framework ensures further improvements in accuracy, some issues are still open, including extrapolation beyond the training interval and the stability and boundedness of the solution. We believe that a possible solution to these issues is to combine the best of the approaches mentioned above, i.e., mathematical frameworks, physics-based modeling, and data-driven approximations. For example, although closure via LSTM allows the trajectory of the FOM solution to be followed over long

Conclusion of Part II

time intervals, better results could be obtained by incorporating conservation laws and physical constraints in the reduced dynamics and closure in the spirit of Part I of this thesis. Similarly, this idea would allow the stability of the reduced solution to be enforced by construction.

Bibliography

- [1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.
- [2] Ralph Abraham and Jerrold E Marsden. *Foundations of mechanics*. Number 364. American Mathematical Soc., 2008.
- [3] Babak Maboudi Afkham and Jan S Hesthaven. Structure preserving model reduction of parametric hamiltonian systems. *SIAM Journal on Scientific Computing*, 39(6):A2616–A2644, 2017.
- [4] Babak Maboudi Afkham, Nicolo Ripamonti, Qian Wang, and Jan S Hesthaven. Conservative model order reduction for fluid flow. In *Quantification of Uncertainty: Improving Efficiency and Technology*, pages 67–99. Springer, 2020.
- [5] Shady E Ahmed, Suraj Pawar, Omer San, Adil Rasheed, Traian Iliescu, and Bernd R Noack. On closures for reduced order models—a spectrum of first-principle to machine-learned avenues. *Physics of Fluids*, 33(9):091301, 2021.
- [6] Nissrine Akkari, Fabien Casenave, and Vincent Moureau. Time stable reduced order modeling by an enhanced reduced order basis of the turbulent and incompressible 3d navier–stokes equations. *Mathematical and computational applications*, 24(2):45, 2019.
- [7] Alessandro Alla and J Nathan Kutz. Nonlinear model order reduction via dynamic mode decomposition. *SIAM Journal on Scientific Computing*, 39(5):B778–B796, 2017.
- [8] Bo O Almroth, Perry Stern, and Frank A Brogan. Automatic choice of global shape functions in structural analysis. *Aiaa Journal*, 16(5):525–528, 1978.
- [9] M Amabili, Abhijit Sarkar, and MP Paidoussis. Reduced-order models for nonlinear vibrations of cylindrical shells via the proper orthogonal decomposition method. *Journal of Fluids and Structures*, 18(2):227–250, 2003.
- [10] D Anderson, R Fedele, and M Lisak. A tutorial presentation of the two stream instability and landau damping. *American Journal of Physics*, 69(12):1262–1266, 2001.
- [11] Harbir Antil, Scott E Field, Frank Herrmann, Ricardo H Nochetto, and Manuel Tiglio. Two-step greedy algorithm for reduced order quadratures. *Journal of Scientific Computing*, 57(3):604–637, 2013.

- [12] Shiri Artstein-Avidan, Roman Karasev, and Yaron Ostrover. From symplectic measurements to the mahler conjecture. *Duke Mathematical Journal*, 163(11):2003–2022, 2014.
- [13] Patricia Astrid, Siep Weiland, Karen Willcox, and Ton Backx. Missing point estimation in models described by proper orthogonal decomposition. *IEEE Transactions on Automatic Control*, 53(10):2237–2251, 2008.
- [14] Nadine Aubry. On the hidden beauty of the proper orthogonal decomposition. *Theoretical and Computational Fluid Dynamics*, 2(5):339–352, 1991.
- [15] Nadine Aubry, Philip Holmes, John L Lumley, and Emily Stone. The dynamics of coherent structures in the wall region of a turbulent boundary layer. *Journal of fluid Mechanics*, 192:115–173, 1988.
- [16] C Audouze, F De Vuyst, and PB Nair. Reduced-order modeling of parameterized pdes using time–space–parameter principal component analysis. *International journal for numerical methods in engineering*, 80(8):1025–1057, 2009.
- [17] Woutijn J Baars and Charles E Tinney. Proper orthogonal decomposition-based spectral higher-order stochastic estimation. *Physics of Fluids*, 26(5):055112, 2014.
- [18] Joan Baiges, Ramon Codina, and Sergio Idelsohn. Reduced-order subscales for pod models. *Computer Methods in Applied Mechanics and Engineering*, 291:173–196, 2015.
- [19] Francesco Ballarin, Andrea Manzoni, Alfio Quarteroni, and Gianluigi Rozza. Supremizer stabilization of pod–galerkin approximation of parametrized steady incompressible navier–stokes equations. *International Journal for Numerical Methods in Engineering*, 102(5):1136–1161, 2015.
- [20] Etienne Balmès. Parametric families of reduced finite element models. theory and applications. *Mechanical systems and signal Processing*, 10(4):381–394, 1996.
- [21] Maxime Barrault, Yvon Maday, Ngoc Cuong Nguyen, and Anthony T Patera. An ‘empirical interpolation’ method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathématique*, 339(9):667–672, 2004.
- [22] A Barrett and G Reddien. On the reduced basis method. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, 75(7):543–549, 1995.
- [23] Peter Benner and Tobias Breiten. Interpolation-based \mathcal{H}_2 -model reduction of bilinear control systems. *SIAM Journal on Matrix Analysis and Applications*, 33(3):859–885, 2012.
- [24] Peter Benner, Ralph Byers, Heike Fassbender, Volker Mehrmann, and David Watkins. *Cholesky-like factorizations of skew-symmetric matrices*. Citeseer, 2000.
- [25] Peter Benner, Daniel Kressner, and Volker Mehrmann. Skew-hamiltonian and hamiltonian eigenvalue problems: Theory, algorithms and applications. In *Proceedings of the Conference on Applied Mathematics and Scientific Computing*, pages 3–39. Springer, 2005.
- [26] Peter Benner, Volker Mehrmann, and Hongguo Xu. A new method for computing the stable invariant subspace of a real hamiltonian matrix. *Journal of computational and applied mathematics*, 86(1):17–43, 1997.

-
- [27] Gerhard Berge. Landau damping in a plasma. 1969.
- [28] Michel Bergmann, C-H Bruneau, and Angelo Iollo. Enablers for robust pod models. *Journal of Computational Physics*, 228(2):516–538, 2009.
- [29] David Bernstein. Optimal prediction of burgers’s equation. *Multiscale Modeling & Simulation*, 6(1):27–52, 2007.
- [30] Nam Parshad Bhatia and Giorgio P Szegö. *Stability theory of dynamical systems*. Springer Science & Business Media, 2002.
- [31] Peter Binev, Albert Cohen, Wolfgang Dahmen, Ronald DeVore, Guergana Petrova, and Przemyslaw Wojtaszczyk. Convergence rates for greedy algorithms in reduced basis methods. *SIAM journal on mathematical analysis*, 43(3):1457–1472, 2011.
- [32] Charles K Birdsall and A Bruce Langdon. *Plasma physics via computer simulation*. CRC press, 2018.
- [33] Åke Björck. Numerics of gram-schmidt orthogonalization. *Linear Algebra and Its Applications*, 197:297–316, 1994.
- [34] Gregory Allan Blaisdell. *Numerical simulation of compressible homogeneous turbulence*. PhD thesis, Stanford University, 1991.
- [35] Christos Boutsidis, Michael W Mahoney, and Petros Drineas. An improved approximation algorithm for the column subset selection problem. In *Proceedings of the twentieth annual ACM-SIAM symposium on Discrete algorithms*, pages 968–977. SIAM, 2009.
- [36] Wernher Brevis and Manuel García-Villalba. Shallow-flow visualization analysis by proper orthogonal decomposition. *Journal of Hydraulic Research*, 49(5):586–594, 2011.
- [37] David S Broomhead and David Lowe. Radial basis functions, multi-variable functional interpolation and adaptive networks. Technical report, Royal Signals and Radar Establishment Malvern (United Kingdom), 1988.
- [38] Patrick Buchfink, Ashish Bhatt, and Bernard Haasdonk. Symplectic model order reduction with non-orthonormal bases. *Mathematical and Computational Applications*, 24(2):43, 2019.
- [39] Patrick Buchfink, Silke Glas, and Bernard Haasdonk. Symplectic model reduction of hamiltonian systems on nonlinear manifolds. *arXiv preprint arXiv:2112.10815*, 2021.
- [40] PATRICK Buchfink, BERNARD Haasdonk, and STEPHAN Rave. Psd-greedy basis generation for structure-preserving model order reduction of hamiltonian systems. In *Proceedings of ALGORITHM*, pages 151–160, 2020.
- [41] Annalisa Buffa, Yvon Maday, Anthony T Patera, Christophe Prud’homme, and Gabriel Turinici. A priori convergence of the greedy algorithm for the parametrized reduced basis method. *ESAIM: Mathematical modelling and numerical analysis*, 46(3):595–603, 2012.
- [42] Marcelo Buffoni, Simone Camarri, Angelo Iollo, and Maria Vittoria Salvetti. Low-dimensional modelling of a confined three-dimensional wake flow. *Journal of Fluid Mechanics*, 569:141–150, 2006.

- [43] T Bui-Thanh, Murali Damodaran, and Karen Willcox. Proper orthogonal decomposition extensions for parametric applications in compressible aerodynamics. In *21st AIAA Applied Aerodynamics Conference*, page 4213, 2003.
- [44] Angelika Bunse-Gerstner. Matrix factorizations for symplectic qr-like methods. *Linear Algebra and its Applications*, 83:49–77, 1986.
- [45] Nicolas Cagniard, Yvon Maday, and Benjamin Stamm. Model order reduction for problems with large convection effects. In *Contributions to partial differential equations and applications*, pages 131–150. Springer, 2019.
- [46] JG Caputo, NK Efremidis, and Chao Hang. Fourier-mode dynamics for the nonlinear schrödinger equation in one-dimensional bounded domains. *Physical Review E*, 84(3):036601, 2011.
- [47] Kevin Carlberg. Adaptive h-refinement for reduced-order models. *International Journal for Numerical Methods in Engineering*, 102(5):1192–1210, 2015.
- [48] Kevin Carlberg, Matthew Barone, and Harbir Antil. Galerkin v. least-squares petrov–galerkin projection in nonlinear model reduction. *Journal of Computational Physics*, 330:693–734, 2017.
- [49] Kevin Carlberg, Youngsoo Choi, and Syuzanna Sargsyan. Conservative model reduction for finite-volume models. *Journal of Computational Physics*, 371:280–314, 2018.
- [50] Kevin Carlberg, Charbel Farhat, Julien Cortial, and David Amsallem. The gnat method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows. *Journal of Computational Physics*, 242:623–647, 2013.
- [51] Kevin Carlberg, Ray Tuminaro, and Paul Boggs. Preserving lagrangian structure in nonlinear model reduction with application to structural dynamics. *SIAM Journal on Scientific Computing*, 37(2):B153–B184, 2015.
- [52] Fernando Casas, Nicolas Crouseilles, Erwan Faou, and Michel Mehrenberger. High-order hamiltonian splitting for the vlasov–poisson equations. *Numerische Mathematik*, 135(3):769–801, 2017.
- [53] Elena Celledoni and Brynjulf Owren. A class of intrinsic schemes for orthogonal integration. *SIAM Journal on Numerical Analysis*, 40(6):2069–2084, 2002.
- [54] Subrahmanyan Chandrasekhar. *Hydrodynamic and hydromagnetic stability*. Courier Corporation, 2013.
- [55] Abhilash J Chandy and Steven H Frankel. The t-model as a large eddy simulation model for the navier–stokes equations. *Multiscale Modeling & Simulation*, 8(2):445–462, 2010.
- [56] Saifon Chaturantabut, Chris Beattie, and Serkan Gugercin. Structure-preserving model reduction for nonlinear port-hamiltonian systems. *SIAM Journal on Scientific Computing*, 38(5):B837–B865, 2016.
- [57] Saifon Chaturantabut and Danny C Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, 2010.

-
- [58] David Chelidze and Wenliang Zhou. Smooth orthogonal decomposition-based vibration mode identification. *Journal of Sound and Vibration*, 292(3-5):461–473, 2006.
 - [59] Mulin Cheng, Thomas Y Hou, and Zhiwen Zhang. A dynamically bi-orthogonal method for time-dependent stochastic partial differential equations ii: Adaptivity and generalizations. *Journal of Computational Physics*, 242:753–776, 2013.
 - [60] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.
 - [61] François Chollet et al. Keras: The python deep learning library. *Astrophysics source code library*, pages ascl-1806, 2018.
 - [62] Alexandre Chorin and Panagiotis Stinis. Problem reduction, renormalization, and memory. *Communications in Applied Mathematics and Computational Science*, 1(1):1–27, 2007.
 - [63] Alexandre J Chorin, Ole H Hald, and Raz Kupferman. Optimal prediction and the mori–zwanzig representation of irreversible processes. *Proceedings of the National Academy of Sciences*, 97(7):2968–2973, 2000.
 - [64] Alexandre J Chorin, Ole H Hald, and Raz Kupferman. Optimal prediction with memory. *Physica D: Nonlinear Phenomena*, 166(3-4):239–257, 2002.
 - [65] Alexandre Joel Chorin and Ole H Hald. *Stochastic tools in mathematics and science*, volume 1. Springer, 2009.
 - [66] Erik Adler Christensen, Morten Brøns, and Jens Nørkær Sørensen. Evaluation of proper orthogonal decomposition-based decomposition techniques applied to parameter-dependent nonturbulent flows. *SIAM Journal on Scientific Computing*, 21(4):1419–1434, 1999.
 - [67] Paul G Constantine. *Active subspaces: Emerging ideas for dimension reduction in parameter studies*. SIAM, 2015.
 - [68] John Cornthwaite. Pressure poisson method for the incompressible navier-stokes equations using galerkin finite elements. 2013.
 - [69] M Couplet, C Basdevant, and P Sagaut. Calibrated reduced-order pod-galerkin system for fluid flow modelling. *Journal of Computational Physics*, 207(1):192–220, 2005.
 - [70] M Couplet, P Sagaut, and C20143141063 Basdevant. Intermodal energy transfers in a proper orthogonal decomposition–galerkin representation of a turbulent separated flow. *Journal of Fluid Mechanics*, 491:275–284, 2003.
 - [71] Nguyen Ngoc Cuong, Karen Veroy, and Anthony T Patera. Certified real-time solution of parametrized partial differential equations. In *Handbook of materials modeling*, pages 1529–1564. Springer, 2005.
 - [72] JP Cusumano, MT Sharkady, and BW Kimble. Experimental measurements of dimensionality and spatial coherence in the dynamics of a flexible-beam impact oscillator. *Philosophical Transactions of the Royal Society of London. Series A: Physical and Engineering Sciences*, 347(1683):421–438, 1994.

- [73] Maurice de Gosson and Franz Luef. Symplectic capacities and the geometry of uncertainty: the irruption of symplectic topology in classical and quantum mechanics. *Physics Reports*, 484(5):131–179, 2009.
- [74] Miguel Fosas de Pando, Peter J Schmid, and Denis Sipp. Nonlinear model-order reduction for compressible flow solvers using the discrete empirical interpolation method. *Journal of Computational Physics*, 324:194–209, 2016.
- [75] Simone Deparis and Gianluigi Rozza. Reduced basis method for multi-parameter-dependent steady navier–stokes equations: applications to natural convection in a cavity. *Journal of Computational Physics*, 228(12):4359–4378, 2009.
- [76] Olivier Desjardins, Guillaume Blanquart, Guillaume Balarac, and Heinz Pitsch. High order conservative finite difference scheme for variable density low mach number turbulent flows. *Journal of Computational Physics*, 227(15):7125–7159, 2008.
- [77] David G Dritschel and Norman J Zabusky. On the nature of vortex interactions and models in unforced nearly-inviscid two-dimensional turbulence. *Physics of Fluids*, 8(5):1252–1256, 1996.
- [78] Zlatko Drmac and Serkan Gugercin. A new selection operator for the discrete empirical interpolation method—improved a priori error bound and extensions. *SIAM Journal on Scientific Computing*, 38(2):A631–A648, 2016.
- [79] Philippe Druault. *Développement d’interfaces expérience/simulation. Application à l’écoulement de couche de mélange plane turbulente*. PhD thesis, Université de Poitiers (France), 1999.
- [80] Daniel Dylewsky, Molei Tao, and J Nathan Kutz. Dynamic mode decomposition for multiscale nonlinear physics. *Physical Review E*, 99(6):063311, 2019.
- [81] Jens L Eftang, David J Knezevic, and Anthony T Patera. An hp certified reduced basis method for parametrized parabolic partial differential equations. *Mathematical and Computer Modelling of Dynamical Systems*, 17(4):395–422, 2011.
- [82] Virginie Ehrlacher and Damiano Lombardi. A dynamical adaptive tensor method for the vlasov–poisson system. *Journal of Computational Physics*, 339:285–306, 2017.
- [83] Virginie Ehrlacher, Damiano Lombardi, Olga Mula, and François-Xavier Vialard. Nonlinear model reduction on metric spaces. application to one-dimensional conservative pdes in wasserstein spaces. *ESAIM: Mathematical Modelling and Numerical Analysis*, 54(6):2159–2197, 2020.
- [84] Lukas Einkemmer and Ilon Joseph. A mass, momentum, and energy conservative dynamical low-rank scheme for the vlasov equation. *Journal of Computational Physics*, page 110495, 2021.
- [85] Richard Everson and Lawrence Sirovich. Karhunen–loève procedure for gappy data. *JOSA A*, 12(8):1657–1664, 1995.
- [86] Evstati G Evstatiev and Bradley A Shadwick. Variational formulation of particle algorithms for kinetic plasma simulations. *Journal of Computational Physics*, 245:376–398, 2013.

-
- [87] Charbel Farhat, Todd Chapman, and Philip Avery. Structure-preserving, stability, and accuracy properties of the energy-conserving sampling and weighting method for the hyper reduction of nonlinear finite element dynamic models. *International journal for numerical methods in engineering*, 102(5):1077–1110, 2015.
- [88] Peter Feldmann and Roland W Freund. Efficient linear circuit analysis by padé approximation via the lanczos process. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 14(5):639–649, 1995.
- [89] Florian Feppon and Pierre FJ Lermusiaux. A geometric approach to dynamical model order reduction. *SIAM Journal on Matrix Analysis and Applications*, 39(1):510–538, 2018.
- [90] HE Fiedler. Coherent structures in turbulent flows. *Progress in Aerospace Sciences*, 25(3):231–269, 1988.
- [91] Alexander Figotin and Jeffrey H Schenker. Hamiltonian structure for dispersive and dissipative dynamical systems. *Journal of Statistical Physics*, 128(4):969–1056, 2007.
- [92] JP Fink and WC Rheinboldt. On the error behavior of the reduced basis technique for nonlinear finite element approximations. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, 63(1):21–28, 1983.
- [93] Maria L Frezzotti, Francesco Nasuti, Cheng Huang, Charles L Merkle, and William E Anderson. Quasi-1d modeling of heat release for the study of longitudinal combustion instability. *Aerospace Science and Technology*, 75:261–270, 2018.
- [94] David Galbally, Krzysztof Fidkowski, Karen Willcox, and Omar Ghattas. Non-linear model reduction for uncertainty quantification in large-scale inverse problems. *International journal for numerical methods in engineering*, 81(12):1581–1608, 2010.
- [95] Ugo Galvanetto and George Violaris. Numerical investigation of a new damage detection method based on proper orthogonal decomposition. *Mechanical Systems and Signal Processing*, 21(3):1346–1361, 2007.
- [96] Alan George and Joseph W Liu. *Computer solution of large sparse positive definite*. Prentice Hall Professional Technical Reference, 1981.
- [97] Massimo Germano, Ugo Piomelli, Parviz Moin, and William H Cabot. A dynamic subgrid-scale eddy viscosity model. *Physics of Fluids A: Fluid Dynamics*, 3(7):1760–1765, 1991.
- [98] Anna-Lena Gerner and Karen Veroy. Reduced basis a posteriori error bounds for the stokes equations in parametrized domains: a penalty approach. *Mathematical Models and Methods in Applied Sciences*, 21(10):2103–2134, 2011.
- [99] Anna-Lena Gerner and Karen Veroy. Certified reduced basis methods for parametrized saddle point problems. *SIAM Journal on Scientific Computing*, 34(5):A2812–A2836, 2012.
- [100] Gennady Gildenblat, Xin Li, Weimin Wu, Hailing Wang, Amit Jha, Ronald Van Langevelde, Geert DJ Smit, Andries J Scholten, and Dirk BM Klaassen. Psp: An advanced surface-potential-based mosfet model for circuit simulation. *IEEE Transactions on Electron Devices*, 53(9):1979–1993, 2006.

Bibliography

- [101] Camille Gillot, Guilhem Dif-Pradalier, Xavier Garbet, Philippe Ghendrih, Virginie Grandgirard, and Yanick Sarazin. Model order reduction approach to the one-dimensional collisionless closure problem. *Physics of Plasmas*, 28(2):022111, 2021.
- [102] Dror Givon, Raz Kupferman, and Ole H Hald. Existence proof for orthogonal dynamics and the mori-zwanzig formalism. *Israel Journal of Mathematics*, 145(1):221–241, 2005.
- [103] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.
- [104] Keith Glover. All optimal hankel-norm approximations of linear multivariable systems and their l_1, ∞ -error bounds. *International journal of control*, 39(6):1115–1193, 1984.
- [105] Yuezheng Gong, Qi Wang, and Zhu Wang. Structure-preserving galerkin pod reduced-order modeling of hamiltonian systems. *Computer Methods in Applied Mechanics and Engineering*, 315:780–798, 2017.
- [106] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.
- [107] William B Gordon. On the completeness of hamiltonian vector fields. *Proceedings of the American Mathematical Society*, pages 329–331, 1970.
- [108] Ayoub Gouasmi, Eric J Parish, and Karthik Duraisamy. A priori estimation of memory effects in reduced-order models of nonlinear systems using the mori–zwanzig formalism. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 473(2205):20170385, 2017.
- [109] Martin A Grepl, Ngoc C Nguyen, Karen Veroy, Anthony T Patera, and Gui R Liu. Certified rapid solution of partial differential equations for real-time parameter estimation and optimization. In *Real-time PDE-constrained optimization*, pages 199–216. SIAM, 2007.
- [110] Martin A Grepl and Anthony T Patera. A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 39(1):157–181, 2005.
- [111] Sebastian Grimberg, Charbel Farhat, and Noah Youkilis. On the stability of projection-based model order reduction for convection-dominated laminar and turbulent flows. *Journal of Computational Physics*, 419:109681, 2020.
- [112] Mikhael Gromov. Pseudo holomorphic curves in symplectic manifolds. *Inventiones mathematicae*, 82(2):307–347, 1985.
- [113] Thomas Hakon Gronwall. Note on the derivatives with respect to a parameter of the solutions of a system of differential equations. *Annals of Mathematics*, pages 292–296, 1919.
- [114] Jacob Grosek and J Nathan Kutz. Dynamic mode decomposition for real-time background/-foreground separation in video. *arXiv preprint arXiv:1404.7592*, 2014.
- [115] Serkan Gugercin, Rostyslav V Polyuga, Christopher Beattie, and Arjan Van Der Schaft. Structure-preserving tangential interpolation for model reduction of port-hamiltonian systems. *Automatica*, 48(9):1963–1974, 2012.

-
- [116] Bernard Haasdonk. Convergence rates of the pod–greedy method. *ESAIM: Mathematical modelling and numerical Analysis*, 47(3):859–873, 2013.
 - [117] Bernard Haasdonk. Reduced basis methods for parametrized pdes—a tutorial introduction for stationary and instationary problems. *Model reduction and approximation: theory and algorithms*, 15:65, 2017.
 - [118] Bernard Haasdonk and Mario Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42(2):277–302, 2008.
 - [119] Ernst Hairer, Marlis Hochbruck, Arieh Iserles, and Christian Lubich. Geometric numerical integration. *Oberwolfach Reports*, 3(1):805–882, 2006.
 - [120] Ernst Hairer, Syvert Paul Norsett, and Gerhard Wanner. *Solving Ordinary, Differential Equations I, Nonstiff problems/E. Hairer, SP Norsett, G. Wanner, with 135 Figures, Vol.: 1*. Number BOOK. 2Ed. Springer-Verlag, 2000, 2000.
 - [121] Ole H Hald and Panagiotis Stinis. Optimal prediction and the rate of decay for solutions of the euler equations in two and three dimensions. *Proceedings of the National Academy of Sciences*, 104(16):6527–6532, 2007.
 - [122] Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011.
 - [123] Carsten Hartmann, Valentina-Mira Vulcanov, and Christof Schütte. Balanced truncation of linear second-order systems: a hamiltonian approach. *Multiscale Modeling & Simulation*, 8(4):1348–1367, 2010.
 - [124] Yang He, Yajuan Sun, Hong Qin, and Jian Liu. Hamiltonian particle-in-cell methods for vlasov-maxwell equations. *Physics of Plasmas*, 23(9):092108, 2016.
 - [125] Maziar S Hemati, Matthew O Williams, and Clarence W Rowley. Dynamic mode decomposition for large and streaming datasets. *Physics of Fluids*, 26(11):111701, 2014.
 - [126] HG Hereward. Landau damping. Technical report, 1977.
 - [127] Jan Hesthaven and Cecilia Pagliantini. Structure-preserving reduced basis methods for poisson systems. *Mathematics of Computation*, 90(330):1701–1740, 2021.
 - [128] Jan S Hesthaven, Cecilia Pagliantini, and Nicolò Ripamonti. Rank-adaptive structure-preserving reduced basis methods for hamiltonian systems. *arXiv preprint arXiv:2007.13153*, 2020.
 - [129] Jan S Hesthaven, Cecilia Pagliantini, and Nicolò Ripamonti. Adaptive symplectic model order reduction of parametric particle-based vlasov-poisson equatio. *arXiv preprint arXiv:2201.05555*, 2022.
 - [130] Jan S Hesthaven, Gianluigi Rozza, Benjamin Stamm, et al. *Certified reduced basis methods for parametrized partial differential equations*, volume 590. Springer, 2016.
 - [131] Hesthaven, Jan S., Pagliantini, Cecilia, and Ripamonti, Nicolò. Rank-adaptive structure-preserving model order reduction of hamiltonian systems. *ESAIM: M2AN*, 56(2):617–650, 2022.

Bibliography

- [132] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
- [133] RW Hockney. Characteristics of noise in a two-dimensional computer plasma. *The Physics of Fluids*, 11(6):1381–1383, 1968.
- [134] Albert E Honein and Parviz Moin. Higher entropy conservation and numerical stability of compressible turbulence simulations. *Journal of Computational Physics*, 201(2):531–545, 2004.
- [135] Kuo-lin Hsu, Hoshin Vijai Gupta, and Soroosh Sorooshian. Artificial neural network modeling of the rainfall-runoff process. *Water resources research*, 31(10):2517–2530, 1995.
- [136] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.
- [137] Angelo Iollo and Damiano Lombardi. Advection modes by optimal mass transfer. *Physical Review E*, 89(2):022923, 2014.
- [138] Martin Kahlbacher and Stefan Volkwein. Galerkin proper orthogonal decomposition methods for parameter dependent elliptic systems. *Discussiones Mathematicae, Differential Inclusions, Control and Optimization*, 27(1):95–117, 2007.
- [139] Irina Kalashnikova, Bart van Bloemen Waanders, Srinivasan Arunajatesan, and Matthew Barone. Stabilization of projection-based reduced order models for linear time-invariant systems via optimization-based eigenvalue reassignment. *Computer Methods in Applied Mechanics and Engineering*, 272:251–270, 2014.
- [140] Michael Karow, Daniel Kressner, and Françoise Tisseur. Structured eigenvalue condition numbers. *SIAM Journal on Matrix Analysis and Applications*, 28(4):1052–1068, 2006.
- [141] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137, 2015.
- [142] NK-R Kevlahan and Marie Farge. Vorticity filaments in two-dimensional turbulence: creation, stability and effect. *Journal of Fluid Mechanics*, 346:49–76, 1997.
- [143] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [144] David J Knezevic and Anthony T Patera. A certified reduced basis method for the fokker-planck equation of dilute polymeric fluids: Fene dumbbells in extensional flow. *SIAM Journal on Scientific Computing*, 32(2):793–817, 2010.
- [145] Birgul Koc, Muhammad Mohebujjaman, Changhong Mou, and Traian Iliescu. Commutation error in reduced order modeling of fluid flows. *Advances in Computational Mathematics*, 45(5):2587–2621, 2019.
- [146] Othmar Koch and Christian Lubich. Dynamical low-rank approximation. *SIAM Journal on Matrix Analysis and Applications*, 29(2):434–454, 2007.

-
- [147] Bernard O Koopman. Hamiltonian systems and transformation in hilbert space. *Proceedings of the national academy of sciences of the united states of america*, 17(5):315, 1931.
- [148] Tanya Kostova, Geoffrey Oxberry, Kyle Chand, and William Arrighi. Error bounds and analysis of proper orthogonal decomposition model reduction methods using snapshots from the solution and the time derivatives. *arXiv preprint arXiv:1501.02004*, 2015.
- [149] Robert H Kraichnan. Inertial ranges in two-dimensional turbulence. *The Physics of Fluids*, 10(7):1417–1423, 1967.
- [150] Robert H Kraichnan. Inertial-range transfer in two-and three-dimensional turbulence. *Journal of Fluid Mechanics*, 47(3):525–535, 1971.
- [151] Robert H Kraichnan. On kolmogorov’s inertial-range theories. *Journal of Fluid Mechanics*, 62(2):305–330, 1974.
- [152] Michael Kraus, Katharina Kormann, Philip J Morrison, and Eric Sonnendrücker. Gempic: geometric electromagnetic particle-in-cell methods. *Journal of Plasma Physics*, 83(4), 2017.
- [153] John A Krommes. Nonequilibrium gyrokinetic fluctuation theory and sampling noise in gyrokinetic particle-in-cell simulations. *Physics of Plasmas*, 14(9):090501, 2007.
- [154] Frances Y Kuo and Ian H Sloan. Lifting the curse of dimensionality. *Notices of the AMS*, 52(11):1320–1328, 2005.
- [155] J Nathan Kutz, Steven L Brunton, Bingni W Brunton, and Joshua L Proctor. *Dynamic mode decomposition: data-driven modeling of complex systems*. SIAM, 2016.
- [156] Sanjay Lall, Petr Krysl, and Jerrold E Marsden. Structure-preserving model reduction for mechanical systems. *Physica D: Nonlinear Phenomena*, 184(1-4):304–318, 2003.
- [157] Hee Sun Lee, Surl-Hee Ahn, and Eric F Darve. Building a coarse-grained model based on the mori-zwanzig formalism. *MRS Online Proceedings Library (OPL)*, 1753, 2015.
- [158] Kookjin Lee and Kevin T Carlberg. Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders. *Journal of Computational Physics*, 404:108973, 2020.
- [159] H Ralph Lewis. Energy-conserving numerical approximations for vlasov plasmas. *Journal of Computational Physics*, 6(1):136–141, 1970.
- [160] Zhen Li, Xin Bian, Xiantao Li, and George Em Karniadakis. Incorporation of memory effects in coarse-grained modeling via the mori-zwanzig formalism. *The Journal of chemical physics*, 143(24):243128, 2015.
- [161] Sarah K Locke and John R Singler. A new approach to proper orthogonal decomposition with difference quotients. *arXiv preprint arXiv:2106.10224*, 2021.
- [162] AE Lovgren, Yvon Maday, and EM Ronquist. A reduced basis element method for complex flow systems. In *ECCOMAS CFD 2006: Proceedings of the European Conference on Computational Fluid Dynamics, Egmond aan Zee, The Netherlands, September 5-8, 2006*. Citeseer, 2006.

Bibliography

- [163] FE Low. A lagrangian formulation of the boltzmann-vlasov equation for plasmas. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 248(1253):282–287, 1958.
- [164] Fei Lu, Kevin K Lin, and Alexandre J Chorin. Data-based stochastic model reduction for the kuramoto–sivashinsky equation. *Physica D: Nonlinear Phenomena*, 340:46–57, 2017.
- [165] John Leask Lumley. The structure of inhomogeneous turbulent flows. *Atmospheric turbulence and radio wave propagation*, 1967.
- [166] Babak Maboudi Afkham and Jan S Hesthaven. Structure-preserving model-reduction of dissipative hamiltonian systems. *Journal of Scientific Computing*, 81(1):3–21, 2019.
- [167] Luc Machiels, Yvon Maday, Ivan B Oliveira, Anthony T Patera, and Dimitrios V Rovas. Output bounds for reduced-basis approximations of symmetric positive definite eigenvalue problems. *Comptes Rendus de l’Académie des Sciences-Series I-Mathematics*, 331(2):153–158, 2000.
- [168] Yvon Maday, Anthony T Patera, and Gabriel Turinici. Global a priori convergence theory for reduced-basis approximations of single-parameter symmetric coercive elliptic partial differential equations. *Comptes Rendus Mathématique*, 335(3):289–294, 2002.
- [169] Giovanni Manfredi. Long-time behavior of nonlinear landau damping. *Physical review letters*, 79(15):2815, 1997.
- [170] Sathesh Mariappan, Anthony Donald Gardner, Kai Richter, and Markus Raffel. Analysis of dynamic stall using dynamic mode decomposition technique. *AIAA journal*, 52(11):2427–2439, 2014.
- [171] Jerrold E Marsden and Tudor S Ratiu. *Introduction to mechanics and symmetry: a basic exposition of classical mechanical systems*, volume 17. Springer Science & Business Media, 2013.
- [172] Jerrold E Marsden and Alan Weinstein. The hamiltonian structure of the maxwell-vlasov equations. *Physica D: nonlinear phenomena*, 4(3):394–406, 1982.
- [173] Daiki Matsumoto and Thomas Indinger. On-the-fly algorithm for dynamic mode decomposition using incremental singular value decomposition and total least squares. *arXiv preprint arXiv:1703.11004*, 2017.
- [174] Charles Meneveau. Germano identity-based subgrid-scale modeling: a brief survey of variations on a fertile theme. *Physics of Fluids*, 24(12):121301, 2012.
- [175] Marcus Meyer and Hermann G Matthies. Efficient model reduction in non-linear dynamics using the karhunen-loeve expansion and dual-weighted-residual methods. *Computational Mechanics*, 31(1):179–191, 2003.
- [176] René Milk, Stephan Rave, and Felix Schindler. pymor–generic algorithms and interfaces for model order reduction. *SIAM Journal on Scientific Computing*, 38(5):S194–S216, 2016.
- [177] Yuto Miyatake. Structure-preserving model reduction for dynamical systems with a first integral. *Japan Journal of Industrial and Applied Mathematics*, 36(3):1021–1037, 2019.

-
- [178] Bruce Moore. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE transactions on automatic control*, 26(1):17–32, 1981.
 - [179] Yohei Morinishi. Skew-symmetric form of convective terms and fully conservative finite difference schemes for variable density low-mach number flows. *Journal of Computational Physics*, 229(2):276–300, 2010.
 - [180] Yohei Morinishi, Thomas S Lund, Oleg V Vasilyev, and Parviz Moin. Fully conservative higher order finite difference schemes for incompressible flow. *Journal of computational physics*, 143(1):90–124, 1998.
 - [181] Youhei Morinishi, Shinji Tamano, and Koichi Nakabayashi. A dns algorithm using b-spline collocation method for compressible turbulent channel flow. *Computers & fluids*, 32(5):751–776, 2003.
 - [182] Philip J Morrison. Hamiltonian field description of the one-dimensional poisson-vlasov equations. Technical report, Princeton Univ., NJ (USA). Plasma Physics Lab., 1981.
 - [183] Eleonora Musharbash, Fabio Nobile, and Eva Vidličková. Symplectic dynamical low rank approximation of wave equations with random parameters. *BIT Numerical Mathematics*, 60(4):1153–1201, 2020.
 - [184] Indranil Nayak, Mrinal Kumar, and Fernando L Teixeira. 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.
 - [185] Bernd R Noack, Paul Papas, and Peter A Monkewitz. The need for a pressure-term representation in empirical galerkin models of incompressible shear flows. *Journal of Fluid Mechanics*, 523:339–365, 2005.
 - [186] Altan Odabasioglu, Mustafa Celik, and Lawrence T Pileggi. Prima: Passive reduced-order interconnect macromodeling algorithm. *IEEE Transactions on computer-aided design of integrated circuits and systems*, 17(8):645–654, 1998.
 - [187] Mario Ohlberger and Stephan Rave. Nonlinear reduced basis approximation of parameterized evolution equations via the method of freezing. *Comptes Rendus Mathématique*, 351(23-24):901–906, 2013.
 - [188] Taku Ohwada and Pietro Asinari. Artificial compressibility method revisited: asymptotic numerical method for incompressible navier–stokes equations. *Journal of Computational Physics*, 229(5):1698–1723, 2010.
 - [189] Cecilia Pagliantini. Dynamical reduced basis methods for hamiltonian systems. *Numerische Mathematik*, pages 1–40, 2021.
 - [190] Chris Paige and Charles Van Loan. A schur decomposition for hamiltonian matrices. *Linear Algebra and its applications*, 41:11–32, 1981.
 - [191] Eric J Parish and Karthik Duraisamy. A dynamic subgrid scale model for large eddy simulations based on the mori–zwanzig formalism. *Journal of Computational Physics*, 349:154–175, 2017.
 - [192] Eric J Parish and Karthik Duraisamy. Non-markovian closure models for large eddy simulations using the mori–zwanzig formalism. *Physical Review Fluids*, 2(1):014604, 2017.

- [193] Eric J Parish and Karthik Duraisamy. A unified framework for multiscale modeling using the mori-zwanzig formalism and the variational multiscale method. *arXiv preprint arXiv:1712.09669*, 2017.
- [194] Anthony T Patera and Einar M Rønquist. Reduced basis approximation and a posteriori error estimation for a boltzmann model. *Computer Methods in Applied Mechanics and Engineering*, 196(29-30):2925–2942, 2007.
- [195] Amnon Pazy. *Semigroups of linear operators and applications to partial differential equations*, volume 44. Springer Science & Business Media, 2012.
- [196] Benjamin Peherstorfer and Karen Willcox. Online adaptive model reduction for nonlinear systems via low-rank updates. *SIAM Journal on Scientific Computing*, 37(4):A2123–A2150, 2015.
- [197] Liqian Peng and Kamran Mohseni. Geometric model reduction of forced and dissipative hamiltonian systems. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 7465–7470. IEEE, 2016.
- [198] Liqian Peng and Kamran Mohseni. Symplectic model reduction of hamiltonian systems. *SIAM Journal on Scientific Computing*, 38(1):A1–A27, 2016.
- [199] Laurent Perret, Erwan Collin, and Joël Delville. Polynomial identification of pod based low-order dynamical system. *Journal of Turbulence*, (7):N17, 2006.
- [200] Janet S Peterson. The reduced basis method for incompressible viscous flow calculations. *SIAM Journal on Scientific and Statistical Computing*, 10(4):777–786, 1989.
- [201] Bérengère Podvin. On the adequacy of the ten-dimensional model for the wall layer. *Physics of fluids*, 13(1):210–224, 2001.
- [202] Bérengère Podvin and John Lumley. A low-dimensional approach for the minimal flow unit. *Journal of Fluid Mechanics*, 362:121–155, 1998.
- [203] Rostyslav V Polyuga and Arjan Van der Schaft. Structure preserving model reduction of port-hamiltonian systems by moment matching at infinity. *Automatica*, 46(4):665–672, 2010.
- [204] Joshua L Proctor and Philip A Eckhoff. Discovering dynamic patterns from infectious disease data using dynamic mode decomposition. *International health*, 7(2):139–145, 2015.
- [205] Bartosz Protas, Bernd R Noack, and Jan Östh. Optimal nonlinear eddy viscosity in galerkin models of turbulent flows. *Journal of Fluid Mechanics*, 766:337–367, 2015.
- [206] Sebastien Prothin, Jean-Yves Billard, and Henda Djeridi. Image processing using proper orthogonal and dynamic mode decompositions for the study of cavitation developing on a naca0015 foil. *Experiments in fluids*, 57(10):1–25, 2016.
- [207] Hong Qin, Jian Liu, Jianyuan Xiao, Ruili Zhang, Yang He, Yulei Wang, Yajuan Sun, Joshua W Burby, Leland Ellison, and Yao Zhou. Canonical symplectic particle-in-cell method for long-term large-scale simulations of the vlasov–maxwell equations. *Nuclear Fusion*, 56(1):014001, 2015.

-
- [208] Alfio Quarteroni, Andrea Manzoni, and Federico Negri. *Reduced basis methods for partial differential equations: an introduction*, volume 92. Springer, 2015.
- [209] Alfio Quarteroni and Alessandro Veneziani. Analysis of a geometrical multiscale model based on the coupling of ode and pde for blood flow simulations. *Multiscale Modeling & Simulation*, 1:173–195, 2003.
- [210] Muruhan Rathinam and Linda R Petzold. A new look at proper orthogonal decomposition. *SIAM Journal on Numerical Analysis*, 41(5):1893–1925, 2003.
- [211] Tomás Chacón Rebollo, Enrique Delgado Avila, Macarena Gómez Mármol, Francesco Ballarin, and Gianluigi Rozza. On a certified smagorinsky reduced basis turbulence model. *SIAM Journal on Numerical Analysis*, 55(6):3047–3067, 2017.
- [212] Julius Reiss, Philipp Schulze, Jörn Sesterhenn, and Volker Mehrmann. The shifted proper orthogonal decomposition: A mode decomposition for multiple transport phenomena. *SIAM Journal on Scientific Computing*, 40(3):A1322–A1344, 2018.
- [213] Julius Reiss and Jörn Sesterhenn. A conservative, skew-symmetric finite difference scheme for the compressible navier–stokes equations. *Computers & Fluids*, 101:208–219, 2014.
- [214] Ricardo Reyes and Ramon Codina. Projection-based reduced order models for flow problems: A variational multiscale approach. *Computer Methods in Applied Mechanics and Engineering*, 363:112844, 2020.
- [215] Werner C Rheinboldt. On the theory and error estimation of the reduced basis method for multi-parameter problems. *Nonlinear Analysis: Theory, Methods & Applications*, 21(11):849–858, 1993.
- [216] Donsub Rim, Benjamin Peherstorfer, and Kyle T Mandli. Manifold approximations via transported subspaces: Model reduction for transport-dominated problems. *arXiv preprint arXiv:1912.13024*, 2019.
- [217] DV Rovas, L Machiels, and Yvon Maday. Reduced-basis output bound methods for parabolic problems. *IMA journal of numerical analysis*, 26(3):423–445, 2006.
- [218] Clarence W Rowley, Igor Mezić, Shervin Bagheri, Philipp Schlatter, and Dan S Henningson. Spectral analysis of nonlinear flows. *Journal of fluid mechanics*, 641:115–127, 2009.
- [219] Gianluigi Rozza, Dinh Bao Phuong Huynh, and Anthony T Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. *Archives of Computational Methods in Engineering*, 15(3):1, 2007.
- [220] David Ryckelynck. A priori hyperreduction method: an adaptive approach. *Journal of computational physics*, 202(1):346–366, 2005.
- [221] Yousef Saad. *Iterative methods for sparse linear systems*. SIAM, 2003.
- [222] Arvind K Saibaba. Randomized discrete empirical interpolation method for nonlinear model reduction. *SIAM Journal on Scientific Computing*, 42(3):A1582–A1608, 2020.
- [223] Ahmed Salam. On theoretical and numerical aspects of symplectic gram–schmidt-like algorithms. *Numerical Algorithms*, 39(4):437–462, 2005.

Bibliography

- [224] Ahmed Salam and Eman Al-Aidarous. Equivalence between modified symplectic gram-schmidt and householder sr algorithms. *BIT Numerical Mathematics*, 54(1):283–302, 2014.
- [225] Themistoklis P Sapsis and Pierre FJ Lermusiaux. Dynamical criteria for the evolution of the stochastic dimensionality in flows with uncertainty. *Physica D: Nonlinear Phenomena*, 241(1):60–76, 2012.
- [226] Taraneh Sayadi, Peter J Schmid, Franck Richecoeur, and Daniel Durox. Parametrized data-driven decomposition for bifurcation analysis, with application to thermo-acoustically unstable systems. *Physics of Fluids*, 27(3):037102, 2015.
- [227] Peter J Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of fluid mechanics*, 656:5–28, 2010.
- [228] Peter J Schmid. Application of the dynamic mode decomposition to experimental data. *Experiments in fluids*, 50(4):1123–1130, 2011.
- [229] Clint Scovel and Alan Weinstein. Finite dimensional lie-poisson approximations to vlasov-poisson equations. *Communications on Pure and Applied Mathematics*, 47(5):683–709, 1994.
- [230] Sugata Sen. Reduced-basis approximation and a posteriori error estimation for many-parameter heat conduction problems. *Numerical Heat Transfer, Part B: Fundamentals*, 54(5):369–389, 2008.
- [231] Yaroslav Shitov. Column subset selection is np-complete. *Linear Algebra and its Applications*, 610:52–58, 2021.
- [232] Lawrence Sirovich. Turbulence and the dynamics of coherent structures. i. coherent structures. *Quarterly of applied mathematics*, 45(3):561–571, 1987.
- [233] Björn Sjögreen and HC Yee. On skew-symmetric splitting and entropy conservation schemes for the euler equations. In *Numerical Mathematics and Advanced Applications 2009*, pages 817–827. Springer, 2010.
- [234] Ian H Sloan and Henryk Woźniakowski. When are quasi-monte carlo algorithms efficient for high dimensional integrals? *Journal of Complexity*, 14(1):1–33, 1998.
- [235] R Smith, M Ellis, G Xia, V Sankaran, W Anderson, and CL Merkle. Computational investigation of acoustics and instabilities in a longitudinal-mode rocket combustor. *AIAA journal*, 46(11):2659–2673, 2008.
- [236] Alessio Spantini. *Preconditioning techniques for stochastic partial differential equations*. PhD thesis, Massachusetts Institute of Technology, 2013.
- [237] Jonathan Squire, Hong Qin, and William M Tang. Geometric integration of the vlasov-maxwell system with a variational particle-in-cell scheme. *Physics of Plasmas*, 19(8):084501, 2012.
- [238] Giovanni Stabile, Francesco Ballarin, Giacomo Zuccarino, and Gianluigi Rozza. A reduced order variational multiscale approach for turbulent flows. *Advances in Computational Mathematics*, 45(5):2349–2368, 2019.

-
- [239] Henry Stark and John W Woods. *Probability, random processes, and estimation theory for engineers*. Prentice-Hall, Inc., 1986.
- [240] Răzvan Ștefănescu, Adrian Sandu, and Ionel M Navon. Comparison of pod reduced order strategies for the nonlinear 2d shallow water equations. *International Journal for Numerical Methods in Fluids*, 76(8):497–521, 2014.
- [241] Panagiotis Stinis. Higher order mori–zwanzig models for the euler equations. *Multiscale Modeling & Simulation*, 6(3):741–760, 2007.
- [242] Panos Stinis. Renormalized mori–zwanzig-reduced models for systems without scale separation. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2176):20140446, 2015.
- [243] Shamima Sultana and Zillur Rahman. Hamiltonian formulation for water wave equation. 2013.
- [244] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27, 2014.
- [245] RD Sydora. Low-noise electromagnetic and relativistic particle-in-cell plasma simulation models. *Journal of computational and applied mathematics*, 109(1-2):243–259, 1999.
- [246] Barna Szabó and Ivo Babuška. Finite element analysis: Method, verification and validation. 2021.
- [247] Daniel B Szyld. The many proofs of an identity on the norm of oblique projections. *Numerical Algorithms*, 42(3):309–323, 2006.
- [248] Tommaso Taddei. A registration method for model order reduction: data compression and geometry reduction. *SIAM Journal on Scientific Computing*, 42(2):A997–A1027, 2020.
- [249] Molei Tao. Explicit symplectic approximation of nonseparable hamiltonians: Algorithm and long time performance. *Physical Review E*, 94(4):043303, 2016.
- [250] William J Thompson. Fourier series and the gibbs phenomenon. *American journal of physics*, 60(5):425–429, 1992.
- [251] Jonathan H Tu. *Dynamic mode decomposition: Theory and applications*. PhD thesis, Princeton University, 2013.
- [252] Tomasz M Tyranowski and Michael Kraus. Symplectic model reduction methods for the vlasov equation. *arXiv preprint arXiv:1910.06026*, 2019.
- [253] Benjamin Unger and Serkan Gugercin. Kolmogorov n-widths for linear dynamical systems. *Advances in Computational Mathematics*, 45(5):2273–2286, 2019.
- [254] Charles Van Loan. A symplectic method for approximating all the eigenvalues of a hamiltonian matrix. *Linear algebra and its applications*, 61:233–251, 1984.
- [255] Luca Venturi, Francesco Ballarin, and Gianluigi Rozza. A weighted pod method for elliptic pdes with random inputs. *Journal of Scientific Computing*, 81(1):136–153, 2019.
- [256] Karen Veroy and Anthony T Patera. Certified real-time solution of the parametrized steady incompressible navier–stokes equations: rigorous reduced-basis a posteriori error bounds. *International Journal for Numerical Methods in Fluids*, 47(8-9):773–788, 2005.

- [257] Karen Veroy, Christophe Prud’Homme, Dimitrios Rovas, and Anthony Patera. A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. In *16th AIAA Computational Fluid Dynamics Conference*, page 3847, 2003.
- [258] Jing Wang, Huafei Sun, and Simone Fiori. A riemannian-steepest-descent approach for optimization on the real symplectic group. *Mathematical Methods in the Applied Sciences*, 41(11):4273–4286, 2018.
- [259] Qian Wang, Jan S Hesthaven, and Deep Ray. Non-intrusive reduced order modeling of unsteady flows using artificial neural networks with application to a combustion problem. *Journal of computational physics*, 384:289–307, 2019.
- [260] Qian Wang, Nicolò Ripamonti, and Jan S Hesthaven. Recurrent neural network closure of parametric pod-galerkin reduced-order models based on the mori-zwanzig formalism. *Journal of Computational Physics*, 410:109402, 2020.
- [261] Zhu Wang, Imran Akhtar, Jeff Borggaard, and Traian Iliescu. Two-level discretizations of nonlinear closure models for proper orthogonal decomposition. *Journal of Computational Physics*, 230(1):126–146, 2011.
- [262] Zhu Wang, Imran Akhtar, Jeff Borggaard, and Traian Iliescu. Proper orthogonal decomposition closure models for turbulent flows: a numerical comparison. *Computer Methods in Applied Mechanics and Engineering*, 237:10–26, 2012.
- [263] David S Watkins. On hamiltonian and symplectic lanczos processes. *Linear algebra and its applications*, 385:23–45, 2004.
- [264] Gerrit Welper. Interpolation of functions with parameter dependent jumps by transformed snapshots. *SIAM Journal on Scientific Computing*, 39(4):A1225–A1250, 2017.
- [265] Rebing Wu, Raj Chakrabarti, and Herschel Rabitz. Optimal control theory for continuous-variable quantum gates. *Physical Review A*, 77(5):052303, 2008.
- [266] Jianyuan Xiao, Hong Qin, Jian Liu, Yang He, Ruili Zhang, and Yajuan Sun. Explicit high-order non-canonical symplectic particle-in-cell algorithms for vlasov-maxwell systems. *Physics of Plasmas*, 22(11):112504, 2015.
- [267] Xuping Xie, David Wells, Zhu Wang, and Traian Iliescu. Approximate deconvolution reduced order modeling. *Computer Methods in Applied Mechanics and Engineering*, 313:512–534, 2017.
- [268] Hongguo Xu. An svd-like matrix decomposition and its applications. *Linear algebra and its applications*, 368:1–24, 2003.
- [269] Süleyman Yildiz, Murat Uzunca, and Bülent Karasözen. Structure-preserving reduced-order modeling of non-traditional shallow water equation. In *Model Reduction of Complex Dynamical Systems*, pages 327–345. Springer, 2021.
- [270] Jian Yu and Jan S Hesthaven. A comparative study of shock capturing models for the discontinuous galerkin method. Technical report, Elsevier, 2017.

- [271] Yen Yu, Stefan Koeglmeier, James Sisco, and William Anderson. Combustion instability of gaseous fuels in a continuously variable resonance chamber (cvrc). In *44th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, page 4657, 2008.
- [272] Yuanran Zhu, Jason M Dominy, and Daniele Venturi. Rigorous error estimates for the memory integral in the mori-zwanzig formulation. *arXiv preprint arXiv:1708.02235*, 2017.
- [273] Yuanran Zhu, Jason M Dominy, and Daniele Venturi. On the estimation of the mori-zwanzig memory integral. *Journal of Mathematical Physics*, 59(10):103501, 2018.
- [274] Yuanran Zhu and Daniele Venturi. Faber approximation of the mori-zwanzig equation. *Journal of Computational Physics*, 372:694–718, 2018.
- [275] Robert Zwanzig. Memory effects in irreversible thermodynamics. *Physical Review*, 124(4):983, 1961.

Nicolò RIPAMONTI

Applied Mathematician

📍 Route de la Maladière 28, 1022 Chavannes-près-Renens, Switzerland 📞 +41 76 437 93 04
✉ nripamont@gmail.com 🌐 linkedin.com/in/nicolò-ripamonti 📧 ripamonti.92
🎓 scholar.google.com 📄 researchgate.com



Research scientist specialized in **scientific computing**, **model order reduction**, **modeling**, and **machine learning**. Passionate for research, development, and design of modern approaches for broadly relevant problems. Strong analytical and communication skills, proactive attitude when facing new challenges.

📁 CORE EXPERIENCE

Present February 2018	Doctoral Assistant in Applied Mathematics <i>Swiss Federal Institute of Technology Lausanne (EPFL), Switzerland</i> <ul style="list-style-type: none">➤ Developed energy-preserving reduced-order models for fluid dynamics➤ Focus on data-driven closure modeling of reduced PDEs using neural networks➤ Research and implementation of dynamical low-rank representations for plasma physics➤ Authored peer-reviewed articles in top-tier journals of scientific computing and modeling (JCP, M2AN)➤ Teaching assistant for Numerical Analysis, Analysis 3, and 4➤ Supervised Master theses and semester projects
July 2016 December 2016	Internship in Research & Development <i>Sony R&D Center Europe Stuttgart, Germany</i> <ul style="list-style-type: none">➤ Optimized coded masked sensors for image acquisition by lensless camera➤ Developed compressive sensing algorithms for image reconstruction➤ Applied numerical methods for the development of prototypal lensless camera➤ Worked in an interdisciplinary team of mathematicians, physicists, and engineers

📁 EDUCATION

PhD in Applied Math 2018-Present	Swiss Federal Institute of Technology Lausanne (EPFL), Switzerland PhD on <i>Stabilization and structure-preservation of reduced order models</i> .
MSc in Computational Science 2015-2017	Swiss Federal Institute of Technology Lausanne (EPFL), Switzerland Double-degree program. Master Thesis on <i>Energy-preserving model reduction of fluid flows</i> .
MSc in Mathematical Engineering 2015-2017	Polytechnic University of Milan (PoliMi), Italy Double-degree program. Master Thesis on <i>Energy-preserving model reduction of fluid flows</i> .
BSc in Mathematical Engineering 2011-2015	Polytechnic University of Milan (PoliMi), Italy Bachelor Thesis on <i>Parametric model order reduction by matrix interpolation</i> .

</> SKILLS

Programming	Python, Matlab, C++, SQL
Libraries	PyTorch, Numpy, TensorFlow, pandas, scipy, matplotlib
OS	MacOS, Linux Ubuntu
Other tools	git, make, LaTeX, Jupyter, bash, Paraview, Doxygen







🗣️ LANGUAGES

Italian	● ● ● ● ●
English	● ● ● ● ●
French	● ○ ○ ○ ○

+ EXPERTISE

- Mathematical Modeling
- Scientific Computing
- Numerical Analysis
- Machine Learning

REFEREED PUBLICATIONS & TALKS

- 2021 | **Structure-preserving model order reduction of Hamiltonian systems.**
J.S. Hesthaven, C. Pagliantini, N. Ripamonti
 [ArXiv](#)
- 2020 | **Rank-adaptive structure-preserving reduced basis methods for Hamiltonian systems.**
J.S. Hesthaven, C. Pagliantini, N. Ripamonti
 [ArXiv](#)
- 2020 | **Recurrent neural network closure of parametric POD-Galerkin reduced-order models based on the Mori-Zwanzig formalism.**
Q.Wang, N. Ripamonti, J.S. Hesthaven
 [Journal of Computational Physics](#)
- 2020 | **Conservative model order reduction for fluid flows.**
B.M. Afkham, N. Ripamonti, Q. Wang, J.S. Hesthaven
 [Quantification of Uncertainty: Improving Efficiency and Technology](#)
- 2021 | **VI Eccomas Young Investigators Conference, Valencia, Spain.**
Title talk : Rank adaptive structure-preserving reduced basis methods for plasma physics.
 [YIC2021](#)
- 2019 | **Model Order Reduction Summer School, Eindhoven, Netherlands.**
Title talk : Hyper-reduction methods for MOR of nonlinear problems.
 [MORSS2019](#)

SELECTED PROJECTS

STRUCTURE-PRESERVING DEEP LEARNING METHODS FOR THE DISCRETIZATION OF HAMILTONIAN SYSTEMS	2021
<i>Supervision of semester project</i>	
Implementation of symplectic neural networks to capture the dynamics of canonically Hamiltonian systems.	
DEEP REINFORCEMENT LEARNING FOR ARTIFICIAL PONG PLAYERS	2020
<i>Project for the course "Artificial Neural Networks"</i>	
Implementation, in TensorFlow, of Reinforcement Learning paradigm for the training of artificial PONG player..	
NUMERICAL APPROXIMATION OF OSCILLATORY GENZ' FUNCTIONS VIA PRUNED SPARSE NEURAL NETWORKS	2020
<i>Supervision of semester project</i>	
Comparison of pruning approaches for the sparsification of feed-forward neural networks for the approximation of oscillatory Genz' functions.	
EXPLICIT STABILIZED METHODS FOR NUMERICAL INTEGRATION OF NEURON'S EQUATIONS	2016
<i>Semester project</i>	
Development of C++ package, based on PETSc, for the integration of ROCK2 solver for stiff problems into the NEURON simulator used by the High Performance Computing team of the Blue Brain Project.	

OUTREACH AND VOLUNTEERING

- 2020-2021 | President of SIAM EPFL student chapter.
- 2020 | Junior Organizer of MORSS, Lausanne, Switzerland.
- 2019 | Junior Organizer of MORSS, Eindhoven, Netherlands.

PERSONAL INFORMATION

Extracurricular Activities	Basketball, Climbing
Other	Italian citizen, Swiss resident permit (B)