

**Development of Reduced Order Modeling Methods for Incompressible
Flows with Heat Transfer and Parametric Boundary Conditions**

Sabrina Kelbij Star

Doctoral dissertation submitted to obtain the academic degree of
Doctor of Electromechanical Engineering

Supervisors

Prof. Joris Degroote, PhD* - Giovanni Stabile, PhD** - Benjamin Sande, PhD***

* Department of Electromechanical, Systems and Metal Engineering
Faculty of Engineering and Architecture, Ghent University

** Scuola Internazionale Superiore di Studi Avanzati, Italy

***Centrum Wiskunde & Informatica, the Netherlands

February 2021



ISBN 978-94-6355-459-6

NUR 919, 928

Wettelijk depot: D/2021/10.500/7

Members of the Examination Board

Chair

Prof. Hennie De Schepper, PhD, Ghent University

Other members entitled to vote

Prof. Andreas Class, PhD, Karlsruhe Institute of Technology, Germany

Prof. Tom Dhaene, PhD, Ghent University

Prof. Geraldine Heynderickx, PhD, Ghent University

Stefano Lorenzi, PhD, Politecnico di Milano, Italy

Prof. Samuele Rubino, PhD, Universidad de Sevilla, Spain

Supervisors

Prof. Joris Degroote, PhD, Ghent University

Giovanni Stabile, PhD, Scuola Internazionale Superiore di Studi Avanzati, Italy

Benjamin Sande, PhD, Centrum Wiskunde & Informatica, the Netherlands

Samenvatting

Voor het MYRRHA project (Engelse afkorting voor Multi-purpose hYbrid Research Reactor for High-tech Applications, oftewel multifunctionele hybride onderzoeksreactor voor hoogtechnologische toepassingen) ontwikkelt and ontwerpt het Belgisch studiecentrum voor de kernenergie SCK CEN een multi-functionele experimentele bestralingsfaciliteit. Het ontwerp bestaat onder andere uit een compact primair systeem van het pooltype dat gekoeld wordt door middel van gesmolten lood-bismut-eutecticum, oftewel een zwaar metaal. Betrouwbare rekenmethoden zijn vereist om het systeemgedrag van de reactor in operationele en ongevalsomstandigheden nauwkeurig te kwantificeren. Het aantal kernreactorsimulaties in een veiligheidsanalyse ligt in de meeste gevallen echter buiten de mogelijkheden van de huidige hardware wanneer computationele vloeistofdynamica wordt gebruikt. Dit heeft de ontwikkeling van modelleringstechnieken met een gereduceerde orde gestimuleerd die het aantal vrijheidsgraden van de zeer nauwkeurige warmtestromingsmodellen verminderen. Wiskundige technieken worden gebruikt om de “karakteristieken” van het complexe model te extraheren, om ze vervolgens te vervangen door een eenvoudiger model. Op die manier wordt de benodigde rekentijd en computergeheugengebruik verminderd. Ondanks de kracht en de toenemende populariteit van gereduceerde modellen voor allerlei soorten stromingstoepassingen, hebben ze vaak problemen met de nauwkeurigheid en vertonen ze numerieke instabiliteit. Uitdagingen met betrekking tot de snelheidsdrukkoppeling en het voldoen aan de randvoorwaarden op het lagere orde-niveau maken het moeilijk om de methoden zodanig te generaliseren dat ze op elk probleem kunnen worden toegepast.

De complexe vloeistofdynamica problemen worden doorgaans numeriek opgelost met behulp van discretisatiemethoden. In dit werk richten we ons op de discretisatiemethode met eindige volumes voor de numerieke oplossing van onsamendrukbare stromingen op gecoluceerde roosters. Om een computationeel efficiënt model van gereduceerde orde te verkrijgen, wordt de procedure idealiter gesplit in een, zogenaamde, offline fase en een online fase. In de offline fase worden de oplossingen van het oorspronkelijke nauwkeurige model verzameld op verschillende tijdstippen en/of voor verschillende parameterwaarden. Ze worden gebruikt om een gereduceerde basis van een veel kleinere orde te genereren dan het model van volledige orde. In dit werk worden de gereduceerde basisruimten overspannen door basisfuncties, of zogenaamde modi, die worden berekend behulp van de hoofcomponentenanalyse (Engels: proper orthogonal decomposition of POD) die gewoonlijk wordt gebruikt voor onsamendrukbare stromingen. Gereduceerde

matrices (lineaire termen) en tensoren (niet-lineaire termen) geassocieerd met de termen van het model van volledige orde worden bepaald tijdens de offline fase, waarvoor in dit werk twee technieken zijn ontwikkeld en onderzocht. De eerste techniek is een niet-intrusieve reductiemethode die de systeemmatrix van lineaire vloeistofdynamica problemen identificeert met een kleinste-kwadratentechniek. Het belangrijkste voordeel van niet-intrusieve methoden is dat ze geen toegang nodig hebben tot het discretisatie- en oplossingsalgoritme van de numerieke simulatiecode. De tweede techniek is de intrusieve Galerkin-projectiebenadering waarbij de vergelijkingen van de volledige orde worden geprojecteerd op de gereduceerde POD-basisruimten. In de online fase wordt het gereduceerde systeem van vergelijkingen opgelost voor dezelfde of nieuwe waarden van de bepaalde parameters tegen lagere rekenkosten in vergelijking met het oplossen van het systeem van volledige orde.

De niet-intrusieve reductiemethode wordt gepresenteerd in het eerste deel van het proefschrift. De methodologie wordt toegepast op de lineaire scalaire convectie-diffusievergelijking voor een tweedimensionaal ingesloten vierkante holte met een verwarmde bovenkant. De (tijdsafhankelijke) randvoorwaarden worden opgelegd in het model van gereduceerde orde met een penalisatiemethode. De resultaten worden vergeleken en de nauwkeurigheid van de modellen van gereduceerde orde wordt beoordeeld aan de hand van de oplossingen van het model van volledige orde. Er wordt aangetoond dat het gereduceerde-orde-model kan worden gebruikt voor gevoeligheidsanalyse door de niet-homogene Dirichlet-randvoorwaarden aan te passen. Voor niet-lineaire problemen moet het vereiste aantal momentopnames worden geschaald met de derde macht van het aantal modi en zijn er minstens evenveel gereduceerde matrices als het aantal gebruikte modi nodig. Daarom is het niet haalbaar om de op POD gebaseerde identificatiemethode te gebruiken voor niet-lineaire problemen. Voor de simulatie van vloeistofstromingen in industriële toepassingen is het echter nodig om gereduceerde-orde-modellen te ontwikkelen voor niet-lineaire problemen, zoals convectie- en opwaartse stromingen. Daarom is het grootste deel van het proefschrift gewijd aan de intrusieve POD-gebaseerde Galerkin-projectiebenadering vanwege zijn toepasbaarheid op niet-lineaire problemen.

POD-Galerkin modellen van gereduceerde orde zijn ontwikkeld waarvan de (tijdsafhankelijke) randvoorwaarden op het lagere orde-niveau worden opgelegd met behulp van twee verschillende strategieën: de liftfunctie-methode, die het doel heeft homogene basisfuncties te verkrijgen voor de gereduceerde basisruimten en de penalisatiemethode waarbij de randvoorwaarden worden opgelegd in het model van gereduceerde orde met behulp van een penalisatiefactor. De penalisatiemethode wordt verbeterd door een iteratieve oplosser te gebruiken voor het bepalen van de penalisatiefactor in plaats van de factor af te stemmen met een gevoeligheidsanalyse of numerieke experimenten. De methoden worden vergeleken en getest voor twee gevallen: het klassieke benchmarkprobleem van een ingesloten vierkante holte met een bewegende bovenkant en de stroming door een Y-junctie met twee inlaatkanalen en één uitlaatkanaal. De resultaten laten zien dat de randvoorwaarden van het model van gereduceerde orde kunnen worden gecontroleerd

met de methodes en dat dezelfde orde van nauwkeurigheid wordt bereikt voor de snelheids- en drukvelden. Het berekenen van de oplossingen kost echter meer tijd in het geval van de liftfunctie-methode, aangezien de gereduceerde basisruimten extra modi bevatten, namelijk de liftfuncties, vergeleken met de penalisatiemethode.

Verder wordt een parametrisch model van gereduceerde orde voor convectieve warmteoverdracht geïntroduceerd. De Boussinesq-benadering wordt gebruikt voor het modelleren van de aangedreven warmteoverdracht. Daarom bestaat er een tweerichtingskoppeling tussen de onsamendrukbare Boussinesq-vergelijkingen en de energievergelijking. Om het gereduceerde-orde-model te verkrijgen, wordt een Galerkin-projectie van de relevante vergelijkingen op de gereduceerde POD-basisruimten uitgevoerd. Het model van gereduceerde orde wordt getest op een tweedimensionaal differentieel verwarmde ingesloten holte waarvan de zijwandtemperaturen worden geparаметriseerd. De parametrisering gebeurt met behulp van de liftfunctie-methode. De liftfuncties worden verkregen door een Laplace-functie voor temperatuur op te lossen. Er was slechts één simulatie met het model van volledige orde nodig voor de creatie van de gereduceerde basis. Het verkregen model van gereduceerde orde is efficiënt en stabiel voor verschillende parametersets waarbij het temperatuurverschil tussen de wanden kleiner is dan de set die werd gebruikt voor het creëren van de POD-basis.

Verder wordt de POD-Galerkin-strategie voor Reynoldsgemiddelde Navier-Stokes simulaties (Engels: Reynolds-Averaged Navier-Stokes of RANS) uitgebreid voor vloeistofstromingen met een laag Prandtl-getal. Het gereduceerde-orde-model is gebaseerd op een model van volledige orde waarbij de effecten van convectieve warmteoverdracht op de stroming worden gekenmerkt door variatie in het Richardson-getal. De Reynolds-spanningen worden berekend met een lineair turbulentie-viscositeitsmodel. Een enkelvoudige gradiëntdiffusiehypothese, samen met een lokale correlatie voor de evaluatie van het turbulente Prandtl-getal, wordt gebruikt om de turbulente warmtefluxen te modelleren. De bijdrage van de turbulente viscositeits- en turbulente thermische diffusiteitsvelden wordt meegenomen in het gereduceerde-orde-model door middel van een op interpolatie gebaseerde datagestuurde methode. Het model van gereduceerde orde wordt getest op een door warmteoverdracht aangedreven turbulente vloeibare natriumstroom over een verticale achterwaarts gerichte trede. Een gelijkmatige warmteflux is aangebracht op de wand stroomafwaarts van de trede. De wandwarmteflux is opgenomen met een Neumann-randvoorwaarde in zowel het volledige-orde-model als het gereduceerde-orde-model. De snelheids- en temperatuurprofielen die met het gereduceerde model worden voorspeld voor dezelfde en nieuwe Richardson-getallen binnen het bereik van de parameterwaarden, komen goed overeen met de RANS-simulaties. Ook worden het lokale Stanton-nummer en de wrijvingsverdeling op de verwarmde wand kwalitatief goed vastgelegd. Ten slotte zijn de simulaties met een gereduceerde orde, uitgevoerd op een enkele processor, ongeveer 10^5 keer sneller dan de volledige RANS-simulaties die op acht processoren werden uitgevoerd.

Het laatste deel van het proefschrift is gewijd aan de ontwikkeling van een

nieuw niet-geparametriseerd model van gereduceerde orde van de onsamendrukbare Navier-Stokes-vergelijkingen op gecoloeerde roosters. Het model is ontwikkeld door een Galerkin-projectie uit te voeren op basis van een volledig (ruimte en tijd) discrete formulatie van het model van volledige orde. Deze ‘discretiseren-dan-projecteren’-benadering vereist geen drukstabilisatietechniek (hoewel de drukterm in het gereduceerde-orde-model aanwezig is) noch een controletechniek om de randvoorwaarden op het gereduceerde orde-niveau op te leggen. Dit zijn twee belangrijke voordelen ten opzichte van bestaande en eerder toegepaste benaderingen. Het volledig discrete model van volledige orde wordt verkregen door een eendige volumediscretisatie van de onsamendrukbare Navier-Stokes-vergelijkingen met een voorwaartse Euler-tijdsdiscretisatie. Twee varianten van de snelheidsdrukkoppeling op het volledig discrete niveau, de inconsistente en consistente fluxmethode, zijn onderzocht. De laatste methode leidt tot divergentie-vrije snelheidsvelden, ook op gereduceerd orde-niveau, terwijl de snelheidsvelden bij de eerste methode slechts bij benadering divergentie-vrij zijn. Voor beide methoden zijn stabiele en nauwkeurige resultaten verkregen voor testgevallen met verschillende soorten randvoorwaarden: een ingesloten vierkante holte met een bewegende bovenkant en een open holte met een inlaat en een uitlaat. Het model van gereduceerde orde verkregen met de consistente fluxmethode, met divergentie-vrije snelheidsvelden, is iets nauwkeuriger maar ook iets duurder om op te lossen in vergelijking met de inconsistente fluxmethode vanwege een extra vergelijking voor de flux die moet worden opgelost. De tijdswinst van het gereduceerde-orde-model ten opzichte van het model van volledige orde is sterk afhankelijk de gebruikte methode, het aantal vrijheidsgraden van het model van volledige orde en het aantal modi dat wordt gebruikt voor de gereduceerde basisruimten.

Een toepassing met een koppeling tussen een thermische systeemcode en een gereduceerd model van een computationele vloeistofdynamica code wordt gepresenteerd in de bijlage van dit werk. De systeemcode en het model van gereduceerde orde zijn gekoppeld door een algoritme voor de decompositie van de domeinen met behulp van een impliciet koppelingsschema. De snelheid die over een koppelingssinterface wordt getransporteerd, wordt in het model van gereduceerde orde opgelegd met behulp van een penalisatiemethode. De gekoppelde modellen worden beoordeeld op open en gesloten configuraties voor stromingen door buizen. De resultaten van de gekoppelde simulaties met het model van gereduceerde orde liggen dicht bij die met de computationele vloeistofdynamica code. Ook voor nieuwe parametersets zijn de gekoppelde modellen met het gereduceerd-orde-model in staat om de oplossingen van de gekoppelde modellen met het volledig-orde-model te voorspellen met een goede nauwkeurigheid. De rekentijd van de simulaties met het model van gereduceerde orde zijn 3-5 keer korter dan die met het model van volledige orde.

Summary

Within the MYRRHA project, which stands for Multi-purpose hYbrid Research Reactor for High-tech Applications, the Belgian Nuclear Research Center SCK CEN is developing and designing a multi-functional experimental fast-spectrum irradiation facility. The MYRRHA design features a compact pool-type primary system cooled by molten Lead-Bismuth Eutectic, i.e. a heavy metal. Reliable computational methods are required to accurately quantify the reactor's primary system behavior in operational and accidental conditions and to handle complex geometries. However, the number of nuclear reactor simulations in a safety analysis is, in the majority of cases, beyond the possibilities of present hardware if a computational fluid dynamics solver is used alone. This has motivated the development of reduced order modeling techniques that reduce the number of degrees of freedom of the high fidelity thermofluids models. Mathematical techniques are used to extract "features" of the complex model in order to replace them by a more simplified model. In that way, the required computational time and computer memory usage is reduced. Despite the potential and increasing popularity of reduced order models for all sorts of flow applications, they tend to have issues with accuracy and exhibit numerical instabilities. Challenges regarding velocity-pressure coupling and satisfying the boundary conditions at the reduced order level make it difficult to generalize the methods such that they can be applied to any problem.

The complex fluid dynamics problems are generally solved numerically using discretization methods. In this work, we focus on the finite volume discretization method for the numerical solution of incompressible fluid flows on collocated grids. To obtain a computationally efficient reduced order model (ROM), the procedure is ideally split into a so-called offline stage and an online stage. In the offline stage, solutions of the high fidelity model are collected at several time instances and/or for different parameter values. They are used to generate a reduced basis of a much smaller order than the full order model (FOM). In this work, the reduced basis spaces are spanned by basis functions, or so-called modes, which are computed using the proper orthogonal decomposition (POD) technique. POD is commonly used for reduced-order modeling of incompressible flows. Reduced matrices (linear terms) and tensors (nonlinear terms) of the ROM associated with the terms of the full order model are determined during the offline stage, for which two techniques are developed and investigated in this work. The first technique is a non-intrusive reduction method that identifies the system matrix of linear fluid dynamical problems with a least-squares technique. The main advantage of non-intrusive methods is that they do not require access to the solver's discretization

and solution algorithm. The second technique is the intrusive Galerkin projection approach for which the full order equations are projected onto the reduced POD basis spaces. In the online stage, the reduced system of equations are solved for the same or new values of the parameters of interest at a lower computational cost compared to solving the full order systems.

The non-intrusive reduction method that identifies the system matrix of linear fluid dynamical problems with a least-squares technique is presented in the first part of the thesis. The methodology is applied to the linear scalar transport convection-diffusion equation for a two-dimensional square cavity problem with a heated lid. The (time-dependent) boundary conditions are imposed in the reduced order model with a penalty method. The results are compared and the accuracy of the reduced order models is assessed against the full order solutions. It is shown that the reduced order model can be used for sensitivity analysis by controlling the non-homogeneous Dirichlet boundary conditions. For nonlinear problems, the required number of snapshots scales with the cube of the number of POD basis functions and at least as many reduced matrices are to be identified as the number of basis functions used. Therefore, it is not feasible to use the POD-based identification method for nonlinear problems. However, for the simulation of fluid flows in (nuclear) engineering applications, it is necessary to develop reduced order models for nonlinear problems, such as convective flows and buoyancy-driven flows. Therefore, the main part of the thesis is dedicated to the intrusive POD-based Galerkin projection approach due to its applicability to nonlinear problems.

POD-Galerkin reduced order models are developed of which the (time-dependent) boundary conditions are imposed at reduced order level using two different boundary control strategies: the lifting function method, whose aim is to obtain homogeneous basis functions for the reduced basis spaces and the penalty method where the boundary conditions are imposed in the reduced order model using a penalty factor. The penalty method is improved by using an iterative solver for the determination of the penalty factor rather than tuning the factor with a sensitivity analysis or numerical experimentation. The boundary control methods are compared and tested for two cases: the classical lid driven cavity benchmark problem and a Y-junction flow case with two inlet channels and one outlet channel. The results show that the boundaries of the reduced order model can be controlled with the boundary control methods and the same order of accuracy is achieved for the velocity and pressure fields. However, computing the ROM solutions takes more time in the case of the lifting function method as the reduced basis spaces contain additional modes, namely the lifting functions, compared to the penalty method.

Furthermore, a parametric reduced order model for buoyancy-driven flow is introduced. The Boussinesq approximation is used for modeling the buoyancy. Therefore, there exists a two-way coupling between the incompressible Boussinesq equations and the energy equation. To obtain the reduced order model, a Galerkin projection of the governing equations onto the reduced POD bases spaces is performed. The ROM is tested on a two-dimensional differentially heated cavity of which the side wall temperatures are parametrized. The parametrization is done using a lifting function method. The lifting functions are obtained by solving a

Laplacian function for temperature. Only one unsteady full order simulation was required for the creation of the reduced bases. The obtained ROM is efficient and stable for different parameter sets for which the temperature difference between the walls is smaller than for the set in the FOM used for the POD basis creation.

In addition, the POD-Galerkin reduced order modeling strategy for steady-state Reynolds averaged Navier–Stokes (RANS) simulation is extended for low-Prandtl number fluid flow. The reduced order model is based on a full order model for which the effects of buoyancy on the flow and heat transfer are characterized by varying the Richardson number. The Reynolds stresses are computed with a linear eddy viscosity model. A single gradient diffusion hypothesis, together with a local correlation for the evaluation of the turbulent Prandtl number, is used to model the turbulent heat fluxes. The contribution of the eddy viscosity and turbulent thermal diffusivity fields are considered in the reduced order model with an interpolation based data-driven method. The ROM is tested for buoyancy-aided turbulent liquid sodium flow over a vertical backward-facing step with a uniform heat flux applied on the wall downstream of the step. The wall heat flux boundary condition is incorporated in both the full order model and the reduced order model. The velocity and temperature profiles predicted with the ROM for the same and new Richardson numbers inside the range of parameter values are in good agreement with the RANS simulations. Also, the local Stanton number and skin friction distribution at the heated wall are qualitatively well captured. Finally, the reduced order simulations, performed on a single core, are about 10^5 times faster than the full order RANS simulations that are performed on eight cores.

The final part of the thesis is dedicated to the development of a novel non-parametric reduced order model of the incompressible Navier-Stokes equations on collocated grids. The reduced order model is developed by performing a Galerkin projection based on a fully (space and time) discrete full order model formulation. This ‘discretize-then-project’ approach requires no pressure stabilization technique (even though the pressure term is present in the ROM) nor a boundary control technique (to impose the boundary conditions at the ROM level). These are two main advantages compared to existing and previously applied approaches. The fully discrete FOM is obtained by a finite volume discretization of the incompressible Navier-Stokes equations with a forward Euler time discretization. Two variants of the velocity-pressure coupling at the fully discrete level, the inconsistent and consistent flux method, have been investigated. The latter leads to divergence-free velocity fields, also on the ROM level, whereas the velocity fields are only approximately divergence-free in the former method. For both methods, stable and accurate results have been obtained for test cases with different types of boundary conditions: a lid-driven cavity and an open cavity with an inlet and outlet. The ROM obtained with the consistent flux method, having divergence-free velocity fields, is slightly more accurate but also slightly more expensive to solve compared to the inconsistent flux method due to an additional equation for the flux. The speedup ratio of the ROM and FOM computation times strongly depends on which method is used, the number of degrees of freedom of the full order model and the number of modes used for the reduced basis spaces.

Finally, an application with the coupling between a system thermal hydraulics code and a reduced order model of a computational fluid dynamics solver is presented in the appendix of this work. The system code and the ROM are coupled by a domain decomposition algorithm using an implicit coupling scheme. The velocity transported over a coupling boundary interface is imposed in the ROM using a penalty method. The coupled models are evaluated on open and closed pipe flow configurations. The results of the coupled simulations with the ROM are close to those with the CFD solver. Also for new parameter sets, the coupled models with the ROM are capable of predicting the results of the coupled models with the FOM with good accuracy. The coupled simulations with the ROM are about 3-5 times faster than those with the FOM.

Preface

This dissertation was submitted to the department of Electromechanical, Systems and Metal Engineering at Ghent university (UGent) in Belgium to fulfill the diploma requirements for the doctorate (Ph.D.) degree. I was engaged in researching and writing this dissertation from October 2017 to November 2020. The research was mostly carried out in the Reactor Safety Analysis (RSA) expert group (split from the Nuclear Systems Physics (NSP) expert group in September 2019) as part of the MYRRHA project at the Belgian Nuclear Research Center SCK CEN. The work presented was partially carried out at the mathLab group of Scuola Internazionale Superiore di Studi Avanzati (SISSA) in Italy and the Scientific Computing group of Centrum Wiskunde en Informatica (CWI) in the Netherlands.

It would have not been possible to complete this dissertation without the support and contributions of numerous people. In this preface, I would like to thank the following people in particular:

Joris Degroote, my main university supervisor at UGent, for his guidance, comprehensive feedback and support that he provided me throughout my Ph.D. time. I appreciate our fruitful collaboration with many insightful discussions. It was certainly a pleasure to be a Ph.D. student under his supervision.

Giovanni Stabile, for explaining me the ins and outs of reduced order modeling, programming and code development. His diligent mindset, his involvement and broad research and coding skills have been an asset to this dissertation. I really enjoyed our collaboration.

Gianluigi Rozza for welcoming me at SISSA. Those six months in Trieste has not only taken my research to the next level, I also had an unforgettable time and I learned much more than just doing research.

Benjamin Sanderse for giving me the opportunity to conduct partially my research at CWI and committing to continue to supervise till the end my doctoral research. I learned a lot from his way of conducting research and I am happy about everything we have achieved in a couple of months time.

Francesco Belloni, my SCK CEN mentor, for writing me praising recommendation letters, which allowed me to go to SISSA for a research exchange and to CWI for a Ph.D. internship.

Gert Van den Eynde, Head of NSP and SCK CEN co-mentor, and Michèle Coeck, Head of the SCK CEN Academy, for providing me with the support and the means to perform my doctoral research.

Sokratia Georgaka for our numerous Skype calls. It was pleasant and super

helpful to share thoughts with someone like-minded, without even ever meeting each other in person.

My master thesis students for their interest in the research topic. I got the opportunity to co-supervise Jef Voorspoels from the KU Leuven at the beginning of my Ph.D. project. Furthermore, I had the pleasure to be the daily supervisor of Giuseppe Spina from the University of Pisa, who carried out his master thesis at SCK CEN and whose results are included in the appendix of this dissertation.

And last, but not least, my colleagues from Mol to Ghent and from Trieste to Amsterdam.

Getting through my numerical research required more than academic support. I want to thank the technical support of the ICT departments, in particular Yves Maenhout (UGent) and Frank Lenaerts (SCK CEN), for providing me the necessary computational environment. Aside from the technical support, a warm thanks to Annie Harri, Annick De Coster and Griet Blondé from UGent and Carine Van Reybroeck from SCK CEN for taking care of all the necessary paperwork for this Ph.D., including all the conferences, workshops and courses that I have been to. I also want to thank Roger Van Hecke from UGent for taking care of the design of the cover of my thesis. In addition, I acknowledge the financial support given by the ENEN+ project that has received funding from the Euratom research and training Work Programme 2016 - 2017 - 1 #755576 for my research stay at SISSA. I also acknowledge the support received from the Commissie Wetenschappelijk Onderzoek (CWO) at Ghent University, which funded my participation to the international M&C 2019 conference in Portland, USA.

Finally, I would also like to thank the committee members for accepting the commitment of evaluating my research work and this dissertation.

Kelbij Star
October 14, 2020

Table of Contents

Samenvatting	vii
Summary	xi
Preface	xv
Nomenclature	xxi
1 Introduction	1
1.1 The MYRRHA project	1
1.1.1 The MYRRHA facility	2
1.2 Motivation for reduced order modeling	5
1.3 The objectives and the framework of the Ph.D.	8
1.3.1 The objectives	8
1.3.2 The framework	9
1.4 Thesis outline	11
1.5 List of publications underlying this thesis	12
1.5.1 Publications in peer-reviewed journals	12
1.5.2 Publications in international conference proceedings	13
2 Reduced order modeling methods in computational fluid dynamics	15
2.1 The incompressible Navier–Stokes equations	15
2.2 The finite volume discretization of the incompressible Navier–Stokes equations	18
2.2.1 Spatial discretization	19
2.2.2 Temporal discretization	21
2.2.3 Velocity–pressure coupling	21
2.3 General overview of reduced order CFD methods	21
2.3.1 Projection based reduced order models	22
2.3.2 Non-intrusive reduced order models	23
2.4 POD–Galerkin method in the finite volume setting	23
2.4.1 Proper orthogonal decomposition	24
2.4.2 Galerkin projection	26
2.5 Challenges of reduced order modeling in the finite volume setting	27
2.5.1 Nonlinear problems	27
2.5.2 Offline–online decomposition	28

2.5.3	Stability	28
2.5.3.1	Pressure stabilization	29
2.5.3.2	Boundary conditions	30
2.5.3.3	Turbulence	31
2.5.3.4	Transient dynamics and long term behavior . . .	33
2.5.4	Parametric problems	34
2.5.5	Sources of errors	35
2.5.5.1	POD truncation error	36
2.5.5.2	Relative error	36
2.5.6	Overview	37
3	POD-identification reduced order model of linear transport equations	39
3.1	Introduction	39
3.2	The convection-diffusion scalar transport equation	40
3.3	POD-ID reduced order model of the scalar transport equation . . .	40
3.3.1	POD-based identification	41
3.3.2	Boundary conditions	43
3.4	Numerical test case of an enclosed heated cavity	45
3.5	Results and analysis	47
3.5.1	Imposing time-independent boundary conditions	50
3.5.2	Imposing time-dependent boundary conditions	51
3.6	Discussion	57
3.7	Conclusion	58
4	Methods to impose boundary conditions in POD-Galerkin reduced order models	61
4.1	Introduction	61
4.2	Exploitation of the pressure Poisson Equation	62
4.3	Boundary control methods at the reduced order level	64
4.3.1	The lifting function method	64
4.3.2	The iterative penalty method	67
4.4	Numerical test cases	70
4.4.1	Lid-driven cavity flow problem	71
4.4.2	Y-junction flow problem	72
4.5	Results and analysis	75
4.5.1	Lid-driven cavity flow problem	75
4.5.2	Y-junction flow problem	81
4.6	Discussion	88
4.7	Conclusion	89
5	POD-Galerkin reduced order model of the Boussinesq approximation for buoyancy-driven enclosed flows	91
5.1	Introduction	91
5.2	The Boussinesq approximation for the Navier-Stokes equations . . .	92

5.3	POD-Galerkin reduced order model of the Boussinesq approximation	92
5.3.1	Imposing boundary conditions with the lifting function method	94
5.4	Numerical test case of an enclosed cavity	94
5.5	Results and analysis	96
5.5.1	POD modes	97
5.5.2	Reduced order simulations for new parameter sets	99
5.6	Discussion	101
5.7	Conclusion	102
6	POD-Galerkin reduced order model of a turbulent convective buoyant flow of sodium over a backward-facing step	103
6.1	Introduction	103
6.2	Full order turbulence model	105
6.2.1	Flow characteristics by non-dimensional numbers	107
6.3	POD-Galerkin reduced order model for turbulent convective buoyant flows	108
6.3.1	Imposing boundary conditions with the penalty method	112
6.4	Numerical set-up for sodium flow over a backward facing step	114
6.5	Results and analysis	117
6.5.1	Determining the penalty factors	118
6.5.2	Comparison of the ROM and RANS solutions	119
6.5.3	Reduced order simulations for new parameter values	122
6.6	Discussion	126
6.7	Conclusion	128
7	A ‘discretize-then-project’ approach for the incompressible Navier-Stokes equations on collocated grids	131
7.1	Introduction	131
7.2	Finite volume discretization on collocated grids	133
7.2.1	Spatial discretization	134
7.2.2	Explicit projection methods (temporal discretization)	138
7.2.2.1	Inconsistent flux method	138
7.2.2.2	Consistent flux method	140
7.3	POD-Galerkin reduced order models for the explicit projection methods	141
7.3.1	Proper orthogonal decomposition	142
7.3.2	Galerkin projection for the inconsistent flux method	143
7.3.3	Galerkin projection for the consistent flux method	144
7.4	Numerical set-up	145
7.4.1	Lid-driven cavity flow problem	146
7.4.2	Open cavity flow problem	147
7.5	Results	148
7.5.1	Lid driven cavity flow problem	149
7.5.2	Open cavity flow problem	157

7.6	Discussion	163
7.7	Conclusions	165
8	Conclusions and outlook	167
8.1	Overall conclusions	167
8.2	Suggestions for future work	169
	References	171
A	Coupling between a system thermal-hydraulics code and a reduced order model	199
A.1	Introduction	199
A.2	Coupling methodology	200
A.2.1	Implicit coupling numerical scheme	201
A.3	Transport of hydraulic quantities over the coupling interfaces of coupled models	203
A.4	The coupled codes' governing equations and models	204
A.4.1	RELAP5-MOD3.3: Thermal-hydraulic modeling	205
A.4.2	OpenFOAM 6: Reynolds-averaged Navier-Stokes equations for incompressible turbulent flow	205
A.5	POD-Galerkin reduced order model for incompressible turbulent flow	205
A.6	Challenges for coupling system thermal-hydraulics codes with a reduced order model	208
A.6.1	Imposing the non-homogeneous time-dependent boundary conditions in the ROM	208
A.6.2	Relative error	209
A.7	Numerical test cases	209
A.7.1	Open pipe flow test	210
A.7.2	Open pipe flow reversal test	211
A.7.3	Closed pipe flow test	212
A.8	Results and analysis	213
A.8.1	Open pipe flow test	213
A.8.2	Open pipe flow reversal test	218
A.8.2.1	Long time integration	221
A.8.3	Closed pipe flow test	224
A.8.4	Convergence history	228
A.9	Discussion	229
A.10	Conclusions and outlook	230
	List of Figures	233
	List of Tables	241

Nomenclature

Acronyms and abbreviations

1D	one-dimensional
2D	two-dimensional
3D	three-dimensional
BC	boundary condition
CFD	computational fluid dynamics
CFM	consistent flux method
CV	cumulative eigenvalues
CWI	Centrum Wiskunde & Informatica
DEIM	discrete empirical interpolation method
DNS	direct numerical simulation
EIM	empirical interpolation method
ENEN	European Nuclear Education Network
ER	expansion ratio
FOM	full order model
FV	finite volume
HPC	high performance computing
IC	initial condition
IFM	inconsistent flux method
ITHACA-FV	In real Time Highly Advanced Computational Applications for Finite Volumes
LBE	lead bismuth eutectic
LES	large eddy simulation
MYRRHA	Multi-purpose hYbrid Research Reactor for High-tech Applications
NIROM	non-intrusive reduced-order method
NS	Navier-Stokes
ODE	ordinary differential equation
OF	OpenFOAM
PDE	partial differential equation
Ph.D.	Doctor of Philosophy
PIMPLE	Pressure-Implicit Method for Pressure-Linked Equations
PISO	Pressure-Implicit with Splitting of Operators
POD-ID	proper orthogonal decomposition with identification
POD	proper orthogonal decomposition
PPE	pressure Poisson equation
RANS	Reynolds-averaged Navier-Stokes

RBF	radial basis functions
RB	reduced basis
RELAP	Reactor Excursion and Leak Analysis Program
RIC	relative information content
ROM	reduced order model
SESAME	thermal-hydraulics Simulations and Experiments for the Safety Assessment of MEtal cooled reactors
SGDH	single gradient diffusion hypothesis
SIMPLE	Semi-Implicit Method for Pressure-Linked Equations
SISSA	Scuola Internazionale Studi Superiori Avanzati
SI	system identification
STH	system thermal-hydraulics
SUP	supremizer enrichment technique
SVD	singular value decomposition
UGent	Ghent University
U	Unsteady

Non-dimensional numbers

Gr	Grashof number
Pe_t	turbulent Peclet number
Pr	Prandtl number
Pr_t	turbulent Prandtl number
Re	Reynolds number
Re_t	turbulent Reynolds number
Ri	Richardson number
St	Stanton number

Greek symbols

α	thermal diffusivity	m^2/s
α_t	turbulence thermal diffusivity	m^2/s
β	thermal expansion coefficient	$1/K$
χ_i	i-th POD basis function for pressure	m^2/s^2
δ	Kronecker delta	
ϵ	error tolerance	
ϵ	relative prediction error	
ϵ_{local}	local time step continuity error	
η_i	i-th POD basis function for eddy viscosity	m^2/s
γ	parameter for radius of RBF kernel	
$\hat{\epsilon}$	basis projection error	
κ	von Karman constant	
λ	thermal conductivity	$W/(mK)$
λ_i	i-th eigenvalue	
μ	dynamic viscosity	$Pa \cdot s$
μ	parameter	
ν	kinematic viscosity	m^2/s

ν_t	eddy viscosity	m^2/s
ϕ_f	face flux	m^3/s
ϕ_i	i-th POD basis function for the transported scalar	K
ψ_i	i-th POD basis function for temperature	K
ρ	fluid density	kg/m^3
ρ_k	kinematic density for buoyancy force	
σ_ε	constant in ε equation	
σ_k	constant in k equation	
τ	penalty factor	
τ_w	wall shear stress	Pa
θ	mean (time or ensemble-averaged) temperature field	K
θ	temperature field	K
θ'	turbulent fluctuating component of temperature	K
$\tilde{\zeta}$	lifting function for temperature	K
ε	dissipation of turbulence kinetic energy	m^2/s^3
ξ_i	i-th POD basis function for thermal diffusivity	m^2/s
$d\Omega$	infinitesimal volume element	
λ	square matrix of eigenvalues	
ψ_i	i-th POD basis function for the face-centered velocity (Chapter 7)	m/s
φ_i	i-th POD basis function for velocity	m/s
ζ	normalized lifting function for velocity	m/s
$\tilde{\zeta}$	lifting function for velocity	m/s
ΔP	difference between two pressure levels	bar
Δt	difference between two time levels	s
$\Delta x, \Delta y$	cell length	m
Γ	boundary of Ω	
Ω	bounded domain	
Ω_h	discrete bounded domain	
Σ	face areas (Chapter 7)	
Θ_i	RBF kernels	
Λ	unit field	
Ω	diagonal matrix with cell-centered volumes (Chapter 7)	
Φ	matrix of cell-centered velocity modes (Chapter 7)	
Φ	matrix of modes	
$\Pi_{p \rightarrow f}$	linear interpolation operator	
Ψ	matrix of face-centered velocity modes (Chapter 7)	
X	matrix of cell-centered pressure modes (Chapter 7)	

Roman symbols

<i>a</i>	velocity coefficients
<i>b</i>	pressure coefficients
<i>b</i>	temperature coefficients (Chapter 5 and 6)
<i>c</i>	eddy viscosity coefficients
d_n	distance vector (Chapter 7)
<i>d</i>	distance vector

d	turbulence thermal diffusivity coefficients	
f	function velocity boundary condition	m/s
g	gravitational acceleration	m/s ²
n	outward pointing unit normal vector	
p_f	face-centered pressure (Chapter 7)	m ² /s ²
p_p	cell-centered pressure (Chapter 7)	m ² /s ²
q_r	ROM boundary vector	
r	position vector	
r_p	FOM boundary vector	
r_r	ROM boundary vector	
s_r	ROM boundary vector	
u'	turbulent fluctuating component of velocity	m/s
u, v	velocity fields	m/s
u_b	boundary velocity	m/s
u_b	vector containing velocity values at boundary (Chapter 7)	m/s
u_f	face-centered velocity (Chapter 7)	m/s
u_p	cell-centered velocity (Chapter 7)	m/s
$\langle u \rangle$	constant velocity field	m/s
c_f	skin friction coefficient	
c_p	specific heat	J/(kgK)
c_{ik}	set of known eddy viscosity coefficients	
d	dimension	
dS	infinitesimal element of a surface area	
f	function temperature boundary condition	K
f	wave frequency	Hz
f_1	damping function	
f_2	damping function	
f_μ	damping coefficient	
f_x	interpolation factor	
h	characteristic dimension	m
h	number of finite volume cells (Chapter 7)	
h	step height (Chapter 6)	m
k	turbulence kinetic energy	m ² /s ²
l_e	turbulence length scale	
m	number of faces (Chapter 7)	
n, e, s, w	compass directions at cell faces	
p	pressure field	m ² /s ²
p'	turbulent fluctuating component of pressure	m ² /s ²
p'_{rgh}	shifted pressure field divided by the reference density	m ² /s ²
p_{rgh}	shifted pressure field	Pa
q	wall heat flux	W/m ²
t	time	s
u_{bc}	wall or inlet velocity	m/s
w	weights RBF approach	
y^+	dimensionless wall distance	

A_r	ROM unknown matrix to be identified using a least-squares approach (Chapter 3)	
A_r	reduced convection term (Chapter 7)	
B_r	ROM matrix associated with the divergence term (continuity equation)	
B	face-to-center discrete divergence operator (Chapter 7)	
B_p	center-to-center discrete divergence operator (Chapter 7)	
C^{corr}	correlation matrix	
C_r	ROM convection matrix (Chapter 3)	
C_r	ROM convection tensor	
C	matrix associated with the convection term	
C_p	discrete convection operator (Chapter 7)	
D_r	ROM diffusion matrix	
DT_r	ROM diffusion matrix	
D	matrix associated with the diffusion term	
D_p	discrete Laplacian operator (Chapter 7)	
E_r	ROM turbulent matrix	
ET_r	ROM turbulent tensor	
F	body force term	kg/(s ² m ²)
G_r	ROM pressure gradient matrix	
G_f	discrete face gradient operator (Chapter 7)	
G_p	discrete gradient operator (Chapter 7)	
G_r	ROM gradient matrix (Chapter 7)	
H_r	ROM matrix associated with the buoyancy term	
I	identity matrix or tensor	
I	input vector (Appendix A)	
$I_{p \rightarrow f}$	linear interpolation operator	
J	Jacobian matrix (Appendix A)	
K_r	ROM convection matrix for the pressure Poisson Equation	
K_r	reduced convection term (Chapter 7)	
L_f	compact stencil Laplacian operator (Chapter 7)	
L_p	wide stencil Laplacian operator (Chapter 7)	
L_r	ROM Laplacian matrix (Chapter 7)	
M_r	ROM mass matrix	
N_r	ROM diffusion matrix for the pressure Poisson Equation	
O_r	matrix associated with the penalty term	
O	output vector (Appendix A)	
P_r	vector associated with the penalty term	
P_r	ROM diffusion matrix (Chapter 7)	
Q_r	ROM convection matrix for the heat equation	
Q	square matrix of eigenvectors	
R	residual	
S	Reynolds-averaged strain rate tensor	m ² /s ³
S_f	outward-pointing face area vector	
T_r	ROM matrix associated with the time-dependent BC for the pressure Poisson Equation	

U	mean (time or ensemble-averaged) velocity field	m/s
V_r	ROM mass matrix for the pressure Poisson Equation	
V_r	ROM mass matrix (Chapter 7)	
W_r	ROM mass matrix for the heat equation	
W_r	ROM mass matrix (Chapter 7)	
X_0	matrix containing the time-dependent coefficients	
X_1	matrix containing the time-dependent coefficients	
Y_r	ROM diffusion matrix for the heat equation	
\hat{A}_r	ROM convection tensor (Chapter 7)	
\hat{C}_r	reduced convection term (Chapter 7)	
\hat{D}_r	ROM diffusion matrix (Chapter 7)	
\hat{G}_r	ROM gradient matrix (Chapter 7)	
\hat{X}_r	projected fields	
A	wave amplitude	
B	wave off-set	
C_μ	model constant	
$C_{\varepsilon 1}$	constant in ε equation	
$C_{\varepsilon 2}$	constant in ε equation	
D	diameter	m
D	source term in k equation	m^2/s^3
E	source term in ε equation	m^2/s^2
E_{POD}	POD basis space	
G	operator (Appendix A)	
H	channel height	m
L	length	m
N	number of a quantity	
N_r	dimension of the reduced basis space	
N_{BC}	number of non-homogenous boundary conditions	
P	mean (time or ensemble-averaged) pressure field	m^2/s^2
P_k	turbulence kinetic energy production term	m^2/s^3
P_{rgh}	mean (time or ensemble-averaged) shifted pressure field	m^2/s^2
S_f	face area	m^2
S_ε	correction term	m^2/s^2
T	total or final simulation time	s
T	transported scalar	K
U	stream-wise velocity component (Chapter 6)	m/s
V	wall-normal velocity component (Chapter 6)	m/s
X	full order fields	
X_r	predicted fields	

Superscripts

'	homogenized
*	(new) input
*	intermediate
<i>corr</i>	correlation

k	k^{th} iteration
n	n^{th} time instance
Γ_1	boundary interface 1 (Appendix A)
Γ_2	boundary interface 2 (Appendix A)
B	divergence (Chapter 7)
C	convection (Chapter 7)
D	diffusion (Chapter 7)
T	transpose

Subscripts

0	initial
ℓ_2	computed with the ℓ_2 norm
τ	time steps penalty method
a	adiabatic
avg	average
b	boundary
b	bulk
c	cold wall
f	face(s) or face-centered
h	degrees of freedom of high fidelity model
h	heater
h	hot wall
in	inlet
k	k^{th} cell
max	maximum
min	minimum
out	outlet
p	cell-centered
r	degrees of freedom of reduced basis
r	reduced
ref	reference
s	snapshot(s)
t	time steps
u	upstream
x	cartesian coordinate along horizontal axis
y	cartesian coordinate along vertical axis
BC	boundary condition value
L_2	computed with the L_2 norm
N	neighboring cell center
P	cell center

Other symbols

∞	infinity
(\cdot, \cdot)	inner product
\mathbb{R}	set of real numbers

\mathcal{O}	order	
\mathcal{P}	parameter space	
\mathcal{D}	diffusion coefficient	m^2/s
\otimes	tensor product	
Δ	laplacian operator	
∇	gradient operator	
$\nabla \cdot$	divergence operator	
$\nabla \times$	curl operator	
$\frac{\partial f}{\partial x}$	partial derivative of a function f with respect to the variable x	

1

Introduction

1.1 The MYRRHA project

The global climate warming as a result of human activities requires large reductions in the emissions of carbon dioxide and other greenhouse gases worldwide [49]. Nuclear energy is an established low-carbon energy source and its expansion can make an important contribution to the effort of decarbonizing the energy supply as indicated by several reports [36, 49, 66]. However, current reactor designs need to be improved regarding safety, sustainability, efficiency and cost.

Therefore, new reactor concepts and designs are being researched in the frame of Generation IV [160, 196, 263]. Generation IV proposals for innovative nuclear systems represent significant improvements with respect to previous generations, especially regarding the improvement of safety, the reduction of the radiotoxicity and the amount of high-level nuclear waste, the optimization of the fuel economy and the resistance to proliferation [94, 162].

A large number of international projects have been promoted and funded to advance the studies on these innovative technologies. One of them that has been recognized as a high priority infrastructure for nuclear research in Europe is MYRRHA, which stands for Multi-purpose hYbrid Research Reactor for High-tech Applications [45].

Within the MYRRHA project, the Belgian Nuclear Research Centre SCK CEN is developing and designing a multi-functional experimental fast-spectrum irradiation facility. Its main purposes can be summarized as [2]:

- flexible fast-spectrum irradiation facility,
- demonstrator for minor actinides transmutation,
- demonstrator for accelerator-driven systems,
- European Technology Pilot Plant for lead-cooled fast reactors.

1.1.1 The MYRRHA facility

The MYRRHA facility is an accelerator-driven system (ADS) featuring a proton accelerator and a compact pool-type reactor that contains a core with mixed oxide fuel which is cooled by lead-bismuth eutectic (LBE), i.e. a heavy metal alloy [1]. The LBE also acts as the spallation target. Figure 1.1 shows a conceptual scheme of the major components of MYRRHA, i.e. proton accelerator, spallation target and reactor.

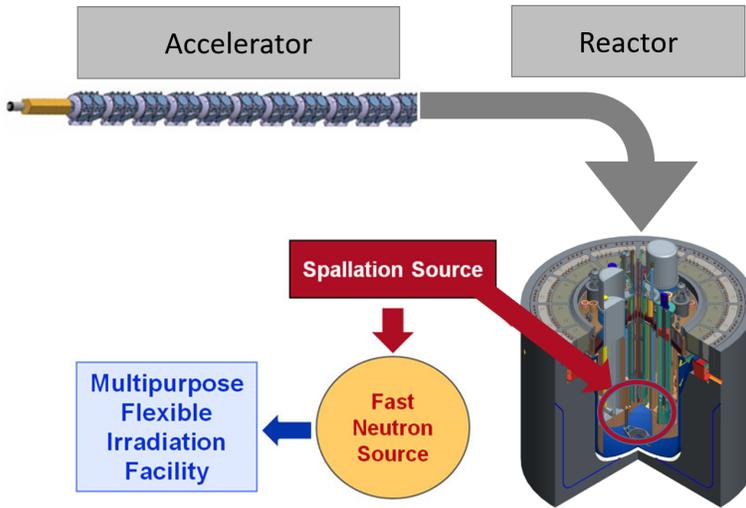


Figure 1.1: Conceptual scheme of the MYRRHA accelerator-driven system. (Figure adapted from [62, 257].)

MYRRHA will be able to operate in both sub-critical and critical mode. In sub-critical mode, a 400 m long linear accelerator provides protons at an energy of 600 MeV to the reactor. At the end of the accelerator beam line, the 4 mA proton beam hits a spallation target, composed by the same LBE acting as primary coolant, in the center of the reactor core (design revision 1.6) [3, 62]. The interaction of

the high energetic protons with the heavy metal target atoms generates a flux of fast neutrons together with other spallation products. This external neutron source maintains the fission reactions when MYRRHA is operated in sub-critical mode. The reaction stops automatically when the accelerator is stopped, which represent a solid safety feature of the MYRRHA reactor. Furthermore, the proton beam can be used for the Fusion Target Station, which is located just below the spallation target and designed to enhance research on accumulated irradiation damage and in-situ testing of materials, especially for fusion applications. Moreover, the ADS can be used to transmute highly radioactive minor actinides into shorter-lived nuclear waste, which lowers the burden on deep geological storage [4, 61, 156]. On the other hand, if the accelerator is decoupled from the reactor core, MYRRHA can operate in critical mode. Critical mode operation foresees adequate reactor shutdown systems to stop the fission chain reaction. The critical mode operation serves as demonstrator for the Lead-cooled Fast Reactor (one of the Generation IV reactor concepts) [1]. Moreover, the accelerator can feed protons to the multi-purpose Proton Target Facility for the production of radioisotopes and related research when it is not coupled to the reactor [4].

The MYRRHA fuel assembly is based on the design of the sodium fast reactor fuel assemblies [212]. It consists of a bundle of 127 cylindrical fuel pins that are arranged in a triangular pattern. The pins are separated from each other by helical wire-spacers which are wrapped around each fuel pin [139]. The cladding temperature of the fuel pins, which is a key parameter for the safe operation of the LBE cooled reactor, is subjected to limitations due chemical interaction of the cladding with the coolant [169]. Moreover, the average coolant bulk velocity is restricted to limit the erosion effects [51, 225].

On the other hand, using LBE as the primary coolant has several advantages: no direct chemical interaction with water and air, low neutronic absorption, low melting point (~ 400 K) and high boiling point (~ 1925 K) [73]. However, a drawback of using LBE as the primary coolant is the accumulation of radioactive isotopes, which could pose difficulties in terms of radiological releases during maintenance or in accidental conditions [261].

MYRRHA's primary cooling system is entirely contained in the primary vessel. Figure 1.2 shows a sketch of the MYRRHA reactor primary cooling system in its latest official design revision. The LBE flows from the lower plenum into the core (~ 543 K) to remove the core power (nominal power: 100 MW) and, from there, into the upper plenum (~ 600 K), which are indicated in Figure 1.3. Four primary heat exchanger units receive the LBE from the upper plenum. The LBE is then reinserted into the lower plenum via two primary pumps (each pump serving two heat exchangers). The lower plenum is separated from the upper plenum by a diaphragm, i.e. an inner vessel structure [45].

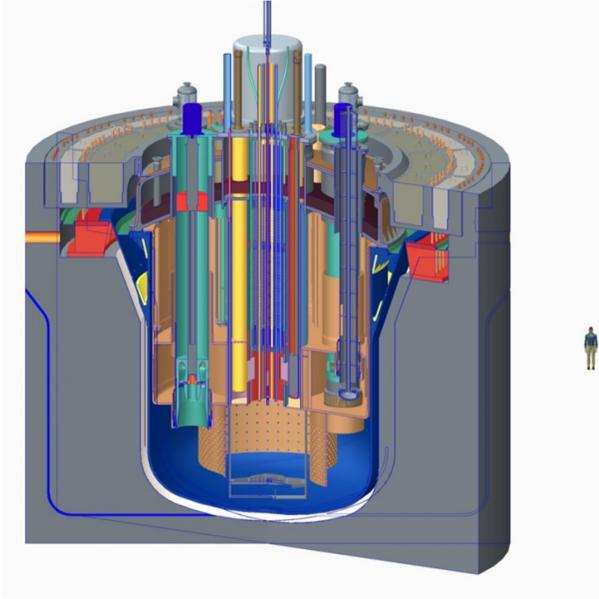


Figure 1.2: Sketch of the MYRRHA reactor design revision 1.6 [76].

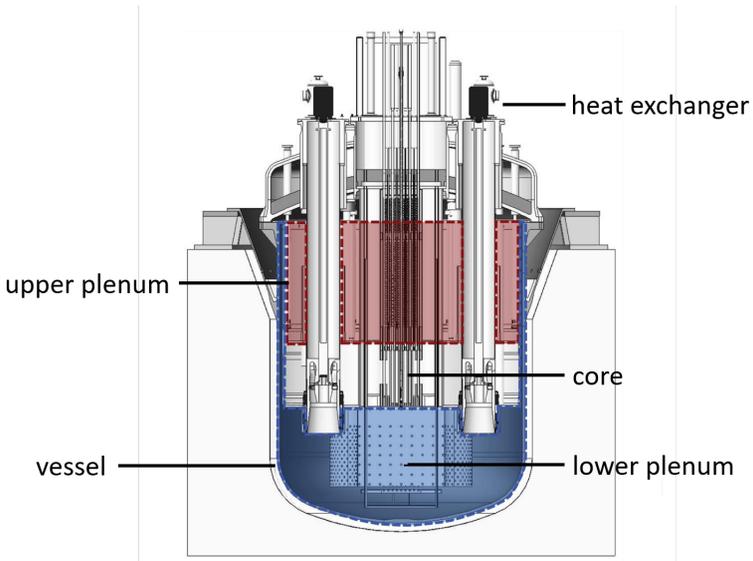


Figure 1.3: Schematic cross-section of the primary system of the MYRRHA reactor. (Figure adopted from [260].)

With regard to the thermal-hydraulic design and safety assessment of the primary cooling system, a major challenge is characterizing the complex transport phenomena in the coolant flow field, e.g. local flow mixing, buoyancy and thermal stratification in the lower and upper plenum of the primary vessel. These phenomena may impact the system response to operational and accidental transients, such as loss of forced flow, on the short-term (transition from forced to natural convection) and the long-term (safe shutdown state). In maintenance and accidental conditions, decay heat is removed passively via natural convection between the heat source at the core and the heat sinks at the primary heat exchangers, which are indicated in Figure 1.3 [62]. This avoids the use of active components and ensures a high level of intrinsic safety, as foreseen by Generation IV requirements. However, the development of the flow, for instance thermal stratification at low flow conditions, during the transition from forced to natural convection may have a detrimental effect on the efficiency of the passive residual heat removal. Therefore, reliable computational methods are required to accurately quantify the reactor's primary cooling system behavior in operational and accidental conditions and to handle complex geometries [260].

1.2 Motivation for reduced order modeling

The work presented in the European Union's Horizon 2020 research projects SESAME (started in 2015) on "Thermal-hydraulics Simulations and Experiments for the Safety Assessment of MEtal cooled reactors" made evident that next generation reactors, such as MYRRHA, require extensive computational characterization [162, 214].

There are two types of numerical codes for thermal-hydraulic analysis that are frequently used in the nuclear industry. The first type of codes are based one-dimensional (1D) models of physical transport phenomena, such as system codes (based mainly on lumped parameter approaches) and sub-channel codes (specifically used for fuel assembly and reactor core analysis). The second type of codes are field codes that are based on three-dimensional (3D) computational fluid dynamics (CFD) models [184].

Figure 1.4 presents the hierarchy of modeling approaches based on the computational cost and the amounts of resolved versus modeled physics. The system codes are, in general, based upon the solution of six balance equations for liquid and vapor. In addition, they use (quasi-steady state) heat transfer correlations to model the heat transfer between a solid, such as tubes or structures, and its surrounding fluid [27, 192]. The lumped parameter system codes and sub-channel codes are placed at the bottom of the pyramid due to the large amount of modeled physics. These codes have their limitations regarding the representation of complex flows due to their low number of degrees of freedom.

On the other hand, CFD codes are used to numerically simulate transient flows to accurately quantify the system behavior in accidental conditions and for geometrically complex cases [214, 260]. (Unsteady) Reynolds-averaged Navier Stokes ((U)RANS) simulation, large eddy simulation (LES) and direct numerical simulation are the most common turbulence models in CFD. DNS resolves all scales of motion, while LES resolves most of the scales and the smallest eddies are modeled. (U)RANS models solve only for the averaged quantities while the entire breakdown of vortices is modeled. Moreover, both DNS and LES always require solving in time and in three-dimensional space, while RANS can be used in steady form and allows solving in one or two-dimensions. The number of degrees of freedom required to solve flow problems using a CFD solver increases from RANS to DNS and, thus, with it the computational cost [107]. Therefore, these CFD methods are represented at the top part of the pyramid depicted in Figure 1.4.

CFD simulations are not only used in nuclear engineering, but are essential in many other engineering fields, among which aerospace, automotive, civil and naval. The CFD solvers are based on discretization methods, such as finite difference, finite element, finite volume or spectral element methods. The finite volume (FV) method is commonly used by commercial software and open-source codes.

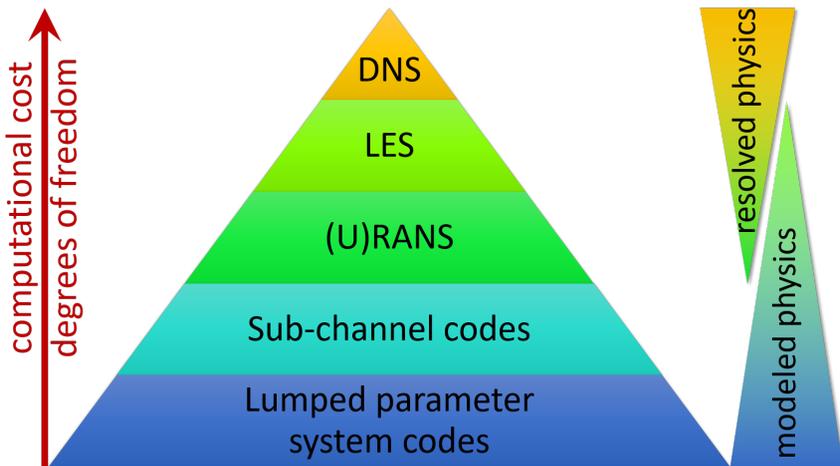


Figure 1.4: A pyramid representing the hierarchy of modeling approaches based on the computational cost and the amounts of resolved versus modeled physics. (Figure adopted from [119, 292].)

Regardless of the discretization strategy, CFD simulations are highly demanding in terms of CPU time and storage, especially for the simulation of turbulent flows, complex geometries, multi-physics phenomena and other types of complex flows. Moreover, the number simulations required for a nuclear reactor safety analysis is, in the majority of cases, beyond the possibilities of present hardware if a CFD code is used alone. Within the SESAME project, reduced and low resolution (coarse grid) CFD approaches were foreseen to lower the computational burden of full core simulations [199, 200, 275, 276]. Also multi-scale approaches, e.g. combining sub-channel codes with CFD codes, were foreseen [214]. Other possible solutions to increase the modeling capabilities are parallelization and High Performance Computing (HPC) [166] or simplifying the underlying mathematical models by introducing assumptions in the physics of the problem. However, HPC is expensive to install and operate. Making physics assumptions or coarse meshing is not always practical and can introduce sources of errors in the model. Also, the gain in computational speed of a STH/CFD coupled model is still too limited for aforementioned purposes [260].

This has motivated the development of reduced order modeling techniques that reduce the number of degrees of freedom of the high fidelity models. Mathematical techniques are used to extract “features” of the complex model in order to replace these by a smaller model. In that way, the required computational time and computer memory usage is reduced [24, 52, 110, 205]. Figure 1.5 depicts schematically the overall goal of reduced order modeling, namely to find the desired compromise between accuracy and the size of the model (i.e the computational cost), with respect to CFD simulations and lumped parameter models.

Rather recently, Lorenzi et al. [161] showed the potential of reduced order models (ROMs) in (nuclear) engineering with their work on performing ROM simulations with the object-oriented Modelica language [84] in the Dymola simulation environment [34, 65]. Modelica is specifically designed for the study of engineering system dynamics, which Lorenzi et al. used for the development of a plant simulator (the ALFRED reactor) for control purposes. They substituted a component of the model in Modelica with the ROM without compromising the rest of the model. Finally, reduced order modeling methods have been applied to a Rayleigh-Benard convection flow problem and to a pool with volumetric heat sinks and sources during the latest development of the SESAME project [71, 175].

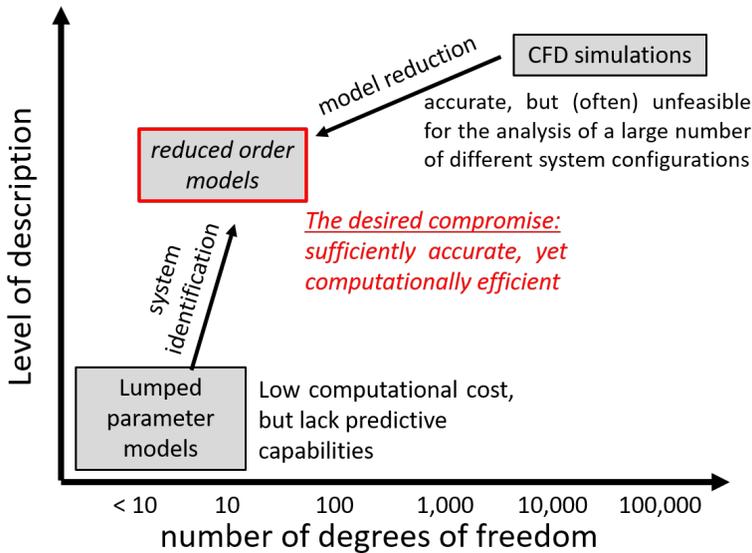


Figure 1.5: Schematic representation of the reduced order modeling approach as a compromise between the level of accuracy and the model size. (Figure adopted from [161, 231].)

1.3 The objectives and the framework of the Ph.D.

The focus of this Ph.D. thesis is the development of reduced order models in various engineering and academic settings not limited to nuclear engineering. Only computational fluid dynamics problems are considered that are based on linear and nonlinear dynamic models. Moreover, it is important to note that the solutions of the full order CFD models are considered to be “the true solutions” in this work. Neither the full order nor the reduced order models have been validated against experimental results or compared to computations from literature.

1.3.1 The objectives

The overall goal of this research is to develop reduced order models for fluid dynamics problems that accurately predict the behavior of the underlying full order model, but require less computational time. The main objectives of this research, addressed in this thesis, are to develop:

- a non-intrusive reduced order CFD model that does not require knowledge of the high fidelity solver’s discretization and solution algorithm,
- reduced order CFD models that are capable to efficiently, yet accurately, simulate nonlinear fluid flow problems,

- reduced order CFD models that can be used for parametric boundary condition problems,
- parametric reduced order CFD models for low Prandtl number fluid flow problems.

1.3.2 The framework

At the start of this Ph.D. project, SCK CEN had some initial experience with reduced order models (ROMs). A Ph.D. project on the real-time simulation of near-range atmospheric dispersion of the radioactive nuclides was carried out by Ververcken [273] in collaboration with the KU Leuven in Belgium. Only linear systems were investigated in this research. Also the Flow, Heat and Combustion Mechanics department (currently the department of Electromechanical, Systems and Metal Engineering) of Ghent University had worked on reduced order models, but for fluid and structural problems. Vierendeels et al. [277] had developed reduced order models to drive a coupling algorithm of partitioned fluid–structure interaction problems to compute the fully coupled solution of a black box fluid solver and a black box structural solver each time step. In addition, Degroote et al. [63], in collaboration with Massachusetts Institute of Technology (USA), investigated interpolation among reduced order matrices to obtain parameterized models for design, optimization and probabilistic analysis.

SCK CEN often uses commercial software, such as ANSYS Fluent, for performing CFD simulations [7, 92, 170, 260, 265]. The disadvantage of commercial software is that the solver’s discretization and solution algorithm are not accessible. Thus, they can basically be considered as black boxes. Given the previous experience of UGent with the development of reduced order models for black box fluid solvers, the first research goal was to develop non-intrusive reduced order models.

For the safety assessment of nuclear facilities, sensitivity studies are carried out to determine how plant parameters are influencing the flow and heat transfer in nuclear reactor components and systems [236]. Boundary conditions are essential for defining the numerical flow problems. However, boundary condition values can be uncertain if they come from measurements and/or they depend on the plant parameters. If so, the sensitivity to the boundary conditions needs to be analyzed. Developing reduced order CFD models that can be used for parametric boundary condition problems is therefore one of the main objectives.

Non-intrusive reduced order models for linear problems with parametric boundary conditions were developed successfully during the first year of this project. However, for the simulation of fluid flows in (nuclear) engineering applications, it is also necessary to develop reduced order models for low Prandtl fluid flow (such as liquid metal flows), buoyancy-driven flows and natural convection. These

phenomena require the developed of nonlinear reduced order models. Therefore, approaches that can also be used for nonlinear problems needed to be investigated.

In November 2017, SISSA's mathLab group (Italy) released ITHACA-FV (In real Time Highly Advanced Computational Applications for Finite Volumes) [240] as an open source code. A huge advantage of ITHACA-FV, other than that it is based on existing libraries, is its applicability to nonlinear fluid flow problems (e.g. the laminar incompressible Navier-Stokes equations). Nowadays, ITHACA-FV consists of the implementation of several reduced order modeling methods and algorithms for parametrized fluid dynamics problems [240].

After obtaining the ENEN+ (European Nuclear Education Network) mobility grant, I got the opportunity to join the mathLab group of Prof. Rozza for a research stay of six months. I closely worked with Dr. Giovanni Stabile, ITHACA-FV's main developer and maintainer and the other researchers working on ITHACA-FV. During this period, we developed reduced order models of buoyancy-driven flows. In addition, we improved some boundary control methods (that impose the boundary conditions in reduced order models).

Sokratia Georgaka, a Ph.D. student from Imperial College London (UK), was also working on ITHACA-FV in collaboration with the mathLab group. Our projects were both related to nuclear engineering. Moreover, we had the common interest of developing reduced order models for turbulent heat transfer problems. This led to a fruitful collaboration on the development of the mathematical framework of the reduced order models as well as on the numerical implementation of some parts of the ROM solvers. This collaboration has led to some joined publications (that are listed in the next section).

After the research stay at SISSA mathLab (April 2019), the collaboration with SISSA continued remotely. The work got extended to reduced order modeling of low Prandtl number fluid flows. Furthermore, nuclear engineering master student Giuseppe Spina from the University of Pisa started his master thesis project under my supervision at SCK CEN in September 2019. He worked during his six-month project on reducing the computational cost of a three-dimensional CFD solver coupled with a one-dimensional system thermal-hydraulics code by replacing the CFD solver with a reduced order model.

Finally, a collaboration with Dr. Benjamin Sanderse from the Scientific Computing group at CWI (The Netherlands) started in March 2020. The three-month Ph.D. internship at CWI was focused on the ROM challenges related to pressure stabilization and imposing the boundary conditions at the ROM level. The goal of the project was to improve the reduced order modeling approach for the incompressible Navier-Stokes equations on collocated grids. The ROMs were developed in the ITHACA-FV framework and the project was carried out in close collaboration with Dr. Giovanni Stabile from SISSA mathLab.

1.4 Thesis outline

The state of the art of reduced order modeling methods in computational fluid dynamics is addressed in Chapter 2. Also a description of the finite volume method is given. Furthermore, the POD-Galerkin method in the finite volume setting, that forms the basis of all reduced order methods developed in this work, is presented and the challenges are discussed.

The subsequent chapters of this dissertation address the aforementioned research goals in Section 1.3.1. These chapters are based on papers published in peer-reviewed journals and international conference proceedings that are listed in the next section.

Chapter 3 is devoted to non-intrusive reduced order modeling for linear CFD problems. A POD-based identification method is proposed, for which the reduced system matrices are identified using a least-squares technique. The ROM is of the same form as if the ROM would have been obtained with the intrusive POD-Galerkin method. Moreover, the penalty method is applied to impose the parametric (time-dependent) non-homogeneous Dirichlet boundary conditions in the ROM. The results of this chapter are published as [245].

Chapter 4 continues on boundary control strategies at the ROM level. The lifting function method is compared with a novel iterative penalty method for (time-dependent) parametric boundary condition problems. However, from this chapter on, intrusive reduced order models are developed of nonlinear flow problems, such as the incompressible Navier-Stokes equations, using the POD-Galerkin projection approach. The results of this chapter are accepted as [248].

Chapter 5 and 6 are devoted to coupling with heat transfer. Chapter 5 is about the development of a POD-Galerkin reduced order model of the Boussinesq approximation for buoyancy driven-enclosed flows. The temperature boundary conditions of the enclosed cavity are parametrized. Chapter 6 is focused on low-Prandtl number fluid turbulent flows. A reduced order model is developed of a turbulent convective buoyant flow of sodium over a backward-facing step. The parametric ROM is used to study solutions for different Richardson numbers, for which the associated heat flux is considered to be the corresponding varying physical parameter. The results of Chapter 5 and Chapter 6 are published as [244] and [249], respectively.

In Chapter 7, a novel reduced order model for incompressible flows is developed by performing a Galerkin projection based on a fully (space and time) discrete full order model formulation. This ‘discretize-then-project’ approach requires no pressure stabilization technique (even though the pressure term is present in the ROM) nor a boundary control technique (which were needed for the reduced order modeling methods described previously to impose the boundary conditions at the ROM level). The results of this chapter are under review as [246].

The last chapter (Chapter 8) contains an overall conclusion and recommendations for future work.

Finally, an application of reduced order modeling in nuclear engineering is given in Appendix A. The computational cost of a three-dimensional CFD solver coupled with a one-dimensional system thermal-hydraulics code is reduced by replacing the CFD solver with a reduced order model. Coupled simulations with the ROM are performed on parametric problems. The results are under review as [247].

1.5 List of publications underlying this thesis

The results of the publications are presented in this dissertation unless stated differently.

1.5.1 Publications in peer-reviewed journals

[245] S.K. Star, F. Belloni, G. Van den Eynde, J. Degroote, “POD-identification reduced order model of linear transport equations for control purposes”, *International Journal for Numerical Methods in Fluids*, **90**, 375–388 (2019).

[88] S. Georgaka, G. Stabile, K. Star, G. Rozza, MJ. Bluck, “A hybrid reduced order method for modelling turbulent heat transfer problems”, *Computers & Fluids*, **208**, 104615 (2020).

The results of this work are not presented in this dissertation.

[249] S.K. Star, G. Stabile, G. Rozza, J. Degroote, “A POD–Galerkin reduced order model of a turbulent convective buoyant flow of sodium over a backward-facing step”, *Applied Mathematical Modelling*, **89**, 486 - 503 (2021).

[248] S.K. Star, G. Stabile, F. Belloni, G. Rozza, J. Degroote, “A novel iterative penalty method to enforce boundary conditions in Finite Volume POD-Galerkin reduced order models for fluid dynamics problems”, *Communications in Computational Physics*, accepted on September 28, 2020.

[247] S.K. Star, G. Spina, F. Belloni, J. Degroote, “Development of a coupling between a system thermal-hydraulic code and a reduced order CFD model”, *Annals of Nuclear Energy*, **153**, 108056 (2021).

[246] S.K. Star, B. Sanderse, G. Stabile, G. Rozza, J. Degroote, “Reduced order models for the incompressible Navier-Stokes equations on collocated grids using a ‘discretize-then-project’ approach”, *International Journal for Numerical Methods in Fluids*, (2020), submitted on October 14, 2020.

1.5.2 Publications in international conference proceedings

[243] K. Star, J. Degroote, J. Vierendeels, G. Van den Eynde, F. Belloni “Reduced order modelling using a POD-based identification method for parameterized PDEs”, *7th European Conference on Computational Fluid Dynamics* (2018).

The results of this work are not presented in this dissertation.

[244] K. Star, G. Stabile, S. Georgaka, F. Belloni, G. Rozza, J. Degroote, “POD-Galerkin reduced order model of the Boussinesq approximation for buoyancy-driven enclosed flows”, *Building theory and applications: proceedings of M&C 2019*, 2452–2461 (2019).

2

Reduced order modeling methods in computational fluid dynamics

Fluid dynamics problems are generally described by the Navier–Stokes (NS) equations, which are a set of partial differential equations. The equations express mathematically the conservation of mass, the conservation of momentum and the conservation of energy. In this work, we focus on incompressible flows, i.e. the density of the fluid is considered constant in any infinitesimal volume of fluid moving in the flow [271]. In the first section, we describe the incompressible Navier–Stokes equations and how they are discretized with the finite volume method, which is partly published as [246]. In the second section, we give a general overview of reduced order modeling methods in fluid dynamics. In the subsequent sections, we explain the POD–Galerkin approach and challenges of reduced order modeling in the finite volume setting.

2.1 The incompressible Navier–Stokes equations

The governing equations to describe the fluid dynamics problem on a geometrical domain Ω , which coincides with the region of flow, are given by the unsteady incompressible Navier–Stokes equations. For a Newtonian flow with constant fluid density ρ , kinematic viscosity ν and thermal diffusivity α and without gravity and body forces, the general equations of mass, momentum and energy conservation

are given, respectively, by

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega, \quad (2.1)$$

$$\frac{\partial \mathbf{u}}{\partial t} = -\nabla \cdot (\mathbf{u} \otimes \mathbf{u}) + \nu \nabla \cdot (\nabla \mathbf{u}) - \nabla p \quad \text{in } \Omega, \quad (2.2)$$

$$\frac{\partial \theta}{\partial t} = -\nabla \cdot (\mathbf{u}\theta) + \alpha \nabla \cdot (\nabla \theta) \quad \text{in } \Omega, \quad (2.3)$$

where $\mathbf{u} = \mathbf{u}(\mathbf{x}, t)$ represents the vectorial velocity field that is evaluated at $\mathbf{x} \in \Omega \subset \mathbb{R}^d$ with $d = 2$ or 3 . Furthermore, $p = p(\mathbf{x}, t)$ is the normalized scalar pressure field, which is divided by the constant fluid density ρ , $\theta = \theta(\mathbf{x}, t)$ is the scalar temperature field and t denotes time. The right hand side of the momentum equations (Equation 2.2) contains a convection, diffusion and pressure gradient term, respectively.

Taking the divergence of both sides of Equation 2.2 and applying the continuity constraint of Equation 2.1 leads to the pressure Poisson equation (PPE):

$$\nabla^2 p = -\nabla \cdot (\nabla \cdot (\mathbf{u} \otimes \mathbf{u})) \quad \text{in } \Omega. \quad (2.4)$$

This equation ensures that continuity is satisfied and can therefore be used as an alternative for the equation of mass conservation (Equation 2.1) by solving for \mathbf{u} and p .

The equations are supplemented with the initial condition for velocity:

$$\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}) \quad \text{in } \Omega, \quad (2.5)$$

which is divergence free, i.e. $\nabla \cdot \mathbf{u}_0 = 0$, and the initial condition for temperature:

$$\theta(\mathbf{x}, 0) = \theta_0(\mathbf{x}) \quad \text{in } \Omega. \quad (2.6)$$

Boundary conditions are required to make the above problem well-posed. There are two main forms of boundary conditions: Dirichlet boundary conditions that prescribe the value of a variable on the boundary and Neumann boundary conditions that prescribe the gradient of a variable normal to the boundary. Other more complicated boundary conditions may specify a mixture of the boundary value and gradient on the boundary [77].

In this work, we consider three types of boundary conditions: wall, inflow and outflow. Correspondingly, we subdivide the boundary into $\Gamma = \Gamma_{wall} \cup \Gamma_{in} \cup \Gamma_{out}$.

Viscous fluids adjacent to a solid boundary such as a wall satisfy the no-slip condition, which states that the velocity of the fluid is equal to the velocity of the boundary:

$$\mathbf{u} = \mathbf{u}_{wall}(\mathbf{x}, t) \quad \text{on } \Gamma_{wall} \quad \text{for } t \geq 0, \quad (2.7)$$

where \mathbf{u}_{wall} is the wall velocity that is assumed to be known. In the case of fixed walls, $\mathbf{u}_{wall} = 0$.

The inflow boundary condition is of the same form as the wall boundary condition:

$$\mathbf{u} = \mathbf{u}_{in}(\mathbf{x}, t) \text{ on } \Gamma_{in} \text{ for } t \geq 0, \quad (2.8)$$

where \mathbf{u}_{in} is the velocity at the inlet boundary Γ_{in} that is assumed to be known.

If the problem contains solely wall/inflow velocity boundary conditions, it is also required that the following compatibility condition, which follows from integrating Equation 2.1 over Ω , is satisfied [97]:

$$\int_{\Gamma} \mathbf{n} \cdot \mathbf{u}_{bc} d\Gamma = 0 \text{ for } t \geq 0, \quad (2.9)$$

where \mathbf{u}_{bc} is either the wall (Equation 2.7) or inlet velocity (Equation 2.8) and \mathbf{n} denotes the outward pointing normal vector on the boundary Γ . As a consequence, the pressure can then only be determined up to a constant. This is remedied by imposing the pressure in a selected point in the domain.

For outflow boundaries, the normal component of the stress tensor is specified:

$$\mathbf{n} \cdot \nu \nabla \mathbf{u} - \mathbf{n} p = 0 \text{ on } \Gamma_{out} \text{ for } t \geq 0. \quad (2.10)$$

If the PPE (Equation 2.4) is used rather than the equation for mass conservation (Equation 2.1), the following boundary conditions apply in addition to Equations 2.7 and 2.8 for the wall and inflow boundary conditions, respectively:

$$\mathbf{n} \cdot \nabla p = \mathbf{n} \cdot (-\nabla \cdot (\mathbf{u} \otimes \mathbf{u})) \text{ on } \Gamma_{wall}, \Gamma_{in} \text{ for } t \geq 0. \quad (2.11)$$

The boundary conditions for the PPE are equal to Equation 2.10 in the case of an outflow boundary condition.

Similar to Equations 2.7 and 2.8, the wall and inflow boundary conditions for temperature can be specified, respectively, as:

$$\theta = \theta_{wall}(\mathbf{x}, t) \text{ on } \Gamma_{wall} \text{ for } t \geq 0, \quad (2.12)$$

and

$$\theta = \theta_{in}(\mathbf{x}, t) \text{ on } \Gamma_{in} \text{ for } t \geq 0, \quad (2.13)$$

where θ_{wall} is the wall temperature and θ_{in} is the temperature of the fluid at the inlet boundary Γ_{in} , which are assumed to be known. Furthermore, a constant heat flux can be prescribed, which is a typical Neumann boundary condition for temperature:

$$\mathbf{n} \cdot \nabla \theta = -\frac{q''_{bc}}{\lambda} \text{ on } \Gamma_{wall}, \Gamma_{out} \text{ for } t \geq 0, \quad (2.14)$$

where q''_{bc} the heat flux on the boundary and λ is the thermal conductivity of the fluid. If the heat flux is set to zero, the boundary condition is called insulated or adiabatic.

2.2 The finite volume discretization of the incompressible Navier-Stokes equations

Finite volume (FV) discretization methods for the incompressible Navier–Stokes equations are mainly applied on two types of grids: staggered and collocated [77]. FV schemes on staggered grids are known to intrinsically conserve mass, momentum and kinetic energy in space and time on Cartesian grids [108, 266]. Another favorable property of staggered grids is that the pressure–velocity coupling is inherently enforced, i.e. preventing odd-even decoupling of the pressure [77, 186]. On the other hand, the collocated grid arrangement offers significant advantages over the staggered grid approach. First of all, the code implementation is generally simpler (easier bookkeeping) [195]. In addition, the collocated grid shortens the computational time and reduces the required memory storage compared to staggered grids on complex solution domains [190, 298]. Therefore, collocated grids are widely used by popular commercial codes such as ANSYS Fluent [11] and STAR-CCM+ [47] and the open source code OpenFOAM [95].

We will solely develop reduced order models for the finite volume discretization method on collocated grids, which is shown in Figure 2.1, in this work. In this section, we describe the spatial and temporal discretization using the finite volume method on a collocated grid. We only discretize the mass and momentum conservation equations (Equations 2.1 and 2.2), i.e. for isothermal incompressible flow, to ease the derivations.

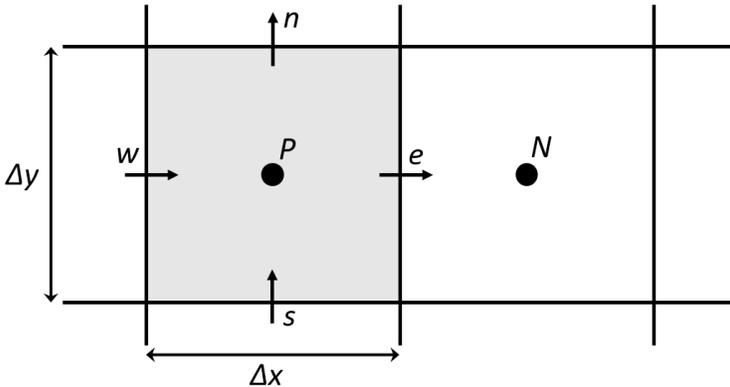


Figure 2.1: Two-dimensional collocated grid with the location of the unknowns at the center of a cell volume P and neighboring cell volume N . Δx and Δy the cell lengths and the arrows indicate the mass fluxes through the cell faces e , w , s and n of cell P .

An integral formulation of the governing equations is imposed to all closed cell volumes such that the conservation laws are satisfied locally [81, 271]. The

integral form of the conservation equations (Equations 2.1 and 2.2) for an arbitrary cell k are given by

$$\int_{\partial(\Omega_h)_k} \mathbf{n} \cdot \mathbf{u} dS = 0, \quad (2.15)$$

$$\int_{(\Omega_h)_k} \frac{\partial \mathbf{u}}{\partial t} d\Omega = - \int_{\partial(\Omega_h)_k} (\mathbf{n} \cdot \mathbf{u}) \mathbf{u} dS + \nu \int_{\partial(\Omega_h)_k} \mathbf{n} \cdot (\nabla \mathbf{u}) dS - \int_{\partial(\Omega_h)_k} \mathbf{n} p dS, \quad (2.16)$$

where $(\Omega_h)_k$ is the volume of cell k and $\partial(\Omega_h)_k$ is its boundary. $d\Omega$ is an infinitesimal volume element and dS is an infinitesimal element of surface area.

2.2.1 Spatial discretization

We detail the spatial discretization of each term of the equations in integral form (Equations 2.15 and 2.16) for an arbitrary cell k .

The discretization of the continuity equation (Equation 2.15) yields

$$\int_{\partial(\Omega_h)_k} \mathbf{n} \cdot \mathbf{u} dS = \sum_i^{N_f} \int_{S_{f,i}} \mathbf{n} \cdot \mathbf{u} dS \approx \sum_{i=1}^{N_f} \mathbf{S}_{f,i} \cdot \mathbf{u}_{f,i} = \sum_{i=1}^{N_f} \phi_{f,i} = 0, \quad (2.17)$$

where N_f is the total number of faces bordering the cell k and $\mathbf{S}_f = \mathbf{n} S_f$ is the outward-pointing face area vector with S_f the area of the particular face. Hence, the divergence-free constraint is not applied to the velocity, but to the face flux ϕ_f [161]. \mathbf{u}_f is the discrete face-centered velocity field.

The discretization of the pressure gradient term of Equations 2.16 yields

$$\int_{\partial(\Omega_h)_k} \mathbf{n} p dS = \sum_{i=1}^{N_f} \int_{S_{f,i}} \mathbf{n} p dS \approx \sum_{i=1}^{N_f} \mathbf{S}_{f,i} p_{f,i}, \quad (2.18)$$

where p_f are the discrete face-centered pressure field. p_f and \mathbf{u}_f are calculated for a cell face i , using appropriate interpolation schemes, as functions of their values at the center of the cells, in the following way, respectively

$$p_{f,i} = f_x p_P + (1 - f_x) p_N, \quad (2.19)$$

in the case of pressure and

$$\mathbf{u}_{f,i} = f_x \mathbf{u}_P + (1 - f_x) \mathbf{u}_N, \quad (2.20)$$

in the case of velocity, where f_x is the interpolation factor. It is assumed that the variables vary linearly between cell centers P and N as is shown in Figure 2.2. The differencing scheme used is called central differencing that is second order accurate (even on nonuniform meshes) [77]. Alternative discretization schemes have been developed such as upwind differencing schemes [264].

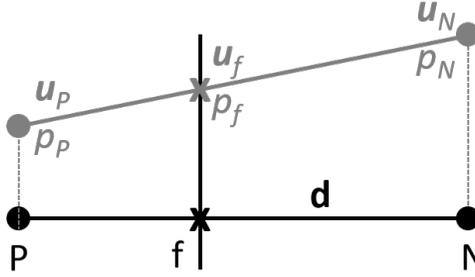


Figure 2.2: Schematic representation of the linear interpolation of velocity and pressure from the cell center to the face of the cell. \mathbf{d} is the distance vector between the cell centers P and N .

Applying a central discretization stencil to the velocity divergence (Equation 2.17) and the pressure gradient (Equation 2.18) together with collocated grids often results in spurious pressure oscillations [140]. These oscillations occur because the compatibility relation between the divergence and gradient operators is not satisfied (in contrast to a staggered grid) [77, 271]. This so-called checkerboard problem is caused by a wide stencil in the PPE, which yields a pressure-velocity decoupling of nearby grid points [60]. The basic solution for this problem is to use the Rhie–Chow interpolation [208] for the face velocities.

The discretization of the diffusion term of the momentum equations for orthogonal meshes yields

$$\int_{\partial(\Omega_h)_k} \mathbf{n} \cdot \nabla \mathbf{u} \, dS = \sum_{i=1}^{N_f} \int_{S_{f,i}} \mathbf{n} \cdot \nabla \mathbf{u} \, dS \approx \sum_{i=1}^{N_f} |\mathbf{S}_{f,i}| \frac{\mathbf{u}_N - \mathbf{u}_P}{|\mathbf{d}|}, \quad (2.21)$$

where \mathbf{d} is the distance vector between any adjacent cell centers P and N to a particular face as shown in Figure 2.2.

Finally, the discretization of the convection term, which is nonlinear, yields

$$\begin{aligned} \int_{\partial(\Omega_h)_k} (\mathbf{n} \cdot \mathbf{u}) \mathbf{u} \, dS &= \sum_{i=1}^{N_f} \int_{S_{f,i}} (\mathbf{n} \cdot \mathbf{u}) \mathbf{u} \, dS \\ &\approx \sum_{i=1}^{N_f} (\mathbf{S}_{f,i} \cdot \mathbf{u}_{f,i}) \mathbf{u}_{f,i} = \sum_{i=1}^{N_f} \phi_{f,i} \mathbf{u}_{f,i}. \end{aligned} \quad (2.22)$$

This shows that convection term depends on the face flux ϕ_f . In this work, the nonlinearity of the convection term is quadratic, because face centered velocity is obtained via linear interpolation of cell centered velocity (Equation 2.20) [129].

2.2.2 Temporal discretization

While the spatial discretization is performed in the spatial domain, the temporal discretization involves the evaluation of the integral of the transient term (left hand side of Equation 2.16). The time integration can be performed using explicit and implicit schemes. Explicit transient schemes compute the values of the quantities of interest based on the values from the previous time step. The Forward Euler method, or simply the Euler method, is a first-order method and the most basic method for numerical integration. Its implicit analog is the Backward Euler Method, which is also first order. Implicit schemes are typically solved by using either direct or iterative methods. Implicit time integration methods are, generally, more stable than explicit schemes [77, 271].

2.2.3 Velocity-pressure coupling

Common implicit solution algorithms for collocated grids that are frequently used in engineering applications are pressure-based methods, also known as pressure correction or segregated methods, such as the SIMPLE and PISO algorithms [77, 271]. PISO stands for Pressure-Implicit with Splitting of Operators [128]. As the name already reveals, the temporal discretization is treated using an implicit differencing scheme. The PISO algorithm [128] consists of a predictor step, where an intermediate velocity field is solved using pressure from the previous time step, followed by a number of corrector steps (often two) for the intermediate computed velocity and pressure fields such that (discretely) divergence free velocity fields are obtained [278].

Another popular algorithm for the velocity-pressure coupling is the PIMPLE algorithm [77], which is a combination of PISO and SIMPLE (Semi-Implicit Method for Pressure-Linked Equations [187]). The PIMPLE algorithm has some advantages (over PISO) when dealing with large time steps (i.e. the maximum Courant number may be above unity) of transient flow simulations or for complex geometries [116].

2.3 General overview of reduced order CFD methods

There exist many types of reduced order modeling methods that can be categorized in different ways [28], e.g. a posteriori vs. a priori ROMs [219], intrusive vs. non-intrusive or physics-based vs. data-driven. Several review papers and books on the development of reduced order modeling techniques can be found in literature. We list here a couple of them: Lucia et al. [163] (2004) reviewed methods based on Volterra series representations, the proper orthogonal decomposition (POD), and

harmonic balance. Moreover, they discussed their applicability to various problems in computational physics. Benner et al. [24] (2015) surveyed the state of the art in projection-based parametric model reduction methods. They highlight the role of parametric model reduction in design, control, optimization, and uncertainty quantification. Recently, Rozza et al. [219] (2018) presented advances in the development of reduced order models with POD, reduced basis (RB) and dynamic mode decomposition (DMD) methods with the focus on computational fluid mechanics and related applications.

2.3.1 Projection based reduced order models

In this work, we make a distinction between reduced order modeling methods that are projection-based and those that are not, such as truncation-based methods [215, 285], goal-oriented methods [35] and low degree-of-freedom models that are based on input-output data [207].

The basic principle of the projection-based methods is to retain the essential physics and dynamics of a high fidelity model by projecting the partial differential equations (PDEs) describing the fluid problem onto a low dimensional space, called the reduced basis space [217, 269]. The result is a physics-based model that is reduced in size [110]. Examples of methods to determine the reduced basis are greedy algorithms [24, 201], the dynamic mode decomposition [147, 226, 256] and the popular proper orthogonal decomposition technique [164, 235].

The proper orthogonal decomposition technique was introduced by Lumley [164] to study the coherent structures in experimental turbulent flows. The technique is also known as the Karhunen–Loeve expansion, principal component analysis or empirical orthogonal functions. POD is used to formulate an optimal basis spanned by modes to represent the most significant features of a dynamical system and is, therefore, widely used in the development of reduced order models. Moreover, the POD technique is commonly used for incompressible flows due to its optimal convergence property and its applicability to nonlinear systems [25, 26].

A classical projection-based method is the POD-Galerkin projection approach for which the reduced basis space is spanned by POD modes [203]. These modes are obtained by applying POD on a set of high fidelity solutions (called snapshots) [110, 203, 235]. The projection-based methods are mostly used in combination with a finite element method as high fidelity model [48]. However, POD-Galerkin methods for finite volume approximations have gained more and more attention in the past years [43, 105, 149, 161, 241] due to the frequent use of the FV method in industry as well as in academics [72, 191, 251]. The FV method owes its popularity to its robustness [72] and its local and global conservation properties [81, 271].

In the modeling and computation of industrial turbulent flows, RANS simula-

tion is often preferred due to its relatively lower computational cost in comparison with the more detailed large eddy simulation and direct numerical simulation. The POD-Galerkin approach has recently been used by Lorenzi et al. [161] and Hijazi et al. [111, 112] to reduce the RANS equations in a finite volume framework. Stabile et al. [238] used a different POD-Galerkin based approach for the turbulence closure, namely the variational multi-scale approach. Other recent efforts that deal with POD-based ROMs using an LES approach for the turbulence modeling can be found in [30, 91, 294]. On the other hand, Carlberg et al. [44] and Xiao et al. [288] presented a Petrov-Galerkin projection approach for the reduced order modeling of the Navier-Stokes equations.

2.3.2 Non-intrusive reduced order models

The main issue of the POD-Galerkin method is that knowledge of the solver's discretization and solution algorithm is required in order to perform the Galerkin projection. Therefore, the method cannot be used for most (commercial) software. Instead, non-intrusive ROMs (NIROM) [290] are using, for instance, a sparse grid collocation approach [286, 289] or interpolation [14, 279, 286, 289, 291] to calculate the POD coefficients. On the other hand, data-driven techniques, such as system identification (SI), are using the input/output data of a dynamical system to identify a low-dimensional system that approximately describes the dynamics of a high-dimensional system [274] with a set of low-order ordinary differential equations (ODEs). Examples are the dynamic mode decomposition, first introduced by Schmid [226] as a method for extracting coherent dynamic flow structures from a set of snapshots, and the Krylov-subspace projection-based methods as shown by Vervecken et al. [272] for the convection - diffusion equation [283]. A disadvantage of SI methods is that consistency issues can occur in the reduced system for parameterized problems [10]. A non-intrusive reduced order model, based on the POD-Galerkin projection method, for linear CFD problems is developed in Chapter 3.

2.4 POD-Galerkin method in the finite volume setting

The reduced order models that are developed in this work are based on the POD-Galerkin technique, although different variations are applied in the next chapters. In this section, we apply the POD technique and the Galerkin projection on the problem described by the governing equations as defined in the previous section (Equations 2.15 and 2.16) to explain the overall procedure to derive a POD-based reduced order model.

First of all, we assume that the full order solutions can be expressed as a linear combination of orthonormal spatial modes, φ_i , multiplied by time-dependent coefficients, a_i [149, 161, 241]. The velocity fields are approximated by

$$\mathbf{u}(\mathbf{x}, t) \approx \mathbf{u}_r(\mathbf{x}, t) = \sum_{i=1}^{N_r^u} \varphi_i(\mathbf{x}) a_i(t), \quad (2.23)$$

with N_r^u the number of velocity modes. The subscript r denotes quantities associated to the ROM.

Similarly, the pressure fields are approximated by

$$p(\mathbf{x}, t) \approx p_r(\mathbf{x}, t) = \sum_{i=1}^{N_r^p} \chi_i(\mathbf{x}) b_i(t), \quad (2.24)$$

where χ_i are the pressure modes and b_i are the corresponding time-dependent coefficients with N_r^p the number of pressure modes.

2.4.1 Proper orthogonal decomposition

The proper orthogonal decomposition method is used to create a reduced set of basis functions, or so-called modes, governing the essential dynamics of the full order model (FOM). Full order solutions are collected at certain time instances, the so-called snapshots. The snapshots can be obtained from detailed numerical simulations, as in this work, but they can also be collected from experimental data [13, 41, 115]. However, the snapshots do not necessarily have to be collected at every time step for which the full order solution is calculated.

The main assumption of POD is that the system's dynamics are governed by a reduced number of dominant modes, N_r , which are orthogonal to each other. Therefore, the POD basis is optimal when the difference between the snapshots and the projection of the snapshots on the basis functions is minimal for a certain norm. The L_2 norm is preferred for discrete numerical schemes [241].

The optimal POD basis space for the velocity, $E_{POD}^u = \text{span}(\varphi_1, \varphi_2, \dots, \varphi_{N_r^u})$, is constructed as

$$E_{POD}^u = \arg \min_{\varphi_1, \dots, \varphi_{N_r^u}} \frac{1}{N_s^u} \sum_{n=1}^{N_s^u} \left\| \mathbf{u}^n - \sum_{i=1}^{N_r^u} (\mathbf{u}^n, \varphi_i)_{L_2(\Omega)} \varphi_i \right\|_{L_2(\Omega)}^2, \quad (2.25)$$

where $\mathbf{u}^n = \mathbf{u}(t^n)$ with t^n the time at the n^{th} time step and $(\cdot, \cdot)_{L_2(\Omega)}$ the L_2 inner product of the functions over the domain Ω . N_s^u is the number of velocity snapshots and $N_r^u \leq N_s^u$ is the number of velocity modes.

Similarly, the optimal POD basis space for the pressure, $E_{POD}^p = \text{span}(\chi_1, \chi_2, \dots, \chi_{N_r^p})$, is given by the minimization problem:

$$E_{POD}^p = \arg \min_{\chi_1, \dots, \chi_{N_r^p}} \frac{1}{N_s^p} \sum_{n=1}^{N_s^p} \left\| p^n - \sum_{i=1}^{N_r^p} (p^n, \chi_i)_{L_2(\Omega)} \chi_i \right\|_{L_2(\Omega)}^2, \quad (2.26)$$

where $p^n = p(t^n)$, N_s^p is the number of pressure snapshots and $N_r^p \leq N_s^p$ is the number of pressure modes.

One way to compute the velocity POD modes is by applying the singular value decomposition (SVD) to the snapshot matrix. However, the SVD approach is often computationally more expensive than solving an eigenvalue problem on the snapshots to determine the basis functions, especially when the dimension of the grid, used to discretize the domain, is increased [239, 241]. Instead, the velocity POD modes are obtained by solving the following eigenvalue problem on a set of velocity snapshots:

$$\mathbf{C}^u \mathbf{Q}^u = \mathbf{Q}^u \boldsymbol{\lambda}^u, \quad (2.27)$$

with

$$C_{ij}^u = (\mathbf{u}^i, \mathbf{u}^j)_{L_2(\Omega)} \quad \text{for } i, j = 1, \dots, N_s^u, \quad (2.28)$$

where $\mathbf{C}^u \in \mathbb{R}^{N_s^u \times N_s^u}$ is the correlation matrix of velocity snapshots, $\mathbf{Q}^u \in \mathbb{R}^{N_s^u \times N_s^u}$ is a square matrix of eigenvectors and $\boldsymbol{\lambda}^u \in \mathbb{R}^{N_s^u \times N_s^u}$ is a diagonal matrix containing the eigenvalues which are sorted in descending order [149]. The velocity POD modes, φ_i , are then constructed as follows

$$\varphi_i = \frac{1}{N_s^u \sqrt{\lambda_i^u}} \sum_{n=1}^{N_s^u} \mathbf{u}^n Q_{in}^u \quad \text{for } i = 1, \dots, N_r^u. \quad (2.29)$$

Similarly, the pressure POD modes, χ_i , are obtained by solving the following eigenvalue problem on a set of pressure snapshots:

$$\mathbf{C}^p \mathbf{Q}^p = \mathbf{Q}^p \boldsymbol{\lambda}^p, \quad (2.30)$$

where $\mathbf{C}^p \in \mathbb{R}^{N_s^p \times N_s^p}$ is the correlation matrix of pressure snapshots, $\mathbf{Q}^p \in \mathbb{R}^{N_s^p \times N_s^p}$ is a square matrix of eigenvectors and $\boldsymbol{\lambda}^p \in \mathbb{R}^{N_s^p \times N_s^p}$ is a diagonal matrix containing the eigenvalues. The pressure POD modes, χ_i , are then constructed as follows

$$\chi_i = \frac{1}{N_s^p \sqrt{\lambda_i^p}} \sum_{n=1}^{N_s^p} p^n Q_{in}^p \quad \text{for } i = 1, \dots, N_r^p. \quad (2.31)$$

The number of POD modes, N_r^u and N_r^p , to span the reduced basis spaces can be chosen according to the eigenvalue decay of eigenvalues $\boldsymbol{\lambda}^u$ and $\boldsymbol{\lambda}^p$ [241].

2.4.2 Galerkin projection

To obtain a reduced order model, the POD is combined with the Galerkin projection, for which the full order momentum Equations 2.2 are projected onto the reduced POD basis space of velocity:

$$\left(\varphi_i, \frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u} \otimes \mathbf{u}) - \nu \nabla \cdot (\nabla \mathbf{u}) + \nabla p \right)_{L_2(\Omega)} = 0. \quad (2.32)$$

Solely projecting the momentum equations onto the reduced velocity basis space does not result in a closed system at reduced order level. Therefore, the continuity equation (Equation 2.1) is projected on the reduced basis space of pressure:

$$(\chi_i, \nabla \cdot \mathbf{u})_{L_2(\Omega)} = 0. \quad (2.33)$$

Substituting the approximations of the velocity and pressure fields (Equations 2.23 and 2.24) into Equations 2.32 and 2.33, the following reduced system of equations, which consists into a system of ODEs, is obtained:

$$\begin{cases} \mathbf{B}_r \mathbf{a} = \mathbf{0}, \\ \mathbf{M}_r \dot{\mathbf{a}} + \mathbf{C}_r(\mathbf{a}) \mathbf{a} - \nu \mathbf{D}_r \mathbf{a} + \mathbf{G}_r \mathbf{b} = \mathbf{0}, \end{cases} \quad (2.34)$$

with

$$\begin{aligned} B_{r_{ij}} &= (\chi_i, \nabla \cdot \varphi_j)_{L_2(\Omega)}, M_{r_{ij}} = (\varphi_i, \varphi_j)_{L_2(\Omega)}, \\ D_{r_{ij}} &= (\varphi_i, \nabla \cdot (\nabla \varphi_j))_{L_2(\Omega)}, G_{r_{ij}} = (\varphi_i, \nabla \chi_j)_{L_2(\Omega)}, \end{aligned} \quad (2.35)$$

where $\mathbf{B}_r \in \mathbb{R}^{N_r^p \times N_r^u}$ is the reduced matrix associated with the divergence term, $\mathbf{M}_r \in \mathbb{R}^{N_r^u \times N_r^u}$ is the reduced matrix associated with the mass term, the 'over-dot' indicates the time derivative, $\mathbf{D}_r \in \mathbb{R}^{N_r^u \times N_r^u}$ is the reduced matrix associated with the diffusion term and $\mathbf{G}_r \in \mathbb{R}^{N_r^u \times N_r^p}$ is the reduced matrix associated with the pressure gradient term. These reduced matrices can be precomputed during an offline stage except for the nonlinear term, $\mathbf{C}_r(\mathbf{a})$, associated with the convection term. Instead, the third order tensor \mathbf{C}_r is stored [204, 239], whose entries are given by

$$C_{r_{ijk}} = (\varphi_i, \nabla \cdot (\varphi_j \otimes \varphi_k))_{L_2(\Omega)}. \quad (2.36)$$

The contribution of the reduced convection term, $\mathbf{C}_r(\mathbf{a}) \mathbf{a}$, to the residual of Equation 2.34, \mathbf{R} , is evaluated at each iteration during the ROM simulations, or so-called online stage, as

$$R_i = \mathbf{a}^T C_{r_{i\bullet\bullet}} \mathbf{a}. \quad (2.37)$$

The dimension of the tensor \mathbf{C}_r (Equation 2.36) is growing with the cube of the number of modes used for the velocity space and therefore this approach may lead

in some cases, especially when a large number of basis functions are employed, to high storage costs. Other approaches, such as EIM-DEIM [23, 287] or Gappy-POD [44] may be more affordable [241].

However, following the approach described above, often leads unphysical results. In the next section, we will specify the challenges of reduced order modeling in the finite volume setting.

2.5 Challenges of reduced order modeling in the finite volume setting

Despite the potential and the increasing popularity of FV-based POD-Galerkin reduced order models for all sorts of applications, they tend to have issues with accuracy and exhibit numerical instabilities [127, 133, 241]. Challenges regarding velocity-pressure coupling and satisfying the boundary conditions at ROM level make it difficult to generalize the ROM methods such that they can be applied to any problem. We list the most common challenges of reduced order modeling for computational fluid dynamics problems (in the finite volume setting) in this section. An overview of which challenges are addressed in the next chapters is provided at the end of this section (Section 2.5.6).

2.5.1 Nonlinear problems

Many of the challenges are related to the nonlinearity of the CFD problems. The Navier–Stokes equations are, generally, a coupled set of nonlinear partial differential equations [22]. There are no general analytical methods for solving nonlinear PDEs. Therefore, each problem must be considered individually. Complex fluid dynamics problems are generally solved using discretization methods such as finite difference, finite element, finite volume or spectral element methods.

The nonlinearity of the problems is, actually, two-fold as we can distinguish between spatial and temporal nonlinearities. The first type of nonlinearity refers to the nonlinear behavior of a spatial state variable [163]. The main nonlinear term of the NS equations is the convection term of the momentum equations that describes the effect of acceleration of a flow with respect to space [115, 180]. Only in some specific cases, such as Stokes flow or the convection-diffusion transport equation, the NS equations can be simplified to linear equations. Chapter 3 is devoted to the linear convection-diffusion equation.

The second type is the nonlinear coupling of the time-varying states in the NS equations. The mass and momentum equations are coupled via the velocity. The pressure appears only as a source term in the momentum equation. Therefore, there is no evolution equation for the pressure. This means that the reduced system of equations are also coupled and need to be solved iteratively at reduced order

level. To ensure the consistency between the FOM and the ROM, next to an initial condition for velocity, often an initial guess for pressure is required for the system to converge more easily [40]. Then there are highly nonlinear problems, such as buoyancy driven flow problems that require an additional coupling between the nonlinear momentum and energy equations. A ROM for buoyancy driven flow is developed in Chapter 5.

Finally, it is important to note that POD is a linear technique in the sense that all the solutions of interest can be approximated via a linear combination of a set of basis functions. Nevertheless, it is possible to capture the nonlinearity of the system with the POD technique, because it can account for the nonlinear coupling of terms acting within the linear space defined by the basis functions [163].

2.5.2 Offline–online decomposition

To obtain an efficient ROM, the procedure is ideally split into a so-called offline phase and an online phase [188, 201].

In the offline stage, solutions of the high fidelity model are obtained, which are called snapshots. In this work, the numerical solutions are obtained with the use of the finite volume discretization technique. The solutions are of the order of N_h number of degrees of freedom. A number of solutions, collected over time or for different parameter values (which are chosen in an optimal way) are then stored. Subsequently, they are used to generate a reduced basis of a much smaller order $N_r \ll N_h$. In this work, the reduced basis spaces are generated using POD. All terms needed for the reduced order model formulation (2.34) are determined during the offline stage. The reduced matrices associated with the linear terms (Equation 2.35) are precomputed and stored during the offline stage. The nonlinear terms, such as the convection term (Equation 2.36), are also precomputed during the offline stage, but are stored as a third order tensor. The construction of the tensor needs to be done in an efficient manner to keep the offline-online splitting.

In the online stage, the reduced system of equations (Equation 2.34) is solved for the velocity and pressure coefficients in certain time period or for a certain parameter value, which we explain further in the next sections.

2.5.3 Stability

Reduced order models often have to deal with numerical instabilities that are usually in the form of unphysical numerical oscillations [124]. Sometimes the numerical instabilities are so strong that the ROM solutions blow up. There are several sources of numerical instability of ROMs for fluid flows [41, 120, 241], among which:

- Spurious pressure modes if the inf-sup condition for the reduced system is not fulfilled;

- Conservation law preservation;
- The non-homogeneous boundary conditions that are, generally, not being satisfied by the ROM;
- The convection-dominated and high Reynolds number regimes;
- Loss of dissipation;
- Transient dynamics;
- The existence of unphysical local phenomena.

We address these sources in the next four subsections. The first subsection is devoted to the numerical realization of the pressure-term representation in the ROM, which is also related to conservation law preservation. The second subsection is about enforcing the boundary conditions in the ROM. In the third subsection, we discuss the high Reynolds number regime, i.e. turbulent flows and the issue related to dissipation. We also explicitly point out the challenges related to low Prandtl number turbulent flows, as these type of flows are of interest in nuclear facilities such as the MYRRHA project. In the last subsection, we underline the issues related to transient dynamics. We do not go deeper into the existence of unphysical local phenomena, such as steep (temperature) gradients, as the local phenomena are very problem specific.

2.5.3.1 Pressure stabilization

A ROM that is based on a coupled scheme does not always fulfill the (discrete) inf-sup condition for the reduced system [241]. In the case of reduced basis methods, the reduced inf-sup condition is a sufficient condition to have a unique solution that depends on the velocity and pressure approximations [41, 149]. Thus, the condition needs to be satisfied to possess the existence and uniqueness of the solution of the problem [33]. In the POD-ROM framework, the discrete inf-sup condition cannot, generally, be checked a priori [41].

Several works on POD-Galerkin reduced order models have shown that the pressure gradient term disappears from the momentum equations when the basis for the velocity-field is (discretely) divergence-free [115, 165, 224] as pressure acts as a Lagrange multiplier of the divergence-free constraint [41]. The resulting ‘velocity-only’ models do not require a pressure stabilization technique. This can avoid the inf-sup stability issues [223]. However, the velocity fields in a collocated grid setting are typically only approximately divergence free. As a result, (pressure) stabilization techniques are often required. Possible stabilization techniques found in literature are:

- The supremizer enrichment of the velocity space in order to meet the inf-sup condition [19];
- Exploitation of a pressure Poisson equation during the projection stage [5, 180];
- A decoupled approach for the reduced velocity-pressure pair [25, 41, 161];

- Constructing a ROM for the pressure that only uses the velocity POD modes [180].
- Adding a pressure-stabilizing Petrov–Galerkin term to account for stabilizing the violated discrete inf-sup condition and to recover the reduced pressure [18, 41, 296];

Two of the pressure stabilization techniques are compared in the work of Stabile et al. [241] in the FV setting: the supremizer enrichment of the velocity space in order to meet the inf-sup condition (SUP) and the exploitation of a pressure Poisson equation during the projection stage. The SUP-ROM performed about an order worse with respect to the PPE-ROM for what concerns the velocity field, but better for what concerns the pressure field. This difference can be explained by the fact that within a supremizer stabilization technique, the POD velocity space is enriched by non-necessary (for the correct reproduction of the velocity field) supremizer modes.

The reduced order models tested Chapter 4 and Appendix A are developed using the PPE approach, while the steady-state ROM described in Chapter 6 is constructed for which the velocity and pressure modes share the same basis.

Moreover, the work of Stabile et al. [241] also shows that even using the supremizer enrichment technique or exploiting the PPE, the ROM velocity and pressure fields are about one or two orders less accurate than the fields obtained by projecting the FOM solutions onto the POD basis spaces. This is even the case for nonparametric laminar flow cases, such as the lid-driven cavity flow problem [241]. Therefore, we develop an efficient nonparametric ROM for the incompressible NS equations on collocated grids that does not require a pressure stabilization in Chapter 7. The novel reduced order models are developed by performing a Galerkin projection based on a fully (space and time) discrete full order model formulation. As a result, the ROM formulations for the incompressible NS equations, which includes pressure, are fully corresponding with the FOM formulations and no additional pressure stabilization method is needed.

2.5.3.2 Boundary conditions

Boundary conditions (BC) are essential for defining flow problems. The ROM needs to be able to satisfy the boundary conditions of the underlying FOM. POD-based reduced order models naturally satisfy the homogeneous BCs as the POD basis functions, which are defined as linear combination of the snapshots, satisfy automatically the homogeneous BCs. The same accounts for the ROM. However, the non-homogeneous BCs are, in general, not satisfied in the ROM as not all linear combinations of snapshots used for the creation of the POD basis will satisfy the non-homogeneous BCs of the FOM. Furthermore, the BCs are not explicitly present in the reduced system and, therefore, they cannot be controlled directly [161]. Two common approaches for handling the BCs are the lifting func-

tion method and the penalty method [93]. The aim of the lifting function method is to have homogeneous POD modes and to enforce the BCs by means of a properly chosen lifting function in the ROM. On the other hand, the penalty method enforces the BCs in the ROM with a penalty factor.

An advantage of the lifting function method is that the functions can be determined in the offline phase. Nevertheless, it may be hard to find functions that will lead to an accurate ROM. The functions are system-specific, need to be physical and they have to satisfy the divergence free constraint in order to retain the divergence-free property of the snapshots [239]. Therefore, extensive testing of ROMs for different functions can be necessary. On the other hand, the penalty factor cannot be determined a priori [93], which is a disadvantage of the penalty method. Moreover, the penalty factor needs to be chosen above a certain threshold to enforce the BCs in the ROM, but can lead to an inaccurate ROM solution if it is too high [234]. Determining (a range of) suitable penalty factors is usually done by numerical experimentation [29, 93, 134, 161]. In Chapter 4 the implementation of an iterative solver to determine the penalty factor to save time compared to performing numerical experimentation manually is proposed. Also, detailed descriptions of both the lifting function method and penalty method are given in Chapter 4.

Furthermore, the boundary control methods can also be applied in the case of parametric boundary condition problems and in the case of time-dependent boundary conditions. An example of a boundary control application from the nuclear field is the coupling of a system thermal-hydraulic codes with a three-dimensional CFD solver [20, 260]. However, the gain in computational time of such a coupled model is still limited by the CFD part. To overcome this burden, the system code can be coupled with a reduced order model of the high fidelity CFD solver. For transient problems, time-dependent boundary conditions of the ROM are then to be controlled based on the BCs obtained from the systems codes. We will continue on this in Appendix A.

Finally, a ‘discretize-then-project’ approach [152], i.e. projecting the fully discrete system, simplifies the treatment of the velocity boundary conditions. A recent study on ROMs on a staggered grid [223] demonstrated that the boundary conditions of the discrete FOM can be inherited by the ROM via the projection of the boundary vectors. With this approach, no additional boundary control method, such as the commonly applied penalty [93, 134, 161, 234] or lifting function methods [79, 93, 239, 262], is needed to handle the BCs at the ROM level. We will investigate this approach on collocated grids in Chapter 7.

2.5.3.3 Turbulence

Projection-based ROMs have been mainly developed for laminar flows [219]. A relatively small number of studies on ROMs for turbulent flows have been reported

in the literature among which, but not limited to, [78, 180]. Only a few studies are based on the finite volume methods [87, 111, 112, 161]. This is in contrast to the full order level for which a large variety of closure models for turbulent flows have been developed [219].

The Reynolds number of a flow, which is defined as the ratio of the inertial forces to the viscous forces within the fluid, indicates whether the flow can be considered laminar or turbulent. At high Reynolds numbers, the flow is dominated by the inertial forces and thus turbulent. Turbulent flows are characterized by three-dimensional eddies at various scales of motion. The large eddies carry most of the energy and extract the energy from the mean flow. These eddies are unstable and break up into smaller eddies. Finally, the eddies reach a sufficiently small length and energy scale that the viscosity can dissipate the kinetic energy into heat. In this way, the energy is transferred from the large scales to the smaller ones. This is known as the theory of the Richardson-Kolomogrov cascade or the turbulence energy cascade [142, 209]. The challenge of developing ROMs for turbulent flows is mainly about resolving the dissipative regime of the turbulence energy cascade.

Usually, only a few POD modes are sufficient to represent the kinetic energy of the flow due to the energetic optimality of the POD basis functions. POD truncation is used to reduce the size of the system. Therefore, only a few modes that contain the largest amount of energy, which are associated to the large eddies of the flow, are included in the ROM. However, the main amount of viscous dissipation takes place in the small eddies represented by the higher order modes, which are the most dissipative. Nevertheless, these higher order modes are left out by the POD truncation as they are less energetic. Therefore, the amount of dissipation in the ROM may be less than that in the FOM, because the high-energy modes that contribute to dissipation are truncated in the POD approach [25]. Consequently, the ROM is not able to dissipate enough energy and can blow up [161]. Another issue when applying the Galerkin projection on RANS equations is that the interaction between the averaged mean flow and the oscillating part of the flow field represented by the POD modes is neglected. This can lead to an unstable ROM [149]. Introducing additional dissipation via a closure model [25] by modeling the interaction between the calculated modes and the non resolved modes have been demonstrated to improve ROM stability [120]. Also in the case of developing an LES-ROM, the derivation of the ROM closure model to improve the accuracy and instability of the standard POD-Galerkin ROM is challenging [185, 280, 293]. Other solutions are numerical dissipation by introducing a fictitious eddy viscosity or modeling a stabilizing dissipative term at ROM level [161].

A specific type of turbulent flows are low-Prandtl number fluid turbulent flows. Especially their associated turbulent heat fluxes are complicated to model numerically as heat conduction through the boundary layer has more dominant effect with respect to convection. Therefore, the thermal boundary layers become thicker

when the Prandtl number is decreased. This means that there is a difference in the range of the spatial (and temporal) scales of temperature and velocity. As a consequence, the conductive heat fluxes near walls become more important. Therefore, it is problematic to apply the Reynolds analogy, which assumes a constant turbulent Prandtl number, Pr_t , close to unity, to calculate the local turbulent heat fluxes [213]. Furthermore, Pr influences not only the temperature field and the heat flux modeling, but also the velocity field and the shear modeling in the case of buoyancy-aided flows [100]. Therefore, heat transfer in liquid metals, compared to common fluids with a Prandtl number around unity, requires additional or different (physical) modeling. We will develop a ROM for low Prandtl number fluid flows in Chapter 6.

2.5.3.4 Transient dynamics and long term behavior

In general, it is challenging to develop a ROM for unsteady flows. The transient behavior, in particular the causality with the evolution over time, should be captured by the snapshots. The snapshots are typically collected at several instances in time during a certain time period [104, 149, 206]. The reduced order models can be used to determine solutions for a time evolution that are different from those of the original simulation or for long time integration.

To enhance the performance of the ROM one should concentrate the snapshots in the time span where the system exhibits the most nonlinear behavior [241]. However, it is difficult to determine the optimal distribution of snapshots in a given time span, which influences the ROM performance. A too low number of snapshots and/or the snapshots are not optimally distributed in time could result in a high (relative) ROM error. We will not research this aspect in this work. Instead, the snapshots are equally distributed in time in this work. Some techniques to optimally select the snapshots are described in [146, 193, 281].

Furthermore, it is difficult to accurately predict long-term transient behavior using reduced order models. The behavior of the flow in the long-term is not always contained in the snapshots and thus not in the modes [149]. Also for periodic regime solutions, the reduced order model might not reproduce the periodic cycles on the long term [239]. The ROM solution can diverge from the periodic solution over time due to different sources of error, even if the ROM is initialized with the correct periodic state [5, 234]. For the long time integration of convection dominated flows some type of closure model might be needed [149]. In [46] the ROM is stabilized by an additional damping term such that the energy transfer between the ROM modes is taken into account.

2.5.4 Parametric problems

Parametric reduced order models are being developed to obtain low-cost but accurate solutions of a system for different values of the parameters. Especially when a large number of different system configurations need to be tested, for instance for control purposes, sensitivity analyses or uncertainty quantification studies, the parametric ROM reduces the computational time. A comprehensive surveys of projection-based model reduction methods for parametric dynamical systems is given in [24]. We highlight here three types of parametrization: physical parametrization, parametric boundary conditions and geometrical parametrization.

Examples of physical parameters are the viscosity or non-dimensional numbers that are characterizing the flow. For this type of parametric problems, the equations governing the system behavior depend on a set of parameters. The approximation properties of the POD bases depend then on the parameter points for which the snapshots are computed. In general, a POD basis computed at a single parameter point is not a good approximation for solutions computed at different parameter points [149]. Therefore, it is required to sample in parameter space. The challenge is to determine the optimal selection of snapshots in parameter space. Often a greedy RB method is applied for parametric problems to select locally optimal sampling points [149, 270] based on the (estimated) error between the FOM and the ROM.

Also the (time-dependent) boundary conditions can be parametrized by assigning a new value to the boundary condition at ROM level. In Chapter 4 it is explained how to parameterize the BCs using the penalty method and the lifting function method.

Geometrical parameterizations are especially challenging as they require modifying the velocity space in order to manage the divergence-free constraint [149]. A common strategy for geometrical parametrization in the context of finite element based reduced order models is mapping different geometries from the physical domain to a common reference domain [242]. This has been widely studied in [123, 216, 218, 220]. However, even if possible, this approach cannot be easily transferred to a finite volume setting [219]. As an alternative, immersed methods can be used in the FV setting. The equations are then written on the same physical domain for all geometries to be studied and the immersed structure is treated as an external forcing term [219]. The most recent advances on geometric parametrization in the finite volume setting can be found in [242]. Despite the numerous of possible applications, such as shape design optimization, we will not consider geometrical parametrization in this research.

The parametric ROMs are, generally, only accurate in the range of parameter values for which the snapshots are collected [109]. If several parameters and/or a wide range of parameter values need to be tested, multiple local reduced order models need to be constructed, which is time consuming. Also if a time-dependent

problem is parametrized, thus a combined time and parameter-dependent problem, the standard POD method can become too expensive as several sets of snapshots for different parameter values are required [87, 141, 241]. A solution is the nested POD method which approximates the global POD space by solving one small eigenvalue problem for each local parameter space. The local POD bases are then weighted by the eigenvalues and a global snapshot matrix is created by appending the local weighted POD basis. A standard POD is then performed on the global snapshot matrix. The method is more efficient than the standard POD method since the computational effort of the former scales with the total number of snapshots while the nested POD method scales with the dimension of the global snapshot matrix [87].

For all types of parametrization, it is important that the parameter value can be adjusted in the ROM during the online phase without the need to recalculate the reduced matrices in order to guarantee the offline-online decomposition. This is not trivial in the case of nonlinear problems.

2.5.5 Sources of errors

As pointed out in the previous subsections, reduced order models are often developed for evaluating solutions on new sets of parameter values or for time evolutions that are different from those of the original simulations [24, 78, 103].

Reduced order models are computationally more efficient than the underlying high fidelity models as they are reduced in size, i.e. the ROMs contain a lower number of degrees of freedom compared to the FOMs. Therefore, the ROM has, generally, a lower accuracy than the FOM [96, 149]. Thus, there is a trade off between efficiency and accuracy.

It is hard to estimate the ROM errors (e.g. due to spatial and or temporal discretization [125, 295]) and to provide meaningful error bounds for the ROM solutions for the (unsteady) Navier-Stokes equations [149]. On top of that, it is especially hard to determine, both a priori and a posteriori, the errors in the framework of finite volume based reduced order modeling for which velocity and pressure belong to discontinuous spaces [241]. A posteriori error bounds in the finite element framework are exploited by Veroy et al. [268] for parametrized steady NS equations. Error estimates for POD Galerkin methods based on the spatial and temporal discretization errors are provided by Kunisch et al. [145]. In the framework of finite volumes, Haasdonk [105] et al. derived rigorous a posteriori error estimates in various norms for linear parabolic and hyperbolic evolution equations in the framework of finite volumes. An a priori error estimate for a ROM for the linearized compressible Euler equations is derived by Kalashnikova et al. [133]. We will not derive error estimates and error bounds in this work. Instead, we focus on the error analysis by quantifying the POD truncation error and computing

the (relative) error of solutions for a specific norm to show the performance of the ROMs regards to predicting the behavior of the underlying FOM.

2.5.5.1 POD truncation error

Ideally, the reduced basis for the ROM that accurately predicts the provided snapshots are identified a priori. However, the high-dimensional, nonlinear, multi-scale nature of the fluid flow causes the system to be highly sensitive to which (and how many) modes are included in the reduced basis spaces. Only the most energetic modes are included in the reduced basis space. The error resulting for leaving out the higher modes is called the POD truncation error. To estimate which modes to include in the reduced basis space, the ratio between the sum of the retained POD modes versus the sum of the whole set of eigenvalues of the correlation matrix can be used. This ratio is also called the relative information content (RIC) and is given by

$$\text{RIC} := \frac{\sum_{i=1}^{N_r} \lambda_i}{\sum_{i=1}^{N_s} \lambda_i}, \quad (2.38)$$

where λ_i is the eigenvalue of a certain mode. The preferred RIC is case dependent. Moreover, if the square root of a retained eigenvalue associated with a certain mode is on the order of numerical precision, it will become close to singular. Then the modes contain mostly numerical noise, which has a detrimental effect on the accuracy of the ROM [149].

2.5.5.2 Relative error

In a standard finite element framework, the H1 norm is preferred for the velocity field since its natural functional space is H1 [241]. However, in a finite volume setting, both velocity and pressure belong to discontinuous spaces. The preferred error norm for discrete numerical schemes and to measure the accuracy of the ROM with respect to the FOM is the L_2 norm [40, 241] as already defined in Section 2.4.1. The relative L_2 error of the fields can be determined for each time step or parameter point to show the performance of the proposed methods. We consider the following three types of fields at a time instance t^n : the full order fields X^n , the projected fields X_r^n , which are obtained by the L_2 projection of the snapshots onto the POD bases and lastly, the predicted fields X_{ROM}^n obtained by solving the ROM. The relative basis projection error for every time instance t^n and/or parameter point μ^k is computed as follows

$$\hat{\epsilon}_{L_2}(t^n, \mu^k) = \frac{\|X(t^n, \mu^k) - X_r(t^n, \mu^k)\|_{L_2(\Omega)}}{\|X(t^n, \mu^k)\|_{L_2(\Omega)}}, \quad (2.39)$$

which is the ‘best possible’ error at every time instance or parameter point. The time or sample-averaged basis projection error can also give a good indication of

the induced error by the POD truncation. The prediction error is determined by

$$\epsilon_{L_2}(t^n, \mu^k) = \frac{\|X(t^n, \mu^k) - X_{ROM}(t^n, \mu^k)\|_{L_2(\Omega)}}{\|X(t^n, \mu^k)\|_{L_2(\Omega)}}. \quad (2.40)$$

The relative prediction error is compared with the basis projection error.

2.5.6 Overview

Table 2.1 summarizes the challenges that are addressed in the consecutive chapters.

Table 2.1: Overview of the reduced order modeling challenges per chapter.

Chapter	3	4	5	6	7	A
Non-linearity (spatial)		x	x	x	x	x
Transient dynamics	x	x	x		x	x
Pressure term present in ROM	x				x	x
Buoyancy			x	x		
Turbulence closure				x		x
Non-homogeneous BCs	x	x	x	x	x	x
Time-dependent BCs	x	x				x
Long time behavior		x				x
Parametric ROM	x	x	x	x	x	x

3

POD-identification reduced order model of linear transport equations

In this chapter, a non-intrusive reduced order modeling method that identifies the system matrix of linear fluid dynamical problems with a least-squares technique is presented. The results of this chapter are published as [245].

3.1 Introduction

As pointed out in Section 2.3.2, the main issue of the POD-Galerkin reduced order modeling method is that knowledge of the high fidelity solver's discretization and solution algorithm is required in order to perform the Galerkin projection. Therefore, the method cannot be used for most (commercial) software.

In this chapter, a POD-based identification (POD-ID) method is proposed, for which reduced system matrices of the same form as in the POD-Galerkin method are identified using a least-squares technique. A set of ordinary differential equations (ODEs), still describing the physical model, is then obtained. The resulting ROM can be used for controlling the (time-dependent) non-homogeneous Dirichlet boundary conditions instead of having to perform a high fidelity simulation for every BC of interest. The boundary conditions are imposed in the ROM with a penalty factor [93].

This chapter is organized as follows: In Section 3.2 the scalar transport equation is introduced for which the full order simulation is performed. The methodol-

ogy of the POD-based identification method is addressed for parametrized boundary conditions together with the boundary control method in Section 3.3. In Section 3.4 the reduced order modeling technique is tested for a numerical experiment and the results are provided and discussed in Sections 3.5 and 3.6, respectively. Finally, conclusions are drawn in Section 3.7.

3.2 The convection-diffusion scalar transport equation

In this work the unsteady convection-diffusion scalar transport equation in incompressible form is considered, given by

$$\begin{cases} \frac{\partial T}{\partial t} + \nabla \cdot (\langle \mathbf{u} \rangle T) - \nabla^2(\mathcal{D}T) = 0 & \text{in } \Omega, \\ T(\mathbf{x}, t) = f(t) & \text{on } \Gamma_{lid}, \\ T(\mathbf{x}, t) = 0 & \text{on } \Gamma_{wall}, \\ T(\mathbf{x}, 0) = T_0 & \text{in } \Omega, \end{cases} \quad (3.1)$$

where $\langle \mathbf{u} \rangle$ is a constant velocity field, T is the transported scalar, T_0 is the initial scalar field, t denotes time and \mathcal{D} is the diffusion coefficient divided by the fluid density and the heat capacity, which are both constant. The boundary of the domain, $\Omega \subset \mathbb{R}^2$, is divided in two parts: $\Gamma = \Gamma_{wall} \cup \Gamma_{lid}$, where $\mathbf{u} = (u_{lid}, 0)$ on Γ_{lid} and $\mathbf{u} = (0, 0)$ on Γ_{wall} . It is important that the given constant background velocity field, $\langle \mathbf{u} \rangle$, satisfies the continuity equation $\nabla \cdot \mathbf{u} = 0$ for incompressible flows. This flow field can be obtained from a standard CFD model and hence its calculation is not described further. Discretizing the transport equation in space and rearranging in matrix form leads to the following system of equations:

$$\begin{cases} \dot{\mathbf{T}} + \mathbf{C}\mathbf{T} - \mathcal{D}\mathbf{D}\mathbf{T} = \mathbf{0}, \\ \mathbf{T}(0) = \mathbf{T}^0, \end{cases} \quad (3.2)$$

where the ‘over-dot’ indicates the time derivative, \mathbf{C} and \mathbf{D} are the matrices associated with the convection and diffusion terms, respectively. \mathbf{T}^0 is the initial condition.

3.3 POD-ID reduced order model of the scalar transport equation

In this section, the methodology of the proposed POD-based identification method, as a non-intrusive reduced order method (NIROM) for linear CFD problems, is described.

3.3.1 POD-based identification

Using the POD technique, the scalar function, T , in Equation 3.1 is replaced by the approximation, T_r , which is given by

$$T(\mathbf{x}, t) \approx T_r(\mathbf{x}, t) = \sum_{i=1}^{N_r} \phi_i(\mathbf{x}) a_i(t), \quad (3.3)$$

where $\phi_i(\mathbf{x})$ are the POD modes, $a_i(t)$ are the time-dependent coefficients and N_r is the dimension of the reduced basis space [145, 205, 284].

The optimal POD basis space, E_{POD} , is constructed by minimizing the difference between the snapshots and their orthogonal projection onto the reduced basis for the L_2 norm:

$$E_{POD} = \arg \min_{\phi_1, \dots, \phi_{N_r}} \frac{1}{N_s} \sum_{n=1}^{N_s} \left\| T(\mathbf{x}, t^n) - \sum_{i=1}^{N_r} (T(\mathbf{x}, t^n), \phi_i(\mathbf{x}))_{L_2} \phi_i(\mathbf{x}) \right\|_{L_2(\Omega)}^2, \quad (3.4)$$

where N_s is the number of snapshots. This procedure is according to the methodology that is described in the Section 2.4.1. To obtain the POD modes, the following eigenvalue problem is solved

$$C^{corr} \mathbf{Q} = \mathbf{Q} \boldsymbol{\lambda}, \quad (3.5)$$

where $C_{ij}^{corr} = (T(\mathbf{x}, t^i), T(\mathbf{x}, t^j))_{L_2(\Omega)}$ is the correlation matrix, \mathbf{Q} is a square matrix of eigenvectors and $\boldsymbol{\lambda}$ is a vector containing the eigenvalues. The POD modes, ϕ_i , can then be constructed as follows

$$\phi_i(\mathbf{x}) = \frac{1}{N_s \sqrt{\lambda_i}} \sum_{n=1}^{N_s} T(\mathbf{x}, t^n) Q_{in} \quad \text{for } i = 1, \dots, N_r. \quad (3.6)$$

By applying the classical Galerkin projection onto the reduced basis, see also Section 2.4.2, the following ROM is obtained

$$\dot{\mathbf{a}} + \mathbf{C}_r \mathbf{a} - \mathcal{D} \mathbf{D}_r \mathbf{a} = \mathbf{0}, \quad (3.7)$$

where

$$D_{r_{ij}} = (\nabla \phi_i, \nabla \phi_j)_{L_2(\Omega)}, \quad (3.8)$$

$$C_{r_{ij}} = (\phi_i, \nabla \cdot (\langle u \rangle \phi_j))_{L_2(\Omega)}. \quad (3.9)$$

However, the main issue of this method is that knowledge of the solver's discretization and solution algorithm is required in order to perform the Galerkin projection. Furthermore, the full order matrices, \mathbf{D} and \mathbf{C} , in Equation 3.2 are not accessible within most CFD codes, due to restricted access to the source code

in commercial software or due to the used solution methodology for open-source codes [272]. Therefore, it is also not an option to apply a Galerkin projection on the matrices of the full order systems in the following way

$$\mathbf{D}_r = \Phi^T \mathbf{D} \Phi, \quad (3.10)$$

$$\mathbf{C}_r = \Phi^T \mathbf{C} \Phi, \quad (3.11)$$

where $\Phi = [\phi_1, \phi_2, \dots, \phi_{N_r}]$. Thence, the POD-based identification method aims at identifying these reduced matrices using a least-squares technique such as normal equations, QR-decomposition or SVD by minimizing the residual, \mathbf{R} ,

$$\mathbf{R} = \dot{\mathbf{a}} + \mathbf{C}_r \mathbf{a} - \mathcal{D} \mathbf{D}_r \mathbf{a}, \quad (3.12)$$

in the following way

$$\left[\hat{\mathbf{D}}_r \hat{\mathbf{C}}_r \right] = \min_{\mathbf{D}_r, \mathbf{C}_r} \|\mathbf{R}\|. \quad (3.13)$$

In Equation 3.12, the time-dependent coefficients are constructed via a projection of modes on the full order solution

$$\mathbf{a}^n = \Phi^T \mathbf{T}^n, \quad (3.14)$$

where $\mathbf{T}^n \approx \mathbf{T}(t^n)$ is the solution at the n^{th} time step.

In addition, when the dynamical system is linear, no sources or sinks are present and the variables, for instance the diffusion coefficient \mathcal{D} , are not a function of a parameter $\mu \in \text{parameter space } \mathcal{P}$, the ROM can be simplified by the assumption that the time-dependent coefficients are related via the linear mapping

$$\mathbf{A}_r \mathbf{a}^{n+1} = \mathbf{a}^n \quad \text{for } n = 0, \dots, N_s - 1, \quad (3.15)$$

where \mathbf{A}_r is an unknown matrix to be identified using a least-squares approach. In order to do that, two matrices, \mathbf{X}_0 and \mathbf{X}_1 , are constructed that contain the known time-dependent coefficients at certain times in the following way

$$\mathbf{X}_0 = [\mathbf{a}^0, \mathbf{a}^1, \dots, \mathbf{a}^{N_s-1}], \quad (3.16)$$

$$\mathbf{X}_1 = [\mathbf{a}^1, \mathbf{a}^2, \dots, \mathbf{a}^{N_s}], \quad (3.17)$$

to satisfy Equation 3.15 as good as possible for each time step in which the snapshot is collected, by minimizing the difference between $\mathbf{A}_r \mathbf{X}_1$ and \mathbf{X}_0 . Therefore, the reduced matrix, \mathbf{A}_r , is computed by minimizing the norm

$$\mathbf{A}_r = \arg \min_{\hat{\mathbf{A}}_r} \left\| \hat{\mathbf{A}}_r \mathbf{X}_1 - \mathbf{X}_0 \right\|, \quad (3.18)$$

using a least-squares technique. A similar approach is applied for the dynamic mode decomposition, where the snapshots are assumed to be related via a linear mapping and the discrete-time linear system is then fitted on the set of snapshots [9]. The POD-ID method differs in the sense that the mapping is done at the reduced level instead of the high-dimensional level at which the snapshots are obtained. The maximum number of modes to be considered, N_r , is of the order $\sqrt{N_s}$ as an overdetermined system is required in order to identify a reduced matrix \mathbf{A}_r of size $N_r \times N_r$.

Finally, three types of scalar fields are considered: the full order fields \mathbf{T} , the projected fields \mathbf{T}_r , which are obtained by the projection of the FOM snapshots onto the POD basis, and the prediction fields \mathbf{T}_{ROM} . For every time instance, n , the relative basis projection error, $\hat{\epsilon}_{\ell_2}$, is given by

$$\hat{\epsilon}_{\ell_2}(t^n) = \frac{\|\mathbf{T}^n - \mathbf{T}_r^n\|_{\ell_2}}{\|\mathbf{T}^n\|_{\ell_2}}, \quad (3.19)$$

with $\mathbf{T}_r^n = \Phi \Phi^T \mathbf{T}^n$ and relative prediction error, ϵ_{ℓ_2} , is given by

$$\epsilon_{\ell_2}(t^n) = \frac{\|\mathbf{T}^n - \mathbf{T}_{ROM}^n\|_{\ell_2}}{\|\mathbf{T}^n\|_{\ell_2}}, \quad (3.20)$$

with $\mathbf{T}_{ROM}^n = \Phi \mathbf{a}^n$. For both errors the ℓ_2 norm is considered.

3.3.2 Boundary conditions

The non-homogeneous BCs are added as an additional constraint to the transport equation in order to weakly enforce the boundary condition for the ROM with a penalty factor [93, 134, 161, 205, 234]. No modification of the snapshots is needed other than adding the boundary points in case these are not present, otherwise the ROM could become unstable. The constraint is added to the transport equation in the following way

$$\frac{\partial T}{\partial t} + \nabla \cdot (\langle \mathbf{u} \rangle T) - \mathcal{D} \nabla^2(T) + \tau \Lambda(T - T_{BC}(t)) = 0, \quad (3.21)$$

where $T_{BC}(t)$ is the (time-dependent) Dirichlet boundary condition, τ the penalty factor and Λ is a null function except on the boundary where the condition is imposed [161]. In order to have an asymptotically stable solution, the penalty factor τ should be larger than 0. In case $\tau \rightarrow \infty$ a strong imposition would be approached and the ROM becomes ill-conditioned. This penalty factor can be found by numerical experimentation [29, 93, 161].

At reduced order level, after a Galerkin projection, this translates to

$$\dot{\mathbf{a}} + \mathbf{C}_r \mathbf{a} - \mathcal{D} \mathbf{D}_r \mathbf{a} + \tau (\mathbf{O}_r \mathbf{a} - \mathbf{P}_r(t)) = \mathbf{0}, \quad (3.22)$$

where \mathbf{P} is the projection of the boundary values on the modes at the boundary and \mathbf{O} the modes projected on the reduced basis at the boundary domain, Γ , given respectively, by

$$P_{ri}(t) = (\phi_i, T_{BC}(t))_{\ell_2(\Gamma)}, \quad (3.23)$$

$$O_{rij} = (\phi_i, \phi_j)_{\ell_2(\Gamma)}. \quad (3.24)$$

For the POD-ID method, without the parametrization of the diffusion coefficient, the reduced system is given by

$$(\mathbf{A}_r + \tau \mathbf{O}) \mathbf{a}^{n+1} = \mathbf{a}^n + \tau \mathbf{P}(t^n), \quad (3.25)$$

that can be solved for \mathbf{a}^{n+1} , depending on the boundary, $T_{BC}(t)$, applied. The initial condition for the ROM is obtained by projecting the full order initial condition for the parametrized BC onto the POD basis as follows

$$\mathbf{a}^0 = \mathbf{\Phi}^T \mathbf{T}^0. \quad (3.26)$$

where \mathbf{T}^0 contains the values of the initial parametrized (time-dependent) Dirichlet boundary condition $T_{BC}(t^0)$.

The overall algorithm for the POD-ID method including the penalty method is given below.

Algorithm: POD-ID method including penalty method

Create ROM with POD-ID method:

- (1) Generate snapshots by solving the linear full order problem of Equation 3.1;
- (2) Perform POD on the snapshots to obtain the POD modes Φ using Equation 3.6;
- (3) Project the snapshots on the modes to obtain the corresponding coefficients \mathbf{a} using Equation 3.14;
- (4) Retrieve the matrices \mathbf{X}_0 and \mathbf{X}_1 from the coefficients with Equations 3.16 and 3.17;
- (5) Identify the reduced matrix \mathbf{A}_r by minimizing the norm $\|\hat{\mathbf{A}}_r \mathbf{X}_0 - \mathbf{X}_1\|$ with a least-squares technique as in Equation 3.18;
- (6) Project the initial field for the parametrized BC onto the POD basis to get the initial condition \mathbf{a}^0 for the ROM using Equation 3.26;

Impose BCs with penalty method:

- (7) Project the values for the parametrized non-homogeneous Dirichlet boundary on the modes to determine \mathbf{P} using Equation 3.23;
- (8) Project the modes on the reduced basis at the same boundary domain of previous step to determine \mathbf{O} using Equation 3.24;
- (9) Set a value for the penalty factor τ ;

Solve reduced order model:

- (10) Solve the reduced order problem of Equation 3.25 for the time period $[t^1, t^{N_s}]$;
 - (11) **if** the boundary is not enforced in ROM solution **then**
 set a new value for τ and go back to step (10).
 end if
 - (12) Reconstruct the full order fields from the obtained coefficients using Equation 3.3;
-

3.4 Numerical test case of an enclosed heated cavity

The POD-ID method is tested for the classical numerical two-dimensional lid-driven cavity benchmark problem [32, 227] with the addition that the moving lid of the cavity is heated [252]. A sketch of the geometry of the square cavity of length $L = 1$ m on which a (512×512) uniform mesh is constructed is depicted in Figure 3.1. The boundary of the domain is divided in two parts: $\Gamma = \Gamma_{wall} \cup \Gamma_{lid}$, where $\mathbf{u} = (1, 0)$ on Γ_{lid} and $\mathbf{u} = (0, 0)$ on Γ_{wall} . A homogeneous Neumann boundary condition for the pressure is applied everywhere on Γ .

The constant background velocity field is precomputed first in the offline phase for isothermal laminar flow with $Re = 1000$. The viscosity, ν , is taken at $1 \cdot 10^{-4}$ m²/s. The initial condition for the velocity, \mathbf{u}_0 , is set to 0 m/s and the reference value for the pressure is set to 0 m²/s² at coordinate (0,0). The calculation of

the flow field is performed in the OpenFOAM environment [283], a finite volume open-source code [129, 177, 271], with the icoFoam solver for $t = 50$ s with time steps of 0.01 s. The velocity field obtained at final simulation time, $t = 50$ s, is taken as the constant background velocity field for the convection-diffusion problem (Equation 3.1). In fact, any constant flow field could be chosen.

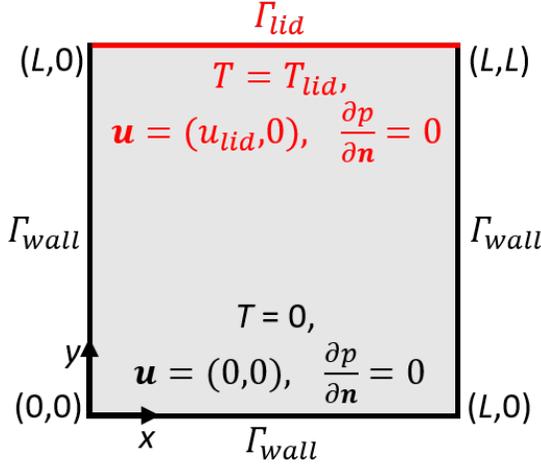


Figure 3.1: A sketch of the geometry of the heated lid-driven cavity problem including boundary conditions.

The unsteady scalar transport equation (Equation 3.1) is solved for the scalar quantity T , which is in this case the dimensionless temperature, with the thermal diffusion constant \mathcal{D} equal to $1 \cdot 10^{-5}$ m²/s and the constant background flow field equal to the one previously obtained. A homogeneous Dirichlet BC is applied on Γ_{wall} and a non-homogeneous BC with $T_{lid}^{FOM} = 1$ on Γ_{lid} . The initial condition is set to $T_0 = 0$. The simulation is performed with the scalarTransportFoam solver of OpenFOAM with an implicit scheme for the time discretization. A constant time step of $\Delta t = 1 \cdot 10^{-3}$ s has been applied. Snapshots of the temperature are collected every 0.1 s, resulting in a total of 1000 snapshots. The POD modes and ROM are constructed according to Sections 2.4.1 and 3.3.1, respectively. The time-independent BC for the ROM, T_{lid}^{ROM} , could be chosen freely, although it should be of the same order as T_{lid}^{FOM} for which the POD has been performed to avoid discrepancies, as the statement “If it is not in the snapshots, it is not in the ROM” by Quarteroni et al. [205] still applies. The BCs are enforced with the penalty method according to Section 3.3.2 and the ROM is tested for $T_{lid}^{ROM} = 0.5, 2$ and 10. As the system is linear, parameterizing the time-independent Dirichlet BC on Γ_{lid} is straightforward as the FOM solutions can be scaled according to a BC of interest and no ROM has to be constructed. Nevertheless, the ROMs are

constructed in order to demonstrate the capability of the POD-ID method.

Finally, the ROM is tested for a time-dependent boundary condition given by

$$T(x, 1, t) = A \sin(2\pi ft) + B, \quad (3.27)$$

where f is the frequency of the wave, A the amplitude and B the off-set. Full order solutions, with this time-dependent BC, are calculated for $f = 0.01 \text{ s}^{-1}$, $A = 1$ and $B = 1$, which will be referred to as the base case (set 1 in Table 3.1). The ROM is tested for five sets of parameters that are defining the Dirichlet BC, summarized as set 2-5 in Table 3.1. To evaluate the accuracy of the POD-ID method for all ROMs, the relative prediction error is calculated according to Equation 3.20.

Table 3.1: Parameter sets for time-dependent Dirichlet BCs defined by Equation 3.27

	f	A	B
Set 1	0.01	1	1
Set 2	0.01	2	1
Set 3	0.01	1	3
Set 4	0.01	2	3
Set 5	0.011	1	1
Set 6	0.02	1	1

In the offline phase, the snapshots are created with OpenFOAM, while in the second part of the offline phase, creating the POD modes and constructing the reduced system of equations (Equation 3.7), is performed with MATLAB [173]. Also the online phase, solving the reduced order systems for different Dirichlet boundary conditions is done with MATLAB. The whole offline-online procedure is carried out with a single core of an Intel Core i5-6300U processor. The ROM's online computational time depends on the number of modes and is no longer dependent on the number of degrees of freedom of the FOM.

3.5 Results and analysis

In this section the accuracy of the ROM is tested for both time-independent and time-dependent Dirichlet BC. Before these tests, the background velocity is pre-computed. The flow field is shown together with the corresponding pressure field in Figure 3.2. Then, the full order simulation for the time-independent Dirichlet BC $T_{lid}^{FOM} = 1$ is performed until $t = 100 \text{ s}$. The evolution of the temperature field in time is shown in Figure 3.3 for $t = 1, 10, 50$ and 100 s .

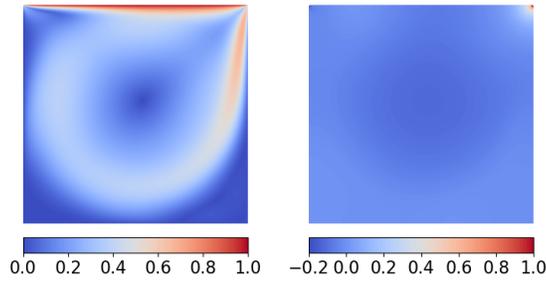


Figure 3.2: (Left) the background velocity field in m/s and (right) the corresponding pressure field in m^2/s^2 for the lid-driven cavity simulation at $t = 50$ s.

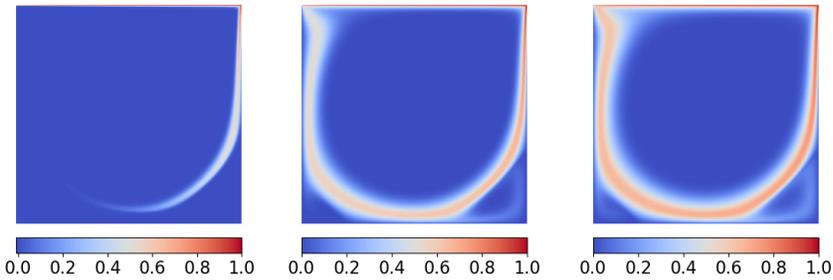


Figure 3.3: Evolution of the temperature with $T_{lid}^{FOM} = 1$ (base case) inside the cavity for time instances $t = 5, 50$ and 100 s.

During the offline phase the reduced basis for the temperature was calculated following the procedure described in Section 3.3. The decay of the normalized POD eigenvalues is plotted in Figure 3.4 in order to determine the number of basis functions needed to create the reduced subspace. The figure shows that 18 basis functions are required to have a truncation error less than 10^{-6} . The cumulative eigenvalues (CV) can be found in Table 3.2 and 36 modes are sufficient to retain more than 99.9% of the energy for temperature. However, as pointed out in Section 3.3.1, the maximum number of modes to be considered is of the order $\sqrt{N_s}$, which is 31 modes for a snapshot matrix containing 1000 snapshots. 31 modes correspond to a truncation error less than 10^{-7} and more than 99.8% of the energy for temperature is retained, so this number is used for the reduced basis.

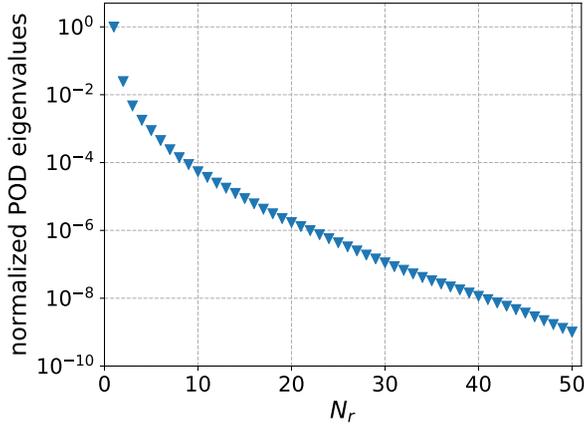


Figure 3.4: Normalized POD eigenvalues of the base case with time-independent boundary conditions.

Table 3.2: Cumulative eigenvalues (CV) of the base case for temperature.

# of modes	CV of T
1	0.71253
2	0.82443
3	0.87342
4	0.90353
5	0.92473
10	0.97132
20	0.99333
31	0.99806
36	0.99901

One reduced matrix, containing all linear terms, is determined as last in the offline phase with the least-squares technique (QR-decomposition) in order to describe the reduced system. The ROM is in the same form as Equation 3.7. In the online phase the ROM is solved for the same initial and boundary conditions as for the FOM and a numerical experimentation is performed on the penalty method on a couple of values for τ , namely 1, 10, 100, 1000 and 10.000. It is found that a penalty factor of 100 enforces the BCs without afflicting the ROM with ill-conditioning problems. Also a factor of 1000 and 10.000 did not lead to unstable solutions. Therefore a penalty factor $\tau = 1000$ is chosen for all ROM simulations.

3.5.1 Imposing time-independent boundary conditions

The constructed ROM is tested for $T_{lid}^{ROM} = 0.5, 1, 2$ and 10 . The ROM simulations are performed until $t = 100$ s for a constant time step of 0.1 s. Full order simulations have been performed for comparison. A cross-section of the temperature field for $x = 0.5$ m at $t = 100$ s is plotted in Figure 3.5. The accuracy of the ROM is analyzed by calculating the relative prediction error of the temperature fields (Equation 3.20) and comparing it with the relative basis projection error (Equation 3.19) of the FOM snapshots onto the POD basis in Figure 3.6. The ROM is describing the system almost as accurate as projecting the first 31 modes for any of the tested BCs.

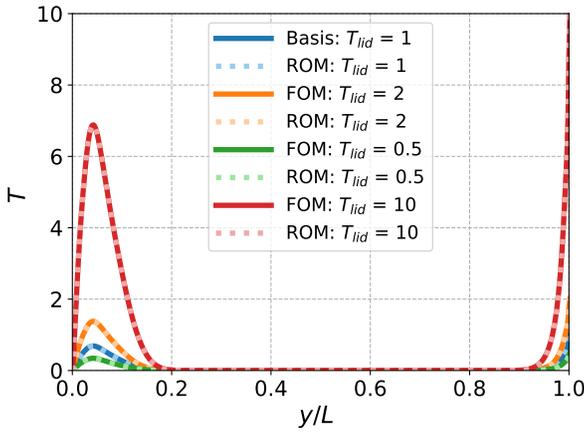


Figure 3.5: Cross-section of the temperature field for $x = 0.5$ m at $t = 100$ s. $T_{lid}^{FOM} = 1$ is the base case and the ROM is tested for $T_{lid}^{ROM} = 0.5, 1, 2$ and 10 with the penalty method.

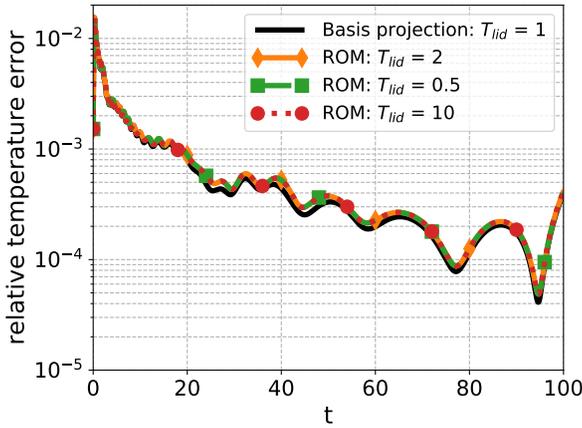


Figure 3.6: Time evolution of the basis projection error and the relative error in the ROM with respect to the FOM for $T_{iid}^{ROM} = 0.5, 1, 2$ and 10 .

For the full order simulation a computational time of 165 minutes is required to collect 1000 snapshots. Generating the POD modes and determining the reduced matrix, \mathbf{A}_r , by solving the least-squares problem in the offline phase requires about 6 s and 0.6 s, respectively.

3.5.2 Imposing time-dependent boundary conditions

In order to construct a ROM for time-dependent BCs, snapshots have to be computed for a FOM with a similar BC as for which the ROM has to be constructed. The time-dependent Dirichlet BC of the FOM is therefore given by Equation 3.27 with amplitude $A = 1$, frequency $f = 0.01$ Hz and offset $B = 1$. The time-dependent BC is enforced in the ROM with the penalty method according to the methodology described in Section 3.3. Snapshots are collected every 0.1 s for the temperature, resulting in a total number of 1000 snapshots. As done previously, 31 modes are used for the ROM construction. No other parametrization is considered and thus only one reduced matrix, containing all linear terms, is determined with the QR-decomposition technique. ROM simulations are carried out for the parameter set 2, 3 and 4 (listed in Table 3.1). For each of the BCs a full order simulation is performed to compare the ROM solution. Figure 3.13 shows the comparison of the temperature fields for $t = 100$ s for the FOM, the corresponding ROM and the relative error between the two. Cross-sections of the temperature field at $x = 0.5$ m and at $y = 0.5$ m in in for $t = 80$ s are plotted in Figure 3.7 and Figure 3.8, respectively.

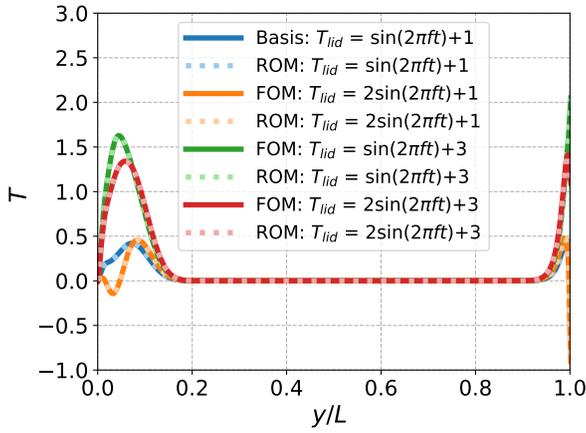


Figure 3.7: Cross-section of the temperature field at $x = 0.5$ m at $t = 80$ s. The time-dependent BCs are listed in the legend with $f = 0.01$ Hz.

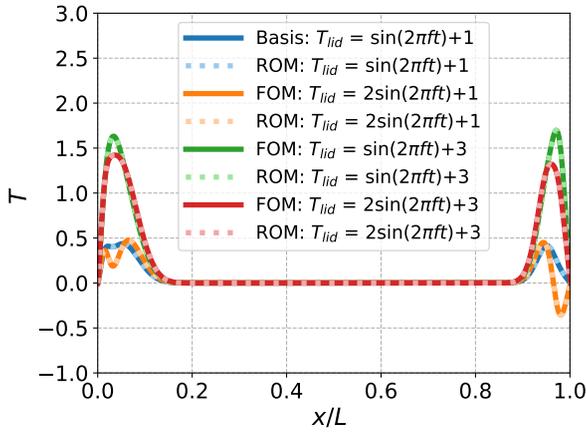


Figure 3.8: Cross-section of the temperature field at $y = 0.5$ m at $t = 80$ s. The time-dependent BCs are listed in the legend with $f = 0.01$ Hz.

These figures show that the full order solution cannot simply be scaled as done previously for the time-independent BCs. Nevertheless, the ROMs are capable of approximating the FOM solutions. For each ROM the relative prediction error is compared with the relative error of the temperature reconstruction from the time-dependent coefficient given by projection of the snapshots onto the POD functions in Figure 3.9. The relative errors for the ROMs are of the same order as the basis projection and decrease, on average, up to about 80 s, meaning that the reduced model is stable in that time interval. The prediction errors slightly increase near the end of the ROM simulation, following the trend of the relative basis projection error. The error can be reduced by adding more modes to the basis. However, as the POD-ID method requires the system to be overdetermined, the maximum number of modes that can be used for identifying the ROM is set by the square root of the number of snapshots, meaning that 31 modes is near the limit when only 1000 snapshots are used for the POD, instead of only one set as in this work. In order to add more modes to the POD basis, one needs to increase the number of snapshots. Each ROM simulation required a computational time of about 2 s. The speed-up is thus of the order $\mathcal{O}(10^{-3})$ compared with simulation the FOM.

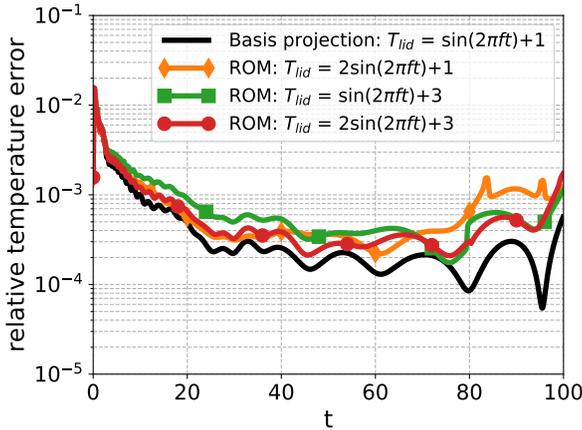


Figure 3.9: Time evolution of the basis projection error and the relative error in the ROM with respect to the FOM. The time-dependent BCs are listed in the legend with $f = 0.01$ Hz.

Finally, the frequency of the time-dependent BC is parametrized and ROM simulations are carried out for the parameter set 5 and 6 (listed in Table 3.1) using the same basis (set 1 in Table 3.1) as for previous cases. Cross-sections of the temperature field at $x = 0.5$ m and $y = 0.5$ m are plotted in Figure 3.10 and Figure 3.11, respectively. The relative errors are plotted in Figure 3.12. Even when the frequency is increased by 10%, the relative error is still of the same order of the basis projection. However, doubling the frequency results in an increase of the relative error by one order after about 10 s of simulation time.

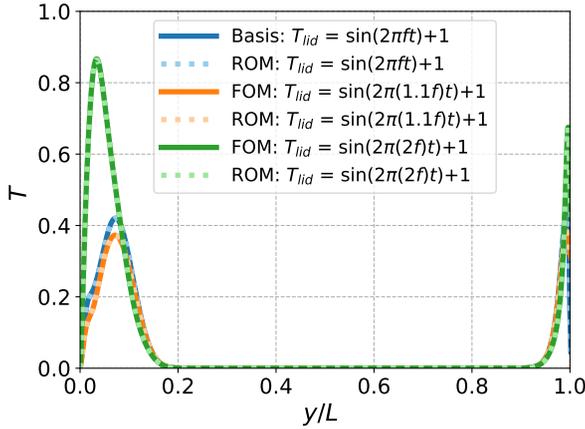


Figure 3.10: Cross-section of the temperature field at $x = 0.5$ m at $t = 80$ s. The time-dependent BCs are listed in the legend with $f = 0.01$ Hz.

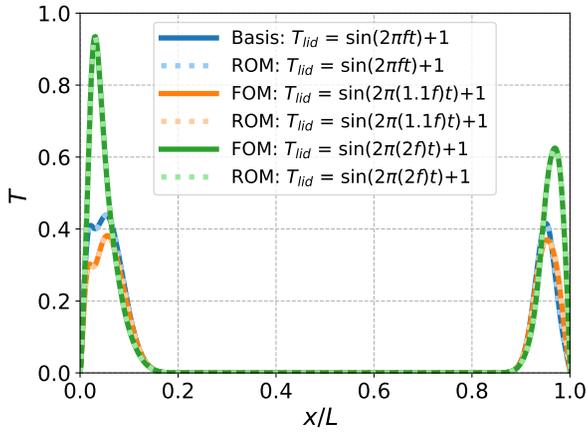


Figure 3.11: Cross-section of the temperature field at $y = 0.5$ m at $t = 80$ s. The time-dependent BCs are listed in the legend with $f = 0.01$ Hz.

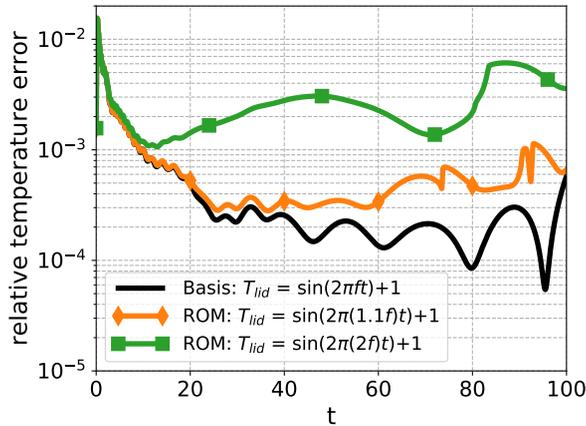


Figure 3.12: Time evolution of the basis projection error and the relative error in the ROM with respect to the FOM for different frequencies.

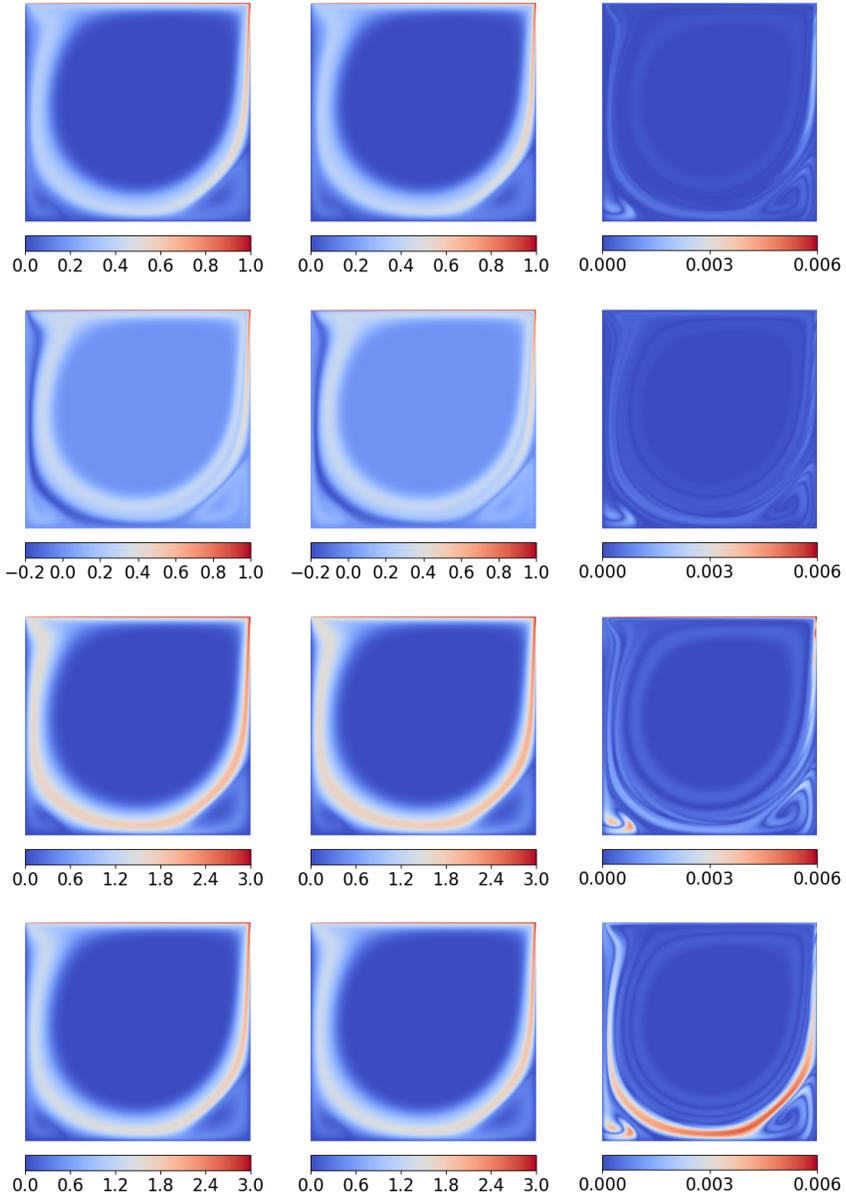


Figure 3.13: Comparison of the temperature field for the FOM (left) and ROM solutions (middle) at $t = 100$ s for parameter set 1 - 4 (from top to bottom). The absolute difference between the ROM and corresponding FOM is plotted on the right. All ROMs are obtained with 31 modes.

3.6 Discussion

The ROMs constructed with the presented POD-ID method for controlling the non-homogeneous time-independent Dirichlet BC are capable of approximating the full order solutions for the linear unsteady convection-diffusion equation as the relative error of the main variable, namely the temperature, is of the same order as the projection error of the snapshots onto the POD basis.

In this study, the penalty factor of 1000 was determined via numerical experimentation. The main advantage of the penalty method is that it can be applied non-intrusively. However, that the factor cannot be determined a priori is a drawback of the method [93]. Also, although it has not been observed here, it is possible that a penalty factor needs to be chosen above a certain threshold to enforce the BC in the ROM, which then will lead to an unstable ROM solution [234]. The range of the factor for which the solution is stable can for instance be determined using Poincaré maps [234]. Also the bounds on the factor that ensure asymptotic stability of the ROM can be derived [134].

The relative prediction error increases when the frequency of the time-dependent BC is doubled. There are two ways to increase the number of snapshots in order to reduce the error and to enhance stability. First of all, more full order sets can be used for the POD. For example, adding full order snapshots for $f = 0.03$ Hz to the snapshot matrix in order to parametrize the frequency in the interval $[0.01, 0.03]$ Hz in the ROM. This can be combined with the second method to increase the snapshot matrix, namely by sampling the full order solution more frequently. Then, more modes can be used to identify the reduced matrix with the least-squares technique as, in order to keep the system overdetermined, the size of the reduced matrix is limited by the square root of the number of snapshots. Even more snapshots would be required in the case of nonlinear systems, because then at least as many reduced matrices are to be identified as there are modes to be stored in the offline phase [161, 239, 241]. For example, the nonlinear convective term of the Navier-Stokes equations can be approximated by $\mathbf{a}^T \mathbf{C}_r \mathbf{a}$, where \mathbf{C}_r is a third order tensor. Then, the reduced problem grows with the cube of the number of modes in order to maintain an offline-online decomposition. Consequently, more snapshots are required to keep the system overdetermined in order to identify all these matrices. In that case, the required number of snapshots scales with the cube of the number of modes required. Otherwise, many matrices will be empty in case the system is not overdetermined and the problem becomes ill-conditioned. Therefore, it is not fully feasible to use the POD-ID method for fully nonlinear problems.

In theory, one can solve the ROM for a different time step, $\Delta \tilde{t}$, than used in the FOM, simply rewriting the problem of Equation 3.15 in the following way: $\mathbf{a}^{n+1} (1 + \mathbf{A}_r \Delta \tilde{t})^{-1} = \mathbf{a}^n$. However, the ROM becomes unstable when approx-

imating the solution for time instances at which no snapshots are collected for the POD, because the POD-ID method identifies a reduced matrix with the least-squares technique that fits the full order snapshot data, like a black-box system, and is not capable of approximating the solution at intermediate time instances.

Besides, it is redundant to construct a ROM in order to impose the time-independent BC of the linear problem investigated in this chapter. One can simply obtain the results for the parametrized BCs at the same time instances for which snapshots were collected, as parameterizing the Dirichlet BC condition only changes the solution with respect to a reference point. This does, however, not apply to the control of time-dependent BCs as then the solutions do not scale linearly with respect to a reference point. In that case, when intermediate results are required, interpolation techniques could be used to approximate the solution at intermediate times. The same applies also for the long time stability of the ROM. Furthermore, parameterizing the BCs could be done for any given background velocity field, including those of turbulent flows. As long as there is no two-way coupling between the fluid flow and the heat transfer, the problem stays linear and could be described by a single reduced matrix. Moreover, other types of parametrization could be applied, for instance, parameterizing the diffusion coefficient. However in that case, not just one, but two reduced matrices have to be identified, one for the diffusion and one for the convection term.

Finally, the speed-up is of the order $\mathcal{O}(10^3)$. Less modes could be used to speed up the calculation even more, but the error will then increase and the ROM could become unstable.

3.7 Conclusion

In this chapter, it has been demonstrated that the proposed POD-based identification method is capable of constructing a ROM that could be used for controlling the non-homogeneous time-(in)dependent Dirichlet BCs of the scalar transport convection-diffusion equations by enforcing the BCs in the ROM with a penalty method rather than having to perform a high fidelity simulation with the finite volume approximation for every BC of interest. However, the ROM could only approximate the solution at the same time instances the snapshots are collected for determining the POD basis. Nevertheless, the POD-ID method together with the penalty method could be applied for linear problem that require boundary control. For instance, determining the heat transport by airflow in a room for transient thermal analysis for buildings where the temperature at a wall is the parameter of interest or for pollution dispersion modeling where the concentration is controlled. However, the main shortcoming of the POD-ID method, at this stage, is that it is not feasible to use the method for fully nonlinear problems as the required number of snapshots scales with the cube of the number of modes and at least as many

reduced matrices are to be identified as the number of modes used. Therefore, the following chapters are devoted to the POD-Galerkin method, which has better nonlinear properties than the POD-ID method. Furthermore, in the case of a larger parameter space to be investigated, one has to perform the POD on snapshots collected for more parameter values and/or one has to sample more frequently in order to capture the full dynamics of the system.

4

Methods to impose boundary conditions in POD-Galerkin reduced order models

Handling the boundary conditions at the reduced order level is one of the challenges that are highlighted in Section 2.5. In this chapter, POD-Galerkin reduced order models are developed of which the (time-dependent) boundary conditions are imposed at the reduced order level using two different strategies: the lifting function method and the penalty method. Moreover, the penalty method, which was applied in the previous chapter, is improved by using an iterative solver for the determination of the penalty factor rather than tuning the factor with a sensitivity analysis or numerical experimentation. The results of this chapter are accepted as [248].

4.1 Introduction

Non-homogeneous boundary conditions are, in general, not satisfied at the reduced order level. Furthermore, the BCs are not explicitly present in the ROM and therefore they cannot be controlled directly [161].

In literature [93, 134, 161, 239], different approaches to control the ROM BCs can be found. Two common approaches are extended and compared in this chapter: the lifting function method and the penalty method. The aim of the lifting function

method [93, 239] is to homogenize the BCs of the basis functions contained in the reduced subspace and to control the boundary condition values at the ROM level using a suitable control function, while the penalty method [93, 134, 161, 234] weakly enforces the BCs in the ROM with a penalty factor. A disadvantage of the penalty method is that it relies on a penalty factor that has to be tuned with a sensitivity analysis or numerical experimentation [234]. Therefore, a novel iterative method is presented for tuning the penalty factor in the context of finite volume based POD-Galerkin reduced order methods. The novelty of this method is that a error tolerance for the enforced BC has to be set instead of an arbitrary value for the factor. Also the factor is determined automatically by iterating rather than manually via numerical experimentation.

The reduced order models investigated in this chapter are constructed with the POD-based Galerkin projection technique according to Section 2.4 and the exploitation of a pressure Poisson equation during the projection stage.

This chapter is organized as follows: in Section 4.2 the full order and reduced order models are presented together with the exploitation of a pressure Poisson equation during the projection stage. In Section 4.3, the two boundary control methods are presented: the lifting function method and the iterative penalty method. In Section 4.4, the set-up of two numerical experiments, a lid-driven cavity and a Y-junction test case, are given and the results are provided and discussed in Section 4.5 and 4.6, respectively. Finally, conclusions are drawn in Section 4.7.

4.2 Exploitation of the pressure Poisson Equation

In this chapter, the POD-Galerkin reduced order models are based on the exploitation of a pressure Poisson equation during the projection stage. At the full order level, the fluid dynamics problem is physically described by the unsteady incompressible Navier–Stokes equations where the incompressibility constraint $\nabla \cdot \mathbf{u} = 0$ is replaced by a pressure Poisson equation. In an Eulerian framework on a domain $\Omega \subset \mathbb{R}^d$ with $d = 2, 3$ and boundary $\Gamma = \Gamma_{wall} \cup \Gamma_{in} \cup \Gamma_{out}$, the governing system of equations is given by

$$\left\{ \begin{array}{ll} \frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u} \otimes \mathbf{u}) - \nabla \cdot (\nu \nabla \mathbf{u}) = -\nabla p + \mathbf{F} & \text{in } \Omega, \\ \Delta p = -\nabla \cdot (\nabla \cdot (\mathbf{u} \otimes \mathbf{u})) + \nabla \cdot \mathbf{F} & \text{in } \Omega, \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}) & \text{in } \Omega, \\ \mathbf{u}(\mathbf{x}, t) = \mathbf{f}(\mathbf{x}, t) & \text{on } \Gamma_{in}, \\ \mathbf{u}(\mathbf{x}, t) = \mathbf{0} & \text{on } \Gamma_{wall}, \\ \mathbf{n} \cdot \nu \nabla \mathbf{u} - np = \mathbf{0} & \text{on } \Gamma_{out}, \\ \mathbf{n} \cdot \nabla p = -\mathbf{n} \cdot \left(\nu \nabla \times \nabla \times \mathbf{u} + \frac{\partial \mathbf{f}}{\partial t} \right) + \mathbf{n} \cdot \mathbf{F} & \text{on } \Gamma, \end{array} \right. \quad (4.1)$$

where $\mathbf{u} = \mathbf{u}(\mathbf{x}, t)$ represents the vectorial velocity field that is evaluated at $\mathbf{x} \in \Omega$ and $p = p(\mathbf{x}, t)$ is the normalized scalar pressure field, which is divided by the constant fluid density ρ , ν is the kinematic viscosity and \mathbf{F} is a body force term. $\mathbf{u}_0(\mathbf{x})$ denotes the initial condition of the velocity at time $t = 0$ s. The (time-dependent) non-homogeneous Dirichlet boundary condition of velocity on Γ_{in} is represented by $\mathbf{f}(\mathbf{x}, t)$ and a no-slip condition is applied on Γ_{wall} . Furthermore, a homogeneous Neumann boundary condition for velocity and a homogenous Dirichlet BC for pressure are applied on Γ_{out} . \mathbf{n} denotes the outward pointing normal vector on the boundary. The equations are presented here in a general format. The problem-specific (boundary) conditions are specified in Section 4.4, in which the numerical experiments are presented. Note that the boundary condition for the pressure differs from the one in Equation 2.4. Different ways to enforce a boundary condition for the pressure term are possible as pointed out in [97, 159]. In this chapter, the same condition as used in the work of Stabile et al. [241] is applied. For more details on the derivation of the PPE the reader is referred to J.-G Liu et al. [159]. The full order equations are discretized with the finite volume method and solved using a PIMPLE [77] algorithm for the pressure-velocity coupling, which is a combination of SIMPLE [187] and PISO [128].

The reduced order model is obtained following the POD-Galerkin procedure described in Section 2.4.1. The projection of the governing equations (Equation 4.1) onto the reduced basis spaces for velocity and pressure results in the following reduced system of equations

$$\begin{cases} M_r \dot{\mathbf{a}} + \mathbf{a}^T C_r \mathbf{a} - \nu D_r \mathbf{a} + G_r \mathbf{b} = \mathbf{0}, \\ V_r \mathbf{b} + \mathbf{a}^T K_r \mathbf{a} - \nu N_r \mathbf{a} - T_r \dot{\mathbf{a}} = \mathbf{0}, \end{cases} \quad (4.2)$$

with

$$\begin{aligned} M_{r_{ij}} &= (\varphi_i, \varphi_j)_{L_2(\Omega)}, D_{r_{ij}} = (\varphi_i, \nabla \cdot (\nabla \varphi_j))_{L_2(\Omega)}, \\ G_{r_{ij}} &= (\varphi_i, \nabla \chi_j)_{L_2(\Omega)}, C_{r_{ijk}} = (\varphi_i, \nabla \cdot (\varphi_j \otimes \varphi_k))_{L_2(\Omega)}, \\ V_{r_{ij}} &= (\nabla \chi_i, \nabla \chi_j)_{L_2(\Omega)}, K_{r_{ijk}} = (\nabla \chi_i, \nabla \cdot (\varphi_j \otimes \varphi_k))_{L_2(\Omega)}, \\ T_{r_{ij}} &= (\chi_i, \mathbf{n} \cdot \varphi_j)_{L_2(\Gamma)}, N_{r_{ij}} = (\mathbf{n} \times \nabla \chi_i, \nabla \times \varphi_j)_{L_2(\Gamma)}, \end{aligned} \quad (4.3)$$

where the last two terms of the reduced PPE are projected on the boundary Γ . Equation 4.3 consists only of first order derivatives as integration by parts of the Laplacian term is used together with exploiting the pressure boundary condition after the PPE is projected onto the POD space spanned by the pressure modes. In that way, the numerical differentiation error can be reduced [241]. Following the same strategy as in Equation 2.37, the nonlinear term in Equation 4.2 is evaluated by storing the third order tensor K_r .

It is important to note that the equations of the reduced system (Equation 4.2) are coupled. The system of nonlinear equations can be solved using an iterative

approach. Moreover, the pressure is only defined up to an arbitrary constant, as in the FOM. Therefore, next to an initial condition for velocity, an initial guess for pressure is required for the system to converge more easily and to ensure the consistency between the FOM and the ROM [40]. The initial conditions and guesses for the reduced order model are obtained by performing a Galerkin projection of the full order initial field onto the POD basis spaces as follows

$$a_i^0 = (\varphi_i, \mathbf{u}(\mathbf{x}, 0))_{L_2(\Omega)}, b_i^0 = (\chi_i, p(\mathbf{x}, 0))_{L_2(\Omega)}, \quad (4.4)$$

for velocity and pressure, respectively.

4.3 Boundary control methods at the reduced order level

The non-homogeneous BCs are, in general, not satisfied in a POD-based ROM as the basis functions, and in the same way their BCs, are a linear combination of the snapshots. Furthermore, the BCs are not explicitly present in the reduced system and therefore they cannot be controlled directly [161]. Two common approaches are presented in this section for handling the BCs: the lifting function and the penalty method [93]. The aim of the lifting function method is to have homogeneous POD modes and to enforce the BCs by means of a properly chosen lifting function in the ROM. On the other side, the penalty method enforces the BCs in the ROM with a penalty factor. In this work only the velocity BCs are controlled with the two methods.

4.3.1 The lifting function method

The lifting function method for the non-homogeneous boundary conditions is often used in the continuous Galerkin finite element setting to reformulate a boundary control problem into a distributed one [31, 54, 64]. The method imposes the non-homogeneous (Dirichlet) conditions to the problem through lifting. This is done by subtracting the lifting function from the unknown variable in the original PDE problem, solving for the modified variable and adding the lifting function to the solution [262].

In a similar way, this method is used to impose non-homogeneous (Dirichlet) boundary conditions in reduced order models for which the lifted fields are projected onto the reduced bases spanned by the POD modes [79].

In this work, the velocity snapshots are made homogeneous by subtracting suitable lifting functions from all of them on which then the POD is performed. The result is a set of velocity modes that individually fulfill the homogeneous BCs as they are linear combinations of the modified velocity snapshots. The lifting functions, which fulfill the original non-homogeneous boundary conditions,

are then added to a linear combination of POD basis functions. As a result, the non-homogeneous Dirichlet boundary conditions are included in the reduced basis space spanned by the POD modes and the lifting functions.

This lifting function method is also known as the “control function method” in literature [5, 93, 148] for PDE problems whose Dirichlet boundary conditions can be parametrized with a single time-dependent coefficient [262]. This is the type of problem that is presented in this work. The method is generalized in [103] for generic functions with multiple parameters at distinct boundary sections.

The functions to be chosen are system-specific and they have to satisfy the divergence free constraint in order to retain the divergence-free property of the snapshots [239].

One way to generate a lifting function, $\tilde{\zeta}(\mathbf{x})$, is by solving a problem as close as possible to the full order problem, where the boundary of interest is set to its value and everywhere else to a homogeneous BC. There are several other ways to compute the lifting function. For instance, the snapshot average can be used, although this does not always lead to a discretely divergence-free function. Alternatively, the solution of the stationary version of the considered problem can be computed [37]. Two other common approaches are solving a non-homogeneous Stokes problem [79, 82, 90] or solving a potential flow problem [69, 113].

As one of the characteristics of the POD modes is that they are orthonormal, the lifting functions are normalized as follows

$$\zeta(\mathbf{x}) = \frac{\tilde{\zeta}(\mathbf{x})}{\|\tilde{\zeta}(\mathbf{x})\|_{L_2(\Omega)}}, \quad (4.5)$$

before subtracting them from all snapshots and applying POD. The snapshots, $\mathbf{u}(\mathbf{x}, t)$, are then modified accordingly

$$\mathbf{u}'(\mathbf{x}, t) = \mathbf{u}(\mathbf{x}, t) - \sum_{j=1}^{N_{BC}} \zeta_j(\mathbf{x}) u_{BC_j}(t), \quad (4.6)$$

where N_{BC} is the number of non-homogeneous BCs, $\zeta(\mathbf{x})$ is a normalized lifting function and u_{BC} is the normalized value of the corresponding Dirichlet boundary condition.

The POD modes, φ'_i , that satisfy the homogeneous boundary conditions are obtained by solving an eigenvalue problem as described in Section 2.4.1 on the homogenized snapshots $\mathbf{u}'(\mathbf{x}, t)$. The lifting functions are then added as additional modes to the reduced velocity basis space

$$E'_u = \left[\zeta_1, \dots, \zeta_{N_{BC}}, \varphi'_1, \dots, \varphi'_{N'_r} \right]. \quad (4.7)$$

Consequently, the velocity field are approximated by

$$\mathbf{u}_r(\mathbf{x}, t) = \sum_{j=1}^{N_{BC}} \zeta_j(\mathbf{x}) u_{BC_j}(t) + \sum_{i=1}^{N_r^u} \varphi'_i(\mathbf{x}) a_i(t), \quad (4.8)$$

which satisfies the boundary conditions of the problem and u_{BC} can be time-dependent. The Dirichlet boundary condition can be parametrized by assigning a new value to u_{BC} in Equation 4.8. In other words, the lifting functions can be scaled by a factor.

Finally, the initial condition of velocity for the reduced order model is obtained by performing a Galerkin projection of the initial full order velocity field with a new value of u_{BC} imposed onto the reduced basis space of velocity E'_u (Equation 4.7).

For more details on the lifting function the reader is referred to [87, 239]. The overall algorithm for the lifting function method is given below.

Algorithm 1: Lifting function method

OFFLINE PHASE**Solve full order model:**

(1) Generate snapshots over a time period $[0, T]$ by solving the full order problem of Equation 4.1;

Obtain the lifting functions:

(2) Generate the lifting functions by solving a flow problem:

```

for  $i = 1$  to  $N_{BC}$  do
  for  $j = 1$  to  $N_{BC}$  do
    if  $i = j$  then
       $u|_{\Gamma_j} = 1$ 
    else
       $u|_{\Gamma_j} = 0$ 
    end for

```

Solve a flow problem for $\tilde{\zeta}_i$

end for;

(3) Normalize the lifting functions to obtain ζ as in Equation 4.5;

(4) Subtract the normalized lifting functions from the velocity snapshots as in Equation 4.6;

Perform POD:

(5) Obtain the POD modes using the homogenized velocity snapshots and pressure snapshots according to Section 2.4.1;

(6) Add the normalized lifting functions, ζ , as additional modes to the set of velocity POD modes φ according to Equation 4.7;

Projection:

(7) Project the full order system onto the obtained reduced bases as done in Section 4.2 and obtain the reduced order model;

ONLINE PHASE**Solve reduced order model:**

(8) Project the initial field of velocity for the parametrized BC onto the POD velocity basis to get the initial condition of velocity for the ROM and the initial guess for pressure according to Equation 4.4;

(9) Solve the reduced order problem of Equation 4.2 in the time period $[t_1, t_{\text{online}}]$;

(10) Reconstruct the full order velocity fields from the obtained coefficients using Equation 4.8 and the pressure fields according to Equation 2.24;

4.3.2 The iterative penalty method

The penalty method was originally proposed in the context of finite element methods [16, 158]. The method transforms a strong non-homogeneous Dirichlet boundary condition into a weak Neumann boundary condition by the means of a small parameter whose inverse is called the penalty factor [197]. Thus, the method

uses a penalty parameter to weakly impose the boundary conditions. In the POD-Galerkin reduced order modeling setting, the penalty method has been first introduced by Sirisup and Karniadakis [234] for the enforcement of boundary conditions. For the penalty method, no modification of the snapshots is needed as the velocity Dirichlet BCs are directly enforced as constraints in the reduced system in the following way:

$$\mathbf{M}_r \dot{\mathbf{a}} + \mathbf{a}^T \mathbf{C}_r \mathbf{a} - \nu \mathbf{D}_r \mathbf{a} + \mathbf{G}_r \mathbf{b} + \sum_{l=1}^{N_{BC}} \tau_l (\mathbf{O}l_r \mathbf{a} - u_{BC_l}(t) \mathbf{P}l_r) = \mathbf{0}, \quad (4.9)$$

where τ is the penalty factor [234] and the additional terms with respect to Equation 2.34 are projected on the boundary as follows

$$\begin{aligned} \mathbf{O}l_{rij} &= (\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_j)_{L_2(\Gamma_l)} \quad \text{for } l = 1, \dots, N_{BC} \text{ and } i, j = 1, \dots, N_r^u, \\ \mathbf{P}l_{ri} &= (\boldsymbol{\varphi}_i, \boldsymbol{\Lambda})_{L_2(\Gamma_l)} \quad \text{for } l = 1, \dots, N_{BC} \text{ and } i = 1, \dots, N_r^u, \end{aligned} \quad (4.10)$$

where $\boldsymbol{\Lambda}$ is a unit field. This minimization problem is formulated at the reduced order level and, therefore, the penalty method does not depend on the full order snapshots.

In order to have an asymptotically stable solution, the penalty factors τ should be larger than 0. If $\tau \rightarrow \infty$ the solution generally converges to a true optimal solution of the original unpenalized problem [117]. Nevertheless, a strong imposition would be approached and the ROM becomes ill-conditioned [70, 161]. Therefore, the penalty factor needs to be chosen above a threshold value for which the method is stable and converges [59, 70]. On the other hand, it is important to find a penalty factor as small as possible to obtain a numerical stable solution. This is usually done by numerical experimentation [29, 93, 134, 161].

Several techniques exist in literature to optimize the numerical experimentation. Kelley [138, page 214] used a simple iteration scheme to optimize the trial-and-error process of the numerical experimentation. With this scheme the penalty value is adjusted each iteration by using the absolute value of the ratio between the constraint violation and a preassigned tolerance as a factor to increase or decrease the values at the end of each iteration. Basically, the idea is that the penalty factor obtained by the iteration scheme is optimal in the sense that it perturbs the original problem by a minimum for the given tolerance [122].

In this work the experimentation is optimized using a first-order iterative optimization scheme [155] to determine the factors that is based on the iteration scheme described in the previous paragraph. The penalty factors, τ , are updated each iteration k , as follows

$$\begin{aligned} \tau_l^{k+1}(t^n) &= \tau_l^k(t^n) \frac{|r_l^k(t^n)|}{\epsilon} \\ &= \tau_l^k(t^n) \frac{|\tilde{u}_{BC_l}^k(t^n) - u_{BC_l}(t^n)|}{\epsilon} \quad \text{for } l = 1, \dots, N_{BC}, \end{aligned} \quad (4.11)$$

with $r^k(t^n)$ the residual between \tilde{u}_{BC}^k , the value of a certain boundary at the k^{th} iteration, and u_{BC} , the enforced boundary condition, at an evaluated time t^n . \tilde{u}_{BC}^k is obtained during the online phase by reconstructing the boundary. $\epsilon > 0$ is the given error tolerance for the residual which has to be set. There is no single approach that can be considered the best for choosing ϵ , as the preferred tolerance depends on the problem and on both physical and geometrical parameters. The eigenvalue truncation error of the POD modes gives a good indication for the value of ϵ . The penalty method is therefore no longer based on an arbitrary value for the penalty factor.

As long as $|\tilde{u}_{BC_l}^k(t^n) - u_{BC_l}(t^n)| > \epsilon$ the penalty factors grow every update and converge to the smallest penalty factors that satisfy the required tolerance. Thus, if the initial guess for the factor is below the minimum value for τ for which the boundary condition is enforced in the ROM, the factor is approached from below using this method. For a time-dependent problem it is not needed to determine a penalty factor for all time steps N_t . Often the factor determined after the first couple of time steps, N_τ , can be used for the whole ROM solution.

The step-by-step demonstration of the iterative function method is given below by Algorithm 2.

Algorithm 2: Iterative penalty method

OFFLINE PHASE**Solve full order model:**

(1) Generate snapshots over a time period $[0, T]$ by solving the full order problem of Equation 4.1;

Perform POD:

(2) Obtain the POD modes using the snapshots according to Section 2.4.1;

Impose BCs with penalty method:

(3) Project the modes on the reduced basis at the boundary of the domain to determine Ol_r and Pl_r for each non-homogeneous Dirichlet boundary condition as in Equation 4.10;

(4) Solve iteratively for the penalty factor using Equation 4.11:

```

for  $i = 1$  to  $N_\tau$  do
  while  $|\tilde{u}_{BC_l}^k(t^i) - u_{BC_l}(t^i)| > \epsilon$  do
     $\tau_l^{k+1}(t^i) = \tau_l^k(t^i) \frac{|\tilde{u}_{BC_l}^k(t^i) - u_{BC_l}(t^i)|}{\epsilon}$ 
  end while
end for;

```

Projection:

(5) Project the full order system onto the obtained reduced bases and obtain the reduced order model according to Section 4.2;

ONLINE PHASE**Solve reduced order model:**

(6) Project the initial fields for the parametrized BC onto the POD bases to get the initial condition/guesses for the ROM using Equation 4.4;

(7) Solve the reduced order model in the time period $[t_1, t_{\text{online}}]$;

(8) Reconstruct the full order fields from the obtained coefficients using Equations 2.23 and 2.24 ;

It is important to note that the penalty factor can affect the number of iterations needed to solve the reduced system and therefore the convergence and cost of the reduced order model [181].

4.4 Numerical test cases

In this section, the set-up of two cases are described for which the boundary control methods, the lifting function method and the iterative penalty method, are tested. The first test case is the classical lid-driven cavity benchmark problem and the second test case is a Y-junction with two inlets and one outlet channel whose time-dependent inlet boundary conditions are controlled.

4.4.1 Lid-driven cavity flow problem

The first test case consists of a lid-driven cavity problem. The simulation is carried out on a two-dimensional square domain of length $L = 0.1$ m on which a (200×200) structured mesh with quadrilateral cells is constructed. The boundary is subdivided into two different parts $\Gamma = \Gamma_{lid} \cup \Gamma_{wall}$. The boundary conditions for velocity and pressure are set according to Figure 4.1. The pressure reference value is set to $0 \text{ m}^2/\text{s}^2$ at coordinate $(0,0)$. At the top of the cavity a constant uniform and horizontal velocity equal to $\mathbf{u} = (u_{lid}, 0) = (1, 0)$ m/s is prescribed. A no slip BC is applied at the walls, Γ_{wall} . The kinematic viscosity is equal to $\nu = 1 \cdot 10^{-4} \text{ m}^2/\text{s}$ and the corresponding Reynolds number is 1000, meaning that the flow is considered laminar.

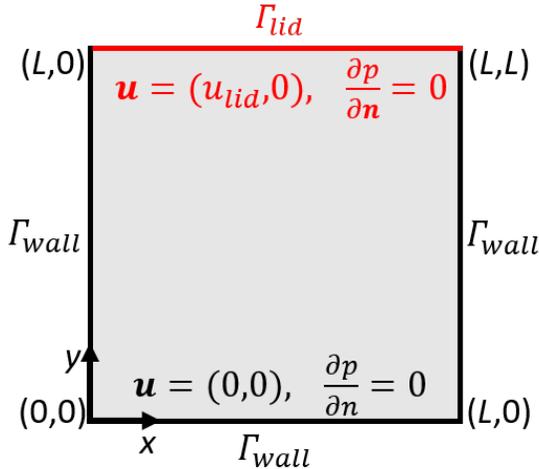


Figure 4.1: Sketch of the geometry of the 2D square cavity with moving top lid including boundary conditions.

The unsteady full order equations are iteratively solved by the FV method with the *pimpleFoam* solver of the open source C++ library OpenFOAM 6 [130]. The PIMPLE algorithm is used for the pressure-velocity coupling [77]. For the full order simulations, the spatial discretization of all terms is performed with a central differencing scheme (linear). The temporal discretization is treated using a second order backward differencing scheme (BDF2). A constant time step of $\Delta t = 5 \cdot 10^{-4}$ s has been applied and the total simulation time is 10 s. Snapshots of the velocity and pressure fields are collected every 0.01 s, resulting in a total of 1001 snapshots (including 1 for the initial condition). The initial condition field with $u_{lid} = 1$ m/s is used as a lifting function.

For this test case the same boundary conditions are applied in the ROM as in

the FOM for which the snapshots are collected. The temporal discretization of the ROM is performed with a first order Newton's method.

POD, projection of the full order solution on the reduced subspace and the reduced order simulations are all carried out with ITHACA-FV, a C++ library based on the finite volume solver OpenFOAM. For more details on the ITHACA-FV code, the reader is referred to [239, 240, 241].

4.4.2 Y-junction flow problem

Junctions are often used for the combination or separation of fluid flows and can be found in all types of engineering applications from gas transport in pipes till micro flow reactors. As a second test case a Y-junction with one outlet channel and two inlet channels is modeled. The angle between each inlet and the horizontal axis is 60 degrees, as shown in Figure 4.2 on the left [250]. The length of the channels is 2 m.

The 2D geometry is split in 6 zones as depicted in Figure 4.2 on the left. On the three rectangular zones a mesh with quadrilateral cells is constructed. The remaining three zones are meshed with hexagonal cells. The different meshes are depicted in Figure 4.2 on the right. The total number of cells is 13046.

The boundary is subdivided into four different parts $\Gamma = \Gamma_{in1} \cup \Gamma_{in2} \cup \Gamma_{out} \cup \Gamma_{wall}$. The two inlets, Γ_{in1} and Γ_{in2} , have a width of 0.5 m, while the outlet, Γ_{out} , has a width of 1 m. The kinematic viscosity is equal to $\nu = 1 \cdot 10^{-2} \text{ m}^2/\text{s}$ meaning that the Reynolds number at the inlet is 50 and the flow is considered laminar. The uniform inlet velocities are time dependent and the velocity magnitude of the flow at the inlets is set according to Figure 4.3.

A homogeneous Neumann boundary condition is applied for pressure at the inlet and wall boundaries. At the outlet, Γ_{out} , $p = 0 \text{ m}^2/\text{s}^2$ together with a homogeneous Neumann BC for velocity. A no slip BC is applied at the walls, Γ_{wall} .

As initial conditions the steady state solution, obtained with the *simpleFoam* solver, for a velocity magnitude of 1 m/s at both inlets is chosen. The other boundary conditions are the same as for the unsteady simulation described above.

As done previously for the lid-driven cavity case, the unsteady governing equations are iteratively solved by the FV method with the *pimpleFoam* solver of OpenFOAM 6 [130]. For the full order simulations, the discretization in space is performed with a central differencing scheme for the diffusive term and a combination of a second order central-differencing and upwind schemes for the convective term. The temporal discretization is treated using a second order backward differencing scheme (BDF2). A constant time step of $\Delta t = 5 \cdot 10^{-4} \text{ s}$ has been applied and the total full order simulation time is 12 s for which snapshots of the velocity and pressure fields are collected every 0.03 s, resulting in a total of 401 snapshots (including 1 for the initial condition).

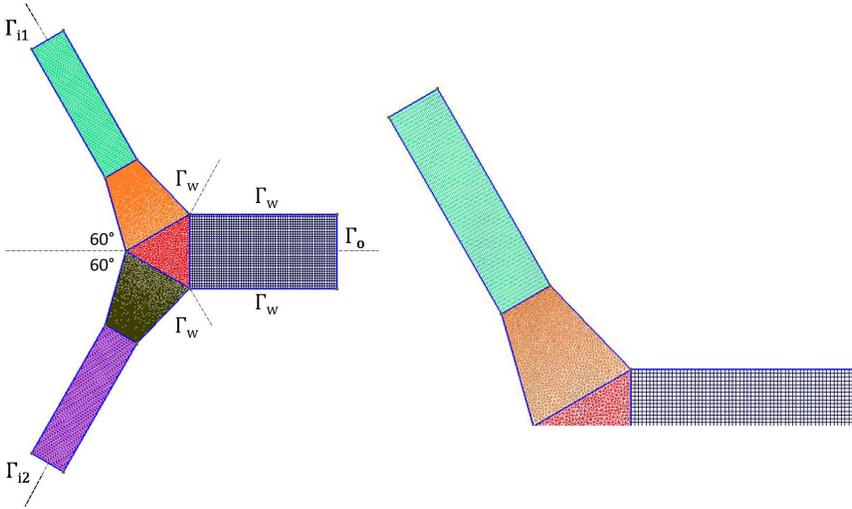


Figure 4.2: (Left) sketch of the geometry and mesh of the Y-junction test case including boundary conditions. (Right) close up of the mesh in different zones.

The inlet velocity BCs are time-dependent and the velocity magnitude of, alternately, inlet 1 or 2 is increased or decreased linearly between 1 m/s to 0.5 m/s as shown in Figure 4.3.

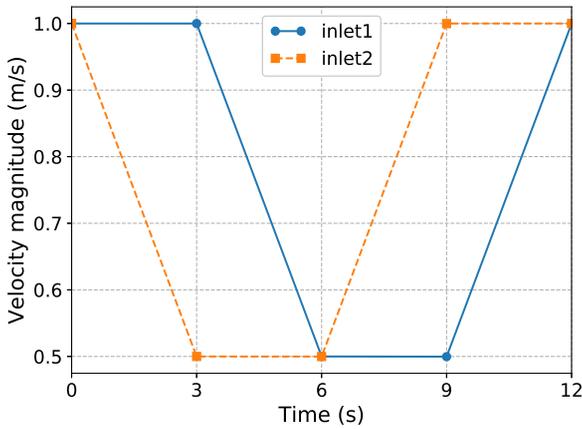


Figure 4.3: Inlet velocity boundary conditions for the full order model for the Y-junction test case.

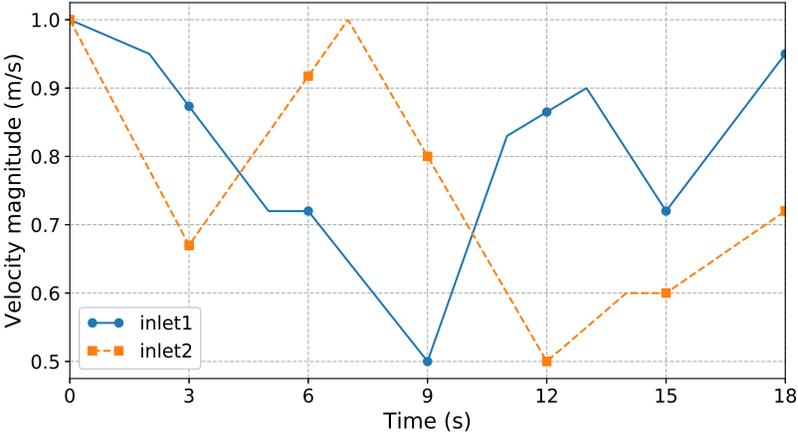


Figure 4.4: Inlet velocity boundary conditions for the reduced order model for the Y-junction test case.

In that way, the ROM is trained for all possible combinations of inlet velocities within the specified range. The inlet boundary conditions of the ROM are then controlled according to Figure 4.4, where the inlet velocity magnitude is increased or decreased linearly over time between the maximum of 1 m/s and the minimum of 0.5 m/s. The magnitude of the inlet velocities of the ROM decreases and increases faster or slower over time compared to the training run. Also the ROM is tested for a longer time period, 18 s, compared to the full order simulation of 12 s. In that way the ROM performance can be tested on the long term.

The temporal discretization of the ROM is performed with a first order Newton's method.

Both the iterative penalty method and lifting function method are tested. The lifting functions are determined by solving for a potential flow field problem given by

$$\left\{ \begin{array}{ll} \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega, \\ \nabla^2 p = 0 & \text{in } \Omega, \\ \mathbf{n} \cdot \nu \nabla \mathbf{u} - n p = \mathbf{0} & \text{on } \Gamma_{out}, \\ (\nabla p(\mathbf{x}, t)) \mathbf{n} = \mathbf{0} & \text{on } \Gamma \not\subseteq \Gamma_{out}, \\ (\nabla \mathbf{u}(\mathbf{x})) \mathbf{n} = \mathbf{0} & \text{on } \Gamma_{wall}, \\ \mathbf{u}(\mathbf{x}) = \mathbf{f1}(\mathbf{x}) & \text{on } \Gamma_{in1}, \\ \mathbf{u}(\mathbf{x}) = \mathbf{f2}(\mathbf{x}) & \text{on } \Gamma_{in2}, \end{array} \right. \quad (4.12)$$

with the magnitude of the inlet velocity at inlet 1, Γ_{in1} , set to 1 m/s while inlet 2, Γ_{in2} , is kept at 0 m/s as shown in Figure 4.5 for the first lifting function. To obtain

the second lifting function, $\|\mathbf{u}\| = 0$ m/s at Γ_{in1} and 1 m/s at Γ_{in2} . Both lifting functions are shown in Figure 4.5.

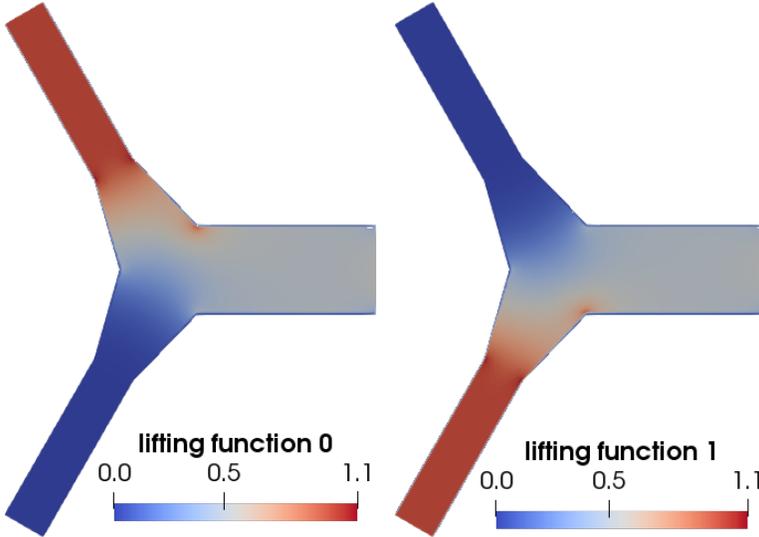


Figure 4.5: The lifting functions for velocity for the Y-junction test case.

The test case of a Y-junction is more complicated than the lid-driven cavity case as not only one, but two boundaries need to be controlled, which are also time dependent. Furthermore, as the channel inlets are placed under an angle, one needs to take into account that the inlet velocity can be decomposed in an x and a y-direction. Therefore, the vectorial lifting functions are split into their components before normalization. Also in the case of the penalty method, four penalty factors are determined; one for each inlet and each direction. This will be further discussed in section 4.6.

4.5 Results and analysis

4.5.1 Lid-driven cavity flow problem

First the full order simulation for the lid-driven cavity test case is performed and 1001 velocity and pressure snapshots are collected, including the initial conditions, which are then used to create the POD basis functions. Stabile and Rozza [241] concluded in their research that 10 velocity and pressure modes are enough to retain 99.99% of the energy contained in the snapshots. Therefore, the same number of modes for the reduced basis creation are used in this work.

Reduced order models are constructed with both the lifting function and penalty

method and compared with a ROM without boundary enforcement. With the use of the iterative procedure a penalty factor of 0.058 is determined within 2 iterations by evaluating only the first five time steps with a maximum error tolerance, ϵ , of 10^{-5} for the value of the boundary condition of the ROM and starting from an initial guess of 10^{-6} . For a similar study of the lid-driven cavity benchmark, Lorenzi et al. [161] had found a factor between 10^{-5} and 10^2 using numerical experimentation. The value found here using the iterative method is thus within the same range and near their minimum value. A higher value for the penalty factor can be used, but it is then more likely that the ROM becomes ill-conditioned.

The obtained ROMs are tested for the same initial and boundary conditions as the high fidelity simulation. The evolution in time of the relative L_2 error between the reconstructed fields and the full order solutions is plotted in Figures 4.6 and 4.7 together with the basis projection for velocity and pressure, respectively.

In case no boundary enforcement method is used the flow field remains zero throughout the simulation and therefore the relative error is 1. When either the lifting function or penalty method is used the relative L_2 error for both the velocity and pressure fields are about the order of 10^{-1} due to the relatively low number of snapshots acquired during the initial part of the transient. The snapshots are equally distributed in time, while this time span exhibits the most nonlinear behavior. Therefore one should concentrate the snapshots in this time span to enhance the performance of the ROM [241]. After about 2 seconds of simulation time, for both boundary control methods, the relative error drops till about the order of 10^{-3} . At the final time of the simulation the penalty method is performing slightly better than the lifting function method, but the order is the same.

Contrary to velocity, the relative error for pressure stays about $4 \cdot 10^{-1}$ after 2 s of simulation time, while the projection error drops till about 10^{-3} . This has been previously acclaimed by Stabile et al. in [239]. The PPE stabilization method is less accurate concerning pressure compared to the supremizer enrichment method. This has also been found by Kean and Schneider [137] in the finite element-based ROM setting. Furthermore, the absolute error between the FOM and the ROMs is shown in Figure 4.9 and 4.10 for velocity magnitude and pressure, respectively.

It is observed that both methods lead, for velocity, to an absolute error between the FOM and the ROM of the order 10^{-2} at the beginning of the simulation and about 10^{-3} once the flow has reached its steady state solution. Furthermore, the velocity error slightly increases between 5 and 10 s of simulation time. This can also be observed in the L_2 error analysis over time in Figure 4.6. For pressure, the error is largest near the top corners of the cavity and are of the order 10^{-3} . Note that the scale does not show the whole range of absolute errors. This is done to better visualize the error. The maximum error for pressure is about $5 \cdot 10^{-2} \text{ m}^2/\text{s}^2$ at the top right corner. As the pressure relative to its reference point at (0,0) plotted in Figure 4.10 is always less than 1 Pa, the relative error plotted in

Figure 4.7 is greater than the absolute error plotted in Figure 4.10. Furthermore, the error distribution, for both the velocity and pressure fields, is similar all over the domain, meaning the methods are performing the same, as previously confirmed by the L_2 error analysis over time in Figures 4.6 and 4.7.

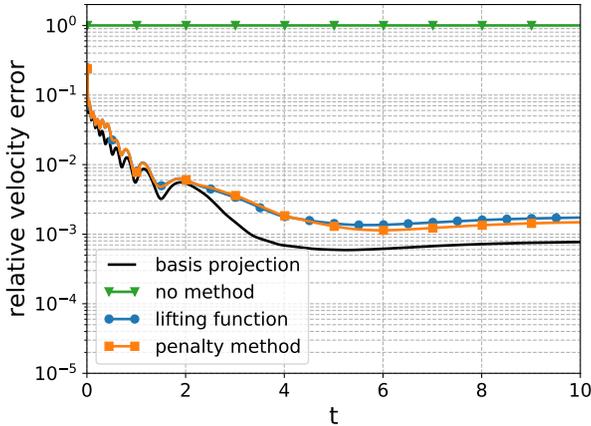


Figure 4.6: Time evolution of the basis projection error and the relative L_2 error of velocity between the FOM and ROM with the lifting function and penalty method for the lid-driven cavity problem.

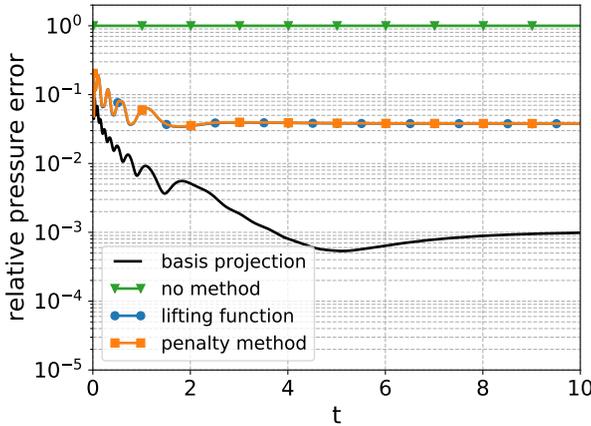


Figure 4.7: Time evolution of the basis projection error and the relative L_2 error of pressure between the FOM and ROM with the lifting function and penalty method for the lid-driven cavity problem.

The relative error for the total kinetic energy is determined and plotted in Figure 4.8. The order is more or less the same for both boundary control methods. From time to time the penalty method is performing slightly better and the other way around and the relative velocity error is less than 10^{-2} for the vast part of the simulation.

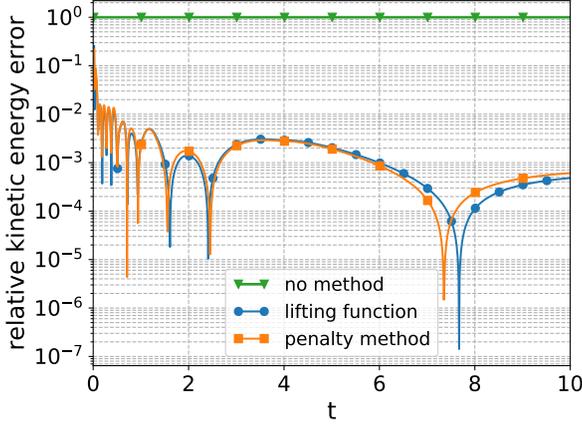


Figure 4.8: Time evolution of the relative L_2 error of kinetic energy between the FOM and ROM with the lifting function and penalty method for the lid-driven cavity problem.

Finally, the computational times for performing the full order simulation, calculating the POD modes, the reduced matrices and performing the simulation at the reduced order level are all listed in Table 4.1. Calculating the POD modes, reduced matrices and the ROM solutions takes more time in the case of the lifting function method as the reduced basis space consists of an additional mode, namely the normalized lifting function for the boundary with the lid, compared to the penalty method. Determining the penalty factor with the iterative method takes only 0.11 s. The speedup ratio between the ROM and the FOM is about 270 times for the lifting method and 308 times for the penalty method.

Table 4.1: Computational time (clock time) for the FOM simulation, POD, calculating reduced matrices offline (Matrices), determining penalty factor with iterative method (Penalty) and ROM simulation.

Method	FOM	POD	Matrices	Penalty factor	ROM
Lifting	37 min.	50 s	8.2 s	-	8.2 s
Penalty	37 min.	45 s	6.8 s	0.11 s	7.2 s

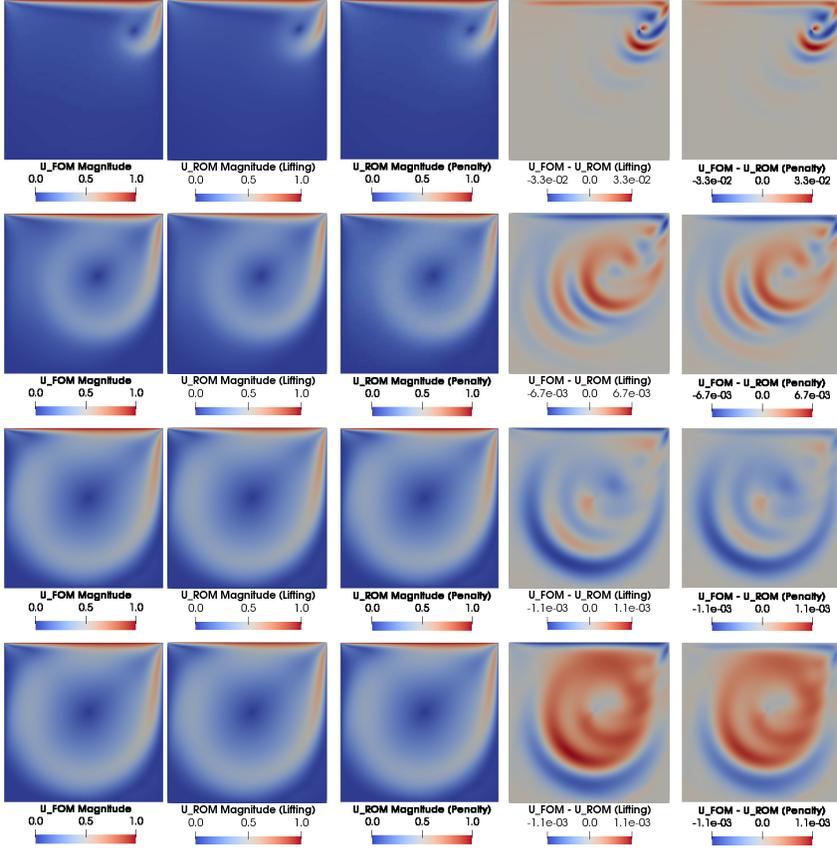


Figure 4.9: Comparison of the full order velocity magnitude fields (1st column), the ROM fields obtained with the lifting function method (2nd column) and penalty method (4th column) and the difference between the FOM and ROM fields obtained with the lifting function method (3rd column) and penalty method (5th column) at $t = 0.2, 1, 5$ and 10 s (from top to bottom) for the lid-driven cavity problem.

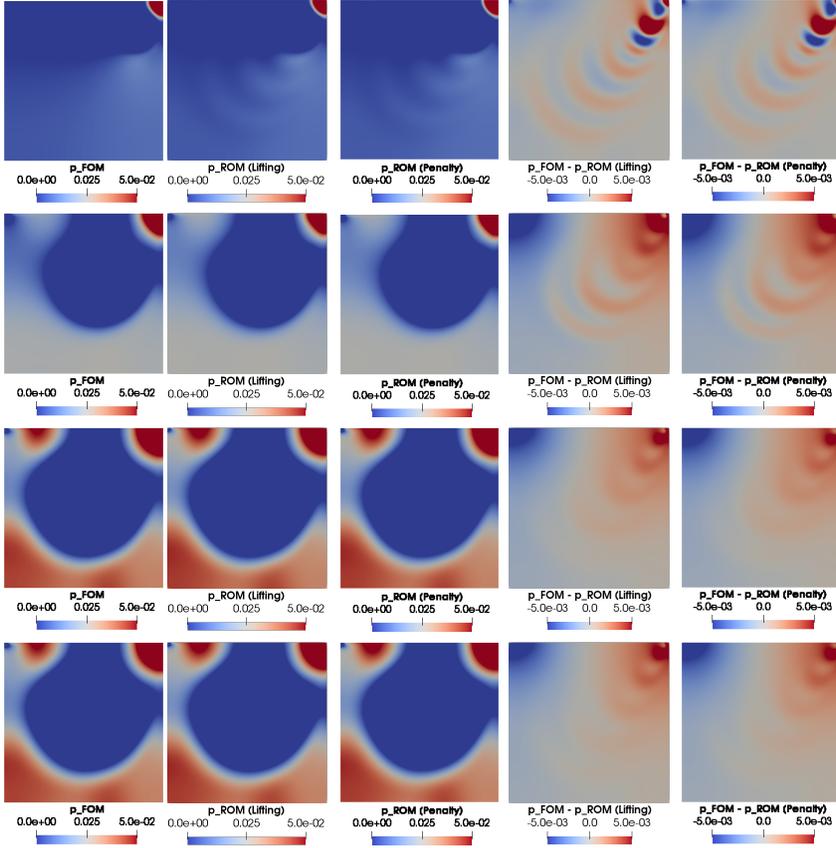


Figure 4.10: Comparison of the full order pressure fields (1st column), the ROM fields obtained with the lifting function method (2nd column) and penalty method (4th column) and the difference between the FOM and ROM fields obtained with the lifting function method (3rd column) and penalty method (5th column) at $t = 0.2, 1, 5$ and 10 s (from top to bottom) for the lid-driven cavity problem.

4.5.2 Y-junction flow problem

A full order simulation is performed for the Y-junction test case with varying inlet velocities (magnitude) according to Figure 4.3. In total 401 velocity and pressure snapshots are collected, which are then used to create the POD basis functions. To determine the number of basis functions necessary for the creation of the reduced subspace, the cumulative eigenvalues (based on the first 20 most energetic POD modes) are listed in Table 4.2.

Table 4.2: The cumulative eigenvalues for the Y-junction test case. The second and third columns report the cumulative eigenvalues (total of the first 20 modes) for the velocity and pressure fields, respectively.

N modes	\mathbf{u}	p
1	0.976478	0.967073
2	0.998492	0.989840
3	0.999724	0.998781
4	0.999859	0.999741
5	0.999924	0.999933
6	0.999967	0.999975
7	0.999989	0.999995
10	0.999999	0.999999

5 velocity and pressure modes are sufficient to retain 99.99% of the energy contained in the snapshots. These first five (homogenized) velocity and pressure modes are plotted in Figure 4.11. The first velocity magnitude modes have a symmetric pattern. In the case of the snapshot set with non-homogeneous BCs, the first mode is similar to the time-averaged solution of velocity snapshots. In the case of the snapshot set with homogeneous BCs, the first mode looks more like a fluctuation around the mean. From the third mode and higher, the modes are more or less alike whether the modes have homogeneous BCs or not.

In Figure 4.12 for each number of modes the time-averaged relative L_2 error between the FOM and the basis projection is plotted, on the left for velocity and on the right for pressure. For velocity this is repeated with a set of homogenized snapshots. As there are two inlet boundary conditions, the first two modes are the normalized lifting functions and all sequential modes are then the homogeneous basis functions obtained with the POD method. Therefore the average L_2 error is still above the order 10^{-1} as these modes do not contain any information about the full order solution. The figure shows that 11 velocity basis functions and 10 pressure basis functions are required to have a truncation error less than 10^{-3} . Taking also into account previous observation, these number of modes are used for calculating the ROM matrices.

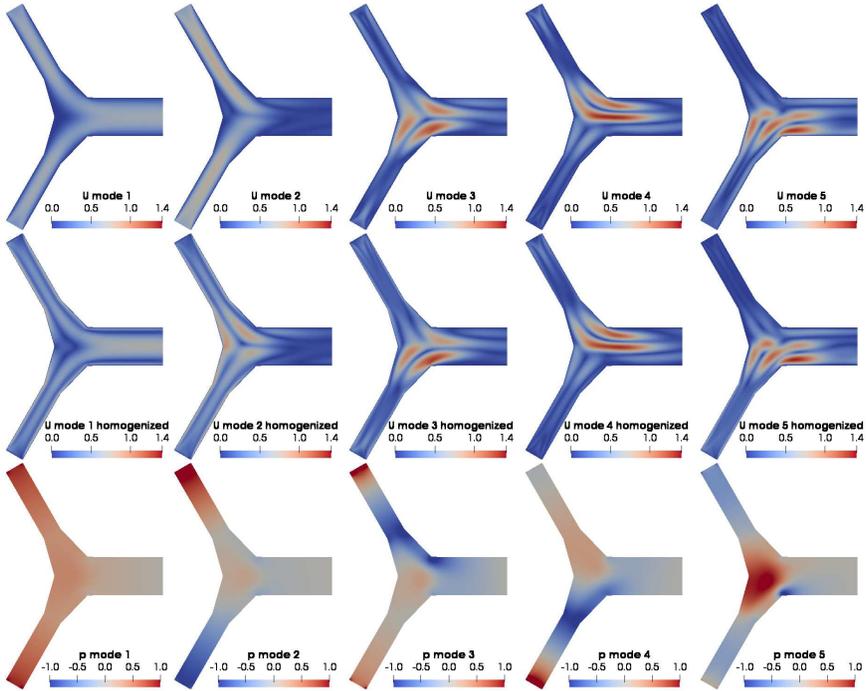


Figure 4.11: First 5 POD modes for (top) velocity, (middle) velocity with homogeneous BCs and (bottom) pressure for the Y-junction flow problem.

After applying the Galerkin projection with the obtained modes, the penalty factors are determined using the iterative procedure. Starting from an initial guess of 10^{-6} the penalty factors found are $5.9 \cdot 10^{-8}$ and 88.3 for inlet 1 and $1.1 \cdot 10^{-7}$ and 125 for inlet 2 in the x-direction and y-direction, respectively. The factors are determined within 41 iterations for an error tolerance of 10^{-5} and only the first five time steps are evaluated. However, it took only 15 iterations to have an error of $1.00009 \cdot 10^{-5}$ with penalty factors 0.0327 , 88.3 , 0.048 , 124.5 . So one could relax the criteria for the error a bit for a faster convergence.

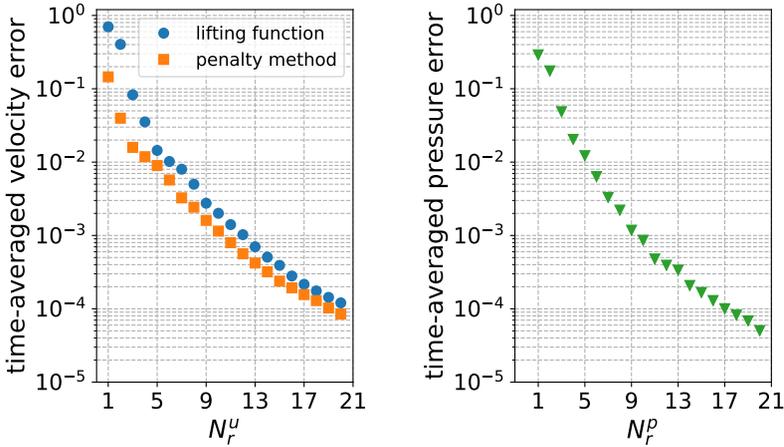


Figure 4.12: The time-averaged relative L_2 error per number of (left) velocity modes, N_r^u , and (right) number of pressure modes, N_r^p , for the Y-junction test case.

Thereafter, three ROMs are obtained; one without boundary enforcement method, one with the lifting function method and one with the penalty method. These are then consecutively tested for the time-dependent boundary conditions of Figure 4.5. The evolution in time of the relative L_2 error between the reconstructed fields is plotted in Figures 4.13 and Figures 4.14 for velocity and pressure, respectively.

In the case no boundary enforcement method is used, the relative error for both velocity and pressure is of the order 1 and larger for the vast part of the simulation.

The relative error is more or less the same for both boundary control methods, as also was observed previously for the lid-driven cavity test case, except around 9 s of simulation time. Then the difference in relative error for pressure between the two methods is the largest; the penalty method is about $2 \cdot 10^{-1}$ larger than the error obtained with the lifting function method. However, on the long term the penalty method performs slightly better. This can also be concluded by having a look at the kinetic energy relative error in Figure 4.15. Other than that, the relative velocity error is of the order 10^{-2} and for pressure 10^{-1} . A possible source for the larger pressure error is that the PIMPLE algorithm, consisting of predictor and correction steps for pressure and velocity, is used at full order level, while the coupled (pressure-velocity) system at the reduced order level is solved with Newton's iterative method. This is causing a discrepancy between the full order and reduced order model formulation. Nevertheless, the difference between the minimum and maximum relative error for both variables is about one order.

Furthermore, the absolute error between the FOM and the ROMs is shown in Figures 4.16 and 4.17 for velocity magnitude and pressure, respectively. For

velocity the absolute error between the FOM and the ROM is of the order 10^{-2} for all plotted simulation times and the absolute error for pressure is of the order 10^{-1} . For pressure, the error is indeed larger in the case of the penalty method compared to the lifting function method at 9 s of simulation time, as previously observed in Figure 4.14, but in general, the error distribution, for both the velocity and pressure fields, is similarly distributed over the domain, and thus the methods are performing the same.

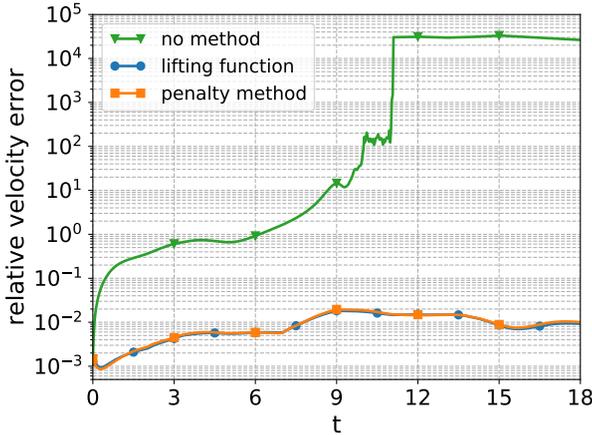


Figure 4.13: Time evolution of the relative L_2 error of velocity between the FOM and ROM with the lifting function and penalty method for the Y-junction flow problem.

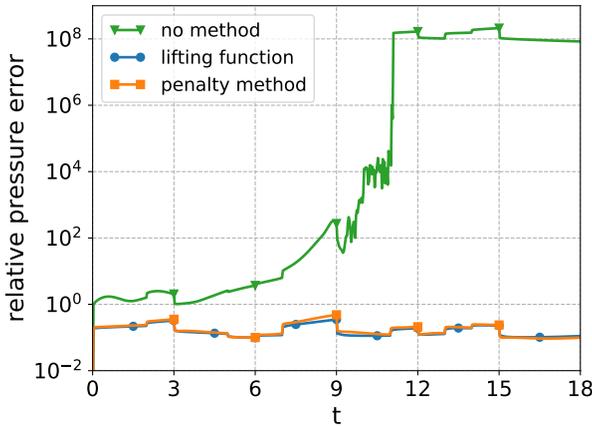


Figure 4.14: Time evolution of the relative L_2 error of pressure between the FOM and ROM with the lifting function and penalty method for the Y-junction flow problem.

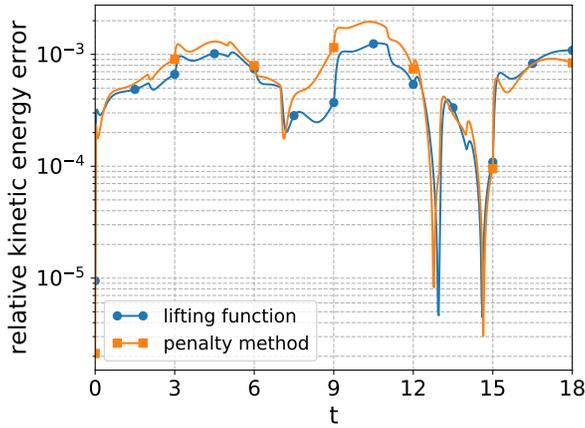


Figure 4.15: Time evolution of the relative L_2 error of kinetic energy between the FOM and ROM with the lifting function and penalty method for the Y-junction flow problem.

Finally, the computational times for performing the full order simulation, calculating the POD modes, the reduced matrices and performing the simulation at the reduced order level are listed in Table 4.3. Calculating the reduced matrices and the ROM solutions takes more time in the case of the lifting function method as the reduced basis space consists of four additional modes, namely the normalized lifting functions, compared to the penalty method. Determining the penalty factor with the iterative method takes 1.4 s. The speedup ratio between the ROM and the FOM is about 13 times for the lifting method and 24 times for the iterative penalty method.

Finally, the computational times for performing the full order simulation, calculating the POD modes, the reduced matrices and performing the simulation at the reduced order level are listed in Table 4.3.

Table 4.3: Computational time (clock time) for the FOM simulation, POD modes, calculating reduced matrices offline (Matrices), determining penalty factor with iterative method (Penalty) and ROM simulation.

Method	FOM	POD	Matrices	Penalty	ROM
Lifting	13 min.	7.6 s	9.2 s	-	59 s
Penalty	13 min.	7.9 s	4.7 s	1.4 s	33 s

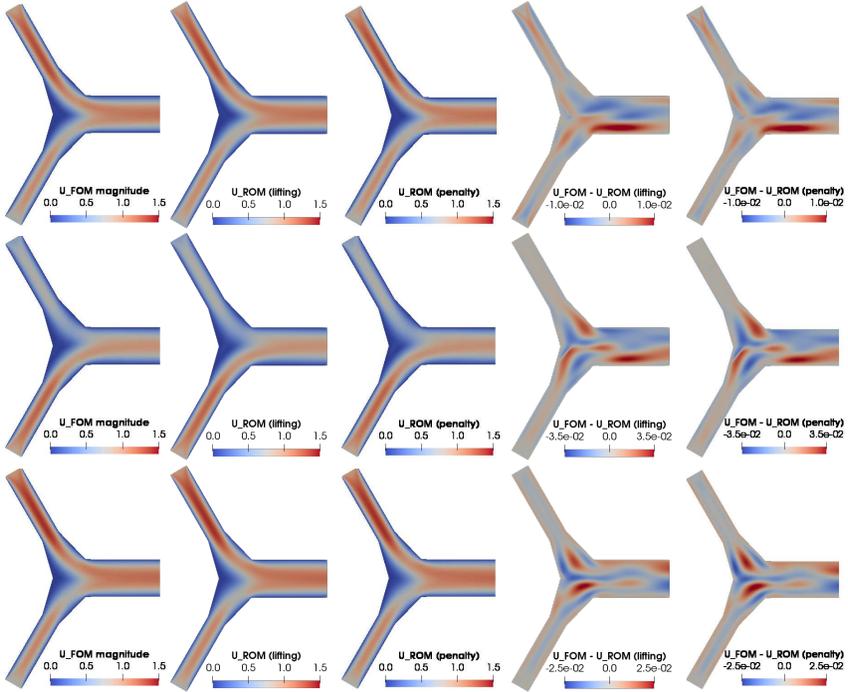


Figure 4.16: Comparison of the full order velocity magnitude fields (1st column), the ROM fields obtained with the lifting function method (2nd column) and penalty method (4th column) and the difference between the FOM and ROM fields obtained with the lifting function method (3rd column) and penalty method (5th column) at $t = 3, 9$ and 18 s (from top to bottom) for the Y-junction flow problem.

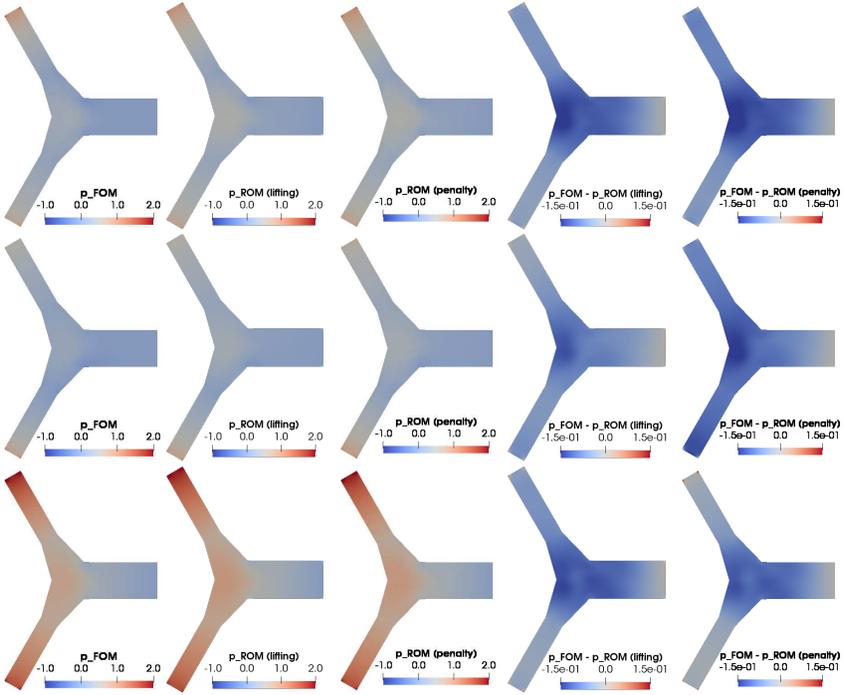


Figure 4.17: Comparison of the full order pressure fields (1st column), the ROM fields obtained with the lifting function method (2nd column) and penalty method (4th column) and the difference between the FOM and ROM fields obtained with the lifting function method (3rd column) and penalty method (5th column) at $t = 3, 9$ and 18 s (from top to bottom) for the Y-junction flow problem.

4.6 Discussion

The results have shown that the lifting function method and penalty method perform equally and lead to similar results. However, they have their own advantages and drawbacks. A disadvantage of the penalty methods is that the penalty factor cannot be determined a priori [93]. The implementation of an iterative solver to determine the penalty factor does however save time compared to performing numerical experimentation manually. On the other hand, even though a lifting function(s) can be determined beforehand, it may be hard to find a function that will lead to an accurate ROM and therefore extensive testing of ROMs for different functions can be needed. In this work, the lifting functions are obtained by solving a potential flow problem and are thus physics-based unlike the penalty factor, which is an arbitrary value. Moreover, this value needs to be chosen above a certain threshold to enforce the BCs in the ROM, but can lead to an inaccurate ROM solution if it is too high [234]. In that case, the penalty method fails for that specific problem.

Finally, an advantage of the penalty method stated in literature [161] is that long-time integration and initial condition issues are less of a problem compared to a lifting function method. Here the ROMs have not been tested for long-term integration, so further research is needed in order to confirm this statement. However, as tested for the Y-junction test case, the ROM is accurate and does not exhibit instabilities even outside the time domain in which snapshots were collected.

For both cases tested in this study, only one full order simulation has been performed for collecting the snapshots. However, snapshots from at least two different full order simulations are required for the penalty method if the BCs of the Y-junction are not time-dependent. The reason for this is that the boundary conditions are a linear combination of snapshots. Therefore, the boundary conditions can only be scaled rather than set to any arbitrary value if only snapshots from one full order simulation with constant BCs are used for the POD. When several sets of snapshots for different boundary values are required, one can optimize the POD procedure by using a nested POD approach [87].

It is important to note that the penalty factor is determined during the online phase and does not depend on high-fidelity data. Therefore, no modification are needed in the case of parametric problems that, for example, use the viscosity as the physical parameter.

In the case of the Y-junction test case, the penalty method can be used to adjust the direction of the inlet flow in the ROM. One penalty factor is enforcing the x-direction and another the y-direction. Nevertheless, new snapshots for different inlet angles are required as the current POD bases do not contain this information. For the lifting function method, it is often problematic to determine suitable lifting functions that are physical. Ideally, the lifting functions are orthogonal to each

other as in the work of Hijazi et al. [113] who studied a flow past an airfoil with parameterized angle of attack and inflow velocity. They used two lifting functions with orthogonal inflow conditions: $\zeta_1 = (0,1)$ and $\zeta_2 = (1,0)$ on Γ_i , respectively. These lifting functions are obtained by solving two linear potential flow problems. In that way, it is possible to adjust the direction of the flow at a inlet by scaling the associate lifting functions accordingly. However, specifying a purely tangential velocity at the inlets of the Y-junction would result in unphysical lifting functions. Thus, this approach is only suitable for a few problems and will not always lead to physical results.

In the case of non-physical lifting functions, the ROM gets unstable or the ROM solutions are polluted with noise. This strongly depends on the chosen lifting functions.

Moreover, both methods can, in theory, also be used for controlling pressure boundary conditions, but this is not studied in this work.

In this study, the exploitation of a pressure Poisson equation has been incorporated in the ROM as a stabilization method. Even though the ROMs are indeed stable, the relative error for pressure is about an order higher than for velocity. Alternatively, the supremizer enrichment of the velocity space technique could be used to stabilize the ROM, which may lead to more accurate pressure fields [19, 137, 241].

Furthermore, the ROMs can be improved by using a second order backward method for the time discretization of the ROM as the FOMs are treated using a second order backward differencing scheme.

Finally, for the Y-junction test case, the full order snapshots and the ROM solutions all have inlet velocities between an identical maximum and minimum value. The ROM could become less stable and accurate in case it is tested for values outside this range. Therefore it is recommended to collect snapshots for the same range as for which the ROM boundary needs to be controlled.

4.7 Conclusion

Two boundary control methods are tested: the lifting function method and the iterative penalty method for controlling the velocity boundary conditions of FV-based POD-Galerkin ROMs. The penalty method has been improved by using an iterative solver for the determination of the penalty factors, rather than using numerical experimentation. The factors are determined by the iterative solver in about a second for both test cases. The results of the reconstructed velocity and pressure fields show that both methods are performing equally. Moreover, the reduced order model of which the boundary conditions are controlled with the iterative penalty method is about two times faster compared to the lifting function method for the Y-junction flow case.

A pressure Poisson equation approach is applied for the reconstruction of the pressure field and to stabilize the ROM. The accuracy of the reconstructed pressure fields can be improved by using a supremizer enrichment approach rather than solving the Pressure Poisson Equation[241].

Finally, a speedup factor, the ratio between the FOM and ROM simulation time, of 308 is obtained with the iterative penalty method and of 270 with the lifting function method for the lid-driven cavity test case. The speedup factors are 24 and 13, respectively, for the Y-junction test case.

5

POD-Galerkin reduced order model of the Boussinesq approximation for buoyancy-driven enclosed flows

In this chapter, a parametric reduced order model for buoyancy-driven laminar flows in an enclosed cavity is developed with the POD-Galerkin method. This work is an extension of the work on the development of a ROM for weakly coupled Navier-Stokes equations with the heat equation performed in [87]. The results of this chapter are published as [244].

5.1 Introduction

In nuclear engineering applications, decay heat can be removed passively via natural circulation in operative and accidental conditions without the use of active components. However, the development of the flow, for instance thermal stratification at low flow conditions, during the transition from forced to natural circulation may have a detrimental effect on the efficiency of the passive residual heat removal. Therefore, reliable computational methods are required to accurately quantify naturally circulating flows and the associated transient phenomena [260].

A two-way coupling between momentum and energy is required to model the complex dynamics of natural circulation. Therefore, the Boussinesq approximation is often applied to simplify the problem by neglecting the effect of local den-

sity differences of the fluid, induced by temperature, except for the influence of the gravitational body force on the flow. This approximation is valid as long as the difference in density is much smaller than the reference density.

5.2 The Boussinesq approximation for the Navier-Stokes equations

The mathematical problem on which this work is focused, i.e. buoyancy-driven enclosed flows, is given by the unsteady incompressible Navier-Stokes equations, without any turbulence treatment, in the presence of the gravity body force and the energy equation. The general form of the equations is

$$\left\{ \begin{array}{ll} \frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) - \nabla \cdot (\mu \nabla \mathbf{u}) = -\nabla p + \rho \mathbf{g} & \text{in } \Omega, \\ \frac{\partial \theta}{\partial t} + (\nabla \cdot \mathbf{u})\theta - \nabla \cdot (\alpha \nabla \theta) = 0 & \text{in } \Omega, \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega, \\ \mathbf{u}(\mathbf{x}, t) = \mathbf{0} & \text{on } \Gamma, \\ \theta(\mathbf{x}, t) = f(\mathbf{x}) & \text{on } \Gamma, \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}) & \text{in } \Omega, \\ \theta(\mathbf{x}, 0) = \theta_0(\mathbf{x}) & \text{in } \Omega, \end{array} \right. \quad (5.1)$$

where \mathbf{u} is the velocity, p is the pressure, θ is the temperature, ρ is the density, μ is the dynamic viscosity, α is thermal diffusivity and \mathbf{g} the gravitational acceleration. $\mathbf{u}_0(\mathbf{x})$ and $\theta_0(\mathbf{x})$ denote the initial conditions of the velocity and temperature, respectively, at time $t = 0$ s. The Boussinesq approximation assumes that ρ is constant for all terms in Equation. 5.1, except for the gravitational term. To avoid numerical issues due to large gradients of the buoyancy force, buoyant flow solvers typically use $p'_{rgh} = p - \rho \mathbf{g} \cdot \mathbf{r}$, with \mathbf{r} the position vector, rather than the static pressure p . The momentum equations with the Boussinesq approximation are then given by

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u} \otimes \mathbf{u}) - \nabla \cdot (\nu \nabla \mathbf{u}) = -\nabla p_{rgh} - (\mathbf{g} \cdot \mathbf{r}) \nabla \rho_k, \quad (5.2)$$

where $\nu = \mu/\rho$ is the kinematic viscosity, $p_{rgh} = (p - \rho \mathbf{g} \cdot \mathbf{r})/\rho_0$ is referred to as a pressure shift and $\rho_k = 1 - \beta(\theta - \theta_0)$ with β the thermal expansion coefficient. The reference state is taken at p_0 , ρ_0 and θ_0 .

5.3 POD-Galerkin reduced order model of the Boussinesq approximation

The proper orthogonal decomposition method is used to create a set of basis functions containing the essential dynamics of the previously described full order model

(Equation 5.1). It is assumed that there exists an approximation of the problem, so that the FOM can be expressed as a linear combination of orthogonal spatial modes multiplied by time-dependent coefficients. For the velocity and temperature, the approximations are given by

$$\mathbf{u}(\mathbf{x}, t) \approx \mathbf{u}_r(\mathbf{x}, t) = \sum_{i=1}^{N_r^u} \varphi_i(\mathbf{x}) a_i(t), \quad \theta(\mathbf{x}, t) \approx \theta_r(\mathbf{x}, t) = \sum_{i=1}^{N_r^\theta} \psi_i(\mathbf{x}) b_i(t), \quad (5.3)$$

where φ_i and ψ_i are the modes of the velocity and temperature, and respectively a_i and b_i the corresponding time-dependent coefficients. N_r^u is the number of velocity modes and N_r^θ is the number of temperature modes. The modes are obtained by solving an eigenvalue problem [239, 241] on snapshots which are generated by sampling the FOM at several moments in time, as described in Section 2.4.1.

To obtain the reduced order model, the POD is combined with the Galerkin projection, for which the full order system is projected onto the reduced subspace of POD modes and the difference between the FOM solution and the approximated one is minimized [205]. For more details about the Galerkin projection method the reader is referred to Section 2.4.2.

For the Boussinesq approximation, special attention is paid to the Galerkin projection of the shifted pressure term, which is given by

$$\begin{aligned} (\varphi_i, \nabla p_{rgh})_{L_2(\Omega)} &= \int_{\Omega} \varphi_i \cdot \nabla p_{rgh} d\Omega \\ &= - \int_{\Omega} p_{rgh} (\nabla \cdot \varphi_i) d\Omega + \int_{\Gamma} p_{rgh} (\varphi_i \cdot \mathbf{n}) d\Gamma. \end{aligned} \quad (5.4)$$

The first term on the right hand side is zero as the POD modes, defined as a linear combination of snapshots, preserve the divergence free property of the flow field [205]. Also the second term on the right hand side is zero as the velocity is zero at the wall in the case of enclosed flow. Therefore, it is not necessary to include a pressure term in the reduced order model for buoyancy-driven enclosed flow [149, 161]. The resulting ROM is then given by

$$\begin{cases} \mathbf{M}_r \dot{\mathbf{a}} + \mathbf{a}^T \mathbf{C}_r \mathbf{a} - \nu \mathbf{D}_r \mathbf{a} + \mathbf{H}_r \mathbf{b} = \mathbf{0}, \\ \mathbf{W}_r \dot{\mathbf{b}} - \alpha \mathbf{Y}_r \mathbf{b} + \mathbf{a}^T \mathbf{Q}_r \mathbf{b} = \mathbf{0}, \end{cases} \quad (5.5)$$

where

$$\begin{aligned} M_{r_{ij}} &= (\varphi_i, \varphi_j)_{L_2(\Omega)}, C_{r_{ijk}} = (\varphi_i, \nabla \cdot (\varphi_j \otimes \varphi_k))_{L_2(\Omega)}, \\ D_{r_{ij}} &= (\varphi_i, \nabla \cdot (\nabla \varphi_j))_{L_2(\Omega)}, \\ H_{r_{ij}} &= (\varphi_i, (\mathbf{g} \cdot \mathbf{r}) \nabla (1 - \beta(\psi_j - \theta_0)))_{L_2(\Omega)}, \\ W_{r_{ij}} &= (\psi_i, \psi_j)_{L_2(\Omega)}, Y_{r_{ij}} = (\psi_i, \nabla \cdot (\nabla \psi_j))_{L_2(\Omega)}, \\ Q_{r_{ijk}} &= (\psi_i, \nabla \cdot (\varphi_j, \psi_k))_{L_2(\Omega)}. \end{aligned} \quad (5.6)$$

The initial conditions for the ROM are given by

$$a_i(0) = (\varphi_i, \mathbf{u}_0(\mathbf{x}))_{L_2(\Omega)}, \quad b_i(0) = (\psi_i, \theta_0(\mathbf{x}))_{L_2(\Omega)}. \quad (5.7)$$

5.3.1 Imposing boundary conditions with the lifting function method

The boundary conditions are not explicitly present in the reduced system and, therefore, they cannot be parametrized directly as discussed in the previous chapter. Whether the lifting function method or the penalty method is most suitable, is problem dependent. The advantage of the lifting function method is that the functions can be determined on beforehand. Moreover, the lifting function is, unlike a penalty factor, physics-based [93, 161]. The lifting function method is an appropriate choice for the parametric reduced order model for buoyancy-driven enclosed flow. The lifting functions can be determined by solving a Laplacian function as no forced convection is applied to the buoyancy-driven enclosed flow problem, which is described in detail in Section 5.4.

The boundary conditions are imposed in the reduced order model of Equation 5.5 using the lifting function method as described in the previous chapter (Section 4.3). The lifting functions, $\tilde{\zeta}(\mathbf{x})$, are obtained by solving a system, as close as possible to the full order system, where the boundary of interest is set to 1 and everywhere else to 0. Then the snapshots are modified as follows

$$\theta'(\mathbf{x}, t) = \theta(\mathbf{x}, t) - \sum_{i=1}^{N_{BC}} \tilde{\zeta}_i(\mathbf{x}) \Theta_{BC_i}, \quad (5.8)$$

where N_{BC} is the number of non-homogeneous BCs and Θ_{BC} is the value of the boundary condition. POD is then applied to these homogeneous snapshots to obtain the homogeneous temperature modes, ψ'_i , as described in Section 4.3.2. Finally, the temperature field is approximated with

$$\theta_r(\mathbf{x}, t) = \sum_{j=1}^{N_{BC}} \tilde{\zeta}_j(\mathbf{x}) \Theta_{BC_j} + \sum_{i=1}^{N_r^\theta} \psi'_i(\mathbf{x}) b_i(t), \quad (5.9)$$

which satisfies the boundary conditions of the problem. Θ_{BC} can be parametric. For more details on the lifting function method, the reader may take a look at [87, 239] or Section 4.3.

5.4 Numerical test case of an enclosed cavity

In this study, a simple configuration for natural convection is studied that consists of a 2D square enclosed cavity with differentially heated walls opposite of each

other. The simulations are carried out on a square domain of length $L = 0.1$ m on which a (100×100) uniform mesh is constructed. A sketch of the geometry is depicted in Figure 5.1.

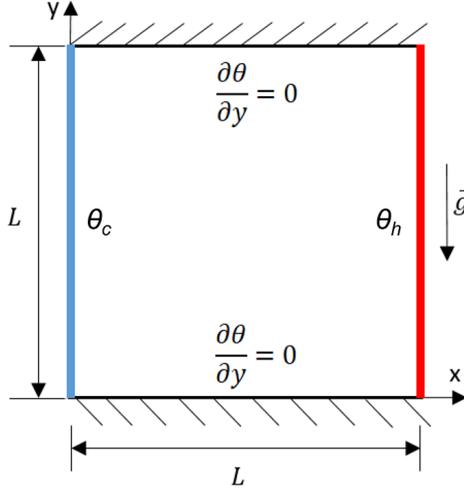


Figure 5.1: A sketch of the geometry of the 2D square cavity with differentially heated walls.

The initial temperature, θ_0 , and initial velocity, \mathbf{u}_0 , are 300 K and $(10^{-4}, 0)$ m/s, respectively. The properties are taken for air with thermal diffusivity, $\alpha = \nu / \text{Pr} = 1.4 \cdot 10^{-5}$ m²/s, kinematic viscosity $\nu = 1 \cdot 10^{-5}$ m²/s and $\text{Pr} = 0.71$. Furthermore, the coefficient of thermal expansion, β , is $3 \cdot 10^{-3}$ K⁻¹. The left side of the cavity is kept at the cold temperature, θ_c , and the right side at the hot temperature, θ_h , with $\theta_h > \theta_c$. Four parameter sets are considered for the temperature BCs, which are listed in Table 5.1.

Table 5.1: Parameter sets for the temperature BCs.

Parameter set #	θ_c [K]	θ_h [K]
0	298.5	301.5
1	299.0	301.0
2	298.0	302.0
3	298.7	301.2

The unsteady governing equations are iteratively solved by the FV method with the *buoyantBoussinesqPimpleFoam* solver of the open source C++ library OpenFOAM [130]. The PIMPLE algorithm is used for the pressure-velocity coupling [177]. For both the full and reduced order simulations, the time discretization is treated using a backward differencing scheme.

A constant time step of $\Delta t = 1 \cdot 10^{-3}$ s is applied and the total simulation time is 10 s. Snapshots of the velocity and temperature fields are collected every 0.01 s, resulting in a total of 1000 snapshots for each parameter set. The full order simulations are performed in OpenFOAM 6, while the reduced order model is solved with ITHACA-FV, a C++ library based on the finite volume solver OpenFOAM. For more details on the ITHACA-FV code, the reader is referred to [239, 240, 241]. A lifting function, for each non-homogeneous BC, is determined by solving a steady state Laplacian function for temperature, $\Delta\theta = 0$, according to the methodology described in Section 5.3.1.

Finally, the relative prediction error between the FOM and ROM velocity and temperature fields and the basis projection error are determined at each time instance according to Section 2.5.5.2.

5.5 Results and analysis

First, a full order simulation is performed for parameter set 0. In total, 1001 snapshots (including 1 for the initial condition) velocity and temperature snapshots are collected. Furthermore, the lifting functions are obtained by solving the steady state Laplacian function and are shown in Figure 5.2. These functions are then used to homogenize the snapshots and to create homogeneous basis functions with the POD method.

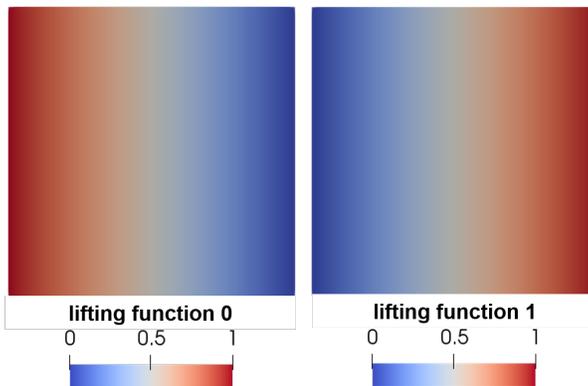


Figure 5.2: The lifting functions for temperature.

5.5.1 POD modes

To determine the number of basis functions needed for the creation of the reduced subspace, the cumulative eigenvalues are plotted in Figure 5.3 on the left. The plot shows that 7 modes are sufficient to retain 99.9% of the energy contained in the snapshots, for both velocity and temperature.

Furthermore, the full order snapshots are projected onto the POD basis for a range of modes, from 1 to 15, to obtain time-dependent coefficients that are then used to reconstruct the fields, called the basis projection. For each number of modes the time-averaged relative basis projection error is calculated and plotted in Figure 5.4 for velocity and temperature.

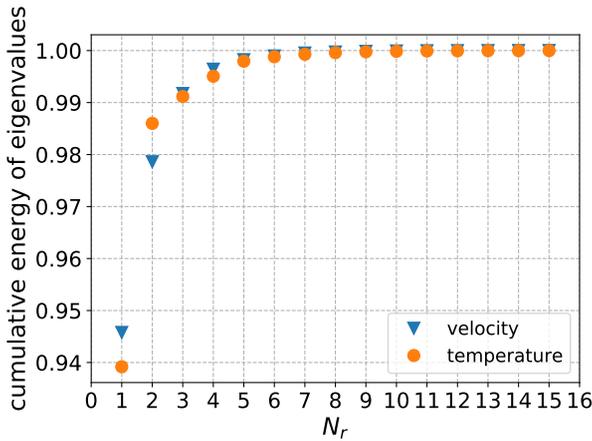


Figure 5.3: Cumulative eigenvalues for velocity and temperature.

It can be seen that the slope of the decay of the time-averaged relative basis projection error for the velocity is the same as for the temperature. However, Figure 5.4 shows that 5 basis functions are required to have a time-averaged truncation error less than 10^{-4} for temperature, while for velocity, 5 basis functions are needed to have an error less than 10^{-1} .

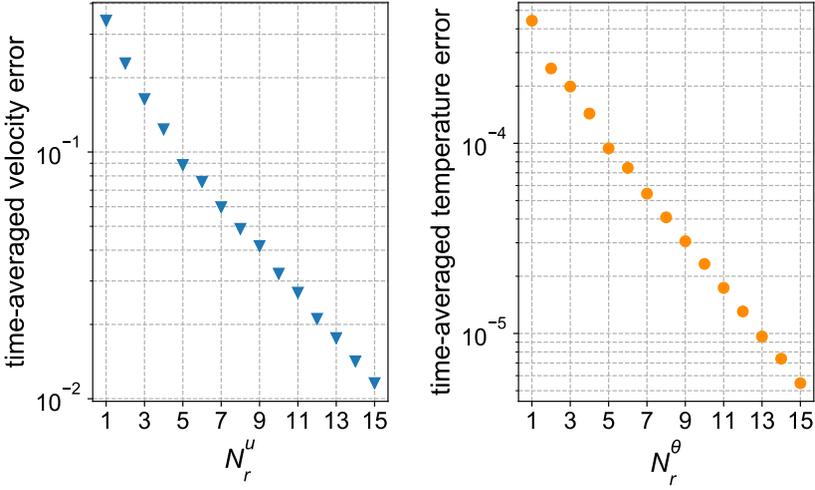


Figure 5.4: The time-averaged relative basis projection error: (left) per number of velocity modes (N_r^u); (right) per number of temperature modes (N_r^θ).

The first five velocity and homogenized temperature modes are plotted in Figure 5.5. The velocity magnitude modes have a symmetric pattern. Furthermore, the first velocity mode is close to the steady state solution of the problem, while the first temperature mode looks like a fluctuation around the mean field due the homogenization by the lifting functions.

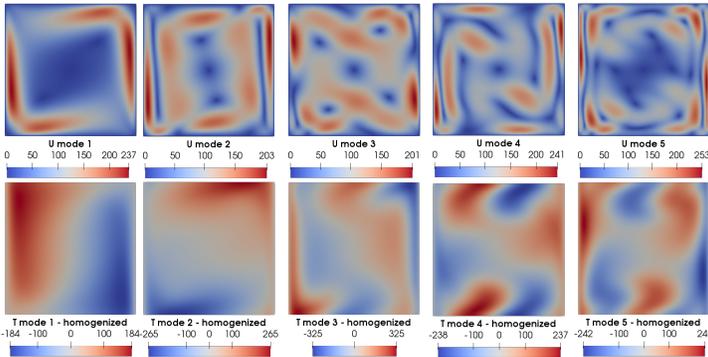
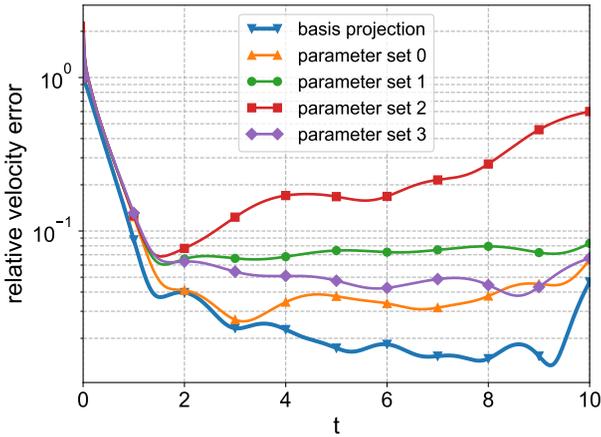


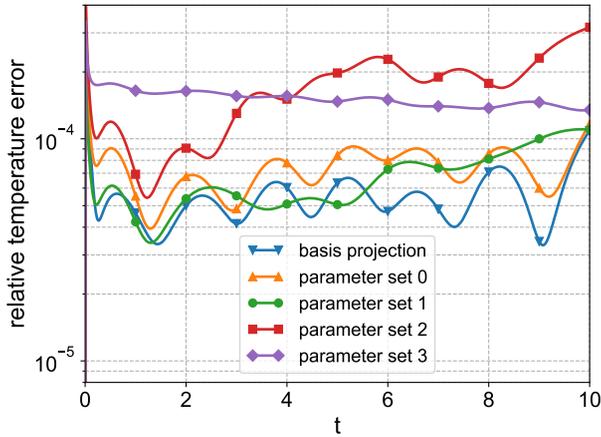
Figure 5.5: First five POD modes for velocity (top) and temperature with homogeneous BCs (bottom).

5.5.2 Reduced order simulations for new parameter sets

Taking also into account previous observation based on the cumulative energy of the eigenvalues, 7 velocity and 7 homogeneous temperature modes are used for the reduced basis creation. Then, the ROM matrices are calculated and the obtained ROM is tested for all parameter sets. The evolution in time of the relative L_2 error between the reconstructed fields and the full order solutions is plotted in Figure 5.6 for velocity and temperature.



(a) velocity



(b) temperature

Figure 5.6: Time evolution of the relative error between the FOM and ROM with the lifting function for all parameter sets and the basis projection error.

For parameter set 0, the prediction errors are of the same order as its basis projection error using 7 modes for both velocity and temperature. The velocity error is more than 1 at the beginning of the ROM simulation, as the flow starts at rest and therefore a small deviation of the initial velocity field leads to a high relative error as the error is magnified by a small velocity magnitude [161]. After about 1 second of simulation time the relative error of the velocity is less than 10^{-1} and for temperature below 10^{-4} . As expected, the performance of the ROM velocity is best when tested for the same parameter set on which the POD has been performed, but increases for each of the following sets. For parameter set 1 and 3, after 2 seconds of simulation time, the error remains of the same order. However, the error of parameter set 3 keeps increasing over time, meaning that the ROM is less stable for those values. The same applies to the relative error of the temperature fields, however at some moments in time the ROM performs better for parameter set 1, where the temperature difference between the walls is smaller compared to set 0, than the basis projection. Then for parameter set 3, where an asymmetric temperature difference is applied on the walls, the ROM is stable, but performs worse than parameter set 1 due to the strong nonlinearity of the flow with respect to the BCs. As no snapshots are collected for a similar case, part of the flow pattern is not contained in the snapshots and therefore also not in the ROM. The full order and reconstructed velocity and temperature fields for parameter set 3 are shown in Figure 5.7 at $t = 1$ s, $t = 5$ s and the final simulation time $t = 10$ s. The absolute error between the FOM and the ROMs is shown in the same figure, which is of the order 10^{-3} m/s for velocity and 10^{-2} K for temperature.

Finally, the computation times on a single Intel Premium CPU G2130 @ 3.20 GHz processor for calculating the FOM, POD modes, the reduced matrices and for the ROM are 563, 15, 0.5 and 27 seconds, respectively. Therefore, the ROM is about 20 times faster than the FOM.

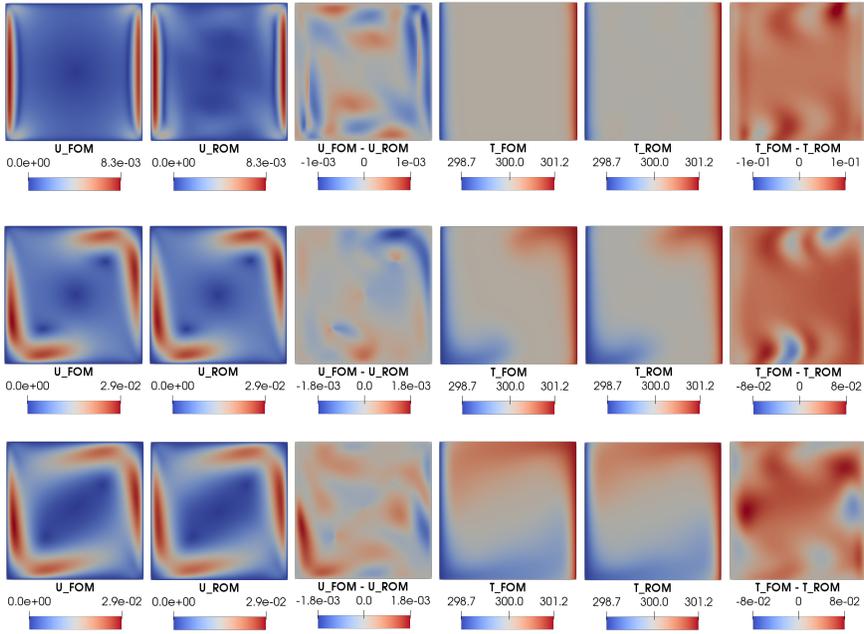


Figure 5.7: Comparison of the full order (FOM) velocity magnitude (1st column) and temperature (4th column) fields with the reduced order (ROM) fields (2nd and 5th column) and the difference between the FOM and ROM fields (3rd and 6th column) for parameter set 3 at $t = 1$ s (top), 5 s (middle) and 10 s (bottom).

5.6 Discussion

Only one full order solution for parameter set 0 was required to perform a ROM with the lifting function method for parameter set 1 and 3. However, the ROM is unstable for parameter set 2, even though the temperature BCs were only deviating by 0.5 K from those of set 0 (like parameter set 1), for which snapshots are collected. However, the temperature difference between the walls is larger for set 2 than the other sets for which the ROM is stable. As for larger temperature differences, part of the flow pattern is not contained in the snapshots, the ROM is not capable of reconstructing those fields. As a solution, when a larger range of parameter values needs to be tested, one can use snapshots from several FOMs to generate the POD modes. Since the POD modes are based on a linear combination of snapshots, while buoyancy-driven flows are highly nonlinear due to the coupling between the nonlinear momentum and energy equations, adding snapshots from different FOMs can make the ROM applicable for a larger range of parameter values.

The relative prediction error for velocity during the first second of the ROM simulation is about one order higher compared to the rest of the simulation. Especially for velocity, the error is more than 1 at the beginning of the ROM simulation. A small deviation of the homogeneous initial field, even for the basis projection using a reduced number of modes, leads to a high relative error compared with the full order solution. The absolute error is of a smaller order, as shown by the absolute difference plot between the FOM and ROM fields at different time instances, and so, depending on the application, the error can be acceptable. Adding more modes to the reduced basis will reduce the error, but then the ROM will become slower. Therefore, there is a trade-off between the accuracy and the gain in computational time.

In this study, the recovery of the pressure has not been incorporated in the ROM. A supremizer enrichment of the velocity space technique or exploitation of a pressure Poisson equation (as presented in Chapter 4) can, for instance, be incorporated in the Galerkin projection to include the pressure in the ROM [241].

Finally, simulations have only been performed for air flow, as special attention is required to low Prandtl number thermal-hydraulics when modeling liquid metal flows.

5.7 Conclusion

In this chapter, a FV-based POD-Galerkin ROM with Boussinesq approximation is presented. The ROM is constructed such that it is consistent with the FOM and both velocity and temperature fields are considered. The reconstruction of the pressure field is, however, not considered as only enclosed flows are investigated in this study. An additional buoyancy term in the ROM induces a two-way coupling between momentum and energy, that is required for buoyancy modeling for nuclear thermal-hydraulic studies and other related industrial problems. A ROM is constructed of which the temperature BCs are parametrized using a lifting function method. The results of the ROMs for a simple 2D differentially heated cavity show that the relative error of temperature is of the order $\mathcal{O}(10^{-4})$ and of velocity $\mathcal{O}(10^{-1})$ for the vast part of the simulations. The ROMs are stable, except when the temperature difference between the walls is larger than the case for which the snapshots, for the reduced basis construction, are collected. The accuracy can be improved by adding more modes to the reduced basis obtained with the POD method. Finally, the ROM is about 20 times faster than the FOM run on a single processor.

In the next chapter, the pressure will be included in the ROM and the model will be extended for turbulent buoyant flows, which will be essential to simulate buoyancy in nuclear reactors.

6

POD-Galerkin reduced order model of a turbulent convective buoyant flow of sodium over a backward-facing step

In this chapter, the POD-Galerkin reduced order modeling strategy for buoyancy-driven flows of Chapter 5 is extended for steady-state Reynolds-averaged Navier-Stokes simulations of turbulent convective buoyant flows of a low-Prandtl number fluid. The results of this chapter are published as [249].

6.1 Introduction

Heat transfer in liquid metals is of interest, for instance, in nuclear facilities that use high-temperature heat transfer media as a coolant. This type of heat transfer fluids is typically characterized by a low-Prandtl (Pr) number, where Pr is the ratio of diffusivity of momentum to diffusivity of heat. Due to the high thermal diffusivity of low-Pr fluids, the influence of buoyancy on the flowfield is present at much higher Reynolds numbers compared to air or water [253]. Therefore, the flow regime between forced and natural convection, where driven flow interacts with buoyancy effects, needs to be studied in many heat transfer applications [179].

Low-Prandtl number fluid turbulent flows, and especially their associated turbulent heat fluxes, are complicated to model numerically as heat conduction through the boundary layer has more dominant effect with respect to convection. There-

fore, the thermal boundary layers become thicker when the Prandtl number is decreased. This means that there is a difference in the range of the spatial (and temporal) scales of temperature and velocity. As a consequence, the conductive heat fluxes near walls become more important. Therefore, it is problematic to apply the Reynolds analogy, which assumes a constant turbulent Prandtl number, Pr_t , close to unity, to calculate the local turbulent heat fluxes [213]. Furthermore, Pr influences not only the temperature field and the heat flux modeling, but also the velocity field and the shear modeling in the case of buoyancy-aided flows [100]. Therefore, heat transfer in liquid metals, compared to common fluids with a Prandtl number around unity, requires additional or different (physical) modeling.

Only a few numerical studies on incompressible turbulent convective buoyant flows for low-Prandtl number fluid flows can be found in literature. Three studies are highlighted here: Cotton and Jackson [56] performed numerical calculations for a buoyancy-aided mixed convective turbulent flow in a vertical pipe for liquid sodium ($Pr = 0.005-0.01$). Niemann and Frohlich [179] investigated a turbulent flow of liquid sodium over a backward-facing step at forced and buoyancy-aided mixed convection using direct numerical simulation. And most recently, Oder et al. presented direct numerical simulation of low-Prandtl fluid flow over a confined backward-facing step [183].

Schumm et al. [229, 230] compared steady-state Reynolds-averaged Navier-Stokes simulations with the direct numerical simulations performed by Niemann and Frohlich [179] and concluded that the predicted velocity, turbulence kinetic energy and Reynolds shear stress profiles are in good agreement with the DNS data. They based the choice of the turbulence model for the Reynolds stresses, namely the Ince and Launder's model [126], on the study of Cotton and Jackson [56]. This turbulence model is basically the model of Launder and Sharma [150] including the near-wall length-scale correction term from Yap [297] in the equation of the dissipation rate of turbulence kinetic energy. The model is widely used due to its algorithmic simplicity and relatively good performance [118, 213] compared to the more advanced model of Hanjalić et al. [106] and second-moment closure models (e.g. Craft et al. [58], Dol et al. [67] and Manceau et al. [168]). Moreover, Schumm et al. modeled the turbulent heat flux with a simple gradient diffusion hypothesis (SGDH). In addition, they evaluated the turbulent Prandtl number locally with the correlation of Kays [135].

In the modeling and computation of industrial turbulent flows, RANS simulation is often preferred due to its relatively lower computational cost in comparison with the more detailed large eddy simulation and direct numerical simulation. However, even RANS simulation is unfeasible for applications that require (almost) in real time modeling or testing of a large number of different system configurations, for instance for control purposes, sensitivity analyses or uncertainty quantification studies. This has motivated the development of reduced order mod-

eling techniques.

The POD-Galerkin approach has recently been used by Lorenzi et al. [161] and Hijazi et al. [111, 112] to reduce the RANS equations in a finite volume framework. Stabile et al. [238] used a different POD-Galerkin based approach for the turbulence closure, namely the variational multi-scale approach. Other recent efforts that deal with POD-based ROMs using a LES approach for the turbulence modeling can be found in [30, 91, 294]. On the other hand, Carlberg et al. [44] and Xiao et al. [288] presented a Petrov-Galerkin projection approach for the reduced order modeling of the Navier-Stokes equations.

Moreover, Georgaka et al. [87] developed a POD-Galerkin reduced order model of weakly coupled parametric Navier-Stokes and energy equations. They also included turbulence modeling in their model [88]. On the other hand, Vergari et al. [267] and also the author of this work [244] developed a reduced order model of buoyancy-driven flow with the employment of the Boussinesq approximation (as described in Chapter 5).

In this Chapter, a reduced order model is developed of which the underlying full order model is corresponding to the model of Schumm et al. [229, 230] for steady-state RANS simulations for a turbulent convective buoyant flow of sodium over a backward-facing step.

6.2 Full order turbulence model

The steady-state governing equations for an incompressible Newtonian fluid, based on the low-Reynolds Launder-Sharma k - ε model [150], for turbulent buoyancy-driven flows in the mixed convection regime are

$$\nabla \cdot \mathbf{U} = 0, \quad (6.1)$$

$$\nabla \cdot (\mathbf{U} \otimes \mathbf{U}) = -\nabla P + \nabla \cdot \left[\nu \left(\nabla \mathbf{U} + \left(\nabla \mathbf{U}^T \right) \right) - \overline{\mathbf{u}'\mathbf{u}'} \right] - \mathbf{g}\beta(\theta - \theta_{ref}), \quad (6.2)$$

$$\nabla \cdot (\mathbf{U}\theta) = \nabla \cdot \left(\alpha \nabla \theta - \overline{\mathbf{u}'\theta'} \right), \quad (6.3)$$

$$\nabla \cdot (\mathbf{U}k) = \nabla \cdot \left[\left(\nu + \frac{\nu_t}{\sigma_k} \right) \nabla k \right] + P_k - D - \varepsilon, \quad (6.4)$$

$$\nabla \cdot (\mathbf{U}\varepsilon) = \nabla \cdot \left[\left(\nu + \frac{\nu_t}{\sigma_\varepsilon} \right) \nabla \varepsilon \right] + \frac{\varepsilon}{k} [C_{\varepsilon 1} f_1 P_k - C_{\varepsilon 2} f_2 \varepsilon] + E, \quad (6.5)$$

where \mathbf{U} , P , and θ are the ensemble averaged fields for velocity, kinematic pressure, which is pressure divided by the fluid density ρ , and temperature, respectively. \mathbf{u}' and θ' are the turbulent fluctuating components for velocity and temperature, respectively. Equations 6.1, 6.2 and 6.3 are the continuity, momentum and energy equations, respectively. Equation 6.4 is the transport equation for turbulence kinetic energy k and Equation 6.5 is the transport equation for the rate of dissipation of turbulence kinetic energy ε . Furthermore, ν is the kinematic viscosity, ν_t is the eddy viscosity and α is the thermal diffusivity. The buoyancy is

considered by the employment of the Boussinesq approximation in the last term of Equation 6.2, where θ_{ref} is a reference temperature, \mathbf{g} the gravitational acceleration and β the thermal expansion coefficient. To avoid numerical issues, due to large gradients of the buoyancy force, buoyant flow solvers typically use the shifted kinematic pressure $P_{rgh} = P - \mathbf{g} \cdot \mathbf{r}$, with \mathbf{r} the position vector, rather than the static kinematic pressure P . The production term of k in Equations 6.4 and 6.5 is given by

$$P_k = -\overline{\mathbf{u}'\mathbf{u}'}\nabla U. \quad (6.6)$$

Note that with this model the effect of buoyancy is not modeled in the turbulence transport equations (Equations 6.4 and 6.5) [167, 172, 210], which is in accordance with the model of Schumm et al. [230]. The values of the constants σ_k , σ_ε , $C_{\varepsilon 1}$, $C_{\varepsilon 2}$ listed in Table 6.1 and the damping functions f_1 and f_2 are listed in Table 6.2.

As low-Reynolds turbulence models are based on damping functions and the extra source terms D and E (listed in Tables 6.1 and 6.2), which enable the integration of the turbulence transport equations up to the wall, the use of turbulence wall functions is avoided. However, two equation-based turbulence models tend to over predict the turbulence length scale in flows at adverse pressure gradients [211] such as those found in detachment, reattachment and impinging regions. Accordingly, Schumm et al. [230] concluded in their study on turbulent flow over a backward-facing step that the turbulence near-wall length scale correction of Yap [297] needs to be added as an additional source term to the right hand side of the transport equation of ε (Equation 6.5). This correction has the form

$$S_\varepsilon = 0.83 \frac{\varepsilon^2}{k} \left(\frac{k^{1.5}}{\varepsilon l_e} - 1 \right) \left(\frac{k^{1.5}}{\varepsilon l_e} \right)^2, \quad (6.7)$$

where the turbulence length scale, l_e , is given by

$$l_e = C_\mu^{-0.75} \kappa y^+, \quad (6.8)$$

where κ is the von Karman constant, C_μ a model constant, both listed in Table 6.1, and y^+ is the dimensionless wall distance.

Furthermore, the unclosed terms that contain products of fluctuating values, namely the Reynolds stress term $\overline{\mathbf{u}'\mathbf{u}'}$, and the turbulence heat transfer tensor $\overline{\mathbf{u}'\theta'}$, need to be modeled. The Reynolds stress term is defined as

$$-\overline{(\mathbf{u}'\mathbf{u}')} = 2\nu_t \mathbf{S} - \frac{2}{3} k \mathbf{I}, \quad (6.9)$$

where \mathbf{S} is the Reynolds-averaged strain rate tensor and \mathbf{I} is the identity tensor. The eddy viscosity, ν_t , is computed by

$$\nu_t = C_\mu f_\mu \frac{k^2}{\varepsilon} \quad (6.10)$$

with f_μ listed in Table 6.2.

The turbulence heat flux tensor is modeled with the single gradient diffusion hypothesis given by the turbulence thermal diffusivity, α_t , and the mean temperature gradient as follows

$$\overline{\mathbf{u}'\theta'} = -\alpha_t \nabla \theta. \quad (6.11)$$

The SGDH expresses the turbulence thermal diffusivity as the ratio between the eddy viscosity, ν_t , and the turbulent Prandtl number, Pr_t , as

$$\alpha_t = \frac{\nu_t}{\text{Pr}_t}. \quad (6.12)$$

Typically, Pr_t is around 0.9 for wall-bounded flows. Here, the local correlation of Kays [135] is applied to have a good fit to DNS of both turbulent flow in ducts and the turbulent external boundary layer of fluids with $0.025 \leq \text{Pr} \leq 0.1$ [230]. Pr_t is defined as

$$\text{Pr}_t = 0.85 + \frac{0.7}{\text{Pe}_t}, \quad (6.13)$$

where Pe_t is the turbulent Peclet number, as function of the Prandtl number and the eddy viscosity divided by the viscosity [282], given by

$$\text{Pe}_t = \frac{\nu_t}{\nu} \text{Pr}. \quad (6.14)$$

Table 6.1: Low-Reynolds Launder-Sharma k - ε model constants and source terms.

C_μ	σ_k	σ_ε	$C_{\varepsilon 1}$	$C_{\varepsilon 2}$	D	E	κ
0.09	1	1.3	1.44	1.92	$2\nu \left(\frac{\partial \sqrt{k}}{\partial x_i} \right)^2$	$2\nu \nu_t \left(\frac{\partial^2 U}{\partial x_i \partial x_i} \right)^2$	0.41

Table 6.2: Damping coefficients with $\text{Re}_t = \frac{k^2}{\nu \varepsilon}$.

f_μ	f_1	f_2
$1-0.3e^{-\text{Re}_t^2}$	1	$e^{\frac{-3.4}{(1+\text{Re}_t/50)^2}}$

6.2.1 Flow characteristics by non-dimensional numbers

The flow characteristics of a fluid can be expressed by non-dimensional numbers. The most relevant ones for turbulent convective buoyant flow are given and explained here.

The ratio of the inertial forces to the viscous forces within the fluid is defined as the Reynolds number (Re)

$$\text{Re} = \frac{U_b h}{\nu}, \quad (6.15)$$

where U_b is the bulk velocity of the fluid and h is the characteristic dimension, which is taken to be the step height. At high Reynolds numbers, the flow is dominated by the inertial forces and is therefore considered turbulent, which is typically for $Re > 4000$ in channel flows.

The Richardson number (Ri) represents the importance of natural convection to the forced convection and is used to determine whether the flow is in the forced, mixed or natural convection regime. In this context, Ri is defined as

$$Ri = \frac{Gr}{Re^2} = \frac{g\beta h^2 q''}{\lambda U_b^2}, \quad (6.16)$$

with Gr the Grashof number defined as

$$Gr = \frac{g\beta h^4 q''}{\nu^2 \lambda}, \quad (6.17)$$

where g is the acceleration due to gravity, q'' the applied wall heat flux and λ is the thermal conductivity of the fluid. Typically, the flow is in the forced convection regime when $Ri < 0.1$, in the natural convection regime when $Ri > 10$, and in the mixed regime when $0.1 < Ri < 10$ [85].

The Stanton number (St) is given by the ratio of the heat transferred into the fluid to the thermal capacity of the fluid itself. Here the Stanton number, as function of the heat flux, is defined as

$$St = \frac{q''}{\rho U_b c_p \Delta\theta} = \frac{q'' \nu}{\rho U_b \lambda Pr \Delta\theta}, \quad (6.18)$$

where $\Delta\theta$ is the characteristic temperature difference and c_p the specific heat of the fluid.

The skin friction coefficient (c_f) is a function of the shearing stress exerted by the fluid on the wall surface over which it flows

$$c_f = \frac{\tau_w}{0.5\rho U_b^2}, \quad (6.19)$$

where τ_w is the wall shear stress. There is a relationship between skin friction and heat transfer for steady flows, which is known in the context of Reynolds analogy [86].

6.3 POD-Galerkin reduced order model for turbulent convective buoyant flows

The proper orthogonal decomposition method is used to create a reduced basis space that is spanned by a number of basis functions, or so-called modes, which capture the essential dynamics of the system [53, 110, 203, 217]. The RB method

assumes that the full order steady-state solutions, the so-called snapshots, of the discretized RANS equations for different parameter values, μ , can be expressed as a linear combination of orthonormal spatial modes multiplied by parameter-dependent coefficients. For velocity, shifted kinematic pressure and temperature the approximations are given by

$$\mathbf{U}(\mathbf{x}, \mu) \approx \mathbf{U}_r = \sum_{i=1}^{N_r} \varphi_i(\mathbf{x}) a_i(\mu), \quad (6.20)$$

$$P_{rgh}(\mathbf{x}, \mu) \approx P_{rgh_r} = \sum_{i=1}^{N_r} \chi_i(\mathbf{x}) a_i(\mu), \quad (6.21)$$

$$\theta(\mathbf{x}, \mu) \approx \theta_r = \sum_{i=1}^{N_r^\theta} \psi_i(\mathbf{x}) b_i(\mu), \quad (6.22)$$

where φ_i , χ_i and ψ_i are respectively the velocity, shifted kinematic pressure and temperature modes. It is assumed that velocity and pressure share the same coefficients $a_i(\mu)$, while $b_i(\mu)$ are the corresponding coefficients for temperature [161, 41]. This eases the development of the reduced order model since it is not needed to project a pressure Poisson equation onto the reduced basis space of pressure as done in Chapter 4. Therefore, only two sets of variables are necessary [267]. N_r is the number of velocity and shifted kinematic pressure modes and N_r^θ is the number of temperature modes.

The above assumptions can be extended to the eddy viscosity fields, ν_t , and the turbulence thermal diffusivity fields, α_t , in the following way

$$\nu_t(\mathbf{x}, \mu) \approx \nu_{t_r} = \sum_{i=1}^{N_r^{\nu_t}} \eta_i(\mathbf{x}) c_i(\mu), \quad (6.23)$$

$$\alpha_t(\mathbf{x}, \mu) \approx \alpha_{t_r} = \sum_{i=1}^{N_r^{\alpha_t}} \xi_i(\mathbf{x}) d_i(\mu), \quad (6.24)$$

with $N_r^{\nu_t}$ the number of eddy viscosity modes and $N_r^{\alpha_t}$ the number of turbulence thermal diffusivity modes, respectively. $\eta_i(\mathbf{x})$ and $\xi_i(\mathbf{x})$ are the eddy viscosity and the turbulence thermal diffusivity modes, respectively, and $c_i(\mu)$ and $d_i(\mu)$ the corresponding coefficients.

The optimal POD basis space for velocity, $E_{POD}^U = \text{span}(\varphi_1, \varphi_2, \dots, \varphi_{N_r})$, is constructed by minimizing the difference between the snapshots and their orthogonal projection onto the reduced basis [205] as follows

$$E_{POD}^U = \arg \min_{\varphi_1, \dots, \varphi_{N_r}} \frac{1}{N_s} \sum_{n=1}^{N_s} \left\| \mathbf{U}^n(\mathbf{x}) - \sum_{i=1}^{N_r} (\mathbf{U}^n(\mathbf{x}), \varphi_i(\mathbf{x}))_{L_2(\Omega)} \varphi_i(\mathbf{x}) \right\|_{L_2(\Omega)}^2, \quad (6.25)$$

where N_s is the number of collected snapshots and $N_s > N_r$. The same approach can be followed for the shifted kinematic pressure to determine the subspace $E_{POD}^{P_{rgh}} = \text{span}(\chi_1, \chi_2, \dots, \chi_{N_r})$. The L_2 norm is preferred for discrete numerical schemes [40, 241] with $(\cdot, \cdot)_{L_2(\Omega)}$ the L_2 inner product of the fields over the domain Ω . Furthermore, as the modes are orthonormal to each other, $(\varphi_i, \varphi_j)_{L_2(\Omega)} = \delta_{ij}$ holds, where δ is the Kronecker delta.

For temperature, the subspace $E_{POD}^\theta = \text{span}(\psi_1, \psi_2, \dots, \psi_{N_r^\theta})$ is obtained by solving a minimization problem similar to Equation 6.25. The same procedure also applies for the subspaces $E_{POD}^{\nu_t} = \text{span}(\eta_1, \eta_2, \dots, \eta_{N_r^{\nu_t}})$ and $E_{POD}^{\alpha_t} = \text{span}(\xi_1, \xi_2, \dots, \xi_{N_r^{\alpha_t}})$.

The velocity POD modes are obtained by solving Equation 6.25 using the following eigenvalue problem on the correlation matrix C^U of the velocity snapshots [235, 239, 241]

$$C^U Q^U = Q^U \lambda^U, \quad (6.26)$$

where $C_{ij}^U = (U^i, U^j)_{L_2(\Omega)}$ for $i, j = 1, \dots, N_s$ is the velocity correlation matrix, Q^U is a square matrix of eigenvectors and λ^U is a diagonal matrix containing the eigenvalues. The velocity POD modes are then constructed in the following way

$$\varphi_i(\mathbf{x}) = \frac{1}{N_s \sqrt{\lambda_i^U}} \sum_{n=1}^{N_s} U^n(\mathbf{x}) Q_{in}^U \quad \text{for } i = 1, \dots, N_r. \quad (6.27)$$

As the same basis for velocity and shifted kinematic pressure are used, no additional stabilization, as the supremizer or Pressure Poisson Equation approach [239, 241], is needed. For the same reason, the shifted kinematic pressure modes are constructed using the previously obtained matrix of eigenvectors Q^U

$$\chi_i(\mathbf{x}) = \frac{1}{N_s \sqrt{\lambda_i^U}} \sum_{n=1}^{N_s} P_{rgh}^n(\mathbf{x}) Q_{in}^U \quad \text{for } i = 1, \dots, N_r, \quad (6.28)$$

where N_s is the number of collected shifted kinematic pressure snapshots. The temperature, eddy viscosity and turbulence thermal diffusivity POD modes are determined by solving a similar eigenvalue problem as Equation 6.26. For more details on obtaining the POD modes, the reader is referred to [241, 267].

To obtain a reduced order model the POD is combined with the Galerkin projection. The momentum equations (Equation 6.2) with substitution according to Equations 6.20, 6.21, 6.22 and 6.23 are projected onto the POD basis space of velocity, $\varphi_i(\mathbf{x})$. The energy equation (Equation 6.3) with substitution according to Equations 6.20, 6.22 and 6.24 is projected onto the temperature spatial basis, $\psi_i(\mathbf{x})$. This results in the following reduced system of ordinary differential equations

$$\mathbf{a}^T C_r \mathbf{a} = -G_r \mathbf{a} + \nu(D_r + DT_r) \mathbf{a} + c^T (E_r + ET_r) \mathbf{a} - H_r \mathbf{b}, \quad (6.29)$$

$$\mathbf{a}^T \mathbf{Q}_r \mathbf{b} = \alpha \mathbf{Y} \mathbf{1}_r \mathbf{b} + \mathbf{d}^T \mathbf{Y} \mathbf{2}_r \mathbf{b}, \quad (6.30)$$

where

$$D_{r_{ij}} = (\boldsymbol{\varphi}_i, \nabla \cdot \nabla \boldsymbol{\varphi}_j)_{L_2(\Omega)}, \quad (6.31)$$

$$DT_{r_{ij}} = (\boldsymbol{\varphi}_i, \nabla \cdot (\nabla \boldsymbol{\varphi}_j^T))_{L_2(\Omega)}, \quad (6.32)$$

$$C_{r_{ijk}} = (\boldsymbol{\varphi}_i, \nabla \cdot (\boldsymbol{\varphi}_j \otimes \boldsymbol{\varphi}_k))_{L_2(\Omega)}, \quad (6.33)$$

$$E_{r_{ijk}} = (\boldsymbol{\varphi}_i, \nabla \cdot \eta_j \nabla \boldsymbol{\varphi}_k)_{L_2(\Omega)}, \quad (6.34)$$

$$ET_{r_{ijk}} = (\boldsymbol{\varphi}_i, \nabla \cdot \eta_j (\nabla \boldsymbol{\varphi}_k^T))_{L_2(\Omega)}, \quad (6.35)$$

$$G_{r_{ij}} = (\boldsymbol{\varphi}_i, \nabla \chi_j)_{L_2(\Omega)}, \quad (6.36)$$

$$H_{r_{ij}} = (\boldsymbol{\varphi}_i, (\mathbf{g} \cdot \mathbf{r}) \nabla (-\beta(\psi_j - \theta_{ref})))_{L_2(\Omega)}, \quad (6.37)$$

$$Y \mathbf{1}_{r_{ij}} = (\psi_i, \nabla \cdot \nabla \psi_j)_{L_2(\Omega)}, \quad (6.38)$$

$$Y \mathbf{2}_{r_{ijk}} = (\psi_i, \nabla \cdot (\xi_j \nabla \psi_k))_{L_2(\Omega)}, \quad (6.39)$$

$$Q_{r_{ijk}} = (\psi_i, \nabla \cdot (\boldsymbol{\varphi}_j \otimes \psi_k))_{L_2(\Omega)}. \quad (6.40)$$

The reduced matrices associated with the linear terms and the third order tensors associated with the nonlinear terms of the governing equation are stored before constructing the reduced order model during a, so called, offline stage. More details on the treatment of the nonlinear terms can be found in [241].

Note that the system of ODEs has $N_r + N_r^\theta + N_r^{\nu_t} + N_r^{\alpha_t}$ unknowns, but only $N_r + N_r^\theta$ equations to solve. Therefore, the coefficients $c_i(\mu^*)$ and $d_i(\mu^*)$ for any new value of an input parameter μ^* are computed with a non-intrusive interpolation procedure using radial basis functions (RBF), as described in [151]. Here the procedure is described for obtaining the eddy viscosity coefficients $c_i(\mu^*)$; the procedure can be applied in a similar fashion to the turbulence thermal diffusivity coefficients $d_i(\mu^*)$.

The RBF approach assumes that the coefficients $c_i(\mu^*)$ can be approximated for any new value of input parameter μ^* as a linear combination of $N_r^{\nu_t}$ chosen RBF kernels Θ_i [279] as follows

$$c_i(\mu^*) = \sum_{j=1}^{N_s^{\nu_t}} w_{ij} \Theta_i (\|\mu^* - \mu_j\|_{L_2}) \text{ for } i = 1, 2, \dots, N_r^{\nu_t}, \quad (6.41)$$

where $N_s^{\nu_t}$ is the number of eddy viscosity snapshots, μ_j are the sampling points corresponding to the eddy viscosity snapshots ν_{t_j} and w_{ij} are the weights that need to be determined. These weights are calculated by solving the following linear system

$$\sum_{j=1}^{N_s^{\nu_t}} w_{ij} \Theta_i (\|\mu_k - \mu_j\|_{L_2}) = c_{ik} \text{ for } i = 1, 2, \dots, N_r^{\nu_t} \text{ and } k = 1, 2, \dots, N_s^{\nu_t}, \quad (6.42)$$

where the output c_{ik} is a set of known eddy viscosity coefficients that are calculated by projecting the eddy viscosity snapshots ν_{t_k} obtained for the parameter inputs μ_k for $k = 1, 2, \dots, N_s^{\nu_t}$ onto the obtained spatial eddy viscosity modes η_i (Equation 6.23) in the following way

$$c_{ik} = (\nu_{t_k}, \eta_i)_{L_2(\Omega)} \text{ for } i = 1, 2, \dots, N_r^{\nu_t} \text{ and } k = 1, 2, \dots, N_s^{\nu_t}. \quad (6.43)$$

Various kernels, Θ_i , can be used for the RBFs. In this work, Gaussian kernels are considered, which have a local response, meaning that their best response is in the area near the center, in contrast to multi-quadratic RBFs which have a global response. The Gaussian kernels are defined as

$$\Theta_i (\|\mu - \mu_j\|_{L_2}) = e^{(-\gamma \|\mu - \mu_j\|_{L_2}^2)} \text{ for } i = 1, 2, \dots, N_r^{\nu_t} \text{ and } j = 1, 2, \dots, N_s^{\nu_t}, \quad (6.44)$$

where γ is the parameter that determines the radius of the kernel. The RBF decreases monotonically away from the center.

Once the coefficients c_i and d_i for new input parameters μ^* are obtained, the set of ODEs, Equation 6.29 and 6.30, can be solved to obtain the coefficients $\mathbf{a}(\mu^*)$ and $\mathbf{b}(\mu^*)$. For more details about using RBF in this type of reduced order modeling setting the reader is referred to [111].

The advantage of determining the coefficients with RBFs is that it is not needed to project the turbulence modeling equations (Equations 6.4 and 6.5) onto the reduced basis spanned by the eddy viscosity modes. These equations are often, even when using open source codes like OpenFOAM [130], challenging to access. Also the turbulence thermal diffusivity coefficients are determined with RBFs as α_t is not directly proportional to ν_t due to the use of the local correlation by Kays (Equation 6.13) for the calculation of Pr_t in the SGDH.

Good initial guesses for the reduced system of ODEs (Equations 6.29 and 6.30) are obtained by projecting, respectively, the velocity and temperature snapshots for a certain parameter value μ that is close to the value of a new input parameter μ^* onto the POD basis spaces as follows

$$a_i(\mu^*) = (\varphi_i(\mathbf{x}), \mathbf{U}(\mathbf{x}, \mu))_{L_2(\Omega)}, \quad (6.45)$$

$$b_i(\mu^*) = (\psi_i(\mathbf{x}), \theta(\mathbf{x}, \mu))_{L_2(\Omega)}. \quad (6.46)$$

6.3.1 Imposing boundary conditions with the penalty method

The POD basis functions are a linear combination of the snapshots and so are their values at the boundaries [149]. Therefore, when using a POD-based reduced order modeling technique, the non-homogeneous boundary conditions are, in general, not satisfied by the ROM [29]. Furthermore, the BCs are not explicitly present in the reduced system and therefore they cannot be controlled directly [161].

In this chapter, a reduced order model is developed for turbulent convective buoyant flows that are characterized by the Richardson number. The Richardson number depends on the wall heat flux (Equation 6.16), which is a Neumann type of boundary condition. Recently, Vergari et al. [267] extended the penalty method for imposing boundary conditions at the reduced order level, which is described in Section 4.3.2, for Neumann boundary conditions.

The penalty method enforces explicitly the BCs in the ROM with a penalty factor [93]. A penalty method was originally proposed by Lions and Magènes [158] in the context of finite element methods. They introduced a penalty parameter to weakly impose the boundary conditions. In the POD-Galerkin reduced order modeling setting, the penalty method has been first introduced by Sirisup and Karniadakis [234] for the enforcement of boundary conditions.

The value of the penalty factor τ is generally chosen arbitrary [29]. Nevertheless, if the penalty factor tends to infinity a strong imposition of the boundary conditions would be approached and the ROM becomes ill-conditioned [161]. On the other hand, small values of the factor result in a weak imposition [234] and the method becomes numerically unstable [70]. Moreover, the penalty factors τ should be larger than 0 in order to have an asymptotically stable solution [161]. Therefore, the penalty factor needs to be chosen above a threshold value for which the method is stable and converges [70]. For these reasons, the (suitable range for the) penalty factor is often determined via a sensitivity study [93, 134, 161].

The value of the velocity Dirichlet BC, U_{BC} , is imposed the reduced momentum equation as follows

$$\begin{aligned} \mathbf{a}^T \mathbf{C}_r \mathbf{a} + \mathbf{G}_r \mathbf{a} - \nu (\mathbf{D}_r + \mathbf{DT}_r) \mathbf{a} - \mathbf{c}^T (\mathbf{E}_r + \mathbf{ET}_r) \mathbf{a} + \mathbf{H}_r \mathbf{b} \\ + \tau_U (\mathbf{O1}_r \mathbf{a} - U_{BC} \mathbf{P1}_r) = \mathbf{0}, \end{aligned} \quad (6.47)$$

where the new terms related to boundary Γ_1 are given by

$$\mathbf{O1}_{r_{ij}} = (\varphi_i, \varphi_j)_{L_2(\Gamma_1)}, \quad (6.48)$$

$$\mathbf{P1}_{ri} = (\varphi_i, \mathbf{\Lambda})_{L_2(\Gamma_1)}, \quad (6.49)$$

and $\mathbf{\Lambda}$ is a unit vector field.

In this work, the Neumann boundary condition is only applied for temperature on a boundary Γ_2 . The BC is related to the heat flux on the boundary, q''_{BC} , in the following way

$$\mathbf{n} \cdot \nabla \theta|_{\Gamma_2} = -\frac{q''_{BC}}{\lambda}. \quad (6.50)$$

The Neumann temperature BC together with a Dirichlet temperature BC, θ_{BC} , are enforced in the energy equation at the reduced order level (Equation 6.30) on,

respectively, boundary Γ_2 and Γ_3 as follows

$$\begin{aligned} \mathbf{a}^T \mathbf{Q}_r \mathbf{b} - \alpha \mathbf{Y} \mathbf{1}_r \mathbf{b} - \mathbf{d}^T \mathbf{Y} \mathbf{2}_r \mathbf{b} + \tau_{\nabla\theta} \left(\mathbf{O} \mathbf{2}_r \mathbf{b} + \frac{q''_{BC}}{\lambda} \mathbf{P} \mathbf{2}_r \right) \\ + \tau_{\theta} (\mathbf{O} \mathbf{3}_r \mathbf{b} - \theta_{BC} \mathbf{P} \mathbf{3}_r) = \mathbf{0}, \end{aligned} \quad (6.51)$$

where the new terms related to boundaries Γ_2 and Γ_3 are

$$\mathbf{O} \mathbf{2}_{r_i j} = (\psi_i, \mathbf{n} \cdot \nabla \psi_j)_{L_2(\Gamma_2)}, \quad (6.52)$$

$$\mathbf{P} \mathbf{2}_{r_i} = (\psi_i, \Lambda)_{L_2(\Gamma_2)}, \quad (6.53)$$

$$\mathbf{O} \mathbf{3}_{r_i j} = (\psi_i, \psi_j)_{L_2(\Gamma_3)}, \quad (6.54)$$

$$\mathbf{P} \mathbf{3}_{r_i} = (\psi_i, \Lambda)_{L_2(\Gamma_3)}, \quad (6.55)$$

where Λ is a unit scalar field.

6.4 Numerical set-up for sodium flow over a backward facing step

In this section the numerical set-up for a backward-facing step is described. Figure 6.1 depicts a sketch of the geometry. The height of the step is h and the channel height is H , which equals $2h$. Consequently, the Expansion Ratio (ER) between inlet and outlet is $ER = H/(H-h) = 2$. The inlet is located $L_u = 4h$ upstream of the step. A constant heat flux is applied on the bottom wall directly downstream of the step over a length $L_h = 20h$ and is referred to as "the heater". This wall is followed by an adiabatic wall of length $L_a = 20h$.

A mesh is constructed in the three-dimensional domain, but can be considered to be two-dimensional as it contains only one layer of cells in the z -direction. The distribution of the cells are described in Figure 6.2 and in Table 6.3. These distributions are taken from the finest mesh of the grid refinement study performed by [230]. Similar to their work, the cells are clustered towards the walls and in stream-wise direction towards the end of the heater where steep changes in the velocity gradients are expected. The cell expansion ratio is 5 in the direction perpendicular to the solid walls and 10 in the direction perpendicular to the inlet and outlet boundaries of the domain. The mesh contains a total number of 585450 hexahedral cells.

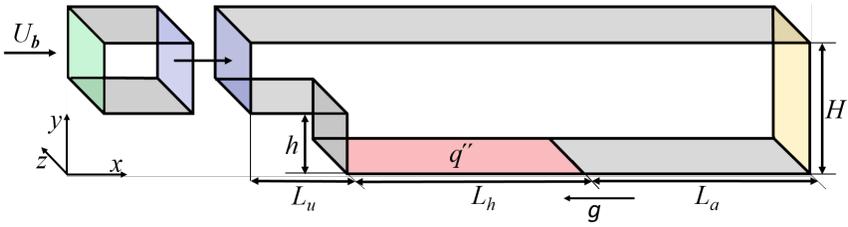


Figure 6.1: A sketch of the geometry of the backward-facing step and the precursor domain.

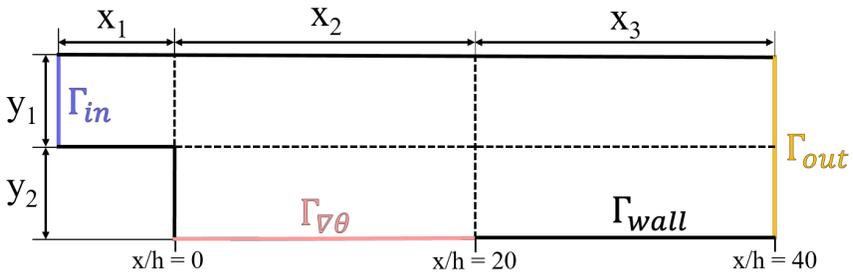


Figure 6.2: A 2D sketch of the geometry of the backward-facing step divided into several zones and including boundaries.

Table 6.3: The number of cells, N , along the horizontal (x_1 , x_2 and x_3) and vertical sides (y_1 , y_2) of each zone depicted in Figure 6.2.

	N_{y_1}	N_{y_2}	N_{x_1}	N_{x_2}	N_{x_3}
Number of cells	225	225	126	900	338

The characteristic dimension of the domain is the step height $h = 0.05$ m. A hydrodynamic fully developed channel flow profile is applied at the inlet boundary Γ_{in} . This inlet velocity profile is generated via a separate simulation of an isothermal channel flow of height h and length $10h$ with the inlet bulk velocity $U_b = 0.1192$ m/s. The flow, characterized by the Reynolds number $Re = 10^4$ (Equation 6.15), is considered to be fully turbulent. The hyperbolic stream-wise velocity profile at the outlet of the channel is set as the inlet velocity profile of the backward-facing step as depicted in Figure 6.1. At the outlet, Γ_{out} , a homogeneous Neumann boundary condition is set for all variables except for pressure. Only the relative pressure is calculated and therefore it is set to $0 \text{ m}^2/\text{s}^2$ at the outlet. At all solid walls a no-slip condition is applied and the turbulence quantities, k and ε , are set to zero. All walls except for the heated one, $\Gamma_{\nabla\theta}$, are adiabatic.

The fluid properties are taken for liquid sodium at a constant inlet temperature $\theta_{in} = 423.15 \text{ K} = 150^\circ\text{C}$ with the kinematic viscosity $\nu = 5.96 \cdot 10^{-7} \text{ m}^2/\text{s}$ and thermal diffusivity $\alpha = 6.8 \cdot 10^{-5} \text{ m}^2/\text{s}$, meaning that the Prandtl number, Pr, is equal to 0.0088.

RANS simulations are performed for Richardson numbers (Equation 6.16) in a range of $\text{Ri} = [0.0, 0.5]$ with steps of 0.05, covering partly the forced- and mixed convection regime. To calculate these Richardson numbers, the thermal expansion coefficient, β is considered to be equal to $2.5644 \cdot 10^{-4} \text{ K}^{-1}$ [179]. Similar to the numerical experiments done by [230], the backward-facing step is placed vertically by having the gravitational acceleration in the downward direction with $\mathbf{g} = (-9.81, 0, 0) \text{ m/s}^2$.

The steady-state RANS equations (Equations 6.1-6.5) are discretized by the Finite Volume method with the open source C++ library OpenFOAM 6 [130]. The simulations are run in parallel on 8 Intel® Xeon® E5-2680 v3 @ 2.50GHz cores. The SIMPLE algorithm for the pressure-velocity coupling is used [77] and blended schemes with an order of accuracy between one and two have been used for the spatial discretization. A solution is assumed to be converged when the scaled residuals of all variables are below 10^{-5} .

The y^+ values at the heater for $\text{Ri} = 0.0, 0.2$ and 0.4 are compared with the values obtained by Schumm et al. [230] for the same distribution of the cells in Table 6.4 and similar values are observed.

Table 6.4: y^+ values at the heater compared with the results of [230].

Ri	y_{min}^+	y_{min}^+ [230]	y_{max}^+	y_{max}^+ [230]	y_{avg}^+	y_{avg}^+ [230]
0.0	$5.9e^{-5}$	$2.5e^{-3}$	$3.1e^{-1}$	$5.6e^{-1}$	$1.7e^{-1}$	$3.3e^{-1}$
0.2	$4.2e^{-3}$	$1.2e^{-3}$	$4.4e^{-1}$	$0.9e^{-1}$	$3.3e^{-1}$	$7.1e^{-2}$
0.4	$7.5e^{-3}$	$1.9e^{-3}$	$6.2e^{-1}$	$1.1e^{-1}$	$4.7e^{-1}$	$9.1e^{-2}$

The calculation of the POD modes, the Galerkin projection of the RANS solutions on the reduced subspace and the ROM simulations are carried out with ITHACA-FV [240] on a single Intel® Xeon® core. ITHACA-FV is a C++ library based on the Finite Volume solver OpenFOAM [130]. For more details on the ITHACA-FV code, the reader is referred to [239, 240, 241].

The ROM is tested for four Richardson numbers $\text{Ri} = 0.12, 0.24, 0.36$ and 0.48 that are all within the aforementioned range. The ROM solutions are compared with the RANS solutions for these Richardson numbers, which are not used in the creation of the ROM basis, to check the consistency of the method. The relative L_2 error between the FOM and ROM velocity and temperature fields and the basis projection error are determined at each parameter value, μ , according to Section 2.5.5.2.

6.5 Results and analysis

Firstly, 10 steady-state RANS simulations are performed for the vertical backward-facing step case for Richardson numbers in the range $[0.05, 0.5]$ with steps of 0.05. The associated heat flux (Equation 6.50) in the range $[112.5, 1125]$ W/m^2 is considered to be the corresponding varying physical parameter. The converged solutions are taken as snapshots, which are then used to create the POD basis functions. Figure 6.3 shows the velocity magnitude, shifted kinematic pressure, and temperature fields for $\text{Ri} = 0.2$ (left) and $\text{Ri} = 0.4$ (right).

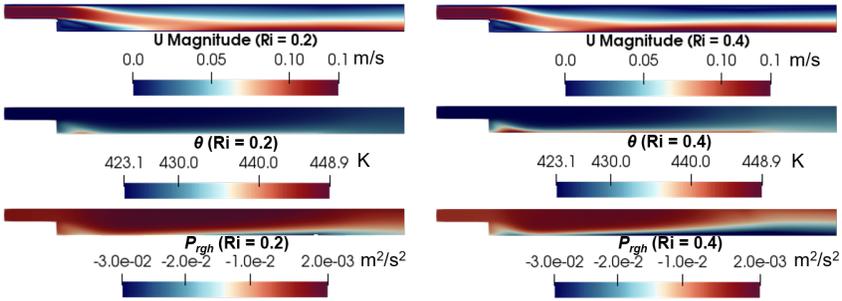


Figure 6.3: Velocity magnitude (top), shifted kinematic pressure (middle) and temperature fields obtained with the RANS simulations for $\text{Ri} = 0.2$ (left) and $\text{Ri} = 0.4$ (right), respectively.

The same figure shows that the effect of buoyancy on the flow field and heat transfer is larger for higher Richardson numbers. For instance, it can be clearly seen that increasing the heat flux results in a decrease of the recirculation zone directly downstream of the step. This is also reported in [178] and [230].

Figure 6.4 shows the projection errors for velocity and temperature up to the first eight modes. The figure shows that for a certain parameter value the error monotonically decreases when the number of modes is increased. These errors act as a lower error bound for the reduced order model. In practice, the prediction error ϵ_{L_2} for the fields obtained by solving the ROM is larger than the projection error $\hat{\epsilon}_{L_2}$.

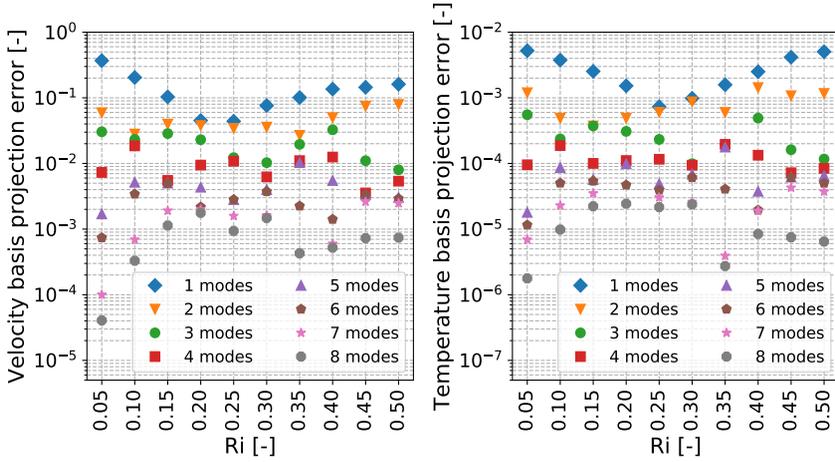


Figure 6.4: Relative basis projection error of all snapshots for different number of modes: (left) velocity relative error; (right) temperature relative error.

In order to retain 99.99% of the energy contained in the snapshots for all physical variables 4 velocity modes, 4 shifted kinematic pressure modes, 1 temperature mode, 6 eddy viscosity modes and 6 turbulence thermal diffusivity modes are needed. Adding more modes can improve the accuracy of the ROM, but this has a detrimental effect on the computational time. Therefore, there is a trade-off between the two options. Based on Figure 6.4, 5 velocity modes, 5 shifted kinematic pressure modes and 5 or 8 temperature modes are used for the construction of the ROM. The projection error is about $\mathcal{O}(10^{-2})$ and $\mathcal{O}(10^{-4})$ for velocity and temperature, respectively. Furthermore, 8 eddy viscosity and turbulent thermal diffusivity modes are used to accurately determine the corresponding coefficients with the RBF approach.

6.5.1 Determining the penalty factors

The penalty factors are determined via a sensitivity study by performing multiple ROM simulations for different values of the factors. To show the effect of the penalty factors, the relative prediction errors for velocity and temperature with $N_r^\theta = 5$ are shown in Figure 6.5 for the following two cases

- A) $\tau_U = 1, \tau_{\nabla\theta} = 1, \tau_\theta = 1,$
- B) $\tau_U = 10^6, \tau_{\nabla\theta} = 10^6, \tau_\theta = 10^6.$

This figure shows that the relative prediction error for both velocity and temperature improves when the penalty factors are larger. The results for other combina-

tions of factors are not shown here as they do not lead to an overall improvement of the prediction error compared to case B.

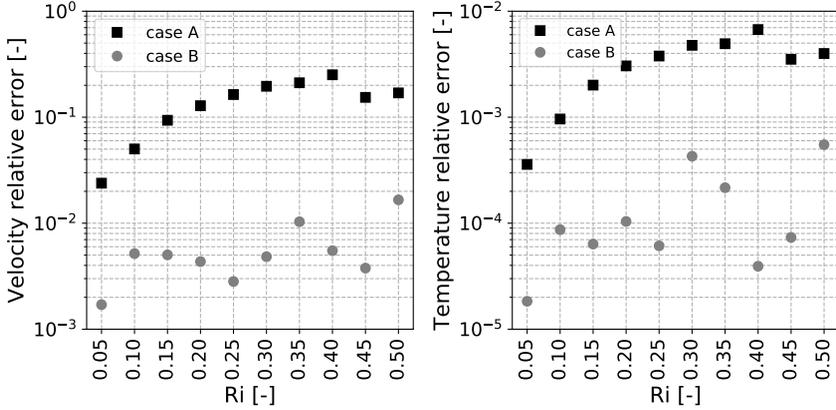


Figure 6.5: Relative prediction error for two sets of penalty factors and different Richardson numbers and $N_r^\theta = 5$. Case A: $\tau_U = 1, \tau_{\nabla\theta} = 1, \tau_\theta = 1$. Case B: $\tau_U = 10^6, \tau_{\nabla\theta} = 10^6, \tau_\theta = 10^6$. (Left) Velocity relative error; (right) temperature relative error.

6.5.2 Comparison of the ROM and RANS solutions

Reduced order simulations are performed for the same parameter values μ for which the snapshots are collected. The results of the ROM simulations are compared with the results of the RANS simulations for the cases of $Ri = 0.2$ and 0.4 . The stream-wise velocity, the wall normal velocity and the non-dimensional temperature profiles are shown in Figure 6.6, 6.7 and 6.8, respectively. Location $x/h = 0$ is at the step as indicated in Figure 6.2. Only the results with $N_r^\theta = 5$ are shown. Good agreement with the RANS data is found for these cases. As also observed by Schumm et al. [230] the recirculation zone is reduced in its stream-wise extent (Figure 6.6) with increasing buoyancy. Also, the velocity profiles have their peak forming above the heater.

Furthermore, the wall normal velocity component, shown in Figure 6.7, is in agreement with the results found in [230]. As a positive wall normal velocity component transports momentum from the heater towards the upper wall, the temperature at the heater is lower for $Ri = 0.4$ compared to $Ri = 0.2$, as can be seen in Figure 6.8. Even though the velocity profiles are in good agreement with literature, the flow near the upper wall starts heating up further downstream at around $x/h = 27$ compared to the cases studied by Schumm et al. [230] where this phenomenon was already present at around $x/h = 15$. This means that less mixing takes place in the thermal field of this study compared to their study, which can be caused by

a lower shear stress.

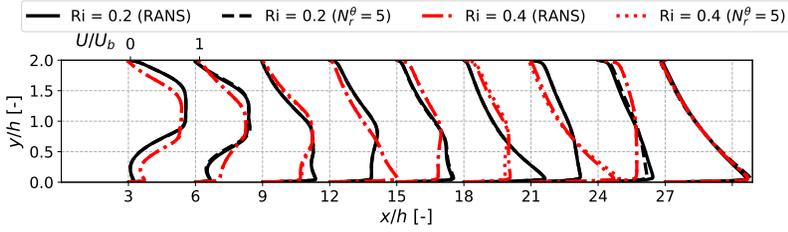


Figure 6.6: Profiles of the normalized stream-wise velocity component at several locations downstream of the step for $Ri = 0.2$ and $Ri = 0.4$, respectively, obtained by performing RANS and ROM simulations. The legend depicts the number of temperature modes, N_r^θ , used for the ROM.

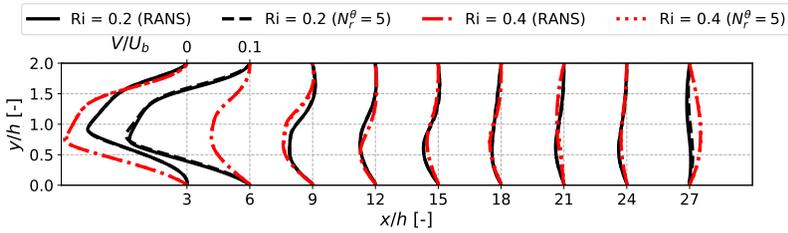


Figure 6.7: Profiles of the normalized wall normal velocity component at several locations downstream of the step for $Ri = 0.2$ and $Ri = 0.4$, respectively, obtained by performing RANS and ROM simulations. The legend depicts the number of temperature modes, N_r^θ , used for the ROM.

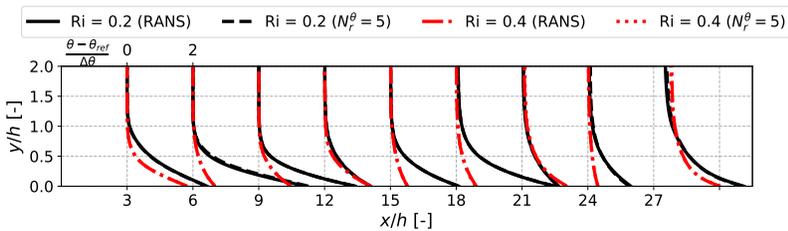


Figure 6.8: Normalized temperature profiles at several locations downstream of the step for $Ri = 0.2$ and $Ri = 0.4$, respectively, obtained by performing RANS and ROM simulations. The legend depicts the number of temperature modes, N_r^θ , used for the ROM.

The local Stanton number profiles, depending on the applied heat flux, along the heater are shown in Figure 6.9 for $Ri = 0.1, 0.2, 0.3, 0.4$ and 0.5 . Figure 6.10 shows the skin friction distribution, depending on the wall shear stress, at the heater and further downstream up to $x/h = 30$ for the same Richardson numbers on the right. Not all cases are shown for the sake of clarity. The distributions obtained by the RANS simulations are in good agreement with the literature. Furthermore, the results, and especially those of the skin friction distribution downstream of the heater, show that buoyancy has a large influence on the flow and heat transfer. Moreover, heat conduction through the boundary layer has more dominant effect with respect to convection as the Prandtl number is below unity. Even though the behavior is nonlinear, the reduced order model is capable of reproducing the RANS results with a good accuracy.

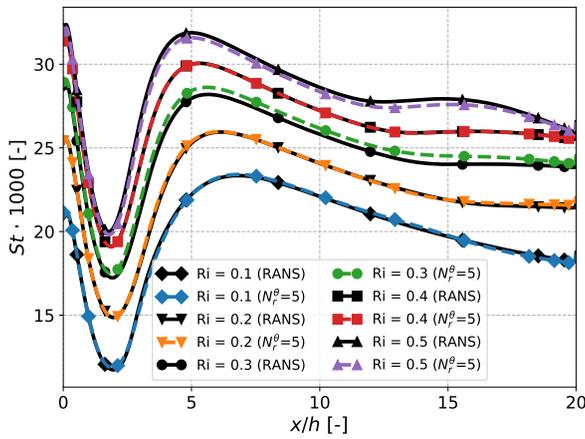


Figure 6.9: Local Stanton number at the heater determined by the RANS and ROM simulations for several Richardson numbers.

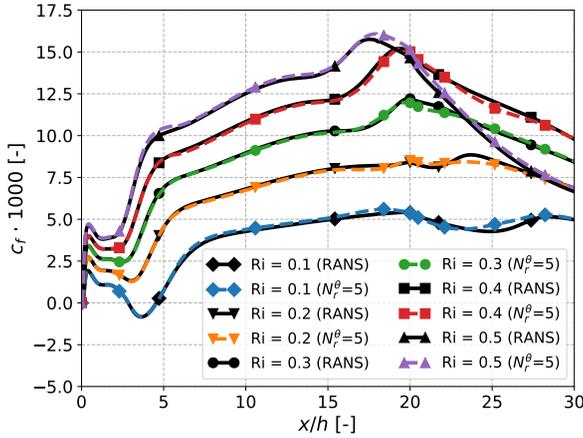


Figure 6.10: Skin friction distribution downstream of the backward-facing step determined by the RANS and ROM simulations for several Richardson numbers.

6.5.3 Reduced order simulations for new parameter values

Besides the parameter for which snapshots are collected, the ROM is tested on four additional Richardson numbers, namely $Ri = 0.12, 0.24, 0.36$ and 0.48 . Figure 6.11 shows the prediction error for these cases with five and eight temperature modes. The figure shows that especially for $Ri = 0.48$ both the velocity and temperature relative error is reduced when the number of temperature modes is increased from five to eight. However, the opposite is true for $Ri = 0.24$. In that case, increasing the number of temperature modes has a detrimental effect on the prediction error of temperature. Therefore, five temperature modes are used further for $Ri = 0.12$ and 0.24 , while eight temperature modes are used for $Ri = 0.36$ and 0.48 .

Figures 6.12 and 6.13 show the local Stanton number and the skin friction distribution downstream of the step respectively. The ROM results for $Ri = 0.24$ and 0.48 are overlapping with the distributions obtained by the RANS simulations. For $Ri = 0.36$ the ROM solution is accurate looking at the local Stanton number. However, the solutions for the skin friction deviates from the RANS solutions downstream of the heater. For $Ri = 0.12$ the ROM over-predicts the local Stanton number and under-predicts the skin friction. An attempt is made to reduce the error for this case by increasing/decreasing the number of modes of all variables and increasing/decreasing the penalty factors. However, all solutions are deviating from the ROM solution by a few percent, while for all other parameter values the deviation is less than 0.1% in case of the local Stanton number.

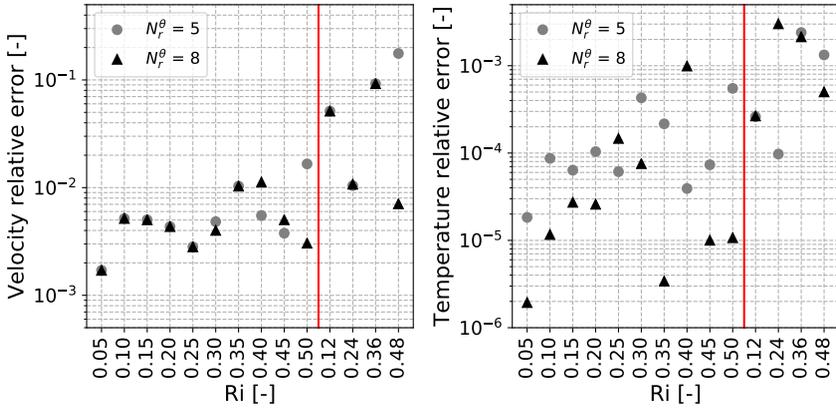


Figure 6.11: Relative prediction error for all Richardson numbers with $N_r^\theta = 5$ and $N_r^\theta = 8$ temperature modes, respectively, used for the construction of the ROM: (left) velocity relative error; (right) temperature relative error.

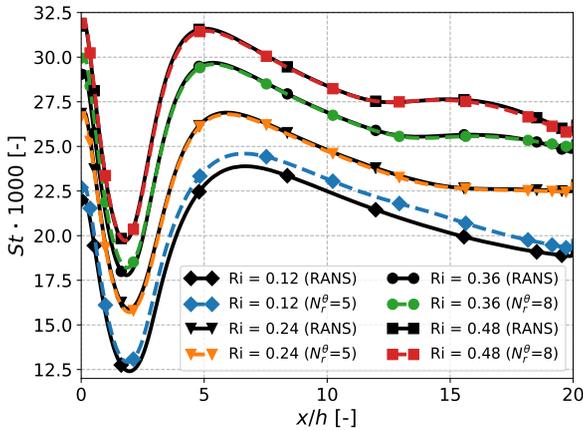


Figure 6.12: Local Stanton number at the heater determined by the ROM compared with those determined by RANS simulations for several Richardson numbers.

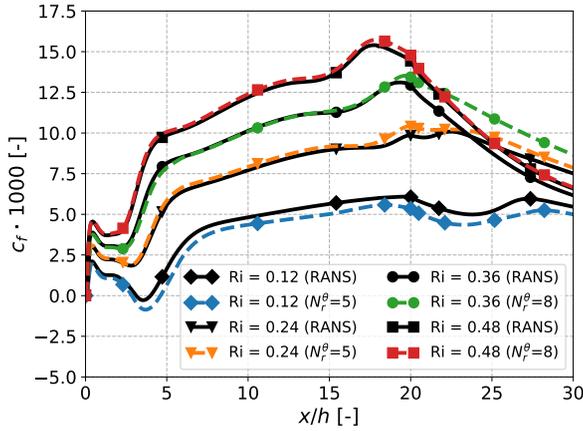


Figure 6.13: Skin friction distribution downstream of the backward-facing step determined by the ROM compared with those determined by RANS simulations for several Richardson numbers.

To see whether this affects the velocity and temperature distribution, the profiles downstream of the step are plotted in Figures 6.14, 6.15 and 6.16. Only the profiles of the wall normal velocity component at $x/h = 3$ show a small deviation between the RANS and ROM solution for $Ri = 0.12$. For all other profiles, the ROM solutions are fully overlapping with the RANS solutions.

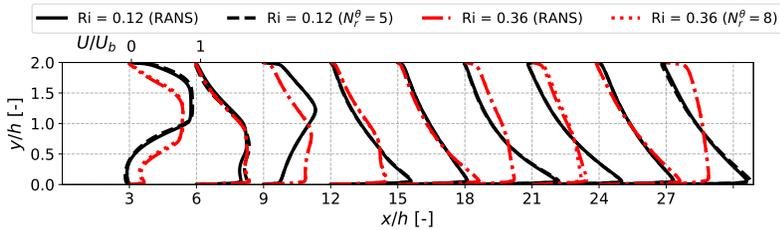


Figure 6.14: Profiles of the normalized stream-wise velocity component at several locations downstream of the step for $Ri = 0.12$ and $Ri = 0.36$, respectively, obtained by performing RANS and ROM simulations. The legend depicts the number of temperature modes, N_r^θ , used for the ROM.

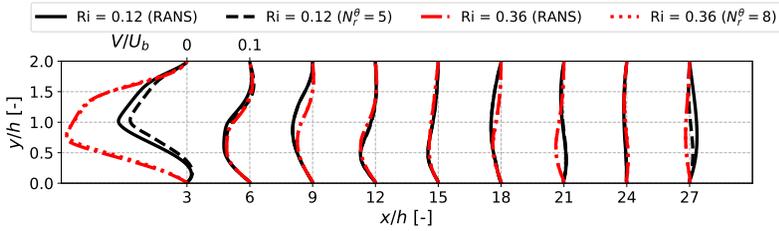


Figure 6.15: Profiles of the normalized wall normal velocity component at several locations downstream of the step for $Ri = 0.12$ and $Ri = 0.36$, respectively, obtained by performing RANS and ROM simulations. The legend depicts the number of temperature modes, N_r^θ , used for the ROM.

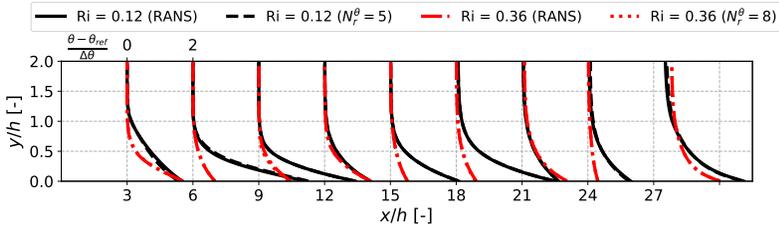


Figure 6.16: Normalized temperature profiles at several locations downstream of the step for $Ri = 0.12$ and $Ri = 0.36$, respectively, obtained by performing RANS and ROM simulations. The legend depicts the number of temperature modes, N_r^θ , used for the ROM.

The performance of the radial basis function interpolation is checked by comparing the ratio of the eddy viscosity to kinematic viscosity at several locations downstream of the step determined by the RANS and ROM simulations for $Ri = 0.12$ and $Ri = 0.36$, as shown in Figure 6.17. Also for these fields, the ROM solutions are fully overlapping with the RANS solutions.

Finally, one RANS simulation takes on average 17 hours to converge on 8 Intel® Xeon® cores to reach a steady state solution. On the other hand, one ROM simulation takes about 1.5 seconds to converge on a single core. Therefore, the speed-up is about the order $\mathcal{O}(10^5)$. The computational cost of the construction of the ROM is not taken into account in the calculation of the speed-up offered by the ROM itself. The whole construction of the ROM, including the collection of snapshots, calculating the POD modes and the reduced matrices, can be done in an offline phase on a high performance computing environment. For this case, the entire offline phase, which is dominated by the time it takes for the RANS simulations to converge, can be done in about 17 hours using parallel calculations on 10 (the number of snapshots) \times 8 cores.

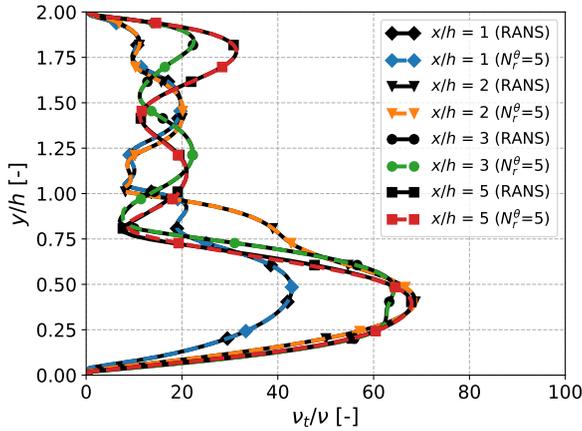
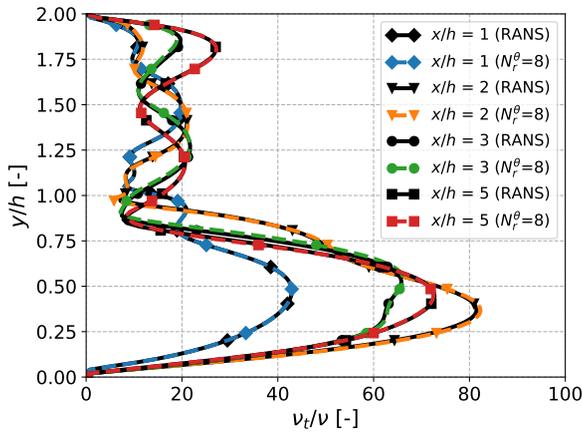
(a) $Ri = 0.12$ (b) $Ri = 0.36$.

Figure 6.17: Comparison of the ratio of eddy viscosity to kinematic viscosity at several locations downstream of the step determined by the RANS and ROM simulations for different Richardson numbers. The legend depicts the number of temperature modes, N_T^θ , used for the ROM.

6.6 Discussion

The results for certain Richardson numbers show that increasing the number of modes for the construction of the reduced basis space does not necessarily result in a more accurate reduced order model. For instance, the relative prediction error of temperature is an order higher if eight instead of five temperature modes are used for $Ri = 0.24$, which can be clearly seen in Figure 6.11. This indicates that the

ROM is not fully consistent with the high-fidelity model.

The discrepancy between the RANS and ROM simulations can have different causes. First of all the SIMPLE algorithm is implemented in the high fidelity model, but not in the reduced order model.

Furthermore, the turbulence transport equations for k and ε (Equations 6.4 and 6.5) are not projected on the reduced basis. Instead, the eddy viscosity and turbulence thermal diffusivity fields are approximated with an RBF interpolation approach. The advantage is that the reduced order model is independent of the turbulence model used in the RANS simulations [111]. Also if the effect of buoyancy is modeled in the turbulence transport equations (Equations 6.4 and 6.5), the reduced system of equations (Equations 6.29 and 6.30) does not have to be adjusted. Another option is to project the equation for the turbulence diffusivity field, Equation 6.12, in combination with Equation 6.13, onto the reduced basis. Then the ν_t and α_t fields can share the same coefficients as α_t is depending on ν_t (Equation 6.12). However, this is not tested in this study.

Moreover, some modes contain more features of the flow solution for higher Richardson numbers than others. This can be seen in Figure 6.5 as the basis projection error is not the same for all Richardson numbers. Also, the projection error stagnates more or less at 7 modes, as can be seen in the same figure. This means that constructing a reduced basis with even more modes can have a detrimental effect on the ROM solution as these higher modes contain only a limited amount of physical information.

The ROM over-predicts the local Stanton number at the heater and under-predicts the skin friction for $Ri = 0.12$ as shown in Figures 6.12 and 6.13, respectively. Neither increasing/decreasing the number of modes of all variables nor increasing/decreasing the penalty factors resulted in a lower error. In this work, the amount of RANS data is limited as only 10 snapshots are used for the construction of the reduced basis. Therefore, a possible solution is to construct a ROM with more snapshots collected in a narrow range of Richardson numbers around the parameter to be tested [109]. However, this approach is time consuming as multiple local reduced order models need to be constructed.

In this study, the penalty factors are found by numerical experimentation. This can, however, be a time consuming process if the results are not satisfactory after a few tries. It remains a question how to properly select the penalty factors for the enforcement of boundary conditions in the reduced order models. This highlights one of the main drawbacks of the penalty methods, namely that the factors cannot be determined a priori [93]. Instead of performing a sensitivity study to determine the penalty factors, the iterative penalty method that is presented in Chapter 4 can be implemented to determine the factors automatically in the ROM setting [248].

The parametric ROM is constructed in this work to study solutions for different Richardson numbers, for which the associated heat flux is considered to be the

corresponding varying physical parameter. The ROM is already set-up in such a way that it can also be used for other parameters. For instance, the constant viscosity is taken outside the reduced matrix for the diffusive term and the inlet velocity that appears as variable in the penalty term of the reduced momentum equations. Nevertheless, the ROM is not trained for these parameters even when staying within the range of Richardson numbers for which snapshots are collected. Therefore, a new ROM needs to be constructed if new or additional snapshots are needed as the POD basis functions are assumed to be based on a linear combination of the snapshots. Furthermore, geometric parametrization, like changing the height of the step, is not possible with this ROM.

The ROM can be extended to unsteady RANS simulation by incorporating a time integration method at the reduced order level. However, standard POD-Galerkin ROMs tend to exhibit instabilities when an iterative algorithm for solving the nonlinear implicit equations is implemented at the reduced order level [5, 18, 25, 78]. An iterative algorithm is required due to the presence of the coupling between pressure and velocity [241]. A pressure stabilization method might be required instead of that velocity and pressure share the same coefficients for the development of the reduced order model. Moreover, the snapshots do not only need to be collected in parameter space, but also at several time instances [146]. In addition, the time-dependency needs to be incorporated into the RBF interpolation method such that the approach is suitable for unsteady cases [112].

Finally, it is known from the literature that simulating unsteady turbulent convective (buoyant) flows is challenging and RANS simulations are inaccurate for large classes of flows [68]. The large eddy simulation method, which is giving access to the fluctuating quantities, is often required [12, 101, 232]. One of the challenges of developing a LES-ROM, other than applying filtering, is the derivation of the ROM closure model to improve the accuracy and instability of the standard POD-Galerkin ROM [185, 280, 293]. More research is needed to extend the current ROM for transient simulations as well as for large eddy simulation.

6.7 Conclusion

A Finite-Volume based POD-Galerkin reduced order modeling strategy for steady-state Reynolds-averaged Navier-Stokes simulations is developed for low-Prandtl number fluid flow. Simulations are performed for sodium flow over a vertical backward-facing step with a heater placed on the wall directly downstream of the step. The results for different Richardson numbers show that buoyancy has large influence on the flow and heat transfer. Even though the behavior is nonlinear, the reduced order model is capable of reproducing the RANS results with good accuracy. The prediction error between the RANS and ROM velocity fields is of the order $\mathcal{O}(10^{-2})$ and below the order $\mathcal{O}(10^{-1})$ for new parameter values inside the

range of Richardson numbers. For temperature, the relative error is about or less than the order $\mathcal{O}(10^{-3})$ for all parameter values. Also, the local Stanton number and skin friction distribution at the heater are qualitatively well captured. Furthermore, the eddy viscosity and turbulence thermal diffusivity coefficients, needed for the turbulence closure modeling at the reduced order level, are approximated with the radial basis function interpolation method. The advantage of this method is that the reduced order model is independent of the turbulence model used in the RANS simulations. Finally, the reduced order simulations performed on a single Intel[®] Xeon[®] core are about 10^5 times faster than the RANS simulations performed on 8 cores.

For further work, the aim is to extend the reduced order model of turbulent convective buoyant flow of low-Prandtl number fluid for the parametrized unsteady RANS equations. An interesting follow-up study would be to develop a ROM for unsteady flow and heat transfer of sodium in an outlet plenum [171]. Furthermore, neural networks [83], instead of using radial basis functions as an interpolation method, could potentially be used to approximate the eddy viscosity and thermal diffusion coefficients conducted in this work [112].

7

A ‘discretize-then-project’ approach for the incompressible Navier-Stokes equations on collocated grids

This chapter introduces the ‘discretize-then-project’ for incompressible fluids on collocated grids. We repeat here part of Chapter 2 on the finite volume discretization method that is essential for the understanding of the ‘discretize-then-project’ approach and due to some changes in notations. The results of this chapter are submitted as [246].

7.1 Introduction

Despite the potential and the increasing popularity of FV-based POD-Galerkin reduced order models for all sorts of applications, they tend to have issues with accuracy and can exhibit numerical instabilities [127, 133, 241]. Challenges regarding velocity-pressure coupling and satisfying the boundary conditions at ROM level make it difficult to generalize the ROM methods such that they can be applied to any problem.

Several works on POD-Galerkin reduced order models have shown that the pressure gradient term disappears from the momentum equations when the basis for the velocity-field is (discretely) divergence-free [115, 165, 223]. However, it is (in contrast to a staggered grid) not straightforward to derive a stable ‘velocity-

only' ROM, since the compatibility relation between divergence and gradient operators is not satisfied on a collocated grid [77, 271]. Typically, a combination of Rhie-Chow interpolation at the level of the full order model [208] and pressure stabilization on the ROM level is required to obtain stable solutions (as described in more detail in Section 2.5.3.1). However, even when stabilization techniques are applied, the ROM velocity and pressure fields are about one or two orders less accurate than the fields obtained by projecting the full order solutions onto the POD basis spaces. This is even the case for non-parametric laminar flow cases, such as the lid driven cavity flow problem [241].

Furthermore, a 'discretize-then-project' approach [152], i.e. projecting the fully discrete system, simplifies the treatment of the velocity boundary conditions. A recent study on ROMs on a staggered grid [223] demonstrated that the boundary conditions of the discrete FOM can be inherited by the ROM via the projection of the boundary vectors. With this approach, no additional boundary control method, such as the penalty [93, 134, 161, 234] or lifting function methods [79, 93, 239, 262] that are described in Chapter 4, is needed to handle the BCs at the ROM level.

To better understand how to deal with the challenges related to projection, pressure stabilization and the boundary conditions at ROM level, one needs to have a deep understanding of the underlying full order models.

In this work, we develop an efficient non-parametric ROM for the incompressible NS equations on collocated grids that does not require a pressure stabilization. We employ explicit time integration methods instead of implicit ones at the FOM and the ROM level in order to ease the derivations [17]. We base our approach on the recent progression on ROMs on staggered grids [223]. First of all, we project the fully discrete system, i.e. we project the discrete FOM operators and boundary vectors onto the POD basis spaces. Furthermore, we evaluate the divergence-freeness of the velocity fields and the necessity of pressure in the ROM formulation to develop a stable ROM. In addition, we derive a boundary condition treatment for which the ROM inherits the boundary conditions specified at discrete FOM level via the projection of the boundary vectors.

This chapter is organized as follows: First, we discuss the spatial and temporal discretization of the Navier-Stokes equations on a collocated grid for two different approaches for the computation of the convecting face fluxes in Section 7.2. In Section 7.3, we apply the POD-Galerkin method at the fully discrete level and show the construction of the ROMs in the online phase. In Section 7.4 the set-up of two numerical test cases, a lid driven cavity flow and an open cavity (with an inlet and an outlet) flow problem, are given and the results are provided and discussed in Sections 7.5 and 7.6, respectively. Finally, conclusions are drawn in Section 7.7 .

7.2 Finite volume discretization on collocated grids

We repeat from Section 2.1 that the general equations of mass and momentum conservation at the continuum level for a Newtonian flow with constant fluid density, constant kinematic viscosity and without body forces are given, respectively, by

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega, \quad (7.1)$$

$$\frac{\partial \mathbf{u}}{\partial t} = -\nabla \cdot (\mathbf{u} \otimes \mathbf{u}) + \nu \nabla \cdot (\nabla \mathbf{u}) - \nabla p \quad \text{in } \Omega, \quad (7.2)$$

where $\mathbf{u} = \mathbf{u}(\mathbf{x}, t)$ represents the vectorial velocity field that is evaluated at $\mathbf{x} \in \Omega \subset \mathbb{R}^d$ with $d = 2$ or 3 . Furthermore, $p = p(\mathbf{x}, t)$ is the normalized scalar pressure field, which is divided by the constant fluid density ρ , and t denotes time. These equations are subjected to initial and boundary conditions as described in Section 2.1.

We discretize the governing partial differential equations, Equations 7.1 and 7.2, using the finite volume method on a collocated grid, which is shown in Figure 7.1. We present both the spatial and temporal discretization. The fully discretized equations are projected on reduced basis spaces in the next section.

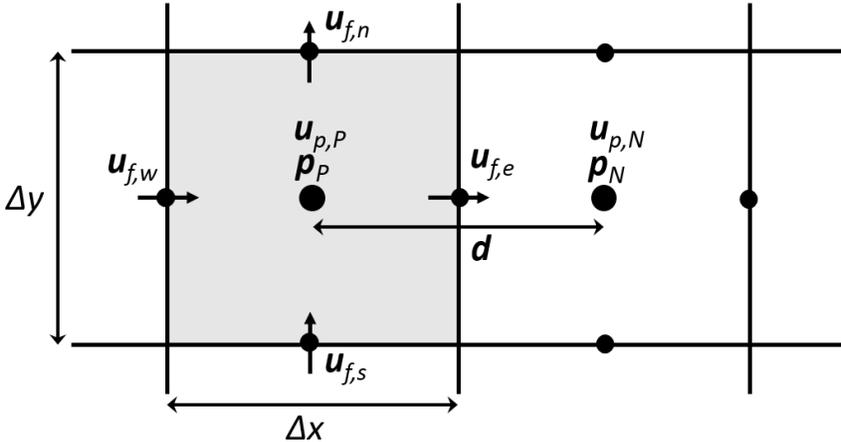


Figure 7.1: Two-dimensional collocated grid with the location of the unknowns at the cell center and the faces of a cell volume.

An integral formulation of the governing equations is imposed to all closed cell volumes such that the conservation laws are satisfied locally [81, 271]. The integral form of the conservation equations (Equations 2.1 and 2.2) for an arbitrary cell k are given by

$$\int_{\partial(\Omega_h)_k} \mathbf{n} \cdot \mathbf{u} dS = 0, \quad (7.3)$$

$$\int_{(\Omega_h)_k} \frac{\partial \mathbf{u}}{\partial t} d\Omega = - \int_{\partial(\Omega_h)_k} (\mathbf{n} \cdot \mathbf{u}) \mathbf{u} dS + \nu \int_{\partial(\Omega_h)_k} \mathbf{n} \cdot (\nabla \mathbf{u}) dS - \int_{\partial(\Omega_h)_k} \mathbf{n} p dS, \quad (7.4)$$

where $(\Omega_h)_k$ is the volume of cell k and $\partial(\Omega_h)_k$ is its boundary. $d\Omega$ is an infinitesimal volume element and dS is an infinitesimal element of surface area.

7.2.1 Spatial discretization

The finite-volume discretization of the governing equations in integral form, Equations 7.3 and 7.4, on an arbitrary collocated mesh consisting of h cells can be written in a matrix-vector notation:

$$\mathbf{B} \mathbf{u}_f = \mathbf{0}, \quad (7.5)$$

$$\frac{d\mathbf{u}_p}{dt} = -\mathbf{C}_p(\mathbf{u}_f, \mathbf{u}_p) - \mathbf{r}_p^C + \nu \mathbf{D}_p \mathbf{u}_p - \mathbf{G}_p \mathbf{p}_p + \nu \mathbf{r}_p^D, \quad (7.6)$$

where $\mathbf{p}_p = (p_{p,1}, p_{p,2}, \dots, p_{p,h})^T \in \mathbb{R}^h$ is the cell-centered pressure and $\mathbf{u}_p \in \mathbb{R}^{dh}$ the cell-centered velocity, which are defined as column vectors containing solely the cell-centered values. For a three-dimensional problem ($d = 3$), \mathbf{u}_p is arranged as $((\mathbf{u}_p)_1, (\mathbf{u}_p)_2, (\mathbf{u}_p)_3)^T$, where each $(\mathbf{u}_p)_i = ((u_{p,1})_i, (u_{p,2})_i, \dots, (u_{p,h})_i)$ for $i = 1, 2, 3$. $(\mathbf{u}_f)_i = ((u_{f,1})_i, (u_{f,2})_i, \dots, (u_{f,m})_i) \in \mathbb{R}^{dm}$, is the velocity evaluated on the cell faces and m the number of faces. Figure 7.2 depicts the location of the variables on a coarse grid. The face-centered velocity field \mathbf{u}_f is related to the cell-centered velocity field \mathbf{u}_p via a linear interpolation operator $\mathbf{I}_{p \rightarrow f} \in \mathbb{R}^{dm \times dh}$:

$$\mathbf{u}_f \equiv \mathbf{I}_{p \rightarrow f} \mathbf{u}_p + \mathbf{u}_b, \quad (7.7)$$

where $\mathbf{u}_b \in \mathbb{R}^{dm}$ is a vector that contains only the velocity values that are defined at boundary faces of the domain. For the two-dimensional example given in Figure 7.2, $(\mathbf{u}_b)_i = (0, 0, \dots, (u_{f,13})_i, \dots, (u_{f,24})_i)$ for $i = 1, 2$ and $13, \dots, 24$ are the indices of the faces at the boundary of the domain. An alternative option to relate \mathbf{u}_f to \mathbf{u}_p will be discussed in Section 7.2.2.2. Furthermore, matrix $\mathbf{B} \in \mathbb{R}^{h \times dm}$ is the face-to-center discrete divergence operator, $\mathbf{D}_p \in \mathbb{R}^{dh \times dh}$ represents the discrete cell-centered Laplacian operator associated with the diffusion term, $\mathbf{C}_p(\mathbf{u}_f, \mathbf{u}_p) \in \mathbb{R}^{dh \times dh}$ represents the nonlinear convection operator and matrix $\mathbf{G}_p \in \mathbb{R}^{dh \times h}$ is the discrete gradient operator. Furthermore, $\mathbf{r}_p^C \in \mathbb{R}^{dh}$ and $\mathbf{r}_p^D \in \mathbb{R}^{dh}$ are boundary vectors that contain the contributions of the convection and diffusion terms, respectively. All operators are scaled with the finite volume sizes.

We now detail the discretization of each term of the equations in integral form (Equations 7.3 and 7.4) for an arbitrary cell k , i.e. we give the details of the operators that are present in Equations 7.5 and 7.6.

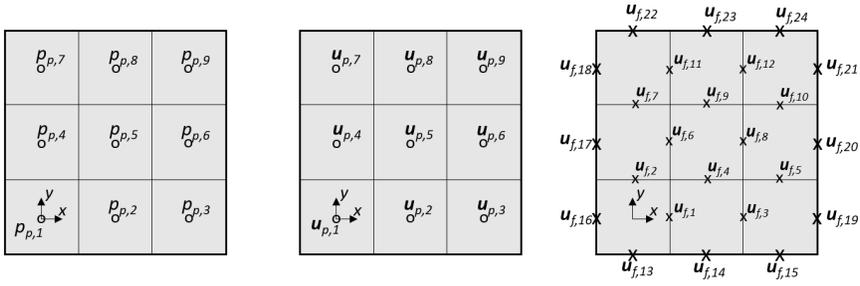


Figure 7.2: Sketch of a two-dimensional collocated grid with the location of the cell-centered pressure (left), the cell-centered velocity (middle) and the face-centered velocity (right).

The discretization of the continuity equation (Equation 7.3) yields

$$\int_{\partial(\Omega_h)_k} \mathbf{n} \cdot \mathbf{u} dS = \sum_i^{N_f} \int_{S_{f,i}} \mathbf{n} \cdot \mathbf{u} dS \approx \sum_{i=1}^{N_f} \mathbf{S}_{f,i} \cdot \mathbf{u}_{f,i} = \sum_{i=1}^{N_f} \phi_{f,i} = 0, \quad (7.8)$$

where N_f is the total number of faces bordering the cell k and $\mathbf{S}_f = \mathbf{n} S_f$ is the outward-pointing face area vector with S_f the area of the particular face. Hence, the face-to-center discrete divergence operator \mathbf{B} of Equation 7.5 consists of the outward pointing face areas associated with all faces of the discrete domain. However, Equation 7.8 shows that the divergence-free constraint is applied to the face flux, $\phi_f = \mathbf{S}_f \cdot \mathbf{u}_f$, and not to the cell-centered velocity \mathbf{u}_p . Therefore, we also need to introduce the center-to-center discrete divergence operator $\mathbf{B}_p \in \mathbb{R}^{h \times dh}$:

$$\mathbf{B}_p \equiv \mathbf{B} \mathbf{I}_{p \rightarrow f}. \quad (7.9)$$

Hence, the semi-discretized continuity Equation 7.5 can also be written as

$$\mathbf{B} \mathbf{u}_f = \mathbf{B} \mathbf{I}_{p \rightarrow f} \mathbf{u}_p + \mathbf{B} \mathbf{u}_b = \mathbf{B}_p \mathbf{u}_p + \mathbf{r}_p^B = \mathbf{0}, \quad (7.10)$$

where the boundary vector $\mathbf{r}_p^B \in \mathbb{R}^h$ is given by

$$\mathbf{r}_p^B \equiv \mathbf{B} \mathbf{u}_b, \quad (7.11)$$

which contains the contributions of the boundary conditions associated with the continuity equation.

The discretization of the pressure gradient term yields

$$\int_{\partial(\Omega_h)_k} \mathbf{n} p dS = \sum_{i=1}^{N_f} \int_{S_{f,i}} \mathbf{n} p dS \approx \sum_{i=1}^{N_f} \mathbf{S}_{f,i} p_{f,i}, \quad (7.12)$$

where, the face-centered pressure field \mathbf{p}_f is related to the cell-centered pressure field \mathbf{p}_p via a linear interpolation operator $\mathbf{\Pi}_{p \rightarrow f} \in \mathbb{R}^{m \times h}$:

$$\mathbf{p}_f \equiv \mathbf{\Pi}_{p \rightarrow f} \mathbf{p}_p. \quad (7.13)$$

Hence, the discretization of the pressure gradient \mathbf{G}_p consists of the face area vectors multiplied by the interpolation factors that are contained in $\mathbf{\Pi}_{p \rightarrow f}$.

Furthermore, the discretization of the diffusion term of the momentum equations for orthogonal meshes yields

$$\int_{\partial(\Omega_h)_k} \mathbf{n} \cdot \nabla \mathbf{u} \, dS = \sum_{i=1}^{N_f} \int_{S_{f,i}} \mathbf{n} \cdot \nabla \mathbf{u} \, dS \approx \sum_{i=1}^{N_f} |\mathcal{S}_{f,i}| \frac{\mathbf{u}_{p,N} - \mathbf{u}_{p,P}}{|\mathbf{d}|}, \quad (7.14)$$

where \mathbf{d} is the distance vector between any adjacent cell centers N and P to a particular face as shown in Figure 7.1. Hence, the discrete diffusion operator \mathbf{D}_p consists of coefficients associated with the face area vectors and the reciprocal of center-to-center distances. If the cell is neighboring a face, b , that is coinciding with the boundary of the computational domain, as shown in Figure 7.3, the discretization associated to that face changes to:

$$|\mathcal{S}_{f,b}| \frac{\mathbf{u}_{f,b} - \mathbf{u}_{p,P}}{|\mathbf{d}_n|}, \quad (7.15)$$

which is split in two terms:

$$\underbrace{|\mathcal{S}_{f,b}| \frac{\mathbf{0} - \mathbf{u}_{p,P}}{|\mathbf{d}_n|}}_{\mathbf{D}_p} + \underbrace{|\mathcal{S}_{f,b}| \frac{\mathbf{u}_{f,b} - \mathbf{0}}{|\mathbf{d}_n|}}_{\mathbf{r}_p^D}, \quad (7.16)$$

where $\mathbf{u}_{f,b}$ is the value of velocity at the boundary face b , \mathbf{d}_n is the distance vector between the face at the boundary of the domain and the center of the cell and $\mathcal{S}_{f,b}$ is the face area vector of b . The first term of Equation 7.16 is contained in the discrete diffusion operator \mathbf{D}_p , while the second term is contained in the boundary vector $\mathbf{r}_p^D \in \mathbb{R}^{dh}$.

Finally, the discretization of the convection term yields

$$\begin{aligned} \int_{\partial(\Omega_h)_k} (\mathbf{n} \cdot \mathbf{u}) \mathbf{u} \, dS &= \sum_{i=1}^{N_f} \int_{S_{f,i}} (\mathbf{n} \cdot \mathbf{u}) \mathbf{u} \, dS \approx \sum_{i=1}^{N_f} (\mathcal{S}_{f,i} \cdot \mathbf{u}_{f,i}) \mathbf{u}_{f,i} \\ &= \sum_{i=1}^{N_f} \phi_{f,i} \mathbf{u}_{f,i}. \end{aligned} \quad (7.17)$$

This shows that the convection operator $\mathbf{C}_p(\mathbf{u}_f, \mathbf{u}_p)$ is a nonlinear operator that depends on the face fluxes ϕ_f . In the case that the cell has a face that corresponds

to the boundary of the domain, as shown in Figure 7.3, the term:

$$\underbrace{(\mathbf{S}_{f,b} \cdot \mathbf{u}_{f,b})}_{\mathbf{r}_p^C} \mathbf{u}_{f,b} \quad (7.18)$$

is contained in the boundary vector $\mathbf{r}_p^C \in \mathbb{R}^{dh}$ instead of the matrix associated with the convection operator. In this work, the nonlinearity of the discretized convection term is quadratic, because \mathbf{u}_f is obtained via linear interpolation of \mathbf{u}_p . Hence, we can redefine the convection operator in terms of a matrix-vector product:

$$\tilde{\mathbf{C}}_p(\mathbf{u}_f) \mathbf{u}_p \equiv \mathbf{C}_p(\mathbf{u}_f, \mathbf{u}_p). \quad (7.19)$$

Finally, substituting Equation 7.10 in the continuity equation (Equation 7.5) and Equations 7.19 in the momentum equations (Equation 7.6) results in the following spatially discretized system of equations:

$$\mathbf{B}_p \mathbf{u}_p + \mathbf{r}_p^B = \mathbf{0}, \quad (7.20)$$

$$\frac{d\mathbf{u}_p}{dt} = -\tilde{\mathbf{C}}_p(\mathbf{u}_f) \mathbf{u}_p + \nu \mathbf{D}_p \mathbf{u}_p - \mathbf{G}_p \mathbf{p}_p + \mathbf{r}_p, \quad (7.21)$$

where $\mathbf{r}_p \in \mathbb{R}^{dh} \equiv -\mathbf{r}_p^C + \nu \mathbf{r}_p^D$. All operators are scaled with the finite volume sizes.

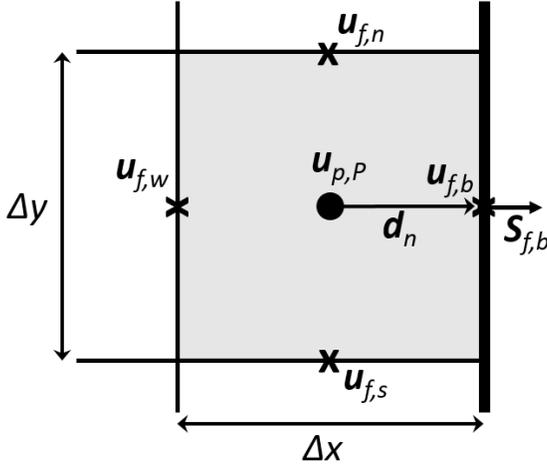


Figure 7.3: Two-dimensional collocated grid with the location of the velocity at the cell center and the faces of a cell volume of a cell near the boundary face of the domain.

7.2.2 Explicit projection methods (temporal discretization)

Applying a central discretization stencil to the velocity divergence (Equation 7.8) and the pressure gradient (Equation 7.12) together with collocated grids generally results in spurious pressure oscillations [140]. These oscillations occur because the compatibility relation between the divergence and gradient operators is not satisfied (in contrast to a staggered grid) [77, 271]. This so-called checkerboard problem is caused by a wide stencil in the PPE, which yields a pressure-velocity decoupling at adjacent cell centers [60]. The typical solution for this problem is to use the Rhie-Chow interpolation [208] for the cell-centered face velocities. The PISO (Pressure-Implicit with Splitting of Operators [128]) solver that is standardly available in OpenFOAM corresponds to this method of Rhie and Chow [144]. However, the use of the Rhie-Chow interpolation is unnecessary even on collocated grids when finite volume projection methods (also called fractional methods and not to be confused with the Galerkin projection method) with explicit time integration methods are employed [114].

We showed in the previous section that in a collocated setting there exist two different velocity fields, namely, the velocities at the cell faces, \mathbf{u}_f , and the cell-centered velocities, \mathbf{u}_p . The cell-centered velocities together with the pressure form the primary solution variables. They can be related to the face-centered velocity via a linear interpolation (Equation 7.7). We call this approach the inconsistent flux method (IFM). The fluxes at the cell centers are only approximately discretely divergence free with this approach, which is shown in the next subsection. We therefore also discuss a second approach for which we have an additional equation for the face velocities. We call this method the consistent flux method (CFM).

Recently, Komen et al. [144] analyzed five numerical algorithms in finite volume collocated grid solvers for the incompressible Navier-Stokes equations for a selection of explicit (and implicit) Runge-Kutta schemes. They demonstrated that the temporal order reduces to approximately one also for the higher order schemes (except for the high-order method of Kazemi [136], which however turns out to be very dissipative). Therefore, and for simplicity reasons as mentioned in the Introduction, we describe both projection methods (IFM and CFM) with the explicit Euler method [278] (also called Forward Euler) i.e. the original Chorin-Temam algorithm [42, 55, 254]. The extension of our approach to multi-stage (Runge-Kutta) methods is straightforward.

7.2.2.1 Inconsistent flux method

We discretize in the time using Forward Euler, which is first order [224], the spatially discretized mass and momentum equations including boundary conditions

(Equations 7.20 and 7.21). Writing them in vector form results in:

$$\mathbf{B}_p \mathbf{u}_p^{n+1} + \mathbf{r}_p^B = \mathbf{0}, \quad (7.22)$$

$$\frac{\mathbf{u}_p^{n+1} - \mathbf{u}_p^n}{\Delta t} = -\tilde{\mathbf{C}}_p(\mathbf{u}_f^n) \mathbf{u}_p^n + \nu \mathbf{D}_p \mathbf{u}_p^n - \mathbf{G}_p \mathbf{p}_p^{n+1} + \mathbf{r}_p, \quad (7.23)$$

where Δt is the time step and $\mathbf{u}_p^n \approx \mathbf{u}_p(t^n)$ is the solution at the n^{th} time step.

As we showed in the previous section, velocity and pressure are coupled. The projection method computes first an intermediate velocity \mathbf{u}_p^* by ignoring the pressure gradient term in the momentum equations:

$$\frac{\mathbf{u}_p^* - \mathbf{u}_p^n}{\Delta t} = -\tilde{\mathbf{C}}_p(\mathbf{u}_f^n) \mathbf{u}_p^n + \nu \mathbf{D}_p \mathbf{u}_p^n + \mathbf{r}_p. \quad (7.24)$$

Only the viscous and convective forces are thus considered in this step. Moreover, \mathbf{u}_p^* is, in general, not divergence free.

Then the projection step follows, where the intermediate velocity field is corrected by the pressure in order to obtain the solution of \mathbf{u}_p at time step $n + 1$:

$$\mathbf{u}_p^{n+1} = \mathbf{u}_p^* - \Delta t \mathbf{G}_p \mathbf{p}_p^{n+1}. \quad (7.25)$$

In order to obtain a divergence free velocity field at the next time step, Equation 7.22, we take the divergence of Equation 7.25:

$$\mathbf{B}_p \mathbf{u}_p^{n+1} + \mathbf{r}_p^B = (\mathbf{B}_p \mathbf{u}_p^* + \mathbf{r}_p^B) - \Delta t \mathbf{B}_p \mathbf{G}_p \mathbf{p}_p^{n+1} = \mathbf{0}. \quad (7.26)$$

Rewriting Equation 7.26 leads to the PPE in fully-discretized form:

$$\mathbf{L}_p \mathbf{p}_p^{n+1} = \frac{1}{\Delta t} (\mathbf{B}_p \mathbf{u}_p^* + \mathbf{r}_p^B), \quad (7.27)$$

where $\mathbf{L}_p \equiv \mathbf{B}_p \mathbf{G}_p \in \mathbb{R}^{h \times h}$ is a wide stencil Laplacian operator. Basically, this operator is based on interpolating the computed cell-centered pressure gradients to the cell faces. As a result, the pressure is decoupled at neighboring cells [140]. Hence, the pressure solution may contain non-physical spurious modes, which is known as the checkerboard problem.

By taking the pressure gradient directly at the cell faces, the checkerboard problem is avoided. This is similar to the way the diffusion operator is discretized in Equation 7.14, for which the direct neighboring cells are used without alternately skipping neighboring cells [271]. This approach corresponds to the original interpolation method of Rhie and Chow. Therefore, we use the compact stencil given by the compact Laplacian operator $\mathbf{L}_f \in \mathbb{R}^{h \times h}$ instead of \mathbf{L}_p . \mathbf{L}_f is also the standard Laplacian operator used in OpenFOAM [278]. However, when using \mathbf{L}_f instead of \mathbf{L}_p , the continuity constraint at the cell centers $\mathbf{B}_p \mathbf{u}_p^{n+1} + \mathbf{r}_p^B = \mathbf{0}$ is no longer satisfied.

Finally, the solution of the PPE is used to correct the cell-centered velocity field as done in Equation 7.25. As a result, the cell-centered velocity fields do not conserve mass and are only approximately divergence free [74, 176]. Moreover, the computation of the face velocity lacks the correction by the flux that appears in the PPE and an incomplete flux term remains [144].

To make the Galerkin projection procedure that will be introduced in Section 7.3 straightforward, we rewrite the fully discrete system of equations (Equations 7.24-7.27) in such a way that we have only one equation for the pressure and one equation for the cell-centered velocity at the next time step, respectively:

$$\mathbf{L}_f \mathbf{p}_p^{n+1} = \frac{1}{\Delta t} (\mathbf{B}_p \mathbf{u}_p^n + \mathbf{r}_p^B) + \mathbf{B}_p \left(-\tilde{\mathbf{C}}_p(\mathbf{u}_f^n) \mathbf{u}_p^n + \nu \mathbf{D}_p \mathbf{u}_p^n + \mathbf{r}_p \right), \quad (7.28)$$

$$\mathbf{u}_p^{n+1} = \mathbf{u}_p^n + \Delta t \left(-\tilde{\mathbf{C}}_p(\mathbf{u}_f^n) \mathbf{u}_p^n + \nu \mathbf{D}_p \mathbf{u}_p^n + \mathbf{r}_p \right) - \Delta t \mathbf{G}_p \mathbf{p}_p^{n+1}. \quad (7.29)$$

For the inconsistent flux method, the velocity at the faces \mathbf{u}_f are approximated using the interpolation operator $\mathbf{I}_{p \rightarrow f}$ of Equation 7.7. Furthermore, the linear system of Equation 7.28 needs to be solved to obtain \mathbf{p}_p^{n+1} , while Equation 7.29 is fully explicit.

7.2.2.2 Consistent flux method

In this method, we use the pressure field obtained by solving the PPE (Equation 7.28) to also correct the face fluxes. We apply first the linear interpolation operator onto Equation 7.29:

$$\mathbf{u}_f^{n+1} = \mathbf{I}_{p \rightarrow f} \left[\mathbf{u}_p^n + \Delta t \left(-\tilde{\mathbf{C}}_p(\mathbf{u}_f^n) \mathbf{u}_p^n + \nu \mathbf{D}_p \mathbf{u}_p^n + \mathbf{r}_p \right) - \Delta t \mathbf{G}_p \mathbf{p}_p^{n+1} \right], \quad (7.30)$$

which is equivalent to Equation 7.7. However, rather than interpolating the cell-centered pressure gradients (using $\mathbf{I}_{p \rightarrow f} \mathbf{G}_p$), we directly evaluate the pressure gradients at the faces using a new discrete face gradient operator $\mathbf{G}_f \in \mathbb{R}^{dm \times h}$. Therefore, Equation 7.30 can be rewritten as:

$$\mathbf{u}_f^{n+1} = \mathbf{I}_{p \rightarrow f} \mathbf{u}_p^n + \Delta t \mathbf{I}_{p \rightarrow f} \left(-\tilde{\mathbf{C}}_p(\mathbf{u}_f^n) \mathbf{u}_p^n + \nu \mathbf{D}_p \mathbf{u}_p^n + \mathbf{r}_p \right) - \Delta t \mathbf{G}_f \mathbf{p}_p^{n+1}. \quad (7.31)$$

The spatial discretization of the pressure gradient term, i.e. the last term of Equation 7.31 on the right hand side, for a cell k is approximated by

$$\sum_{i=1}^{N_f} \mathbf{S}_{f,i} \frac{p_{p,N} - p_{p,P}}{|\mathbf{d}|}. \quad (7.32)$$

Hence, the gradient operator \mathbf{G}_f consists of coefficients associated with the surface normal vectors and the reciprocal of center-to-center distances. \mathbf{G}_f directly uses

the cell-centered pressure to calculate the gradient, while G_p (Equation 7.12) is based on the linear interpolation of the pressure in the cell centers.

If we then take the divergence of Equation 7.31 according to 7.10:

$$\begin{aligned} B\mathbf{u}_f^{n+1} &= (B_p\mathbf{u}_p^n + \mathbf{r}_p^B) \\ &+ B_p \left[\Delta t \left(-\tilde{C}_p(\mathbf{u}_f^n)\mathbf{u}_p^n + \nu D_p\mathbf{u}_p^n + \mathbf{r}_p \right) \right] - \Delta t B G_f p_p^{n+1}, \end{aligned} \quad (7.33)$$

we see that the combination of the first two terms on the right hand side is equal to the right hand side of Equation 7.28 (multiplied by Δt). Therefore, substituting the pressure computed with Equation 7.28, proves that the face velocity fields are discretely divergence free as the right hand side of Equation 7.33 is zero.

The system of equations for the consistent flux method is then formed by Equations 7.28, 7.29 and 7.31, which are solved in this particular order to obtain the solution for \mathbf{u}_f , \mathbf{u}_p and p_p at t^{n+1} .

7.3 POD-Galerkin reduced order models for the explicit projection methods

We apply the POD-Galerkin method [149, 235] directly on the fully discrete formulations given by Equations 7.28 and 7.29 for the inconsistent flux method and Equations 7.28, 7.29 and 7.31 for the consistent flux method. Therefore, the full order models and reduced order models are both first order in time (Forward Euler).

We assume that the FOM solutions can be expressed as a linear combination of orthonormal spatial modes multiplied by time-dependent coefficients [149]. The discrete cell-centered velocity fields, \mathbf{u}_p , are approximated by

$$\mathbf{u}_p \approx \mathbf{u}_{p,r} = \Phi \mathbf{a}, \quad (7.34)$$

where $\Phi = (\varphi_1, \varphi_2, \dots, \varphi_{N_r^u}) \in \mathbb{R}^{dh \times N_r^u}$ is a matrix containing the cell-centered velocity modes $\varphi \in \mathbb{R}^{dh}$. For a three-dimensional problem ($d = 3$), φ is arranged as $((\varphi)_1, (\varphi)_2, (\varphi)_3)^T$, where each $(\varphi)_i = ((\varphi_1)_i, (\varphi_2)_i, \dots, (\varphi_h)_i)$ for $i = 1, 2, 3$. $\mathbf{a} = (a^1, a^2, \dots, a^{N_r^u})^T \in \mathbb{R}^{N_r^u}$ are the corresponding time-dependent coefficients with N_r^u the number of velocity modes. The subscript r denotes quantities associated to the ROM.

Similarly, the discrete pressure fields are approximated by

$$\mathbf{p}_p \approx \mathbf{p}_{p,r} = \mathbf{X} \mathbf{b}, \quad (7.35)$$

where $\mathbf{X} = (\chi_1, \chi_2, \dots, \chi_{N_r^p}) \in \mathbb{R}^{h \times N_r^p}$ is a matrix containing the cell-centered pressure modes $\chi = (\chi_1, \chi_2, \dots, \chi_h)^T \in \mathbb{R}^h$ and $\mathbf{b}^n = (b^1, b^2, \dots, b^{N_r^p})^T \in \mathbb{R}^{N_r^p}$

the corresponding time-dependent coefficients with N_r^p the number of pressure modes.

Finally, the discrete face velocity fields are approximated by

$$\mathbf{u}_f \approx \mathbf{u}_{f,r} = \Psi \mathbf{c}, \quad (7.36)$$

where $\Psi = (\boldsymbol{\psi}_1, \boldsymbol{\psi}_2, \dots, \boldsymbol{\psi}_{N_r^u}) \in \mathbb{R}^{dm \times N_r^u}$ is a matrix containing the face velocity modes $\boldsymbol{\psi} \in \mathbb{R}^m$ and $\mathbf{c}(t) = (c^1, c^2, \dots, c^{N_r^u})^T \in \mathbb{R}^{N_r^u}$ the corresponding time-dependent coefficients. For a three-dimensional problem ($d = 3$), $\boldsymbol{\psi}$ is arranged as $((\boldsymbol{\psi})_1, (\boldsymbol{\psi})_2, (\boldsymbol{\psi})_3)^T$, where each $(\boldsymbol{\psi})_i = ((\boldsymbol{\psi}_1)_i, (\boldsymbol{\psi}_2)_i, \dots, (\boldsymbol{\psi}_m)_i)$ for $i = 1, 2, 3$.

7.3.1 Proper orthogonal decomposition

The optimal POD basis space for the cell-centered velocity, $E_{POD}^{u_p} = \text{span}(\boldsymbol{\varphi}_1, \boldsymbol{\varphi}_2, \dots, \boldsymbol{\varphi}_{N_r^u})$ is constructed by minimizing the difference between the snapshots, i.e. the discrete solutions at several time instances, and their orthogonal projection onto the reduced basis for the L_2 -norm:

$$E_{POD}^{u_p} = \arg \min_{\boldsymbol{\varphi}_1, \dots, \boldsymbol{\varphi}_{N_r^u}} \frac{1}{N_s^u} \sum_{n=1}^{N_s^u} \left\| \mathbf{u}_p^n - \sum_{i=1}^{N_r^u} (\mathbf{u}_p^n, \boldsymbol{\varphi}_i)_{L_2(\Omega_h)} \boldsymbol{\varphi}_i \right\|_{L_2(\Omega_h)}^2, \quad (7.37)$$

subjected to the orthogonality constraint $\Phi^T \Omega \Phi = \mathbf{I}$, where $\Omega \in \mathbb{R}^{dh \times dh}$ is a diagonal matrix with the cell-centered control volumes and $\mathbf{I} \in \mathbb{R}^{N_r^u \times N_r^u}$ is the identity matrix. N_s^u is the number of velocity snapshots and $N_r^u \leq N_s^u$. $(\cdot, \cdot)_{L_2(\Omega_h)}$ is the discrete L_2 -inner product of the fields over the whole discrete domain Ω_h . The L_2 -norm is the preferred norm for discrete numerical schemes [40, 241] with

$$(\mathbf{u}_p^n, \boldsymbol{\varphi}_i)_{L_2(\Omega_h)} \equiv \sum_{j=1}^h \mathbf{u}_{p,j}^n \cdot \boldsymbol{\varphi}_{i,j}(\Omega_h)_j. \quad (7.38)$$

The optimal POD basis space for the cell-centered pressure, $E_{POD}^p = \text{span}(\boldsymbol{\chi}_1, \boldsymbol{\chi}_2, \dots, \boldsymbol{\chi}_{N_r^p})$ is constructed in a similar way.

For the face velocity, $E_{POD}^{u_f} = \text{span}(\boldsymbol{\psi}_1, \boldsymbol{\psi}_2, \dots, \boldsymbol{\psi}_{N_r^u})$ is constructed as follows:

$$E_{POD}^{u_f} = \arg \min_{\boldsymbol{\psi}^1, \dots, \boldsymbol{\psi}_{N_r^u}} \frac{1}{N_s^u} \sum_{n=1}^{N_s^u} \left\| \mathbf{u}_f^n - \sum_{i=1}^{N_r^u} (\mathbf{u}_f^n, \boldsymbol{\psi}_i)_{L_2(\Sigma)} \boldsymbol{\psi}_i \right\|_{L_2(\Sigma)}^2, \quad (7.39)$$

where the discrete inner product $(\cdot, \cdot)_{L_2(\Sigma)}$ is defined over all face areas Σ :

$$(\mathbf{u}_f^n, \boldsymbol{\psi}_i)_{L_2(\Sigma)} \equiv \sum_{j=1}^m \mathbf{u}_{f,j}^n \cdot \boldsymbol{\psi}_{i,j}(S_f)_j. \quad (7.40)$$

The minimization problem mentioned in Equation 7.37 is equivalent to solving the following eigenvalue problem on a set of snapshots:

$$\mathbf{C}^u \mathbf{Q}^u = \mathbf{Q}^u \boldsymbol{\lambda}^u, \quad (7.41)$$

with

$$\mathbf{C}_{ij}^u = (\mathbf{u}_p^i, \mathbf{u}_p^j)_{L_2(\Omega_h)} \quad \text{for } i, j = 1, \dots, N_s^u, \quad (7.42)$$

where $\mathbf{C}^u \in \mathbb{R}^{N_s^u \times N_s^u}$ is the correlation matrix of velocity snapshots, $\mathbf{Q}^u \in \mathbb{R}^{N_s^u \times N_s^u}$ is a square matrix of eigenvectors and $\boldsymbol{\lambda}^u \in \mathbb{R}^{N_s^u \times N_s^u}$ is a diagonal matrix containing the eigenvalues. The POD modes, φ_i , are then constructed as follows

$$\varphi_i = \frac{1}{N_s^u \sqrt{\lambda_i^u}} \sum_{n=1}^{N_s^u} \mathbf{u}_p^n Q_{in}^u \quad \text{for } i = 1, \dots, N_r^u. \quad (7.43)$$

The cell-centered velocity modes φ are only approximately discretely divergence free like the cell-centered velocity \mathbf{u}_p . As a consequence, it is necessary to include pressure in the ROM formulations to develop a stable ROM.

The most energetic (dominant) POD modes are selected based on the decay of the eigenvalues λ_i^u . The procedure is the same for obtaining the pressure modes and the face velocity modes using the appropriate inner products.

7.3.2 Galerkin projection for the inconsistent flux method

The approximations of the discrete velocity and pressure fields (Equations 7.34 and 7.35) are substituted into the FOM of the inconsistent flux method (Equations 7.28 and 7.29). The PPE (Equation 7.28) is then projected onto the reduced basis spanned by the pressure modes by left-multiplying with $\mathbf{X}^T \boldsymbol{\Omega}$:

$$\begin{aligned} \mathbf{X}^T \boldsymbol{\Omega} \mathbf{L}_f \mathbf{X} \mathbf{b}^{n+1} &= \frac{1}{\Delta t} \left(\mathbf{X}^T \boldsymbol{\Omega} \mathbf{B}_p \boldsymbol{\Phi} \mathbf{a}^n + \mathbf{X}^T \boldsymbol{\Omega} \mathbf{r}_p^B \right) \\ &\quad + \mathbf{X}^T \boldsymbol{\Omega} \mathbf{B}_p \left(-\tilde{\mathbf{C}}_p(\mathbf{I}_{p \rightarrow f} \boldsymbol{\Phi} \mathbf{a}^n) \boldsymbol{\Phi} \mathbf{a}^n + \nu \mathbf{D}_p \boldsymbol{\Phi} \mathbf{a}^n + \mathbf{r}_p \right). \end{aligned} \quad (7.44)$$

Rewriting leads to the following ROM formulation for the equation for pressure:

$$\mathbf{L}_r \mathbf{b}^{n+1} = \frac{1}{\Delta t} (\mathbf{B}_r \mathbf{a}^n + \mathbf{q}_r^B) - \hat{\mathbf{A}}_r(\mathbf{a}^n) \mathbf{a}^n + \nu \hat{\mathbf{D}}_r \mathbf{a}^n + \mathbf{q}_r, \quad (7.45)$$

where the reduced matrices associated with the linear terms, $\mathbf{L}_r = \mathbf{X}^T \boldsymbol{\Omega} \mathbf{L}_f \mathbf{X} \in \mathbb{R}^{N_r^p \times N_r^p}$, $\mathbf{B}_r = \mathbf{X}^T \boldsymbol{\Omega} \mathbf{B}_p \boldsymbol{\Phi} \in \mathbb{R}^{N_r^p \times N_r^u}$ and $\hat{\mathbf{D}}_r = \mathbf{X}^T \boldsymbol{\Omega} \mathbf{B}_p \mathbf{D}_p \boldsymbol{\Phi} \in \mathbb{R}^{N_r^p \times N_r^u}$ and the reduced vector $\mathbf{q}_r = \mathbf{X}^T \boldsymbol{\Omega} \mathbf{B}_p \mathbf{r}_p \in \mathbb{R}^{N_r^p}$, can all be determined during the offline stage. The nonlinear convection term $\hat{\mathbf{A}}_r(\mathbf{a}) \in \mathbb{R}^{N_r^p \times N_r^u \times N_r^u}$ is also pre-computed during the offline stage and is stored as a third order tensor. Therefore, $\hat{\mathbf{A}}_r$ consists of N_r^u components $\hat{\mathbf{A}}_{r,i} \in \mathbb{R}^{N_r^p \times N_r^u}$ and is constructed as:

$$\hat{\mathbf{A}}_{r,i} = \mathbf{X}^T \boldsymbol{\Omega} \mathbf{B}_p \tilde{\mathbf{C}}_p(\mathbf{I}_{p \rightarrow f} \boldsymbol{\Phi}_i) \boldsymbol{\Phi}. \quad (7.46)$$

During the online stage, the term $\hat{\mathbf{A}}_r(\mathbf{a}^n)\mathbf{a}^n$ of Equation 7.45 is evaluated as

$$\sum_{i=1}^{N_r^u} (\mathbf{a}^n)^T \hat{\mathbf{A}}_{r,i} \mathbf{a}^n. \quad (7.47)$$

This only holds when the interpolation operator $I_{p \rightarrow f}$ is linear, e.g. the convection term is quadratic and discretized with a linear discretization scheme.

Similarly, the discrete momentum equations 7.29 are projected onto the reduced basis spanned by the velocity modes by left-multiplying with $\Phi^T \Omega$:

$$\begin{aligned} \Phi^T \Omega \Phi \mathbf{a}^{n+1} &= \Phi^T \Omega \Phi \mathbf{a}^n + \Delta t \Phi^T \Omega \left(-\tilde{\mathcal{C}}_p(I_{p \rightarrow f} \Phi \mathbf{a}^n) \Phi \mathbf{a}^n + \nu D_p \Phi \mathbf{a}^n + \mathbf{r}_p \right) \\ &\quad - \Delta t \Phi^T \Omega \mathbf{G}_p \mathbf{X} \mathbf{b}^{n+1}. \end{aligned} \quad (7.48)$$

Rewriting this in matrix-vector notation leads to the following ROM formulation for the momentum equations:

$$\mathbf{a}^{n+1} = \mathbf{a}^n + \Delta t \left(-\hat{\mathcal{C}}_r(\mathbf{a}^n) \mathbf{a}^n + \nu D_r \mathbf{a}^n + \mathbf{r}_r \right) - \Delta t \hat{\mathbf{G}}_r \mathbf{b}^{n+1}, \quad (7.49)$$

where the reduced matrices $D_r = \Phi^T \Omega D_p \Phi \in \mathbb{R}^{N_r^u \times N_r^u}$ and $\hat{\mathbf{G}}_r = \Phi^T \Omega \mathbf{G}_p \mathbf{X} \in \mathbb{R}^{N_r^u \times N_r^p}$ and the reduced vector $\mathbf{r}_r = \Phi^T \Omega \mathbf{r}_p \in \mathbb{R}^{N_r^u}$ can all be determined during the offline stage. The equation is simplified by $\Phi^T \Omega \Phi = \mathbf{I}$. Similar to Equation 7.46, the nonlinear convection term $\hat{\mathcal{C}}_r(\mathbf{a}) \in \mathbb{R}^{N_r^u \times N_r^u \times N_r^u}$ is precomputed during the offline stage and stored as a third order tensor.

During the online stage, the linear system of Equation 7.45 can be solved for the pressure coefficients \mathbf{b}^{n+1} as all terms of the right hand side depend solely on the solutions at time step t^n . This vector of coefficients is then used to calculate the velocity coefficients \mathbf{a}^{n+1} at the new time step t^{n+1} from Equation 7.49. The boundary conditions are incorporated in the ROM (Equations 7.44 and 7.49) as the boundary vector \mathbf{r}_p is also projected onto the reduced bases. Therefore, no additional boundary control method is needed.

In many POD-Galerkin ROMs it is assumed that the POD velocity modes satisfy the strong divergence free constraint and that the pressure only enters the ROM on the boundary [149, 161]. Then, the pressure gradient term completely vanishes in the case of enclosed flow. This is not true for the inconsistent flux method as the discrete cell-centered velocity field is only approximately discretely divergence free. Therefore the divergence free constraint is also not fully satisfied neither at the FOM nor at the ROM level.

7.3.3 Galerkin projection for the consistent flux method

We obtain the ROM for the consistent flux method by following the same Galerkin projection procedure for the inconsistent flux method as described in the previ-

ous subsection. The approximations of the discrete cell-centered velocity, face-centered velocity and pressure fields (Equations 7.34, 7.35 and 7.36) are substituted into the FOM of the consistent flux method (Equations 7.28, 7.29 and 7.31). This results in the following reduced system of equations in matrix-vector notation:

$$\mathbf{L}_r \mathbf{b}^{n+1} = \frac{1}{\Delta t} (\mathbf{B}_r \mathbf{a}^n + \mathbf{q}_r^B) - \mathbf{A}_r(\mathbf{c}^n) \mathbf{a}^n + \nu \hat{\mathbf{D}}_r \mathbf{a}^n + \mathbf{q}_r, \quad (7.50)$$

$$\mathbf{a}^{n+1} = \mathbf{a}^n + \Delta t (-\mathbf{C}_r(\mathbf{c}^n) \mathbf{a}^n + \nu \mathbf{D}_r \mathbf{a}^n + \mathbf{r}_r) - \Delta t \hat{\mathbf{G}}_r \mathbf{b}^{n+1}, \quad (7.51)$$

$$\mathbf{W}_r \mathbf{c}^{n+1} = \mathbf{V}_r \mathbf{a}^n + \Delta t (-\mathbf{K}_r(\mathbf{c}^n) \mathbf{a}^n + \nu \mathbf{P}_r \mathbf{a}^n + \mathbf{s}_r) - \Delta t \mathbf{G}_r \mathbf{b}^{n+1}, \quad (7.52)$$

with $\mathbf{W}_r = \Psi^T \Sigma \Psi \in \mathbb{R}^{N_r^u \times N_r^u}$, $\mathbf{V}_r = \Psi^T \Sigma \mathbf{I}_{p \rightarrow f} \Phi \in \mathbb{R}^{N_r^u \times N_r^u}$, $\mathbf{P}_r = \Psi^T \Sigma \mathbf{D}_p \Phi \in \mathbb{R}^{N_r^u \times N_r^u}$, $\mathbf{G}_r = \Psi^T \Sigma \mathbf{G}_f \mathbf{X} \in \mathbb{R}^{N_r^u \times N_r^p}$ and the reduced vector $\mathbf{s}_r = \Psi^T \Sigma \mathbf{r}_p \in \mathbb{R}^{N_r^u}$. The matrix $\Sigma \in \mathbb{R}^{dm \times dm}$ contains the face areas of the cells. The reduced convection terms $\mathbf{A}_r(\mathbf{a}^n) \in \mathbb{R}^{N_r^p \times N_r^u \times N_r^u}$, $\mathbf{C}_r(\mathbf{a}^n) \in \mathbb{R}^{N_r^u \times N_r^u \times N_r^u}$ and $\mathbf{K}_r(\mathbf{a}^n) \in \mathbb{R}^{N_r^u \times N_r^u \times N_r^u}$ are determined, respectively, by

$$\mathbf{A}_{r,i} = \mathbf{X}^T \Omega \mathbf{B}_p \tilde{\mathbf{C}}_p(\Psi_i) \Phi, \quad (7.53)$$

$$\mathbf{C}_{r,i} = \Phi^T \Omega \tilde{\mathbf{C}}_p(\Psi_i) \Phi, \quad (7.54)$$

$$\mathbf{K}_{r,i} = \Psi^T \Sigma \tilde{\mathbf{C}}_p(\Psi_i) \Phi. \quad (7.55)$$

As the face-centered velocity fields are discretely divergence free, also the POD flux modes are discretely divergence free [41]. Therefore, the pressure gradient term of Equation 7.52 completely vanishes in the case of enclosed flow [149].

The reduced system of the CFM (Equations 7.50-7.52) differs from the reduced system of the IFM (Equations 7.45-7.49) in three ways. First of all, the reduced equation for the coefficients of the face-centered velocity is added to the CFM-ROM in the same way that the CFM-FOM also has an additional equation for \mathbf{u}_f at the new time step. Secondly, the convection terms of Equations 7.50-7.52 depend on the face-centered velocity coefficients \mathbf{c} instead of the cell-centered velocity coefficients \mathbf{a} . Thirdly, more reduced matrices need to be precomputed during the offline stage, which results in additional storage and CPU costs compared to the IFM.

7.4 Numerical set-up

In this section the numerical set-up of two cases is described. The first test case is the classical lid driven cavity benchmark, which is a closed flow problem. The second test case consists of an open cavity flow problem featuring an inlet and outlet boundary. This is an important test case for testing the projection of the

boundary vectors. Both cases are modeled on a two-dimensional domain. Full order simulations are carried out for both the consistent and inconsistent flux method that have been implemented in ITHACA-FV [240], which is an open source C++ library based on OpenFOAM [130]. The libraries of OpenFOAM 6 are used in this work. For the full order simulations, the spatial discretization is performed using central differencing schemes. For the open cavity, an upwind discretization scheme is used for the convective term due to a higher Peclet number of this case and to test the methods for different numerical schemes.

We focus on the non-parametric case. Therefore, the same boundary conditions are applied in the ROM as in the FOM for which the snapshots are collected. The time step, total simulation time and the Reynolds number are also identical for the FOM and the ROM.

7.4.1 Lid-driven cavity flow problem

Figure 7.4 depicts a sketch of the geometry of the two-dimensional lid driven cavity problem. The length of the square cavity, L , equals 1.0 m. A (64×64) structured mesh with quadrilateral cells is constructed on the domain. A tangential uniform velocity $U_{lid} = 1.0$ m/s is prescribed at the top wall and non-slip conditions are applied to the other walls. The Reynolds number based on the velocity of the lid and the cavity characteristic length is 100 and the flow is considered laminar. The pressure reference value is set to $0 \text{ m}^2/\text{s}^2$ at coordinate $(0,0)$ at the lower left corner of the cavity. The initial condition for the cell-centered velocity is a zero field: $\mathbf{u}_0 = \mathbf{0}$ m/s. Simulations are run with a constant time step of $\Delta t = 5 \cdot 10^{-3}$ s and for a total simulation time, T , of 1.0 s.

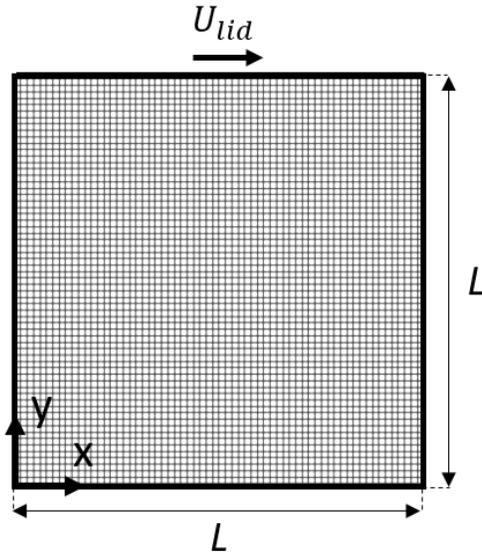


Figure 7.4: Sketch of the geometry and mesh of the 2D square cavity with a moving top lid.

7.4.2 Open cavity flow problem

The second test case consists of a two-dimensional square cavity problem with an inlet and outlet along the top [21, 198, 233]. Figure 7.5 depicts a sketch of the geometry. The height of the cavity equals its length $L = 1.0$ m. The fluid enters from the left of the domain at a uniform velocity $U_\infty = 1.0$ m/s. The inlet is located $L_u = 1.2L$ upstream of the cavity and the exit $L_d = 1.5L$ downstream of the cavity. The outflow boundary condition of Equation 2.10 is considered at the outlet. The no-slip boundary condition is applied to all walls. The pressure reference value is set to 0 Pa at coordinate $(0,0)$. The computational domain is divided into 7125 quadrilateral cells. The Reynolds number based on the free-stream velocity U_∞ and the cavity characteristic length L is 200.

The initial condition for the cell-centered velocity is determined by solving a potential flow problem subjected to the problem’s boundary conditions is given by

$$\begin{cases} \nabla \cdot \mathbf{u}_0 = 0 & \text{in } \Omega, \\ \nabla^2 p = 0 & \text{in } \Omega. \end{cases} \quad (7.56)$$

The total simulation time is $T = 2.0$ s with a time step $\Delta t = 2.5 \times 10^{-3}$ s. Snapshots of the flow fields are collected every time step. Table 7.1 summarizes the computational details for the open cavity flow problem.

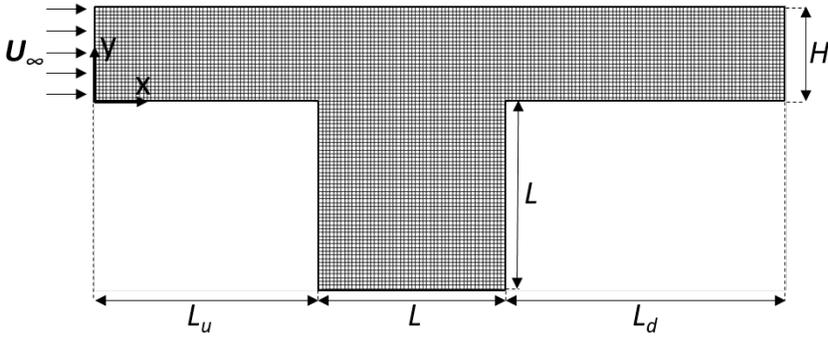


Figure 7.5: Sketch of the geometry and mesh of the 2D open cavity.

Table 7.1: Computational details for the lid driven cavity and open cavity flow problems.

Variables	Lid driven cavity	Open cavity
Number of cells	4096	7125
Cavity length L	1.0 m	1.0 m
U_{lid}, U_∞	1.0 m/s	1.0 m/s
Viscosity ν	0.01 m ² /s	0.005 m ² /s
Reynolds number	100	200
Simulation time T	1.0 s	2.0 s
Time step Δt	0.005 s	0.0025 s
Spatial scheme convection	Linear (central differencing)	Linear (upwind)
Temporal scheme	Forward Euler	Forward Euler

7.5 Results

In this section, we show the full order and reduced order results of two test cases: the lid driven cavity flow problem and the open cavity flow problem. These open and closed flow test cases are excellent test cases to demonstrate the difference in the treatment of the (non-homogeneous) boundary conditions: In the case of the closed cavity, a tangential boundary condition is applied on the top wall of the cavity, while an inflow and outflow boundary condition are applied for the open flow problem.

One of the main goals of this work is to reproduce the FOM results with our developed reduced order models in a stable and accurate way. Therefore, rather than validating the models against experimental results and/or other numerical models, we directly compare the ROM results with the corresponding FOM results.

We analyze and compare the FOM and ROM results of the inconsistent flux

method and the consistent flux method. The main difference between the two projection methods is that mass conserving face fluxes are obtained with the CFM, while the fluxes are only approximately discretely divergence free in the case of the IFM. Therefore, we compare the summation of the local continuity errors for every cell at all time instances as they give an indication of how well the continuity equation is satisfied in the simulations. The local time step continuity error is calculated, according to the definition used by OpenFOAM [95], as follows for the FOM fields:

$$\epsilon_{local}(t^n) = \sum_{k=1}^h \frac{\Delta t}{(\Omega_h)_k} \left| \left[\sum_{i=1}^{N_f} \phi_{f,i}(t^n) \right]_k \right|. \quad (7.57)$$

Similarly, the local time step continuity error can be determined for the POD velocity modes and the fields obtained with the ROMs.

Furthermore, we compute the relative error of the cell-centered fields at each time step to show the performance of the proposed methods. For this we consider the following three types of fields at a time instance t^n : the full order fields \mathbf{u}_p^n and p_p^n , the projected fields $\hat{\mathbf{u}}_{p,r}^n = \Phi \Phi^T \Omega \mathbf{u}_p^n$ and $\hat{p}_{p,r}^n = \mathbf{X} \mathbf{X}^T \Omega p_p^n$, which are obtained by the L_2 -projection of the snapshots onto the POD bases and lastly, the predicted fields $\mathbf{u}_{p,r}^n$ and $p_{p,r}^n$ obtained by solving the ROMs. For every time instance, t^n , the relative basis projection error is given by

$$\hat{\epsilon}_{L_2}^u(t^n) = \frac{\|\mathbf{u}_p^n - \hat{\mathbf{u}}_{p,r}^n\|_{L_2(\Omega_h)}}{\|\mathbf{u}_p^n\|_{L_2(\Omega_h)}}, \quad (7.58)$$

and the relative prediction error is determined by

$$\epsilon_{L_2}^u(t^n) = \frac{\|\mathbf{u}_p^n - \mathbf{u}_{p,r}^n\|_{L_2(\Omega_h)}}{\|\mathbf{u}_p^n\|_{L_2(\Omega_h)}}. \quad (7.59)$$

Similarly, $\hat{\epsilon}_{L_2}^p(t^n)$ and $\epsilon_{L_2}^p(t^n)$ are computed for the pressure fields. For each of the cases and methods we compare the relative prediction error with the basis projection error, which is the ‘best possible’ error at every time instance.

Finally, we determine the speedup in computational time, which is defined as the FOM CPU time divided by the ROM CPU time.

7.5.1 Lid driven cavity flow problem

Full order simulations are performed for the lid driven cavity problem according to Section 7.4.1. The velocity and pressure profiles at the centerlines of the cavity at final simulation time are shown in Figures 7.6 and 7.7, respectively. These figures show that the full order solutions obtain with the inconsistent flux method are close to the consistent-flux solutions.

The local continuity errors (Equation 7.57) of the IFM-FOM is of the order $\mathcal{O}(10^{-6})$, while it is of the order $\mathcal{O}(10^{-16})$ in the case of the CFM-FOM. Nevertheless, this difference can be considered negligible in this particular case as the Figures 7.6 and 7.7 show that the methods perform equally.

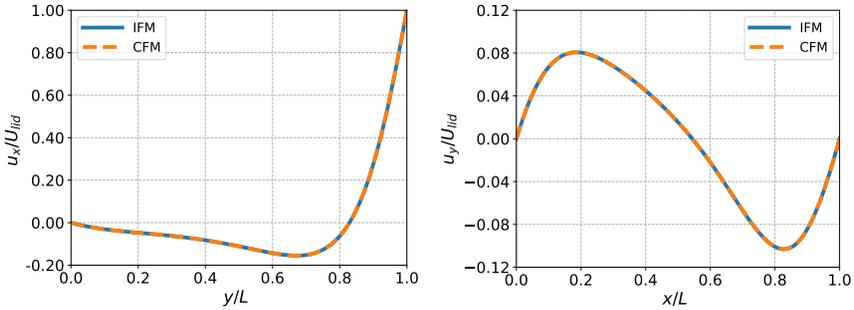


Figure 7.6: Velocity profiles for the lid driven cavity flow case at final simulation time: (left) normalized velocity component in the x -direction at $x/L = 0.5$; (right) normalized velocity component in the y -direction at $y/L = 0.5$.

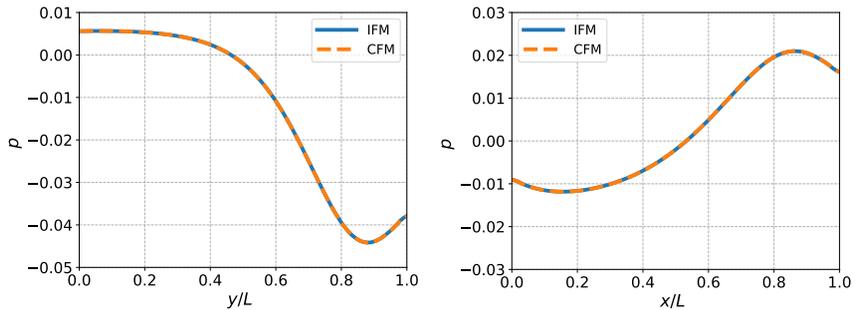
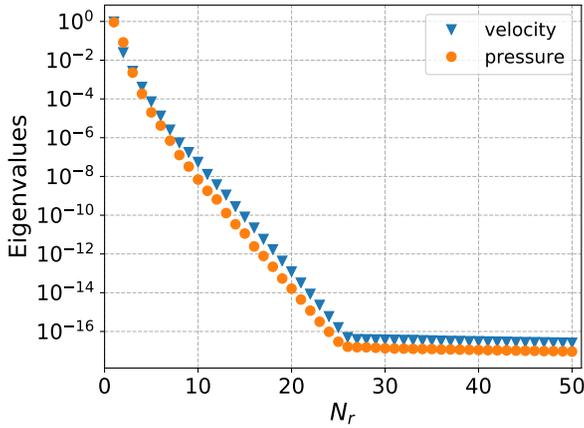
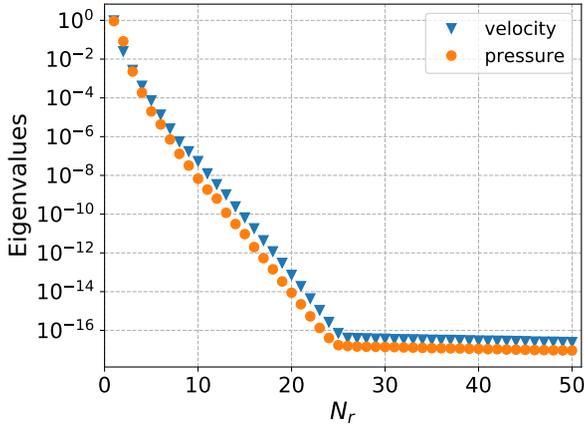


Figure 7.7: Normalized pressure profiles for the lid driven cavity flow case at final simulation time: (left) at $x/L = 0.5$; (right) at $y/L = 0.5$.

The POD eigenvalues of the cell-centered velocity and pressure modes are shown in Figure 7.8 for the IFM and CFM. The eigenvalues are approximately the same for both projection methods. For both velocity and pressure, the values decay rapidly for increasing number of modes. Therefore, the problem is suited for dimension reduction. A plateau is reached at about 25 modes due to the machine precision. As the slope of eigenvalue decay is almost the same for pressure and velocity, we take an equal number of modes N_r for the reduced pressure basis and reduced velocity basis: $N_r = N_r^u = N_r^p$.



(a) inconsistent flux method

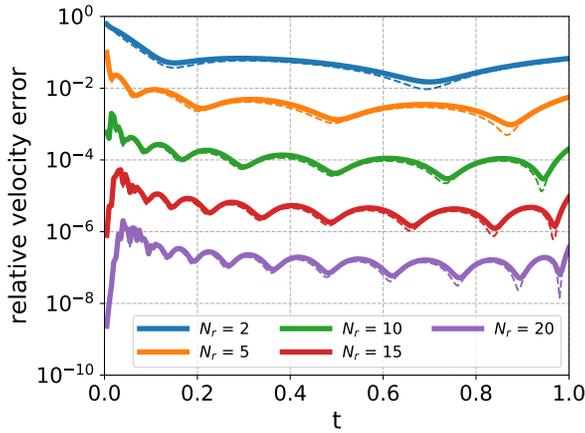


(b) consistent flux method

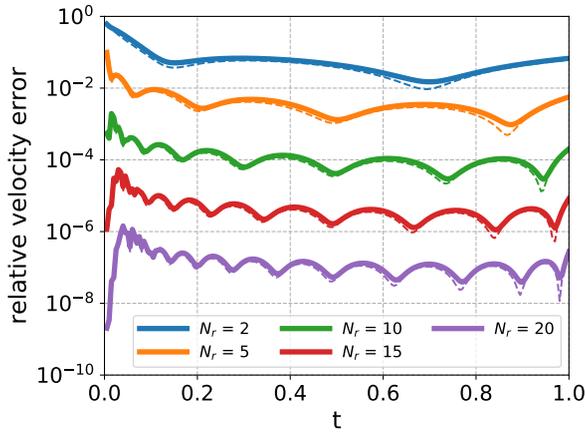
Figure 7.8: Eigenvalues as function of the number of modes for the lid driven cavity flow case.

We study the effect of increasing the number of modes on the accuracy of the cell-centered velocity field, while using the full snapshot set as basis for the POD. We take $N_r = 2; 5; 10; 15; 20$.

The relative prediction and basis projection errors are plotted in Figure 7.9 for velocity and Figure 7.10 for pressure. We clearly see how the accuracy increases when increasing the number of modes. The relative error for a certain number of modes appears to be almost the same for both projection methods. This means that both ROMs are consistent with the FOMs used for the snapshot collection [99].



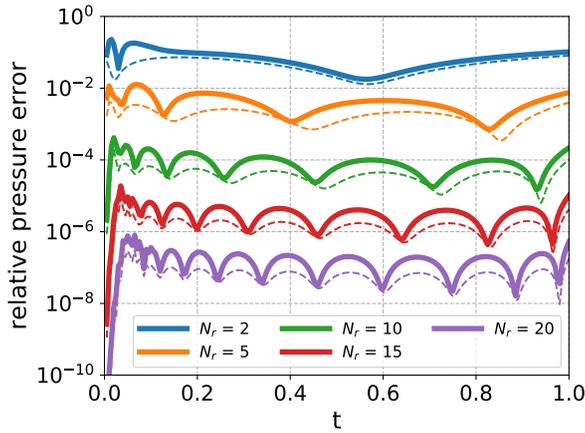
(a) inconsistent flux method



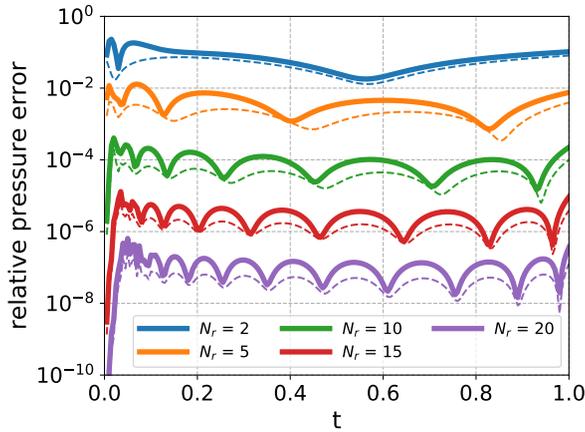
(b) consistent flux method

Figure 7.9: Relative cell-centered velocity error as a function of time for different number of modes for the lid driven cavity flow case. Dashed lines: basis projection error (projecting snapshots onto truncated basis).

Furthermore, for both ROM methods, the relative velocity errors (Equation 7.59) are very close to the relative basis projection errors (Equation 7.58) as they are almost overlapping. In addition, the relative pressure errors are of the same order as the velocity errors for the same number of modes. This is also shown in Figure 7.11 in which we plotted the time-averaged basis projection errors (Equation 7.58) and time-averaged ROM prediction errors (Equation 7.59) for velocity and pressure.



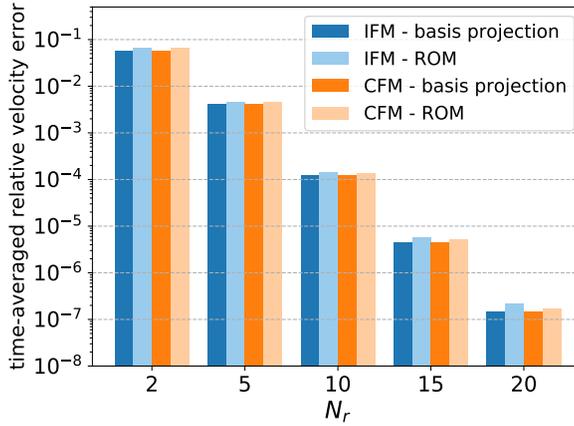
(a) inconsistent flux method



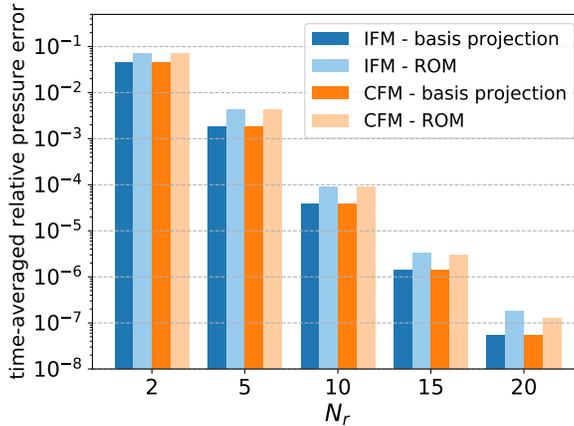
(b) consistent flux method

Figure 7.10: Relative pressure error as a function of time for different number of modes for the lid driven cavity flow case. Dashed lines: basis projection error (projecting snapshots onto truncated basis).

In all cases, stable ROM results were obtained with the proposed explicit projection method. This indicates that additional pressure stabilization methods, such as the supremizer enrichment technique, the exploitation of a pressure Poisson equation during the projection stage or the novel local projection stabilization methods [222], are not required. Moreover, in this test case a relative error of about $\mathcal{O}(10^{-4})$, which is accurate enough for many engineering applications, is obtained with only 10 velocity and 10 pressure modes (plus 10 face velocity modes in the case of the consistent flux method).



(a) velocity

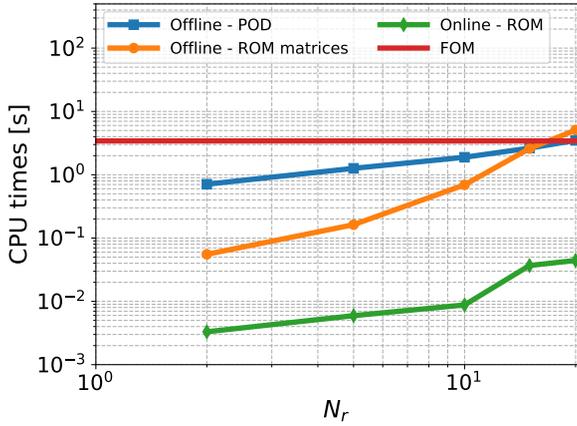


(b) pressure

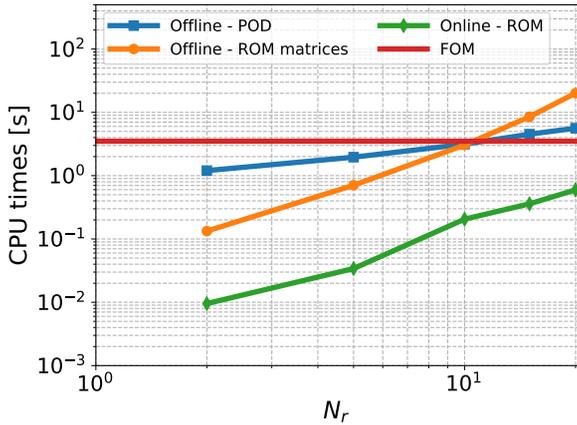
Figure 7.11: Time-averaged relative basis projection and prediction errors of the lid driven cavity flow problem.

Furthermore, the local continuity errors computed for the IFM-POD velocity modes and the IFM-ROM are of the $\mathcal{O}(10^{-6})$ (regardless the number of modes used). On the other hand, the local continuity errors are of the order $\mathcal{O}(10^{-16})$ for the CFM, which is of the order of the machine precision. They are of the same order as for the corresponding FOMs. Thus, the discrete face velocity is only approximately discretely divergence free in the case of the IFM, whereas it is discretely divergence free with the consistent flux method.

Finally, the computational time required by the ROMs is compared to the FOM CPU times in Figure 7.12 for the IFM and CFM. The plotted computational times are the average times of two simulations.



(a) inconsistent flux method



(b) consistent flux method

Figure 7.12: Computational times in seconds as function of number of modes for the lid driven cavity flow case.

For both methods, the speedup ratio between the ROM and the FOM is shown in Figure 7.13, which depend strongly on the number of modes used for the ROMs. In the case of the CFM, an additional equation for the face velocity (Equation 7.30) needs to be solved at the reduced order level, which explains the lower speedup compared to the IFM. Moreover, the larger the number of modes, the more time it takes to precompute the reduced matrices. This especially applies to those related to the convection operators as the dimension of the tensors increases with the cube of the number of POD modes. The cost is higher for the CFM than the IFM as more matrices need to be precomputed due to the additional equation for the face velocity (Equation 7.30). Therefore, the time to compute the POD modes is also higher for the consistent flux method. The POD is relatively expensive compared to the ROM simulation time. However, the POD modes only need to be determined once during the offline phase.

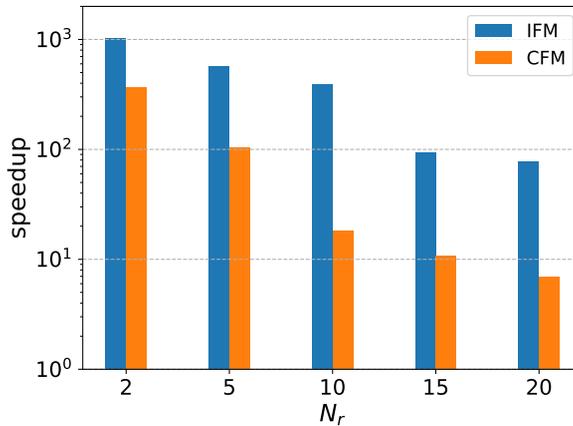


Figure 7.13: Speedup in computational time of the ROM compared to the FOM in seconds as function of number of modes for the lid driven cavity flow case.

7.5.2 Open cavity flow problem

Full order simulations are performed for the open cavity problem according to Section 7.4.2. The cell-centered velocity (magnitude) and pressure snapshots at $t = 0; 0.5; 1.0; 2.0$ s that are obtained with the consistent flux method are shown in Figure 7.14; these snapshots look similar for the IFM. This figure shows that the problem is unsteady for the simulated time span.

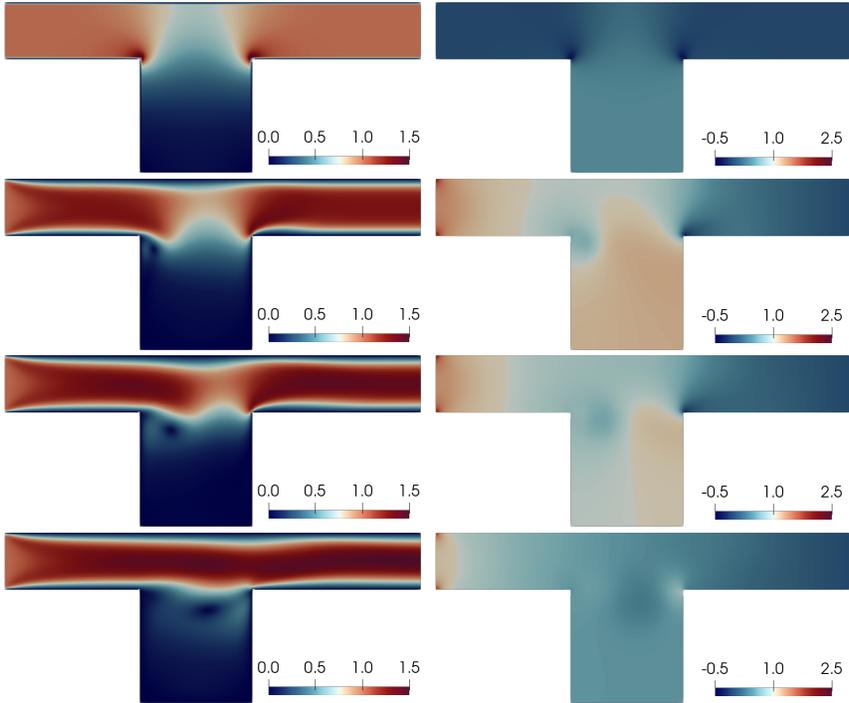
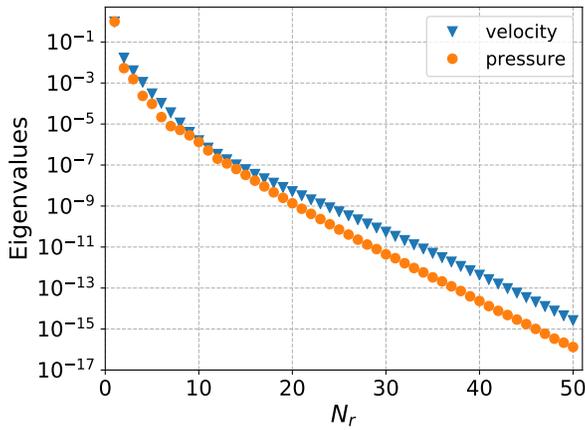
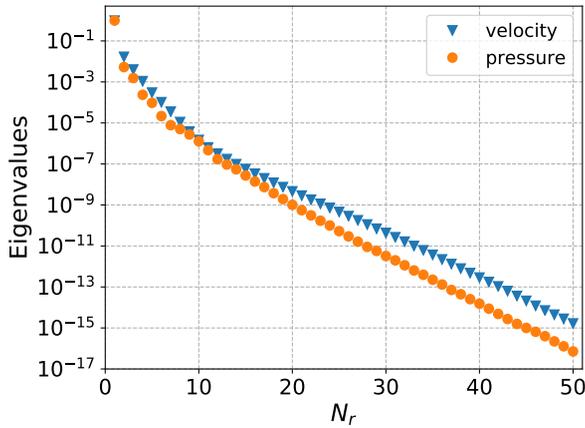


Figure 7.14: From top to bottom: Snapshots obtained at $t = 0; 0.5; 1.0; 2.0$ s with the consistent flux method: (left) cell-centered velocity magnitude in m/s; (right) pressure in Pa.

The POD eigenvalues of the cell-centered velocity and pressure modes are shown in Figure 7.15. The eigenvalues are approximately the same for both projection methods. For both velocity and pressure, the rate of decay of the first ten modes is steeper than the rate of decay of the higher modes. The eigenvalues also decay less rapidly for increasing number of modes compared to the lid driven cavity case (Figure 7.8), which indicates that more POD modes are needed to approximate the FOM solutions accurately. As the slope of eigenvalue decay is more or less the same for pressure and velocity, we take equal numbers of modes $N_r = 2; 5; 10; 15; 20$ for the reduced pressure basis and reduced velocity basis.



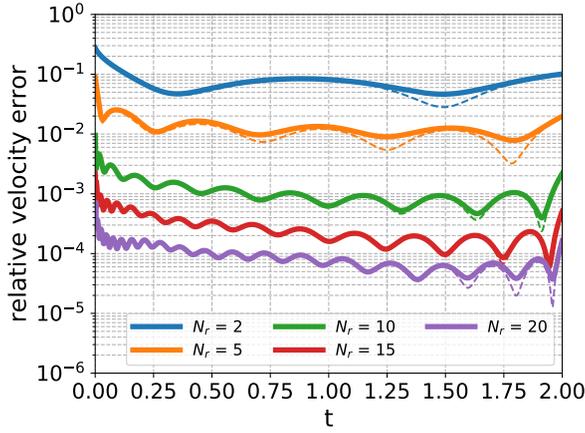
(a) inconsistent flux method



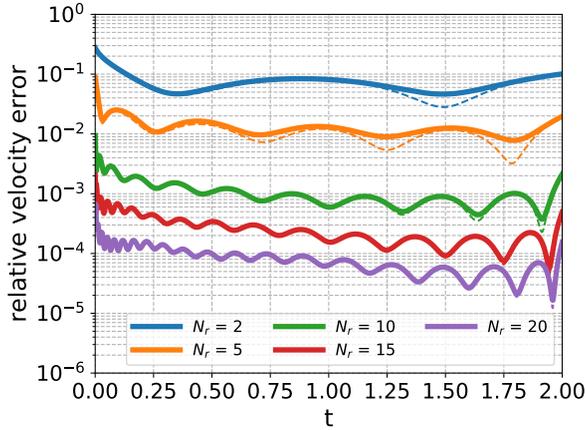
(b) consistent flux method

Figure 7.15: Eigenvalues for the open cavity flow case.

The relative prediction and basis projection errors (Equations 7.58 and 7.59) are plotted in Figure 7.16 for velocity and Figure 7.17 for pressure. These figures show that the errors decrease when increasing the number of modes for both the IFM-ROM and CFM-ROM. Figure 7.16 shows that the relative velocity errors (Equation 7.59) are very close to the basis projection errors (Equation 7.58) as they are almost overlapping. However, after about 1.5 seconds of simulation time the prediction error for 20 modes start slightly deviating from the projection error for the same number of modes in the case of the inconsistent flux methods, while the errors are almost overlapping in the case of the consistent flux method.



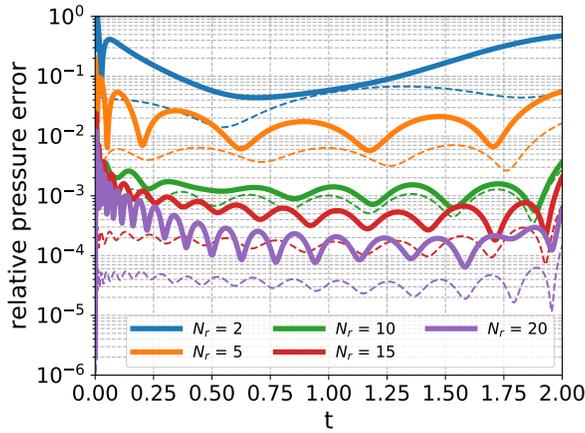
(a) inconsistent flux method



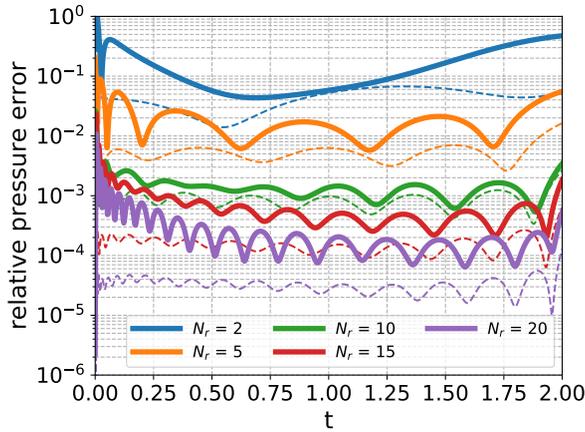
(b) consistent flux method

Figure 7.16: Relative cell-centered velocity error as a function of time for different number of modes for the open cavity flow case. Dashed lines: basis projection error (projecting snapshots onto truncated basis).

The relative pressure errors plotted in Figure 7.17 are of the same order as the velocity errors in Figure 7.16 for the same number of modes. Also for pressure, the prediction error at around 1.8 s of simulation time is higher for the IFM-ROM compared to the CFM-ROM for 20 modes.



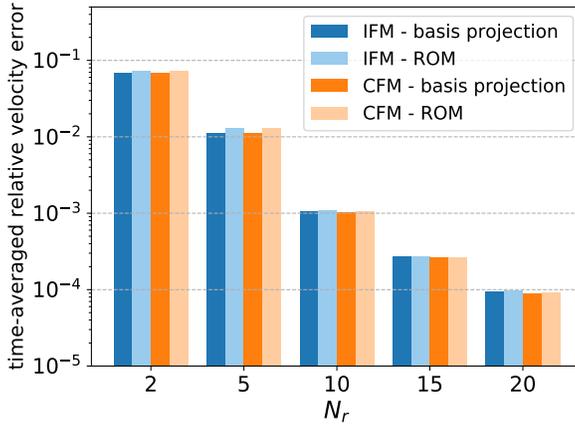
(a) inconsistent flux method



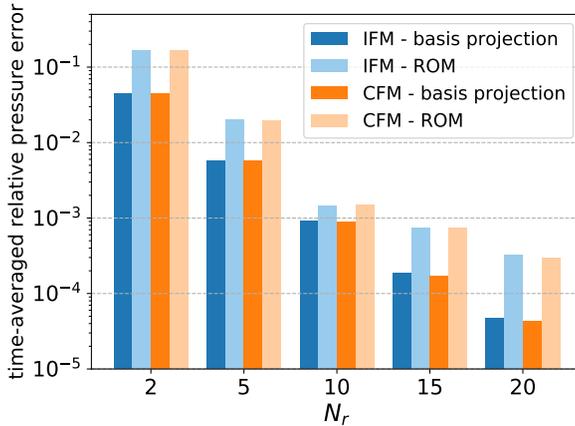
(b) consistent flux method

Figure 7.17: Relative pressure error as a function of time for different number of modes for the open cavity flow case. Dashed lines: basis projection error (projecting snapshots onto truncated basis).

Moreover, the difference between the prediction and projection errors is the smallest for 10 modes as is also shown in Figure 7.18 in which we plotted the time-averaged basis projection errors (Equation 7.58) and time-averaged ROM prediction errors (Equation 7.59) for velocity and pressure, respectively.



(a) velocity



(b) pressure

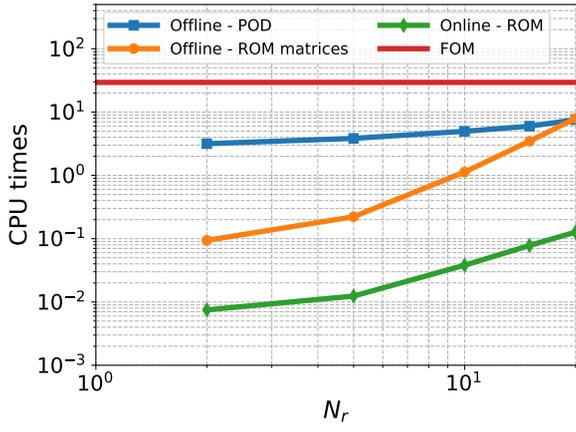
Figure 7.18: Time-averaged relative basis projection and prediction errors of the open cavity flow problem.

In all cases, stable ROM results were obtained with the proposed explicit projection method, indicating that additional pressure stabilization methods are not required (as discussed in the Introduction (Section 7.1)). Moreover, a relative error of about $\mathcal{O}(10^{-3})$ is obtained with only 10 velocity and 10 pressure modes (plus 10 face velocity modes in the case of the consistent flux method) in this test case.

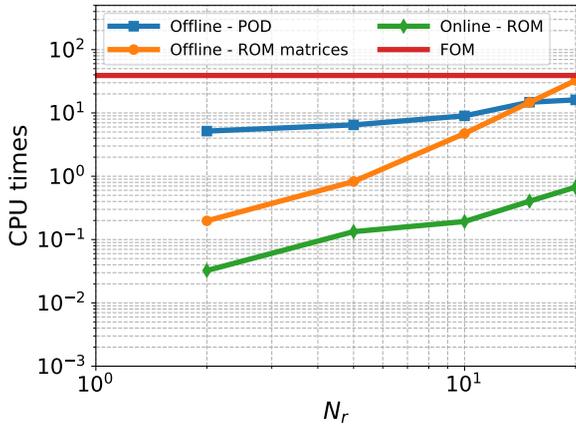
Furthermore, the local continuity errors (Equation 7.57) of the IFM-FOM is of the order $\mathcal{O}(10^{-5})$. Also the local continuity errors computed for the POD

velocity modes and the IFM-ROM are of the order $\mathcal{O}(10^{-5})$ (regardless the number of modes used). On the other hand, the local continuity errors are of the order $\mathcal{O}(10^{-16})$ for the CFM, which is of the order of the machine precision. Thus, the discrete face velocity is only approximately discretely divergence free in the case of the IFM, whereas the constraint is fully satisfied with the CFM.

Finally, the computational times required by the ROMs is compared to the FOM CPU times in Figure 7.19.



(a) inconsistent flux method



(b) consistent flux method

Figure 7.19: Computational times in seconds as function of number of modes the open cavity flow case.

The plotted computational times are the average times of two simulations. The speedup is plotted in Figure 7.20 and is between about 2×10^2 and 4×10^3 , depending on the number of modes used for the IFM-ROM, while the speedup is between about 6×10^1 and 1×10^3 for the CFM-ROM. This is according to expectations as an additional equation for the face velocity (Equation 7.30) needs to be determined at ROM level. For the same reason, more matrices need to be precomputed for the CFM, which explains the higher cost. Moreover, the larger the number of modes, the more time it takes to precompute the reduced matrices.

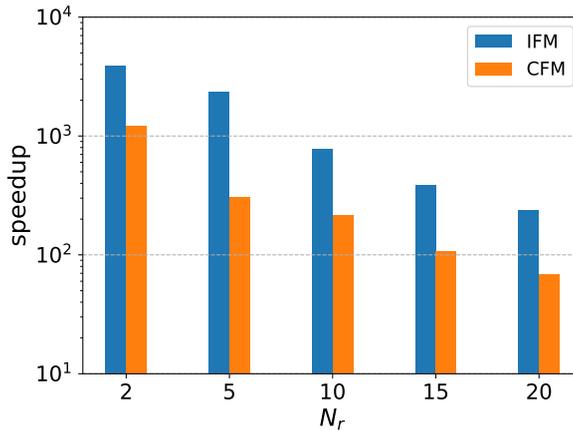


Figure 7.20: Speedup in computational time of the ROM compared to the FOM in seconds as function of number of modes for the open cavity flow case.

7.6 Discussion

The main difference between the two projection methods, the inconsistent flux method and the consistent flux method, is the divergence freeness of the fluxes. Whereas the fluxes are discretely divergence free in the case of the CFM, they are only approximately discretely divergence free in the case of the IFM. Nevertheless, the difference in the cell-centered velocity and pressure solutions can be considered negligible in our test cases.

However, in both methods the cell-centered velocity fields are only approximately discretely divergence free. As a consequence, velocity and pressure are also coupled at the reduced order level. Therefore, pressure needs to be included in the ROM formulation for the incompressible NS equations and cannot be simply recovered in a post-processing step, in contrast to ‘velocity-only’ ROMs [41, 82, 137].

The CFM-FOM and CFM-ROM simulation take more computational time than the equivalent models with the IFM as shown in Figures 7.12 and 7.19, for the lid driven cavity and open cavity test cases, respectively. This is mostly due to the additional equation that needs to be solved for the fluxes at FOM and ROM level as well as computing the reduced POD basis space for the face-centered velocity. Therefore, it is plausible to prefer the IFM method despite the fact that the velocity fields are only approximately discretely free.

On the other hand, as observed for the open cavity case, the IFM-ROM is slightly less accurate than the CFM-ROM towards the end of the ROM simulation when a large number of modes is used for the construction of the reduced bases. For different test cases than the cases studied in this work, the divergence error of the IFM could be potentially much larger, leading to possibly non-physical or inaccurate results.

Another possible cause of the slight difference, which only occurred for large number of modes, is that the modes with smaller eigenvalues are dominated by numerical noise. Therefore, the drop in eigenvalue magnitude does not always provide a reliable identification of a reduced basis of high quality [153].

We have only investigated first order explicit temporal discretization. Moreover, we first discretize in space and in time before performing the Galerkin projection. Therefore, the ROM formulations are fully corresponding with the FOM formulations. Higher-order explicit (Runge-Kutta) methods, such as those analyzed by Komen et al. [144], are generally more accurate than the Forward Euler scheme used in this work. However, to keep the ROM and the FOM consistent with each other, higher-order methods would require the implementation of the different stages also at reduced order level. This is, in contrast to the FOM level, not straightforward at the ROM level.

Moreover, the disadvantage of explicit schemes is that the systems become unstable for Courant numbers larger than unity. This can form a severe limitation for the time step [130]. The standard OpenFOAM method is PISO, which is an implicit pressure-based scheme for the NS equations. The segregated nature of PISO induces a decoupling between mass and momentum equations. The PISO algorithm has similarities with the consistent flux method presented in this work. Therefore, it would be an asset to extend the CFM to implicit schemes. However, a number of corrections of the pressure and velocity fields are needed to enforce the velocity-pressure coupling at each time step and to minimize the errors. Therefore, the same challenge as for higher order (explicit) Runge-Kutta schemes applies, namely keeping the ROM and the FOM consistent with each other.

Furthermore, the results have shown that with the current approach of projecting the boundary vectors onto the reduced bases, it is not needed to use a penalty method or a lifting function method to enforce the boundary conditions in the ROM. This approach can also be implemented for PISO or PIMPLE al-

gorithms for collocated grids that are more frequently used in engineering applications as the implicit time discretization is, generally, more stable than explicit schemes [77, 271].

The methodology can be extended to parametric problems as the ROM formulations are already written in such a way that viscosity is not part of the diffusion operator and the associated boundary vector (when projecting the boundary vectors, \mathbf{r}_p^C and \mathbf{r}_p^D , onto the reduced basis spaces separately).

Finally, the speedup is higher for the open cavity case compared to the lid driven cavity case as the FOM contains a larger number degrees of freedom. With an increasing number of modes, the precomputing phase (in particular assembling the reduced convection operator) becomes the dominant factor in the ROM execution. In our test cases this is not a concern, as the number of modes is typically sufficient before the precomputing phase becomes a dominant factor. Nevertheless, one could reduce the complexity of the convection operator (a third order tensor) by using hyper-reduction techniques such as the discrete empirical interpolation method [50].

7.7 Conclusions

The novel reduced order models are developed using a ‘discretize-then-project’ approach. The ROM formulations are fully corresponding to the discrete FOM formulations of the incompressible NS equations on collocated grids. No pressure stabilization method is needed, even though the pressure term is present in the ROM. Moreover, the boundary conditions at the ROM level are imposed via the projection of the boundary vectors that are specified at the discrete FOM level. Therefore, it is not needed to use a boundary control method such as the penalty method or lifting function method.

We considered two variants of a Forward Euler time discretization: the inconsistent flux method, for which the velocity at the cell centers are considered only approximately discretely divergence free and the consistent flux method, for which the face velocities are discretely divergence free.

The ROMs predict well the underlying FOMs as stable and accurate results are obtained with the proposed methods for the lid driven cavity and open cavity flow cases. The ROMs obtained with the consistent flux method, having divergence-free velocity fields, are slightly more accurate compared to the inconsistent flux method.

However, the speedup of the ROM compared to the FOM is lower for the consistent flux method due to the additional equation for the face velocity (Equation 7.30) that also needs to be solved at the ROM level. Furthermore, the speedup strongly depends on the number of modes used for the reduced basis spaces. For any number of modes, the speedup is the highest for the open cavity test case with

the inconsistent flux method as it contains more degrees of freedom than the lid driven cavity case at full order level.

8

Conclusions and outlook

This dissertation aimed to present novel reduced order models for computational fluid dynamics and heat transfer problems. This chapter provides the overall conclusions of the research and an outlook for future work.

8.1 Overall conclusions

It has been demonstrated that the proposed non-intrusive POD-based identification method is capable of constructing a reduced order model that can be applied to linear transport problems for new boundary condition values of interest. The non-homogeneous time-(in)dependent Dirichlet boundary conditions are imposed in the ROM with a penalty method. The main shortcoming of the POD-ID method, at this stage, is that it is not feasible to use the method for nonlinear problems. A drawback is that the required number of snapshots scales with the cube of the number of modes. Moreover, at least as many reduced matrices are to be identified as the number of modes used.

As most (industrial) fluid flow and heat transfer problems are described by non-linear partial differential equations, the limitations of the non-intrusive method were not acceptable. Therefore, intrusive reduced order models have been developed using a finite volume based POD-Galerkin projection technique. This intrusive reduced order modeling technique requires the knowledge of the high fidelity solver's discretization and solution algorithm. Moreover, a boundary control method needs to be used to impose the non-homogeneous boundary conditions at

the reduced order level. Two boundary control methods have been tested: the lifting function method and the penalty method. Moreover, the penalty method has been improved by using an iterative solver for the determination of the penalty factors, rather than using numerical experimentation. The factors were determined by the iterative solver in about a second for both test cases. The ROM results show that the boundary control methods perform equally well and can be used for parametric boundary problems.

Thermal-hydraulic studies and other industrial applications require thermal buoyancy modeling for natural and mixed convection regimes. In this work, reduced order models have been developed for buoyancy-driven flows that include the Boussinesq approximation and energy equation. The additional buoyancy term in the ROM induces a two-way coupling between momentum and energy. Furthermore, a reduced order model has been developed for steady-state Reynolds-averaged Navier-Stokes simulations of low-Prandtl number fluid flow. Simulations are performed for sodium flow over a vertical backward-facing step with a heater placed on the wall directly downstream of the step. The results for different Richardson numbers show that buoyancy has a large influence on the flow and heat transfer. Even though the behavior is nonlinear, the reduced order model is capable of reproducing the RANS results with good accuracy also for new parameter values inside the range of Richardson numbers. Also, the local Stanton number and skin friction distribution at the heater are qualitatively well captured. The eddy viscosity and turbulence thermal diffusivity coefficients, needed for the turbulence closure modeling at the reduced order level, are approximated with the radial basis function interpolation method. The advantage of this method is that the reduced order model is independent of the turbulence model used in the RANS simulations.

The main challenges of reduced order modeling for fluid flow problems are related to (pressure) stabilization and the treatment of boundary conditions at reduced order level. Therefore, novel reduced order models of the incompressible Navier-Stokes equations on collocated grids have been developed using a ‘discretize-then-project’ approach. The formulations of the reduced order models are fully corresponding to the discrete full order model formulations. No pressure stabilization method is needed, even though the pressure term is present in the ROM. Moreover, the boundary conditions at the ROM level are imposed via the projection of the boundary vectors that are specified at the discrete FOM level. This is a big improvement compared to all other reduced order models presented in this dissertation that required a boundary control method, such as the penalty method or lifting function method, to impose the boundary conditions in the ROMs.

Finally, all developed reduced order models are efficient in terms of computational time and storage, due to the offline-online decomposition. The reduced

matrices for the linear terms and third order tensors for the nonlinear terms of all reduced order models were precomputed and stored during the offline stage. The reduced system of equations were solved during the online phase. The speedup factor, the ratio between the full order model and reduced order model simulation time, is highly problem dependent. The simulation time depends on the number of degrees of freedom at full and reduced order level, the complexity of the problem, e.g turbulence or buoyancy modeling, and on the pressure stabilization and boundary control methods used.

8.2 Suggestions for future work

The reduced order models developed in this study can be improved in several ways. In this outlook, we propose some suggestions for future investigations.

Progress has been made in the last part of the Ph.D. project with the work on a ‘discretize-then-project’ approach that requires no pressure stabilization technique nor a boundary control technique. First, the approach of projecting the full order boundary vectors containing the contributions of the boundary conditions needs to be extended to parametric (time-dependent) boundary conditions. The approach can then be applied to all other intrusive POD-Galerkin reduced order models developed in this dissertation. That way, boundary control methods such as the (iterative) penalty and lifting function method become redundant. Moreover, the ‘discretize-then-project’ approach should be extended to higher-order explicit and implicit time discretization methods to be widely applicable to industrial fluid flow and heat transfer problems. Then, the reduced order models developed with the ‘discretize-then-project’ approach can be compared to the reduced order models developed with the approach of exploiting a pressure Poisson equation during the projection stage in terms of performance and limitations.

Regards to (nuclear) engineering applications, it would be an asset to extend the reduced order model of turbulent convective buoyant flow of low-Prandtl number fluids for the parametrized unsteady RANS equations. In addition, neural networks [83], instead of using radial basis functions as an interpolation method, could potentially be used to approximate the eddy viscosity and thermal diffusion coefficients (for turbulence modeling) [112].

An interesting follow-up study would be to develop a reduced order model for unsteady flow and heat transfer of sodium in an outlet plenum [171]. Thereafter, it would be of interest to develop reduced order models based on transient simulations with the MyrrhaFOAM solver, which is currently being developed by the Von Karman Institute [143].

Lastly, this work has not been focused on determining the optimal reduced basis spaces. However, to improve the accuracy of the reduced order models, the effect of the variation of the number of basis functions needs to be investigated.

In addition, the sampling in time of the snapshot selection procedure needs to be optimized, which would also reduce the number of expensive high fidelity simulations that typically need to be performed in the offline phase. Also, the validity of the reduced order solutions away from the parametric snapshots can be improved by sampling the parameter spaces carefully. A priori error estimates and greedy strategies need to be investigated in order to optimize the snapshots selection procedures. Future work could include the use of (data-driven) techniques to adapt the ROM while the reduced order simulation proceeds [189, 202].

References

- [1] H. Aït Abderrahim. Multi-purpose hYbrid Research Reactor for High-tech Applications a multipurpose fast spectrum research reactor. *International Journal of Energy Research*, 36(15):1331–1337, 2012.
- [2] H. Aït Abderrahim. Contribution of the European Commission to a European strategy for HLW management through partitioning & transmutation. In *Nuclear Back-end and Transmutation Technology for Waste Disposal*, pages 59–71. Springer, 2015.
- [3] H. Aït Abderrahim, P. Baeten, D. De Bruyn, and R. Fernandez. MYRRHA—A multi-purpose fast spectrum research reactor. *Energy Conversion and Management*, 63:4–10, 2012.
- [4] H. Aït Abderrahim, P. Baeten, A. Sneyers, M. Schyns, P. Schuurmans, A. Kochetkov, G. Van den Eynde, and J.-L. Biarrotte. Partitioning and transmutation contribution of MYRRHA to an EU strategy for HLW management and main achievements of MYRRHA related FP7 and H2020 projects: MYRTE, MARISA, MAXSIMA, SEARCH, MAX, FREYA, ARCAS. *EPJ Nuclear Sciences & Technologies*, 6:33, 2020.
- [5] I. Akhtar, A. H. Nayfeh, and C. J. Ribbens. On the stability and extension of reduced-order Galerkin models in incompressible flows. *Theoretical and Computational Fluid Dynamics*, 23(3):213–237, 2009.
- [6] N. Anderson, Y. Hassan, and R. Schultz. Analysis of the hot gas flow in the outlet plenum of the very high temperature reactor using coupled RELAP5-3D system code and a CFD code. *Nuclear Engineering and Design*, 238(1):274–279, 2008.
- [7] F. Andreoli, D. Bisogni, F. Belloni, and G. Scheveneels. Influence of the 3-D phenomena on the safety parameters during a ULOF accident in the MYRRHA reactor. In *NURETH-16: International Topical Meeting on Nuclear Reactor Thermal Hydraulics*, pages 4703–4716, 2015.

- [8] M. Angelucci, D. Martelli, G. Barone, I. Di Piazza, and N. Forgione. STH-CFD codes coupled calculations applied to HLM loop and pool systems. *Science and Technology of Nuclear Installations*, 2017, 2017.
- [9] J. Annoni, P. Gebraad, and P. Seiler. Wind farm flow modeling using an input-output reduced-order model. In *American Control Conference*, pages 506–512. American Control Conference, IEEE, 2016.
- [10] J. Annoni and P. Seiler. A method to construct reduced-order parameter-varying models. *International Journal of Robust and Nonlinear Control*, 27(4):582–597, 2017.
- [11] ANSYS Fluent. 14.0 user’s manual. *ANSYS Inc., Canonsburg, PA*, 2011.
- [12] C. Argyropoulos and N. Markatos. Recent advances on the numerical modelling of turbulent flows. *Applied Mathematical Modelling*, 39(2):693–732, 2015.
- [13] N. Aubry, P. Holmes, J. L. Lumley, and E. Stone. The dynamics of coherent structures in the wall region of a turbulent boundary layer. *Journal of Fluid Mechanics*, 192:115–173, 1988.
- [14] C. Audouze, F. De Vuyst, and P. Nair. Nonintrusive reduced-order modeling of parametrized time-dependent partial differential equations. *Numerical Methods for Partial Differential Equations*, 29(5):1587–1628, 2013.
- [15] D. Aumiller, E. Tomlinson, and R. Bauer. A coupled RELAP5-3D/CFD methodology with a proof-of-principle calculation. *Nuclear Engineering and Design*, 205(1-2):83–90, 2001.
- [16] I. Babuška. The finite element method with penalty. *Mathematics of Computation*, 27(122):221–228, 1973.
- [17] J. Baiges, R. Codina, and S. Idelsohn. Explicit reduced-order models for the stabilized finite element approximation of the incompressible Navier–Stokes equations. *International Journal for Numerical Methods in Fluids*, 72(12):1219–1243, 2013.
- [18] J. Baiges, R. Codina, and S. R. Idelsohn. Reduced-order modelling strategies for the finite element approximation of the incompressible Navier–Stokes equations. In *Numerical Simulations of Coupled Problems in Engineering*, pages 189–216. Springer, 2014.
- [19] F. Ballarin, A. Manzoni, A. Quarteroni, and G. 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] G. Bandini, M. Polidori, A. Gerschenfeld, D. Pialla, S. Li, W. Ma, P. Kudinov, M. Jeltsov, K. Kööp, K. Huber, et al. Assessment of systems codes and their coupling with CFD codes in thermal–hydraulic applications to innovative reactors. *Nuclear Engineering and Design*, 281:22–38, 2015.
- [21] A. Barbagallo, D. Sipp, and P. J. Schmid. Closed-loop control of an open cavity flow using reduced-order models. *Journal of Fluid Mechanics*, 641:1–50, 2009.
- [22] C. Bardos. A Basic Example of Non Linear Equations: The Navier-Stokes Equations. *Mathematics: Concepts, and Foundations in Encyclopedia of Life Support Systems*, 2005.
- [23] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An ‘empirical interpolation’ method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, Nov. 2004.
- [24] P. Benner, S. Gugercin, and K. Willcox. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM review*, 57(4):483–531, 2015.
- [25] M. Bergmann, C.-H. Bruneau, and A. Iollo. Enablers for robust POD models. *Journal of Computational Physics*, 228(2):516–538, 2009.
- [26] G. Berkooz, P. Holmes, and J. L. Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual Review of Fluid Mechanics*, 25(1):539–575, 1993.
- [27] D. Bertolotto, A. Manera, S. Frey, H.-M. Prasser, and R. Chawla. Single-phase mixing studies by means of a directly coupled CFD/system-code tool. *Annals of Nuclear Energy*, 36(3):310–316, 2009.
- [28] A. Bērziņš, J. Helmig, F. Key, and S. Elgeti. Standardized non-intrusive reduced order modeling using different regression models with application to complex flow problems. *arXiv preprint arXiv:2006.13706*, 2020.
- [29] K. Bizon and G. Continillo. Reduced order modelling of chemical reactors with recycle by means of POD-penalty method. *Computers & Chemical Engineering*, 39:22–32, 2012.
- [30] J. Borggaard, A. Duggeby, A. Hay, T. Iliescu, and Z. Wang. Reduced-order modeling of turbulent flows. In *Proceedings of MTNS*, 2008.
- [31] G. Borgia, M. Gunzburger, and S. Manservigi. A distributed control approach for the boundary optimal control of the steady MHD equations. *Communications In Computational Physics*, 14(3):722–752, 2013.

- [32] O. Botella and R. Peyret. Benchmark spectral results on the lid-driven cavity flow. *Computers & Fluids*, 27(4):421–433, 1998.
- [33] F. Brezzi. On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers. *Publications Mathématiques et Informatique de Rennes*, (S4):1–26, 1974.
- [34] D. Brück, H. Elmqvist, S. E. Mattsson, and H. Olsson. Dymola for multi-engineering modeling and simulation. In *2nd International Modelica Conference*, pages 55–1–55–8, 2002.
- [35] T. Bui-Thanh, K. Willcox, O. Ghattas, and B. van Bloemen Waanders. Goal-oriented, model-constrained optimization for reduction of large-scale systems. *Journal of Computational Physics*, 224(2):880–896, 2007.
- [36] J. Buongiorno, J. E. Parsons, D. A. Petti, and J. Parsons. The future of nuclear energy in a carbon-constrained world. *IEEE Power and Energy Magazine*, 17:69–72, 2019.
- [37] J. Burkardt, M. Gunzburger, and H.-C. Lee. POD and CVT-based reduced-order modeling of Navier–Stokes flows. *Computer Methods in Applied Mechanics and Engineering*, 196(1-3):337–355, 2006.
- [38] A. Burns, I. Jones, J. Kightley, and N. Wilkes. *Harwell-FLOW3D, Release 2.1: User Manual*. Harwell, UK, 1988.
- [39] T. Bury. Coupling of CFD and lumped parameter codes for thermal-hydraulic simulations of reactor containment. *Computer Assisted Methods in Engineering and Science*, 20(3):195–206, 2017.
- [40] S. Busto, G. Stabile, G. Rozza, and M. E. Vázquez-Cendón. POD–Galerkin reduced order methods for combined Navier–Stokes transport equations based on a hybrid FV-FE solver. *Computers & Mathematics with Applications*, 79(2):256–273, 2020.
- [41] A. Caiazzo, T. Iliescu, V. John, and S. Schyschlowa. A numerical investigation of velocity–pressure reduced order models for incompressible flows. *Journal of Computational Physics*, 259:598–616, 2014.
- [42] C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang. *Spectral methods: fundamentals in single domains*. Springer Science & Business Media, 2007.
- [43] K. Carlberg, Y. Choi, and S. Sargsyan. Conservative model reduction for finite-volume models. *Journal of Computational Physics*, 371:280–314, 2018.

- [44] K. Carlberg, C. Farhat, J. Cortial, and D. 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.
- [45] D. Castelliti and G. Lomonaco. A preliminary stability analysis of MYRRHA Primary Heat Exchanger two-phase tube bundle. *Nuclear Engineering and Design*, 305:179–190, 2016.
- [46] W. Cazemier, R. Verstappen, and A. Veldman. Proper orthogonal decomposition and low-dimensional models for driven cavity flows. *Physics of Fluids*, 10(7):1685–1699, 1998.
- [47] CD-Adapco. STAR CCM+ user guide version 12.04. *CD-Adapco: New York, NY, USA*, 2017.
- [48] R. Chakir and Y. Maday. A two-grid finite-element/reduced basis scheme for the approximation of the solution of parameter dependent PDE. In *9e Colloque national en calcul des structures*. CSMA, 2009.
- [49] I. Change et al. Mitigation of climate change. *Contribution of Working Group III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change*, pages 1–11, 2014.
- [50] S. Chaturantabut and D. C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, 2010.
- [51] F. Chen, X. Huai, J. Cai, X. Li, and R. Meng. Investigation on the applicability of turbulent-Prandtl-number models for liquid lead-bismuth eutectic. *Nuclear Engineering and Design*, 257:128–133, 2013.
- [52] F. Chinesta, A. Huerta, G. Rozza, and K. Willcox. *Model Order Reduction*. Wiley Encyclopedia of Computational Mechanics, 2nd edition, 2016.
- [53] F. Chinesta, P. Ladeveze, and E. Cueto. A short review on model order reduction based on proper generalized decomposition. *Archives of Computational Methods in Engineering*, 18(4):395, 2011.
- [54] L. Chirco, A. Chierici, R. Da Vià, V. Giovacchini, and S. Manservigi. Optimal Control of the Wilcox turbulence model with lifting functions for flow injection and boundary control. *Journal of Physics: Conference Series*, 1224(1):012006, 2019.
- [55] A. J. Chorin. Numerical solution of the Navier–Stokes equations. *Mathematics of Computation*, 22(104):745–762, 1968.

- [56] M. Cotton and J. Jackson. Vertical tube air flows in the turbulent mixed convection regime calculated using a low-Reynolds-number $k-\epsilon$ model. *International Journal of Heat and Mass Transfer*, 33(2):275–286, 1990.
- [57] M. Couplet, P. Sagaut, and C. Basdevant. Intermodal energy transfers in a proper orthogonal decomposition-Galerkin representation of a turbulent separated flow. *Journal of Fluid Mechanics*, 491:275, 2003.
- [58] T. Craft, N. Ince, and B. Launder. Recent developments in second-moment closure for buoyancy-affected flows. *Dynamics of Atmospheres and Oceans*, 23(1-4):99–114, 1996.
- [59] X. Dai, P. Tang, and M. Wu. Analysis of an iterative penalty method for Navier–Stokes equations with nonlinear slip boundary conditions. *International Journal for Numerical Methods in Fluids*, 72(4):403–413, 2013.
- [60] A. Date. Solution of Navier-Stokes equations on non-staggered grid. *International Journal of Heat and Mass Transfer*, 36(7):1913–1922, 1993.
- [61] D. De Bruyn, H. Aït Abderrahim, P. Baeten, and P. Leysen. The MYRRHA ADS project in Belgium enters the front end engineering phase. *Physics Procedia*, 66:75–84, 2015.
- [62] D. De Bruyn, H. Aït Abderrahim, and M. Schyns. Recent progress and perspectives in the Belgian MYRRHA ADS programme. In *RRFM 2019 - European research reactor conference*, 2019.
- [63] J. Degroote, J. Vierendeels, and K. Willcox. Interpolation among reduced-order matrices to obtain parameterized models for design, optimization and probabilistic analysis. *International Journal for Numerical Methods in Fluids*, 63(2):207–230, 2010.
- [64] L. Demkowicz. *Computing with hp-adaptive finite elements: volume 1 one and two dimensional elliptic and Maxwell problems*. CRC press, 2006.
- [65] M. Dempsey. Dymola for Multi-Engineering Modelling and Simulation. In *2006 IEEE Vehicle Power and Propulsion Conference*, pages 1–6, 2006.
- [66] P. Denholm, J. C. King, C. F. Kutcher, and P. P. Wilson. Decarbonizing the electric sector: Combining renewable and nuclear energy using thermal storage. *Energy Policy*, 44:301–311, 2012.
- [67] H. Dol, K. Hanjalić, and T. Versteegh. A DNS-based thermal second-moment closure for buoyant convection at vertical walls. *Journal of Fluid Mechanics*, 391:211–247, 1999.

- [68] D. Drikakis, M. Hahn, A. Mosedale, and B. Thornber. Large eddy simulation using high-resolution and high-order methods. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 367(1899):2985–2997, 2009.
- [69] J. L. Eftang and E. M. Rønquist. Evaluation of flux integral outputs for the reduced basis method. *Mathematical Models and Methods in Applied Sciences*, 20(03):351–374, 2010.
- [70] Y. Epshteyn and B. Rivière. Estimation of penalty parameters for symmetric interior penalty Galerkin methods. *Journal of Computational and Applied Mathematics*, 206(2):843–872, 2007.
- [71] J. Y. Escanciano and A. G. Class. POD-Galerkin modeling of a heated pool. *Progress in Nuclear Energy*, 113:196 – 205, 2019.
- [72] R. Eymard, T. Gallouët, and R. Herbin. Finite volume methods. *Handbook of Numerical Analysis*, 7:713–1018, 2000.
- [73] C. Fazio, V. Sobolev, A. Aerts, S. Gavrilov, K. Lambrinou, P. Schuurmans, A. Gessi, P. Agostini, A. Ciampichetti, L. Martinelli, et al. Handbook on lead-bismuth eutectic alloy and lead properties, materials compatibility, thermal-hydraulics and technologies. Technical report, Organisation for Economic Co-Operation and Development, 2015.
- [74] F. N. Felten and T. S. Lund. Kinetic energy conservation issues associated with the collocated mesh scheme for incompressible flow. *Journal of Computational Physics*, 215(2):465–484, 2006.
- [75] T. Feng, D. Zhang, P. Song, W. Tian, W. Li, G. Su, and S. Qiu. Numerical research on water hammer phenomenon of parallel pump-valve system by coupling FLUENT with RELAP5. *Annals of Nuclear Energy*, 109:318 – 326, 2017.
- [76] R. Fernandez, D. De Bruyn, P. Baeten, and H. Ait Abderrahim. The evolution of the primary system design of the MYRRHA facility. In *FR17 - IAEA International Conference on Fast Reactors and Related Fuel Cycles: Next Generation Nuclear Systems for Sustainable Development*. IAEA-CN245-358, 2017.
- [77] J. H. Ferziger and M. Perić. *Computational methods for fluid dynamics*, volume 3. Springer, 2002.
- [78] L. Fick, Y. Maday, A. T. Patera, and T. Taddei. A reduced basis technique for long-time unsteady turbulent flows. *arXiv preprint arXiv:1710.03569*, 2017.

- [79] L. Fick, Y. Maday, A. T. Patera, and T. Taddei. A stabilized POD model for turbulent flows over a range of Reynolds numbers: Optimal parameter sampling and constrained projection. *Journal of Computational Physics*, 371:214–243, 2018.
- [80] C. Fiorina, I. Clifford, M. Aufiero, and K. Mikityuk. GeN-Foam: a novel OpenFOAM® based multi-physics solver for 2D/3D transient analysis of nuclear reactors. *Nuclear Engineering and Design*, 294:24–37, 2015.
- [81] C. Fletcher. *Computational Techniques for Fluid Dynamics Vol. I. Fundamental and General Techniques*. Springer-Verlag, 2nd edition, 1991.
- [82] E. Fonn, H. van Brummelen, T. Kvamsdal, and A. Rasheed. Fast divergence-conforming reduced basis methods for steady Navier–Stokes flow. *Computer Methods in Applied Mechanics and Engineering*, 346:486–512, 2019.
- [83] M. Frank, D. Drikakis, and V. Charissis. Machine-learning methods for computational science and engineering. *Computation*, 8(1):15, 2020.
- [84] P. Fritzson and V. Engelson. Modelica - A unified object-oriented language for system modeling and simulation. In *European Conference on Object-Oriented Programming*, pages 67–90. Springer, 1998.
- [85] O. Garbrecht. *Large eddy simulation of three-dimensional mixed convection on a vertical plate*. PhD thesis, RWTH Aachen University, 2017.
- [86] C. J. Geankoplis. *Transport processes and separation process principles (includes unit operations)*. Prentice Hall Professional Technical Reference, 2003.
- [87] S. Georgaka, G. Stabile, G. Rozza, and M. J. Bluck. Parametric POD-Galerkin model order reduction for unsteady-state heat transfer problems. *Communications in Computational Physics*, 27(1):1–32, 2018.
- [88] S. Georgaka, G. Stabile, K. Star, G. Rozza, and M. J. Bluck. A hybrid reduced order method for modelling turbulent heat transfer problems. *Computers & Fluids*, page 104615, 2020.
- [89] H. Gibeling and J. Mahaffy. Benchmarking simulations with CFD to 1-D coupling. In *Joint IAEA/OECD Technical Meeting on Use of CFD Codes for Safety Analysis of Reactor Systems, Including Containment*, 2002.
- [90] V. Girault and L. R. Scott. Analysis of a two-dimensional grade-two fluid model with a tangential boundary condition. *Journal de Mathématiques Pures et Appliquées*, 78(10):981–1011, 1999.

- [91] M. Girfoglio, A. Quaini, and G. Rozza. A Finite Volume approximation of the Navier-Stokes equations with nonlinear filtering stabilization. *Computers & Fluids*, 187:27–45, 2019.
- [92] K. Gladinez, K. Rosseel, J. Lim, A. Marino, G. Heynderickx, and A. Aerts. Formation and transport of lead oxide in a non-isothermal lead-bismuth eutectic loop. *Nuclear Engineering and Design*, 349:78–85, 2019.
- [93] W. Graham, J. Peraire, and K. Tang. Optimal control of vortex shedding using low-order models. Part I - open-loop model development. *International Journal for Numerical Methods in Engineering*, 44(7):945–972, 1999.
- [94] S. Grape, S. Jacobsson Svård, C. Hellesen, P. Jansson, and M. Åberg Lindell. New perspectives on nuclear power-Generation IV nuclear energy systems to strengthen nuclear non-proliferation and support nuclear disarmament. *Energy Policy*, 73:815 – 819, 2014.
- [95] C. J. Greenshields. *OpenFOAM User Guide version 6*. OpenFOAM Foundation Ltd, 2018.
- [96] M. A. Grepl, Y. Maday, N. C. Nguyen, and A. T. Patera. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 41(3):575–605, 2007.
- [97] P. M. Gresho and R. L. Sani. *Incompressible flow and the finite element method. Volume 1: Advection-diffusion and isothermal laminar flow*. John Wiley and Sons, Inc., New York, NY (United States), 1998.
- [98] D. Grgić, N. Cavlina, T. Bajš, L. Oriani, and L. Conway. Coupled RELAP5/GOTHIC model for IRIS SBLOCA analysis. In *5th International Conference on Nuclear Option in Countries with Small and Medium Electricity Grids, Dubrovnik, Croatia*, 2004.
- [99] S. Grimberg, C. Farhat, and N. Youkilis. On the stability of projection-based model order reduction for convection-dominated laminar and turbulent flows. *Journal of Computational Physics*, 419:109681, 2020.
- [100] G. Grötzbach. Challenges in low-Prandtl number heat transfer simulation and modelling. *Nuclear Engineering and Design*, 264:41–55, 2013.
- [101] G. Grötzbach and M. Wörner. Direct numerical and large eddy simulations in nuclear applications. *International Journal of Heat and Fluid Flow*, 20(3):222–240, 1999.

- [102] T. P. Grunloh and A. Manera. A novel domain overlapping strategy for the multiscale coupling of CFD with 1D system codes with applications to transient flows. *Annals of Nuclear Energy*, 90:422–432, 2016.
- [103] M. D. Gunzburger, J. S. Peterson, and J. N. Shadid. Reduced-order modeling of time-dependent PDEs with multiple parameters in the boundary data. *Computer Methods in Applied Mechanics and Engineering*, 196(4-6):1030–1047, 2007.
- [104] M. Guo and J. S. Hesthaven. Data-driven reduced order modeling for time-dependent problems. *Computer Methods in Applied Mechanics and Engineering*, 345:75–99, 2019.
- [105] B. Haasdonk and M. Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42(2):277–302, 2008.
- [106] K. Hanjalić, S. Kenjereš, and F. Durst. Natural convection in partitioned two-dimensional enclosures at higher Rayleigh numbers. *International Journal of Heat and Mass Transfer*, 39(7):1407–1427, 1996.
- [107] K. Hanjalić, S. Kenjereš, M. Tummers, and H. Jonker. *Analysis and modelling of physical transport phenomena*. VSSD, 2007.
- [108] F. H. Harlow and J. E. Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *The Physics of Fluids*, 8(12):2182–2189, 1965.
- [109] M. W. Hess, A. Alla, A. Quaini, G. Rozza, and M. Gunzburger. A localized reduced-order modeling approach for PDEs with bifurcating solutions. *Computer Methods in Applied Mechanics and Engineering*, 351:379 – 403, 2019.
- [110] J. Hesthaven, G. Rozza, and B. Stamm. *Certified reduced basis methods for parametrized partial differential equations*. Springer International Publishing, 1st edition, 2016.
- [111] S. Hijazi, S. Ali, G. Stabile, F. Ballarin, and G. Rozza. The effort of increasing Reynolds number in projection-based reduced order methods: from laminar to turbulent flows. *Lecture Notes in Computational Science and Engineering*, 136, 2019.
- [112] S. Hijazi, G. Stabile, A. Mola, and G. Rozza. Data-driven POD-Galerkin reduced order model for turbulent flows. *Journal of Computational Physics*, 416:109513, 2020.

- [113] S. Hijazi, G. Stabile, A. Mola, and G. Rozza. Non-intrusive polynomial chaos method applied to full-order and reduced problems in computational fluid dynamics: a comparison and perspectives. In *Quantification of Uncertainty: Improving Efficiency and Technology*, pages 217–240. Springer, 2020.
- [114] C. Hirsch. *Numerical computation of internal and external flows: The fundamentals of computational fluid dynamics*. Elsevier, 2007.
- [115] P. Holmes, J. L. Lumley, G. Berkooz, and C. W. Rowley. *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge university press, 2012.
- [116] T. Holzmann. *Mathematics, numerics, derivations and OpenFOAM®*. Leoben, 2017.
- [117] L. Hou and S. Ravindran. Numerical approximation of optimal flow control problems by a penalty method: Error estimates and numerical results. *SIAM Journal on Scientific Computing*, 20(5):1753–1777, 1999.
- [118] K. Hsieh and F. Lien. Numerical modeling of buoyancy-driven turbulent flows in enclosures. *International Journal of Heat and Fluid Flow*, 25(4):659–670, 2004.
- [119] R. Hu and T. Fanning. Progress report on development of intermediate fidelity full assembly analysis methods. Technical report, Argonne National Lab, Argonne, IL (United States), 2011.
- [120] C. Huang, K. Duraisamy, and C. L. Merkle. Investigations and improvement of robustness of reduced-order models of reacting flow. *AIAA Journal*, 57(12):5377–5389, 2019.
- [121] Z. Huang and W. Ma. Performance evaluation of passive containment cooling system of an advanced PWR using coupled RELAP5/GOTHIC simulation. *Nuclear Engineering and Design*, 310:83–92, 2016.
- [122] C. Huettel and A. Matzenmiller. Minimum penalty parameter for given constraint violation tolerance. In *European Congress on Computational Methods in Applied Sciences and Engineering - ECCOMAS 2000*, 2000.
- [123] L. Iapichino, A. Quarteroni, and G. Rozza. Reduced basis method and domain decomposition for elliptic problems in networks and complex parametrized geometries. *Computers & Mathematics with Applications*, 71(1):408–430, 2016.

- [124] T. Iliescu, H. Liu, and X. Xie. Regularized reduced order models for a stochastic Burgers equation. *International Journal of Numerical Analysis and Modeling*, 15, 07 2016.
- [125] T. Iliescu and Z. Wang. Variational multiscale proper orthogonal decomposition: Navier-stokes equations. *Numerical Methods for Partial Differential Equations*, 30(2):641–663, 2014.
- [126] N. Ince and B. Launder. On the computation of buoyancy-driven turbulent flows in rectangular enclosures. *International Journal of Heat and Fluid Flow*, 10(2):110–117, 1989.
- [127] A. Iollo, S. Lanteri, and J.-A. Désidéri. Stability properties of POD–Galerkin approximations for the compressible Navier–Stokes equations. *Theoretical and Computational Fluid Dynamics*, 13(6):377–396, 2000.
- [128] R. I. Issa. Solution of the implicitly discretised fluid flow equations by operator-splitting. *Journal of Computational Physics*, 62(1):40–65, 1986.
- [129] H. Jasak. *Error analysis and estimation for the finite volume method with applications to fluid flows*. PhD thesis, Imperial College London (University of London), 1996.
- [130] H. Jasak, A. Jemcov, and Ž. Tuković. OpenFOAM: A C++ library for complex physics simulations. International workshop on coupled methods in numerical dynamics, IUC, Dubrovnik, Croatia, 2007.
- [131] M. Jeltsov, K. Kööp, P. Kudinov, and W. Villanueva. Development of a domain overlapping coupling methodology for STH/CFD analysis of heavy liquid metal thermal-hydraulics. In *15th International Topical Meeting on Nuclear Reactor Thermal Hydraulics (NURETH-15)*, 2013.
- [132] J.-J. Jeong, S. K. Sim, and S. Y. Lee. Development and assessment of the COBRA/RELAP5 code. *Journal of Nuclear Science and Technology*, 34(11):1087–1098, 1997.
- [133] I. Kalashnikova and M. Barone. Stable and efficient Galerkin reduced order models for non-linear fluid flow. In *6th AIAA Theoretical Fluid Mechanics Conference*, page 3110, 2011.
- [134] I. Kalashnikova and M. Barone. Efficient non-linear proper orthogonal decomposition/Galerkin reduced order models with stable penalty enforcement of boundary conditions. *International Journal for Numerical Methods in Engineering*, 90(11):1337–1362, 2012.

- [135] W. M. Kays. Turbulent Prandtl number—where are we? *Journal of Heat Transfer*, 116(2):284–295, 1994.
- [136] V. Kazemi-Kamyab, A. Van Zuijlen, and H. Bijl. Analysis and application of high order implicit Runge–Kutta schemes to collocated finite volume discretization of the incompressible Navier–Stokes equations. *Computers & Fluids*, 108:107–115, 2015.
- [137] K. Kean and M. Schneier. Error analysis of supremizer pressure recovery for POD based reduced-order models of the time-dependent Navier–Stokes equations. *SIAM Journal on Numerical Analysis*, 58(4):2235–2264, 2020.
- [138] H. J. Kelley. Method of gradients. In *Mathematics in Science and Engineering*, volume 5, pages 205–254. Elsevier, 1962.
- [139] G. Kennedy, K. Van Tichelen, and H. Doolaard. Experimental investigation of the pressure loss characteristics of the full-scale MYRRHA fuel bundle in the COMPLIT LBE facility. In *NURETH-16: International Topical Meeting on Nuclear Reactor Thermal Hydraulics*, page 61, 2015.
- [140] C. M. Klaij. On the stabilization of finite volume methods with co-located variables for incompressible flow. *Journal of Computational Physics*, 297:84–89, 2015.
- [141] D. J. Knezevic, N.-C. Nguyen, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for the parametrized unsteady Boussinesq equations. *Mathematical Models and Methods in Applied Sciences*, 21(07):1415–1442, 2011.
- [142] A. N. Kolmogorov. The local structure of turbulence in incompressible viscous fluid for very large Reynolds numbers. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, 434(1890):9–13, 1991.
- [143] L. Koloszar, S. Buckingham, P. Planquart, and S. Keijers. MyrrhaFoam: A CFD model for the study of the thermal hydraulic behavior of MYRRHA. *Nuclear Engineering and Design*, 312:256–265, 2017.
- [144] E. Komen, E. Frederix, T. Coppen, V. D’Alessandro, and J. Kuerten. Analysis of the numerical dissipation rate of different Runge–Kutta and velocity interpolation methods in an unstructured collocated finite volume method in OpenFOAM®. *Computer Physics Communications*, page 107145, 2020.
- [145] K. Kunisch and S. Volkwein. Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics. *SIAM Journal on Numerical Analysis*, 40(2):492–515, 2002.

- [146] K. Kunisch and S. Volkwein. Optimal snapshot location for computing POD basis functions. *ESAIM: Mathematical Modelling and Numerical Analysis*, 44(3):509–529, 2010.
- [147] J. N. Kutz, S. L. Brunton, B. W. Brunton, and J. L. Proctor. *Dynamic mode decomposition: data-driven modeling of complex systems*. SIAM, 2016.
- [148] I. Lasiecka. Ritz–Galerkin approximation of the time optimal boundary control problem for parabolic systems with Dirichlet boundary conditions. *SIAM Journal on Control and Optimization*, 22(3):477–500, 1984.
- [149] T. Lassila, A. Manzoni, A. Quarteroni, and G. Rozza. Model order reduction in fluid dynamics: challenges and perspectives. In *Reduced Order Methods for modeling and computational reduction*, pages 235–273. Berlin: Springer, 2014.
- [150] B. E. Launder and B. Sharma. Application of the energy-dissipation model of turbulence to the calculation of flow near a spinning disc. *Letters in Heat and Mass Transfer*, 1(2):131–137, 1974.
- [151] D. Lazzaro and L. B. Montefusco. Radial basis functions for the multivariate interpolation of large scattered data sets. *Journal of Computational and Applied Mathematics*, 140(1-2):521–536, 2002.
- [152] K. Lee and K. T. Carlberg. Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders. *Journal of Computational Physics*, 404:108973, 2020.
- [153] M. W. Lee and E. H. Dowell. On the Importance of Numerical Error in Constructing POD-based Reduced-Order Models of Nonlinear Fluid Flows. In *AIAA Scitech 2020 Forum*. AIAA Scitech 2020 Forum, 2020.
- [154] S. Y. Lee, J. J. Jeong, S.-H. Kim, and S. H. Chang. COBRA/RELAP5: a merged version of the COBRA-TF and RELAP5/MOD3 codes. *Nuclear Technology*, 99(2):177–187, 1992.
- [155] G. Leitmann. *Optimization techniques: with applications to aerospace systems*, volume 5. Academic Press, 1962.
- [156] W. Leysen, D. Terentyev, and A. Stankovskiy. Fusion target station on MYRRHA facility: Baseline concept. *Journal of Nuclear Materials*, 538:152242, 2020.
- [157] W. Li, X. Wu, D. Zhang, G. Su, W. Tian, and S. Qiu. Preliminary study of coupling CFD code FLUENT and system code RELAP5. *Annals of Nuclear Energy*, 73:96–107, 2014.

- [158] J. Lions and E. Magenes. *Non-homogeneous boundary value problems and applications*, volume 1. Springer, 1973.
- [159] J.-G. Liu, J. Liu, and R. L. Pego. Stable and accurate pressure approximation for unsteady incompressible viscous flow. *Journal of Computational Physics*, 229(9):3428–3453, 2010.
- [160] G. Locatelli, M. Mancini, and N. Todeschini. Generation IV nuclear reactors: Current status and future prospects. *Energy Policy*, 61:1503 – 1520, 2013.
- [161] S. Lorenzi, A. Cammi, L. Luzzi, and G. Rozza. POD-Galerkin method for finite volume approximation of Navier–Stokes and RANS equations. *Computer Methods in Applied Mechanics and Engineering*, 311:151–179, 2016.
- [162] P. Lorusso, S. Bassini, A. Del Nevo, I. Di Piazza, F. Giannetti, M. Tarantino, and M. Utili. GEN-IV LFR development: status & perspectives. *Progress in Nuclear Energy*, 105:318–331, 2018.
- [163] D. J. Lucia, P. S. Beran, and W. A. Silva. Reduced-order modeling: new approaches for computational physics. *Progress in Aerospace Sciences*, 40(1-2):51–117, 2004.
- [164] J. L. Lumley. Coherent structures in turbulence. In *Transition and turbulence*, pages 215–242. Elsevier, 1981.
- [165] X. Ma and G. E. Karniadakis. A low-dimensional model for simulating three-dimensional cylinder flow. *Journal of Fluid Mechanics*, 458:181–190, 2002.
- [166] M. H. Malik. *Reduced Order Modeling for Smart Grids Simulation and Optimization*. PhD thesis, École centrale de Nantes; Universitat politècnica de Catalunya, 2017.
- [167] G. D. Mallinson and S. E. Norris. 6 Fundamentals of Computational Fluid Dynamics. *Mathematical Modeling of Food Processing*, page 125, 2010.
- [168] R. Manceau, S. Parneix, and D. Laurence. Turbulent heat transfer predictions using the v_2 - f model on unstructured meshes. *International Journal of Heat and Fluid Flow*, 21(3):320–328, 2000.
- [169] A. Marino, J. Lim, S. Keijers, J. Deconinck, and A. Aerts. Numerical modeling of oxygen mass transfer in a wire wrapped fuel assembly under flowing lead bismuth eutectic. *Journal of Nuclear Materials*, 506:53–62, 2018.

- [170] A. Marino, J. Lim, S. Keijers, J. Van den Bosch, J. Deconinck, F. Rubio, K. Woloshun, M. Caro, and S. Maloy. Temperature dependence of dissolution rate of a lead oxide mass exchanger in lead–bismuth eutectic. *Journal of Nuclear Materials*, 450(1-3):270–277, 2014.
- [171] N. C. Markatos. Transient flow and heat transfer of liquid sodium coolant in the outlet plenum of a fast nuclear reactor. *International Journal of Heat and Mass Transfer*, 21(12):1565–1579, 1978.
- [172] N. C. Markatos and K. Pericleous. Laminar and turbulent natural convection in an enclosed cavity. *International Journal of Heat and Mass Transfer*, 27(5):755–772, 1984.
- [173] MATLAB. *Release 2017a*. The MathWorks Inc., 2017.
- [174] H. G. Matthies and J. Steindorf. Strong coupling methods. In *Analysis and simulation of multifield problems*, pages 13–36. Springer, 2003.
- [175] V. Moreau, M. Profir, A. Alemberti, et al. Pool CFD modelling: lessons from the SESAME project. *Nuclear Engineering and Design*, 355:110343, 2019.
- [176] Y. Morinishi, T. S. Lund, O. V. Vasilyev, and P. Moin. Fully conservative higher order finite difference schemes for incompressible flow. *Journal of Computational Physics*, 143(1):90–124, 1998.
- [177] F. Moukalled, L. Mangani, M. Darwish, et al. The finite volume method in computational fluid dynamics. *An Advanced Introduction with OpenFOAM and Matlab*, pages 3–8, 2016.
- [178] M. Niemann and J. Fröhlich. Buoyancy-affected backward-facing step flow with heat transfer at low prandtl number. *International Journal of Heat and Mass Transfer*, 101:1237–1250, 2016.
- [179] M. Niemann and J. Fröhlich. Turbulence budgets in buoyancy-affected vertical backward-facing step flow at low Prandtl number. *Flow, Turbulence and Combustion*, 99(3-4):705–728, 2017.
- [180] B. R. Noack, P. Papas, and P. A. Monkewitz. The need for a pressure-term representation in empirical Galerkin models of incompressible shear flows. *Journal of Fluid Mechanics*, 523:339, 2005.
- [181] B. Nour-Omid and P. Wriggers. A note on the optimum choice for penalty parameters. *Communications in Applied Numerical Methods*, 3(6):581–585, 1987.

- [182] Nuclear Safety Analysis Division. RELAP5/MOD3.3 Code Manual. Technical report, Idaho Falls: Information Systems Laboratories, Inc., 2003.
- [183] J. Oder, A. Shams, L. Cizelj, and I. Tiselj. Direct numerical simulation of low-Prandtl fluid flow over a confined backward facing step. *International Journal of Heat and Mass Transfer*, 142:118436, 2019.
- [184] OECD/NEA. State-of-the-art report on containment thermal-hydraulics and hydrogen distribution, 1999.
- [185] J. Östh, B. R. Noack, S. Krajnović, D. Barros, and J. Borée. On the need for a nonlinear subscale turbulence term in POD models as exemplified for a high-Reynolds-number flow over an Ahmed body. *Journal of Fluid Mechanics*, 747:518–544, 2014.
- [186] S. Patankar. *Numerical heat transfer and fluid flow*. Taylor & Francis, 1980.
- [187] S. V. Patankar and D. B. Spalding. A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. In *Numerical Prediction of Flow, Heat Transfer, Turbulence and Combustion*, pages 54–73. Elsevier, 1983.
- [188] A. Patera, D. Rovas, and L. Machiels. Reduced-basis output-bound methods for elliptic partial differential equations. *SIAG/OPT Views-and-News*, 11(1):1–6, 2000.
- [189] B. Peherstorfer and K. Willcox. Dynamic data-driven reduced-order models. *Computer Methods in Applied Mechanics and Engineering*, 291:21–41, 2015.
- [190] M. Perić, R. Kessler, and G. Scheuerer. Comparison of finite-volume numerical methods with staggered and collocated grids. *Computers & Fluids*, 16(4):389–403, 1988.
- [191] R. Petrova. *Finite Volume Method: Powerful Means of Engineering Design*. BoD–Books on Demand, 2012.
- [192] A. Petruzzi and F. D’Auria. Thermal-hydraulic system codes in nuclear reactor safety and qualification procedures. *Science and Technology of Nuclear Installations*, 2008, 2008.
- [193] P. Phalippou, S. Bouabdallah, P. Breitkopf, P. Villon, and M. Zarroug. ‘On-the-fly’ snapshots selection for Proper Orthogonal Decomposition with application to nonlinear dynamics. *Computer Methods in Applied Mechanics and Engineering*, 367:113120, 2020.

- [194] D. Pialla, D. Tenchine, S. Li, P. Gauthé, A. Vasile, R. Baviere, N. Tauveron, F. Perdu, L. Maas, F. Cocheme, K. Huber, and X. Cheng. Overview of the system alone and system/CFD coupled calculations of the PHENIX Natural Circulation Test within the THINS project. *Nuclear Engineering and Design*, 290:78 – 86, 2015. Thermal-Hydraulics of Innovative Nuclear Systems.
- [195] M. Piller and E. Stalio. Finite-volume compact schemes on staggered grids. *Journal of Computational Physics*, 197(1):299–340, 2004.
- [196] I. Pioro. *Handbook of generation IV nuclear reactors*. Woodhead Publishing, 2016.
- [197] A. Placzek, D.-M. Tran, and R. Ohayon. Hybrid proper orthogonal decomposition formulation for linear structural dynamics. *Journal of Sound and Vibration*, 318(4-5):943–964, 2008.
- [198] C. Poussot-Vassal and D. Sipp. Parametric reduced order dynamical model construction of a fluid flow control problem. *IFAC-PapersOnLine*, 48(26):133–138, 2015.
- [199] D. P. Prill and A. G. Class. Non-linear Proper Orthogonal Decomposition Reduced Order Modeling: Bifurcation study of a natural convection circuit. *PAMM*, 13(1):285–286, 2013.
- [200] D. P. Prill, M. Stokmaier, and A. G. Class. Methodology of ROMing applying POD of BWR fuel assemblies applied to introductory examples. *PAMM*, 12(1):237–238, 2012.
- [201] C. Prud’Homme, D. V. Rovas, K. Veroy, L. Machiels, Y. Maday, A. T. Patera, and G. Turinici. Reliable real-time solution of parametrized partial differential equations: Reduced-basis output bound methods. *Journal of Fluids Engineering*, 124(1):70–80, 2002.
- [202] L. Pyta and D. Abel. Online model adaption of reduced order models for fluid flows. *IFAC-PapersOnLine*, 50(1):11138–11143, 2017.
- [203] A. Quarteroni, A. Manzoni, and F. Negri. *Reduced basis methods for partial differential equations: an introduction*, volume 92. Springer, 2015.
- [204] A. Quarteroni and G. Rozza. Numerical solution of parametrized Navier–Stokes equations by reduced basis methods. *Numerical Methods for Partial Differential Equations: An International Journal*, 23(4):923–948, 2007.
- [205] A. Quarteroni and G. Rozza. *Reduced order methods for modeling and computational reduction*, volume 9 of *MS&A*. Springer, 1st edition, 2014.

- [206] M.-L. Rapún and J. M. Vega. Reduced order models based on local POD plus Galerkin projection. *Journal of Computational Physics*, 229(8):3046–3063, 2010.
- [207] S. S. Ravindran. A reduced-order approach for optimal control of fluids using proper orthogonal decomposition. *International Journal for Numerical Methods in Fluids*, 34(5):425–448, 2000.
- [208] C. Rhie and W. L. Chow. Numerical study of the turbulent flow past an airfoil with trailing edge separation. *AIAA Journal*, 21(11):1525–1532, 1983.
- [209] L. F. Richardson. *Weather prediction by numerical process*. Cambridge university press, 2007.
- [210] W. Rodi. Examples of turbulence models for incompressible flows. *AIAA Journal*, 20(7):872–879, 1982.
- [211] W. Rodi and G. Scheuerer. Scrutinizing the k - ε turbulence model under adverse pressure gradient conditions. *Journal of Fluids Engineering*, 108(2):174–179, 1986.
- [212] F. Roelofs, V. R. Gopala, S. Jayaraju, A. Shams, and E. Komen. Review of fuel assembly and pool thermal hydraulics for fast reactors. *Nuclear Engineering and Design*, 265:1205–1222, 2013.
- [213] F. Roelofs, A. Shams, I. Otic, M. Böttcher, M. Duponcheel, Y. Bartosiewicz, D. Lakehal, E. Baglietto, S. Lardeau, and X. Cheng. Status and perspective of turbulence heat transfer modelling for the industrial application of liquid metal flows. *Nuclear Engineering and Design*, 290:99–106, 2015.
- [214] F. Roelofs, A. Shams, J. Pacio, et al. European outlook for LMFR thermal hydraulics. In *NURETH-16: International Topical Meeting on Nuclear Reactor Thermal Hydraulics*, 2015.
- [215] C. W. Rowley. Model reduction for fluids, using balanced proper orthogonal decomposition. *International Journal of Bifurcation and Chaos*, 15(03):997–1013, 2005.
- [216] G. Rozza. Reduced basis approximation and error bounds for potential flows in parametrized geometries. *Communications in Computational Physics*, 9(1):1–48, 2011.
- [217] G. Rozza, D. B. P. Huynh, and A. 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, 2008.

- [218] G. Rozza, D. P. Huynh, and A. Manzoni. Reduced basis approximation and a posteriori error estimation for Stokes flows in parametrized geometries: roles of the inf-sup stability constants. *Numerische Mathematik*, 125(1):115–152, 2013.
- [219] G. Rozza, M. Malik, N. Demo, M. Tezzele, M. Girfoglio, G. Stabile, and A. Mola. Advances in reduced order methods for parametric industrial problems in computational fluid dynamics. In *ECCOMAS Proceedings, Glasgow, UK*, 2018.
- [220] G. Rozza and A. Manzoni. Model order reduction by geometrical parametrization for shape optimization in computational fluid dynamics. In *Proceedings of the ECCOMAS CFD 2010, V European Conference on Computational Fluid Dynamics*, 2010.
- [221] G. Rozza and K. Veroy. On the stability of the reduced basis method for Stokes equations in parametrized domains. *Computer methods in applied mechanics and engineering*, 196(7):1244–1260, 2007.
- [222] S. Rubino. Numerical analysis of a projection-based stabilized POD-ROM for incompressible flows. *SIAM Journal on Numerical Analysis*, 58(4):2019–2058, 2020.
- [223] B. Sanderse. Non-linearly stable reduced-order models for incompressible flow with energy-conserving finite volume methods. *Journal of Computational Physics*, 421:109736, 2020.
- [224] B. Sanderse and B. Koren. Accuracy analysis of explicit Runge–Kutta methods applied to the incompressible Navier–Stokes equations. *Journal of Computational Physics*, 231(8):3041–3063, 2012.
- [225] M. Sarotto, D. Castelliti, R. Fernandez, D. Lamberts, E. Malambu, A. Stankovskiy, W. Jaeger, M. Ottolini, F. Martin-Fuertes, L. Sabathé, et al. The MYRRHA-FASTEF cores design for critical and sub-critical operational modes (EU FP7 Central Design Team project). *Nuclear Engineering and Design*, 265:184–200, 2013.
- [226] P. J. Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656:5–28, 2010.
- [227] R. Schreiber and H. B. Keller. Driven cavity flows by efficient numerical techniques. *Journal of Computational Physics*, 49(2):310–333, 1983.
- [228] R. R. Schultz and W. L. Weaver. Coupling the RELAP5-3d advanced systems analysis code with commercial and advanced CFD software. Technical report, International Atomic Energy Agency (IAEA), 2003.

- [229] T. Schumm, B. Frohnapfel, and L. Marocco. Numerical simulation of the turbulent convective buoyant flow of sodium over a backward-facing step. *Journal of Physics: Conference Series*, 745(3):032051, 2016.
- [230] T. Schumm, B. Frohnapfel, and L. Marocco. Investigation of a turbulent convective buoyant flow of sodium over a backward-facing step. *Heat and Mass Transfer*, 54(8):2533–2543, 2018.
- [231] B. Shapiro. Creating compact models of complex electronic systems: An overview and suggested use of existing model reduction and experimental system identification tools. *IEEE Transactions on Components and Packaging Technologies*, 26(1):165–172, 2003.
- [232] J.-p. Simoneau, J. Champigny, and O. Gelineau. Applications of large eddy simulations in nuclear field. *Nuclear Engineering and Design*, 240(2):429–439, 2010.
- [233] D. Sipp and A. Lebedev. Global stability of base and mean flows: a general approach and its applications to cylinder and open cavity flows. *Journal of Fluid Mechanics*, 593:333–358, 2007.
- [234] S. Sirisup and G. Karniadakis. Stability and accuracy of periodic flow solutions obtained by a POD-penalty method. *Physica D: Nonlinear Phenomena*, 202(3-4):218–237, 2005.
- [235] L. Sirovich. Turbulence and the dynamics of coherent structures. I. Coherent structures. *Quarterly of Applied Mathematics*, 45(3):561–571, 1987.
- [236] B. Smith, M. Andreani, U. Bieder, F. Ducros, D. Bestion, E. Graffard, M. Heitsch, M. Scheuerer, M. Henriksson, T. Hoehne, et al. Assessment of CFD Codes for Nuclear Reactor Safety Problems-Revision 2. Technical report, Organisation for Economic Co-Operation and Development, 2015.
- [237] B. Smith, P. Bjorstad, and W. Gropp. *Domain decomposition: parallel multilevel methods for elliptic partial differential equations*. Cambridge university press, 2004.
- [238] G. Stabile, F. Ballarin, G. Zuccarino, and G. Rozza. A reduced order variational multiscale approach for turbulent flows. *Advances in Computational Mathematics*, 45:2349–2368, 2019.
- [239] G. Stabile, S. Hijazi, A. Mola, S. Lorenzi, and G. Rozza. POD-Galerkin reduced order methods for CFD using Finite Volume Discretisation: vortex shedding around a circular cylinder. *Communications in Applied and Industrial Mathematics*, 8(1):210–236, 2017.

- [240] G. Stabile and G. Rozza. ITHACA-FV - In real Time Highly Advanced Computational Applications for Finite Volumes. www.mathlab.sissa.it/ithaca-fv. Accessed 2020-10-16.
- [241] G. Stabile and G. Rozza. Finite volume POD-Galerkin stabilised reduced order methods for the parametrised incompressible Navier–Stokes equations. *Computers & Fluids*, 173:273–284, 2018.
- [242] G. Stabile, M. Zancanaro, and G. Rozza. Efficient geometrical parametrization for finite-volume-based reduced order methods. *International Journal for Numerical Methods in Engineering*, 121(12):2655–2682, 2020.
- [243] K. Star, J. Degroote, J. Vierendeels, G. Van den Eynde, and F. Belloni. Reduced order modelling using a POD-based identification method for parameterized PDEs. In *7th European Conference on Computational Fluid Dynamics*, 2018.
- [244] K. Star, G. Stabile, S. Georgaka, F. Belloni, G. Rozza, and J. Degroote. POD-Galerkin reduced order model of the Boussinesq approximation for buoyancy-driven enclosed flows. In *Building theory and applications: proceedings of M&C 2019*, pages 2452–2461. American Nuclear Society (ANS), 2019.
- [245] S. K. Star, F. Belloni, G. Van den Eynde, and J. Degroote. POD-identification reduced order model of linear transport equations for control purposes. *International Journal for Numerical Methods in Fluids*, 90(8):375–388, 2019.
- [246] S. K. Star, B. Sanderse, G. Stabile, G. Rozza, and J. Degroote. Reduced order models for the incompressible Navier-Stokes equations on collocated grids using a ‘discretize-then-project’ approach. *Submitted to International Journal for Numerical Methods in Fluids*, 2020.
- [247] S. K. Star, G. Spina, F. Belloni, and J. Degroote. Development of a coupling between a system thermal-hydraulic code and a reduced order CFD model. *Annals of Nuclear Energy*, 153:108056, 2021.
- [248] S. K. Star, G. Stabile, F. Belloni, G. Rozza, and J. Degroote. A novel iterative penalty method to enforce boundary conditions in Finite Volume POD-Galerkin reduced order models for fluid dynamics problems. *Communications in Computational Physics (accepted)*, 2020.
- [249] S. K. Star, G. Stabile, G. Rozza, and J. Degroote. A POD-Galerkin reduced order model of a turbulent convective buoyant flow of sodium over a backward-facing step. *Applied Mathematical Modelling*, 89:486 – 503, 2021.

- [250] D. Stephenson. Creating openFOAM meshes with Gmsh. <https://www.micronanoflows.ac.uk/creating-openfoam-meshes-with-gmsh/>, 2015. Accessed 2019-08-22.
- [251] A. Syrakos, S. Varchanis, Y. Dimakopoulos, A. Goulas, and J. Tsamopoulos. A critical analysis of some popular methods for the discretisation of the gradient operator in finite volume methods. *Physics of Fluids*, 29(12):127103, 2017.
- [252] M. Taher, S. Saha, Y. Lee, and H. Kim. Numerical study of lid-driven square cavity with heat generation using LBM. *American Journal of Fluid Dynamics*, 3(2):40–47, 2013.
- [253] D. Taler. Heat transfer in turbulent tube flow of liquid metals. *Procedia Engineering*, 157:148–157, 2016.
- [254] R. Temam. Une méthode d’approximation de la solution des équations de Navier-Stokes. *Bulletin de la Société Mathématique de France*, 96:115–152, 1968.
- [255] The RELAP5-3D Code Development Team. RELAP5-3D Code Manual. Technical report, Idaho Falls: Idaho National Laboratory, 2018.
- [256] G. Tissot, L. Cordier, N. Benard, and B. R. Noack. Model reduction using dynamic mode decomposition. *Comptes Rendus Mécanique*, 342(6-7):410–416, 2014.
- [257] A. Toti. *Development and Validation of a System Thermal-Hydraulic/CFD Codes Coupling Methodology for Multi-Scale Transient Simulations of Pool-Type Reactors*. PhD thesis, Ghent University, 2017.
- [258] A. Toti, J. Vierendeels, and F. Belloni. Development and preliminary validation of a STH-CFD coupling method for multiscale thermal-hydraulic simulations of the MYRRHA reactor. In *24th International Conference on Nuclear Engineering*. American Society of Mechanical Engineers, 2016.
- [259] A. Toti, J. Vierendeels, and F. Belloni. Improved numerical algorithm and experimental validation of a system thermal-hydraulic/CFD coupling method for multi-scale transient simulations of pool-type reactors. *Annals of Nuclear Energy*, 103:36–48, 2017.
- [260] A. Toti, J. Vierendeels, and F. Belloni. Coupled system thermal-hydraulic/CFD analysis of a protected loss of flow transient in the MYRRHA reactor. *Annals of Nuclear Energy*, 118:199–211, 2018.

- [261] K. Tuček, S. Hermsmeyer, L. Ammirabile, D. Blanc, E. Wattelle, L. Burgazzi, M. Frogheri, L. Mansani, S. Ehster-Vignoud, B. Carlucci, et al. Identification and categorisation of safety issues for ESNII reactor concepts. Part I: Common phenomena related to materials. *Annals of Nuclear Energy*, 87:411–425, 2016.
- [262] S. Ullmann. *POD-Galerkin modeling for incompressible flows with stochastic boundary conditions*. PhD thesis, Technische Universität Darmstadt, 2014.
- [263] US DOE Nuclear energy research advisory committee and the generation IV international forum. Technology roadmap for generation iv nuclear energy systems. Technical report, US DOE, 2002.
- [264] B. Van Leer. Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov’s method. *Journal of Computational Physics*, 32(1):101–136, 1979.
- [265] K. Van Tichelen, F. Mirelli, M. Greco, and G. Viviani. E-SCAPE: A scale facility for liquid-metal, pool-type reactor thermal hydraulic investigations. *Nuclear Engineering and Design*, 290:65–77, 2015.
- [266] O. V. Vasilyev. High order finite difference schemes on non-uniform meshes with good conservation properties. *Journal of Computational Physics*, 157(2):746–761, 2000.
- [267] L. Vergari, A. Cammi, and S. Lorenzi. Reduced order modeling approach for parametrized thermal-hydraulics problems: inclusion of the energy equation in the POD-FV-ROM method. *Progress in Nuclear Energy*, 118:103071, 2020.
- [268] K. Veroy and A. 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.
- [269] K. Veroy, C. Prud’Homme, and A. T. Patera. Reduced-basis approximation of the viscous Burgers equation: rigorous a posteriori error bounds. *Comptes Rendus Mathématique*, 337(9):619–624, 2003.
- [270] K. Veroy, C. Prud’Homme, D. Rovas, and A. 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.

- [271] H. K. Versteeg and W. Malalasekera. *An introduction to computational fluid dynamics: the finite volume method*. Pearson Education, 2007.
- [272] L. Vervecken, J. Camps, and J. Meyers. Stable reduced-order models for pollutant dispersion in the built environment. *Building and Environment*, 92:360–367, 2015.
- [273] L. Vervecken, C. Gueibe, J. Camps, and J. Meyers. *Real-time simulation of the near-range atmospheric dispersion using Computational Fluid Dynamics*. PhD thesis, KU Leuven, 2015.
- [274] M. Viberg. Subspace-based methods for the identification of linear time-invariant systems. *Automatica*, 31(12):1835–1851, 1995.
- [275] M. Viellieber and A. Class. Coarse-grid-CFD for the thermal hydraulic investigation of Rod-Bundles. *PAMM*, 15(1):497–498, 2015.
- [276] M. Viellieber, P. Dietrich, and A. G. Class. Coarse-grid-CFD for a wire wrapped fuel assembly. *PAMM*, 13(1):317–318, 2013.
- [277] J. Vierendeels, L. Lanoye, J. Degroote, and P. Verdonck. Implicit coupling of partitioned fluid–structure interaction problems with reduced order models. *Computers & Structures*, 85(11-14):970–976, 2007.
- [278] V. Vuorinen, J.-P. Keskinen, C. Duwig, and B. Boersma. On the implementation of low-dissipative Runge–Kutta projection methods for time dependent flows using OpenFOAM®. *Computers & Fluids*, 93:153–163, 2014.
- [279] S. Walton, O. Hassan, and K. Morgan. Reduced order modelling for unsteady fluid flow using proper orthogonal decomposition and radial basis functions. *Applied Mathematical Modelling*, 37(20-21):8930–8945, 2013.
- [280] Z. Wang, I. Akhtar, J. Borggaard, and T. Iliescu. Proper orthogonal decomposition closure models for turbulent flows: a numerical comparison. *Computer Methods in Applied Mechanics and Engineering*, 237:10–26, 2012.
- [281] Z. Wang, B. McBee, and T. Iliescu. Approximate partitioned method of snapshots for POD. *Journal of Computational and Applied Mathematics*, 307:374–384, 2016.
- [282] B. Weigand, J. Ferguson, and M. Crawford. An extended Kays and Crawford turbulent Prandtl number model. *International Journal of Heat and Mass Transfer*, 40(17):4191–4196, 1997.
- [283] H. G. Weller, G. Tabor, H. Jasak, and C. Fureby. A tensorial approach to computational continuum mechanics using object-oriented techniques. *Computers in Physics*, 12(6):620–631, 1998.

- [284] J. Weller, E. Lombardi, M. Bergmann, and A. Iollo. Numerical methods for low-order modeling of fluid flows based on POD. *International Journal for Numerical Methods in Fluids*, 63(2):249–268, 2010.
- [285] K. Willcox and J. Peraire. Balanced model reduction via the proper orthogonal decomposition. *AIAA Journal*, 40(11):2323–2330, 2002.
- [286] D. Xiao, F. Fang, A. Buchan, C. Pain, I. Navon, and A. Muggerridge. Non-intrusive reduced-order modelling of the Navier-Stokes equations based on RBF interpolation. *International Journal for Numerical Methods in Fluid*, 79:580–595, 2015.
- [287] D. Xiao, F. Fang, A. G. Buchan, C. C. Pain, I. M. Navon, J. Du, and G. Hu. Non-linear model reduction for the Navier–Stokes equations using residual DEIM method. *Journal of Computational Physics*, 263:1–18, 2014.
- [288] D. Xiao, F. Fang, J. Du, C. Pain, I. Navon, A. Buchan, A. H. Elsheikh, and G. Hu. Non-linear Petrov–Galerkin methods for reduced order modelling of the Navier–Stokes equations using a mixed finite element pair. *Computer Methods In Applied Mechanics and Engineering*, 255:147–157, 2013.
- [289] D. Xiao, F. Fang, C. Pain, and I. Navon. A parameterized non-intrusive reduced order model and error analysis for general time-dependent nonlinear partial differential equations and its applications. *Computer Methods In Applied Mechanics And Engineering*, 317:868–889, 2017.
- [290] D. Xiao, C. Heaney, L. Mottet, F. Fang, W. Lin, I. Navon, Y. Guo, O. Matar, A. Robins, and C. Pain. A reduced order model for turbulent flows in the urban environment using machine learning. *Building and Environment*, 148:323–337, 2019.
- [291] D. Xiao, P. Yang, F. Fang, J. Xiang, C. Pain, I. Navon, and M. Chen. A non-intrusive reduced-order model for compressible fluid and fractured solid coupling and its application to blasting. *Journal of Computational Physics*, 330:221–224, 2016.
- [292] H. Xiao and P. Cinnella. Quantification of model uncertainty in RANS simulations: A review. *Progress in Aerospace Sciences*, 108:1–31, 2019.
- [293] X. Xie, M. Mohebjjaman, L. G. Rebholz, and T. Iliescu. Data-driven filtered reduced order modeling of fluid flows. *SIAM Journal on Scientific Computing*, 40(3):B834–B857, 2018.
- [294] X. Xie, D. Wells, Z. Wang, and T. Iliescu. Approximate deconvolution reduced order modeling. *Computer Methods in Applied Mechanics and Engineering*, 313:512–534, 2017.

-
- [295] X. Xie, D. Wells, Z. Wang, and T. Iliescu. Numerical analysis of the Leray reduced order model. *Journal of Computational and Applied Mathematics*, 328:12–29, 2018.
- [296] M. Yano. A space-time Petrov–Galerkin certified reduced basis method: Application to the Boussinesq equations. *SIAM Journal on Scientific Computing*, 36(1):A232–A266, 2014.
- [297] C. Yap. *Turbulent heat and momentum transfer in recirculating and impinging flows*. PhD thesis, The University of Manchester (United Kingdom), 1987.
- [298] Y. Zang, R. L. Street, and J. R. Koseff. A non-staggered grid, fractional step method for time-dependent incompressible Navier-Stokes equations in curvilinear coordinates. *Journal of Computational Physics*, 114(1):18–33, 1994.
- [299] K. Zhang. The multiscale thermal-hydraulic simulation for nuclear reactors: A classification of the coupling approaches and a review of the coupled codes. *International Journal of Energy Research*, 44(5):3295–3315, 2020.



Coupling between a system thermal-hydraulics code and a reduced order model

This appendix deals with the coupling of a reduced order model of a computational fluid dynamics model with a system thermal hydraulics code. The results are submitted as [247].

A.1 Introduction

For the design and safety assessment of a new generation of nuclear reactors, such as the MYRRHA reactor, computer codes have been developed for the thermal-hydraulic analyses of the reactor's primary system in operational and accidental conditions. There are two main types of numerical codes for thermal-hydraulic analyses used in the nuclear industry: the system codes, also called the lumped parameter codes, based on one-dimensional models of physical transport phenomena and the field codes, based on three-dimensional computational fluid dynamics models [184].

The flow in many reactor primary components exhibit phenomena as natural circulation, mixing and stratification that cannot be modeled by system codes adequately. CFD codes are therefore used to numerically simulate these types of transient flows to accurately quantify the system behavior in accidental conditions and to handle complex geometries [260]. However, the number of nuclear reactor simulations in a safety analysis is, in the majority of cases, beyond the possibilities of present hardware if a CFD code is used alone.

Thus, to get the best out of both worlds, coupling between system and CFD codes has been postulated as a new method for thermal-hydraulic analyses. The nuclear community has performed extensive research on interfacing CFD codes with the traditional system codes. Gibling and Mahaffy [89] were among the first to study the transition between the

1D and 3D descriptions at an interface.

A well recognized system thermal-hydraulics code by many nuclear authorities for safety analyses is the RELAP5 series. The RELAP5 series have been coupled with several computer codes as the sub-channel code COBRA-TF [132, 154], the containment analysis code GOTHIC [98, 121] and CFD codes ANSYS-CFX (previous called CFDS-FLOW3D) [15, 38], ANSYS Fluent [6, 8, 75, 157, 228] and Star-CCM+ [131]. Recently, work has been conducted in the framework of the THINS project of the 7th Framework EU Program on nuclear fission safety [20, 194].

SCK CEN uses the RELAP5-3D [255] version for MYRRHA safety studies that allows the use of LBE as a working fluid. Moreover, SCK CEN has developed a numerical algorithm to couple RELAP5-3D with ANSYS Fluent for multi-scale transient simulations of pool-type reactors [260]. Another CFD code that has been coupled already to several STH codes [80, 194, 299], but to the best of the authors' knowledge not yet with RELAP5, is the open source code OpenFOAM [130].

Even though coupled systems require considerably less computational resources and time than stand-alone CFD codes, the gain in computational effort is still limited by the CFD part [39]. To overcome this burden, this work proposes to couple the system code with a reduced order model of the high fidelity CFD code. The STH code RELAP5-MOD3.3 [182] and a reduced order CFD model that is constructed using the libraries of the open source code OpenFOAM 6 are coupled, which is called the RELAP5/ROM model hereafter. The codes are coupled using a domain decomposition coupling algorithm, which is explained in Section A.2. The exchange of the hydraulic quantities between the coupled domains at the coupling interfaces is explained in Section A.3. The CFD and ROM formulations for an incompressible Newtonian fluid are described in Sections A.4 and A.5, respectively. In Section A.6, some challenges for coupling STH codes with reduced order models are presented. The set-up of three numerical test cases, the open pipe flow test, the open pipe flow reversal test and the closed pipe flow test, are described in Section A.7. Then in Section A.8, the coupling methodology is first evaluated by comparing the results of a coupled RELAP5/CFD model with RELAP5 stand-alone results. Consecutively, the coupled RELAP5/ROM model is tested on a series of parametric problems that are evaluated against the coupled RELAP5/CFD model and the results are discussed in Section A.9. Finally, conclusions are drawn and an outlook for further improvements is provided in Section A.10.

A.2 Coupling methodology

A methodology is developed for a coupling approach [174] together with a domain decomposition method [237] in which the different domains are resolved separately by independent solvers. The whole simulation domain is split into sub-domains; where the one-dimensional approximation is deemed accurate enough for the given problem, the sub-domain is allocated to the STH code and if not to the CFD code. The number of coupling faces between the sub-domains is identified.

At each coupling interface between two sub-domains, thermal-hydraulic quantities, like mass flow rate, pressure and temperature, are exchanged between the solvers. By treating the sub-domains as black boxes, the following input-output relations hold at each coupling interface:

$$O_{STH} = G_{STH}(I_{STH}), \quad (\text{A.1})$$

$$O_{CFD} = G_{CFD}(I_{CFD}), \quad (\text{A.2})$$

where \mathbf{I} and \mathbf{O} are the input and output vectors, respectively. These vectors are either obtained by the STH code or the CFD code. G is the associated operator. Hence, the output \mathbf{O} is obtained from the given input \mathbf{I} .

Then, as the thermal-hydraulic quantities are exchanged between the sub-domains, the following relation holds between the inputs and outputs at the coupling interfaces:

$$\mathbf{I}_{CFD} = \mathbf{O}_{STH}, \quad (\text{A.3})$$

$$\mathbf{I}_{STH} = \mathbf{O}_{CFD}. \quad (\text{A.4})$$

Based on Equations A.1-A.4, the STH/CFD coupled problem can be expressed in its fixed-point formulation as follows:

$$\mathbf{I}_{CFD} = G_{STH}(\mathbf{I}_{STH}) = G_{STH}(G_{CFD}(\mathbf{I}_{CFD})). \quad (\text{A.5})$$

For time-dependent problems, the coupling can either be done by an explicit or implicit coupling method. Explicit coupling procedures are appealing in terms of efficiency as only one (or a few) solution of the sub-problems per time step are needed. However, the numerical stability of the scheme can be drastically compromised, especially when dealing with incompressible fluids. Consequently, the time step size needs to be restricted. For more details and the analytical explanation, the reader is referred to [102]. Moreover, it is known from the work of Toti et al. [258] that the implicit coupling scheme is numerically more stable than the explicit coupling schemes. As reduced order models are sensitive to numerical instabilities [5, 25, 234], the coupling is only done by an implicit coupling method in this work.

A.2.1 Implicit coupling numerical scheme

In order to assure a global conservation of transported quantities over the interface, an implicit numerical scheme, which determines the solution of the fixed-point problem of Equation A.5, is implemented. The exchange of data between the codes is repeated through an iterative procedure within a time step until a defined convergence criterion is met. In this way, an equilibrium is reached at the coupling boundary interfaces and the numerical stability is improved. The coupled problem of Equation A.5 is reformulated as a root finding problem:

$$\mathbf{R}(\mathbf{I}_{CFD}) = G_{STH}(G_{CFD}(\mathbf{I}_{CFD})) - \mathbf{I}_{CFD} = \mathbf{0}, \quad (\text{A.6})$$

where \mathbf{R} is the residual vector. This residual is approximated by a first order Taylor expansion around the current solution at each coupling iteration, k , within time step n as expressed below

$${}^n \mathbf{R}^{k+1}({}^n \mathbf{I}_{CFD}^{k+1}) = {}^n \mathbf{R}^k({}^n \mathbf{I}_{CFD}^k) + {}^n \mathbf{J}^k ({}^n \Delta \mathbf{I}_{CFD}^k) = \mathbf{0}, \quad (\text{A.7})$$

where \mathbf{J} is the Jacobian matrix which contains the partial derivatives of the residual vector $\mathbf{R} = [r_1, r_2, \dots, r_m]$ with respect to the terms of the CFD input vector $\mathbf{I}_{CFD} = [I_1, I_2, \dots, I_m]$. The Jacobian at the n^{th} time step and k^{th} iteration is given by

$${}^n \mathbf{J}^k = \begin{bmatrix} \frac{{}^n \partial r_1^k}{{}^n \partial I_1^k} & \frac{{}^n \partial r_1^k}{{}^n \partial I_2^k} & \dots & \frac{{}^n \partial r_1^k}{{}^n \partial I_m^k} \\ \frac{{}^n \partial r_2^k}{{}^n \partial I_1^k} & \frac{{}^n \partial r_2^k}{{}^n \partial I_2^k} & \dots & \frac{{}^n \partial r_2^k}{{}^n \partial I_m^k} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{{}^n \partial r_m^k}{{}^n \partial I_1^k} & \frac{{}^n \partial r_m^k}{{}^n \partial I_2^k} & \dots & \frac{{}^n \partial r_m^k}{{}^n \partial I_m^k} \end{bmatrix}. \quad (\text{A.8})$$

The unknown terms of the Jacobian are approximated by finite differences:

$$\frac{{}^n \partial r_i^k}{{}^n \partial I_j^k} \approx \frac{{}^n r_i^k - {}^n r_i^{k-1}}{{}^n I_j^k - {}^n I_j^{k-1}}. \quad (\text{A.9})$$

Once the Jacobian is known, ${}^n \Delta \mathbf{I}_{CFD}^k$ is calculated based on Equation A.7 and is added to the CFD input vector of the current iteration k to get the input vector for the next iteration $k + 1$:

$${}^n \mathbf{I}_{CFD}^{k+1} = {}^n \mathbf{I}_{CFD}^k + {}^n \Delta \mathbf{I}_{CFD}^k. \quad (\text{A.10})$$

This method is also known as the interface Quasi-Newton method. The computed Jacobian matrix for a certain time step n obtained for the first iteration $k = 0$ can be used for several following coupling iterations as long as the following condition is met

$$\|{}^n \mathbf{R}^k\| < \frac{\|{}^n \mathbf{R}^0\|}{10}, \quad (\text{A.11})$$

where 10 is a heuristic value that is introduced to assure a quick convergence [260]. If the condition is not met, the Jacobian needs to be recomputed. For a more detailed description of the procedure to create and to evaluate the Jacobian the reader is referred to [260].

Figure A.1 shows a simplified flowchart of the implicit coupling algorithm.

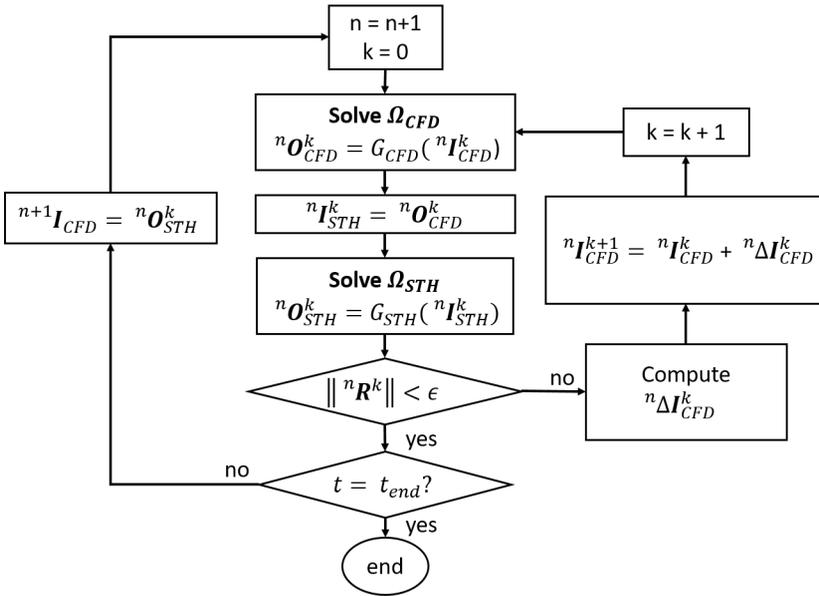


Figure A.1: Flowchart of the generic implicit coupling numerical algorithm with the interface Quasi-Newton algorithm. $\epsilon > 0$ is the given tolerance, t denotes time and t_{end} is the final simulation time.

A.3 Transport of hydraulic quantities over the coupling interfaces of coupled models

The transport of hydraulic quantities over the coupling interfaces of a coupled RELAP5 with OpenFOAM (RELAP5/CFD) model is explained in this section. The procedure is the same when RELAP5 is coupled with the reduced order model.

As introduced previously, the coupling method is based on a domain decomposition technique. In this work, the computational domain Ω is divided into several non-overlapping sub-domains: the STH sub-domain(s), Ω_{STH} , attributed to RELAP5 and the CFD sub-domain(s), Ω_{CFD} , attributed to OpenFOAM. This work is limited to simple configurations with only two interfaces, Γ_1 and Γ_2 , between the STH and CFD sub-domain as depicted in Figure A.2. However, the methodology is straightforward to expand to more interfaces.

Two hydraulic quantities are transported over the coupling interfaces: velocity and kinetic pressure. This is done in such a way that mass and momentum are conserved. The average velocity determined at the single junction \mathbf{U}_{STH} of the STH sub-domain at coupling interface 1 is implemented as a uniform inlet velocity profile onto the inlet boundary of the CFD sub-domain. At coupling interface 2, the area-averaged velocity \mathbf{U}_{CFD} at the outlet boundary of the CFD sub-domain is transported to the single junction of the STH sub-domain.

The transport of pressure over the interfaces is done differently. OpenFOAM uses the kinematic pressure, which is pressure divided by the fluid density ρ . Moreover, the pressure is only calculated relative to a reference level and is therefore set to 0 Pa at the outlet of the CFD sub-domain. RELAP5 on the other hand calculates the absolute pressure at the center of all cells in the STH sub-domain. To determine the pressure at the coupling interface 2, the volume-centered pressures from the first two neighboring cell centers of the STH sub-domain, P_1 and P_2 , as depicted in Figure A.3, are extrapolated to the center of the boundary of the STH sub-domain as follows

$$P_{STH}^{\Gamma_2} = P_1 + \frac{P_1 - P_2}{2}, \quad (A.12)$$

where it is assumed that these neighboring cells have the same size.

The outlet boundary of the STH sub-domain at coupling interface 1 is then updated based on the pressure determined at coupling interface 2 and the area-averaged pressure drop over the CFD sub-domain in the following way:

$$P_{STH}^{\Gamma_1} = \Delta P_{CFD} + P_{STH}^{\Gamma_2}. \quad (A.13)$$

As the area-averaged velocity is transferred in one direction and the area-averaged pressure in the opposite direction at the coupling interfaces, transient simulations of reverse flows can also be performed using this approach [260].

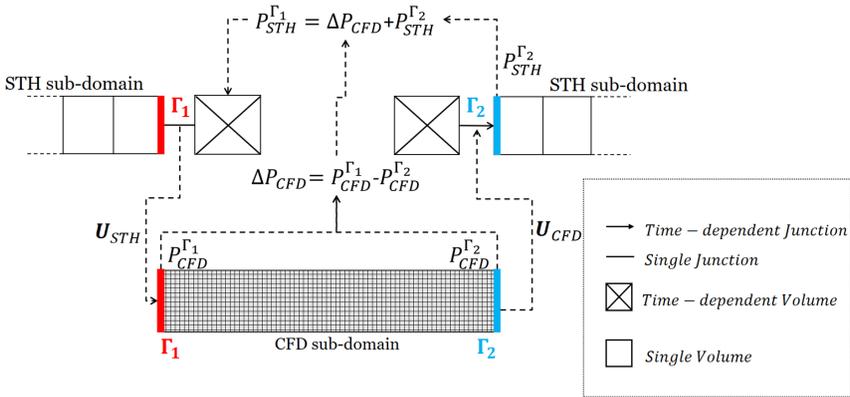


Figure A.2: Variable exchanges over the coupling interfaces between the STH sub-domains and the CFD sub-domain.

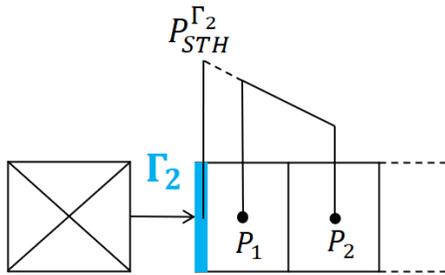


Figure A.3: Pressure extrapolation at coupling interface 2. The legend of Figure A.2 applies.

A.4 The coupled codes' governing equations and models

This section presents a brief description of the best-estimate system thermal-hydraulics code RELAP5-MOD3.3. Furthermore, the governing equations that are discretized and solved with the CFD code OpenFOAM are described as those equations are projected onto a reduced basis in order to construct the ROM.

A.4.1 RELAP5-MOD3.3: Thermal-hydraulic modeling

The RELAP5-MOD3.3 code [182] is developed at the Idaho National Engineering & Environmental Laboratory for the U.S. Nuclear Regulatory Commission. The code makes use of a two-fluid model in 1D form. The computational domain is subdivided in volumes that are joined by junctions. In the RELAP5 approach, the equations of mass and energy are solved in the control volumes and momentum equations are solved in the junction components i.e. across the two volumes. Therefore, the quantities that come from the solution of the mass and energy equations, like pressure and temperature, are evaluated at the center of the nodes. Instead, the quantities that come from the solution of the momentum equation, like velocity and mass flow rate, are evaluated at the interface between two adjoining volumes (junction).

A.4.2 OpenFOAM 6: Reynolds-averaged Navier-Stokes equations for incompressible turbulent flow

Industrial turbulent flows are often described with the Reynolds-averaged Navier-Stokes (RANS) equations. The governing unsteady RANS equations for an incompressible Newtonian flow without gravity and body forces are given by

$$\begin{cases} \nabla \cdot \mathbf{U} = 0 & \text{in } \Omega_{CFD}, \\ \frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot (\mathbf{U} \otimes \mathbf{U}) = -\nabla P + \nabla \cdot [(\nu + \nu_t)(\nabla \mathbf{U} + (\nabla \mathbf{U}^T))] & \text{in } \Omega_{CFD}, \end{cases} \quad (\text{A.14})$$

where \mathbf{U} are the time-averaged values for velocity and P is the time-averaged kinematic pressure, which is pressure divided by the fluid density ρ , ν is the kinematic viscosity and t denotes time. Often one or two equation turbulence models are used to model incompressible turbulent flows. This work uses the k - ε turbulence model [150], according to the previous work done by Toti et al. [260]. In this model the eddy viscosity, ν_t , is a function of the two variables k and ε , which stand, respectively, for the turbulence kinetic energy and the turbulence dissipation rate.

OpenFOAM uses a finite volume discretization method for which the computational domain is broken into smaller regions that are called control volumes [177]. In this work, the RANS equations A.14 are discretized and solved using a PIMPLE [77] algorithm for the pressure-velocity coupling, which is a combination of SIMPLE [187] and PISO [128].

A.5 POD-Galerkin reduced order model for incompressible turbulent flow

The reduced order model for the full order CFD code is constructed using a POD-Galerkin technique. POD stands for proper orthogonal decomposition and is used to reduce the dimensionality of a system by transforming the original set of N_h degrees of freedom into a new set of N_r degrees of freedom, so-called modes, where $N_r \ll N_h$. These modes are ordered in such a way that the first few modes retain most of the energy present in the original solution [149]. For more details about POD and other reduced basis techniques, the reader is referred to [53, 110, 203, 217].

Flow field solutions obtained by solving the unsteady RANS equations, so-called snapshots, are collected at certain time instances. As the finite volume discretization is used

on collocated grids, the variables velocity and pressure are known at discrete points in the spatial domain, which is at the center of the control volumes. POD assumes that these solutions can be expressed as a linear combination of spatial modes multiplied by time-dependent coefficients. The L_2 norm is preferred for discrete numerical schemes [40, 241] with $(\cdot, \cdot)_{L_2(\Omega_{CFD})}$ the L_2 inner product of the fields over the sub-domain Ω_{CFD} . As the POD modes are orthonormal to each other, $(\varphi_i, \varphi_j)_{L_2(\Omega_{CFD})} = \delta_{ij}$ holds, where δ is the Kronecker delta.

For the velocity and pressure fields, the approximations are given, respectively, by

$$\mathbf{U}(\mathbf{x}, t) \approx \mathbf{U}_r = \sum_{i=1}^{N_r^U} \varphi_i(\mathbf{x}) a_i(t), \quad (\text{A.15})$$

$$P(\mathbf{x}, t) \approx P_r = \sum_{i=1}^{N_r^P} \chi_i(\mathbf{x}) b_i(t), \quad (\text{A.16})$$

where φ_i and χ_i are the modes of the velocity and pressure, and respectively a_i and b_i the corresponding time-dependent coefficients. N_r^U is the number of velocity modes and N_r^P is the number of pressure modes.

The above assumptions are extended to the eddy viscosity fields, ν_t as follows

$$\nu_t(\mathbf{x}, t) \approx \nu_{t_r} = \sum_{i=1}^{N_r^{\nu t}} \eta_i(\mathbf{x}) c_i(t), \quad (\text{A.17})$$

with η_i the eddy viscosity modes and c_i the corresponding time-dependent coefficients. $N_r^{\nu t}$ is the number of eddy viscosity modes.

The optimal POD basis space for velocity, $E_{POD}^U = \text{span}(\varphi_1, \varphi_2, \dots, \varphi_{N_r^U})$ is then constructed by minimizing the difference between the snapshots and their orthogonal projection onto the basis for the L_2 norm [205]. This gives the following minimization problem

$$E_{POD}^U = \arg \min_{\varphi_1, \dots, \varphi_{N_r^U}} \frac{1}{N_s} \sum_{n=1}^{N_s} \left\| \mathbf{U}^n - \sum_{i=1}^{N_r^U} (\mathbf{U}^n, \varphi_i)_{L_2(\Omega_{CFD})} \varphi_i \right\|_{L_2(\Omega_{CFD})}^2, \quad (\text{A.18})$$

where N_s is the number of collected velocity snapshots and $N_s > N_r^U$. The POD modes are then obtained from this minimization problem by solving the following eigenvalue problem on the snapshots [235, 239, 241]:

$$\mathbf{C}^U \mathbf{Q}^U = \mathbf{Q}^U \boldsymbol{\lambda}^U, \quad (\text{A.19})$$

where $\mathbf{C}_{ij}^U = (\mathbf{U}^i, \mathbf{U}^j)_{L_2(\Omega_{CFD})}$ for $i, j = 1, \dots, N_s$ is the correlation matrix, \mathbf{Q}^U is a square matrix of eigenvectors and $\boldsymbol{\lambda}^U$ is a diagonal matrix containing the eigenvalues. The POD modes, φ_i , can then be constructed as follows

$$\varphi_i(\mathbf{x}) = \frac{1}{N_s \sqrt{\lambda_i^U}} \sum_{n=1}^{N_s} \mathbf{U}^n(\mathbf{x}) Q_{in}^U \quad \text{for } i = 1, \dots, N_r^U, \quad (\text{A.20})$$

of which the most energetic (dominant) modes are selected. The above assumptions can be extended to obtain the pressure and eddy viscosity modes.

To obtain a reduced order model, the POD is combined with the Galerkin projection, for which the momentum equations of the set of governing equations A.14 are projected onto the reduced POD basis space. The following reduced system of momentum equations is then obtained

$$\mathbf{M}_r \dot{\mathbf{a}} + \mathbf{a}^T \mathbf{C}_r \mathbf{a} + \mathbf{A}_r \mathbf{b} - \nu (\mathbf{D}_r + \mathbf{DT}_r) \mathbf{a} - \mathbf{c}^T (\mathbf{E}_r + \mathbf{ET}_r) \mathbf{a} = \mathbf{0}, \quad (\text{A.21})$$

where the 'over-dot' indicates the time derivative and

$$M_{r_{ij}} = (\varphi_i, \varphi_j)_{L_2(\Omega_{CFD})}, \quad (\text{A.22})$$

$$A_{r_{ij}} = (\varphi_i, \nabla \chi_j)_{L_2(\Omega_{CFD})}, \quad (\text{A.23})$$

$$D_{r_{ij}} = (\varphi_i, \nabla \cdot \nabla \varphi_j)_{L_2(\Omega_{CFD})}, \quad (\text{A.24})$$

$$DT_{r_{ij}} = (\varphi_i, \nabla \cdot (\nabla \varphi_j^T))_{L_2(\Omega_{CFD})}, \quad (\text{A.25})$$

$$C_{r_{ijk}} = (\varphi_i, \nabla \cdot (\varphi_j \otimes \varphi_k))_{L_2(\Omega_{CFD})}, \quad (\text{A.26})$$

$$E_{r_{ijk}} = (\varphi_i, \nabla \cdot \boldsymbol{\eta}_j \nabla \varphi_k)_{L_2(\Omega_{CFD})}, \quad (\text{A.27})$$

$$ET_{r_{ijk}} = (\varphi_i, \nabla \cdot \boldsymbol{\eta}_j (\nabla \varphi_k^T))_{L_2(\Omega_{CFD})}. \quad (\text{A.28})$$

These reduced matrices and third order tensors are stored while constructing the reduced order model during a, so-called, off-line stage. More details on the POD and Galerkin projection method can be found in [87, 239, 241].

Standard Galerkin projection-based reduced order models are unreliable when applied to the nonlinear unsteady Navier-Stokes equations [5, 25, 41, 149, 221, 234]. This is mainly caused by the fact that there is no dedicated equation for pressure in Equations A.14. Therefore, for fluid problems that are solved numerically using a finite volume discretization technique [177, 271], a Pressure Poisson Equation (PPE) is solved rather than the equation for mass conservation (Equation A.14a). For more details on the derivation of the PPE the reader is referred to J.-G Liu et al. [159]. Alternative methods are possible, which are discussed in [241].

Moreover, reduced order models for scale resolving turbulent flow simulations are often affected by the energy blow up due to the truncation of POD modes. The occurrence of unstable time behavior in the reduced order model can then be explained by the concept of the energy cascade [57]. A closure modeling between the POD-based ROM and the full order scale resolving turbulence representation is then required to improve the accuracy and instability of POD-Galerkin ROMs [185, 280, 293]. ROMs based on RANS simulation typically include a closure model based on the eddy viscosity at ROM level as eddy viscosity models are already present in the full order model [238, 271].

Also the reduced system of equations constructed in this work, consisting of the momentum equations (Equation A.21) together with the PPE, is not sufficient to determine all unknown coefficients (of velocity, pressure and eddy diffusivity). Therefore, the coefficients $c_i(t)$, that are used in the approximation of the eddy viscosity fields, are computed with a data-driven non-intrusive interpolation procedure using Radial Basis Functions (RBF) as described in [151]. The advantage of approximating the eddy viscosity coefficients with RBFs is that the turbulence transport equations for k and ε of the system of Equations A.14 do not need to be projected onto the POD basis space spanned by the eddy viscosity modes. Therefore, the reduced order model is independent of the turbulence model used in the RANS simulations [111].

The initial conditions for the reduced system of ordinary differential equations (Equation A.21) are obtained by performing a Galerkin projection of the initial conditions for the RANS simulations onto the POD basis spaces as follows

$$a_i(0) = (\varphi_i(\mathbf{x}), \mathbf{U}(\mathbf{x}, 0))_{L_2(\Omega_{CFD})}, \quad (\text{A.29})$$

$$b_i(0) = (\chi_i(\mathbf{x}), P(\mathbf{x}, 0))_{L_2(\Omega_{CFD})}, \quad (\text{A.30})$$

for velocity and kinematic pressure, respectively.

A.6 Challenges for coupling system thermal-hydraulics codes with a reduced order model

In the previous section and in Section A.2, we noted that POD-Galerkin reduced order models are, in general, sensitive to numerical instabilities [5, 25, 234]. This is one of the main challenges of reduced order modeling for fluid flow problems [149]. Therefore, only an implicit coupling scheme is considered in this work, which is numerically more stable than the explicit coupling schemes [258].

The Quasi-Newton coupling algorithm of Figure A.1 for coupling a system thermal-hydraulics code with a CFD solver is the same for the coupling with a reduced order model. Nevertheless, there are a couple of challenges that need to be taken into account when replacing the CFD part with a ROM.

The velocity and pressure snapshots, required for the creation of the reduced basis spaces of the ROM, are typically collected by performing high fidelity simulations. For the RELAP5/ROM coupled model, the snapshots need to be collected by performing coupled RELAP5/CFD simulations. If there are perturbations present in the coupled RELAP5/CFD solutions, they can have a detrimental effect on the performance of the RELAP5/ROM coupled model.

Moreover, reduced order models are only capable of predicting solutions for new parameter values and for long time integration if the flow features of these new cases are contained in the reduced basis spaces spanned by the POD modes [149]. It is challenging to construct the optimal reduced basis spaces, especially for parametric ROMs. We will not focus on this challenge in this work. For more details on constructing the optimal reduced basis spaces we refer the reader to [25, 149].

A challenge related to the exchange of hydraulic quantities over the coupling interfaces is that non-homogeneous time-dependent boundary conditions need to be imposed in the ROM using a boundary control method [93]. Another challenge is related to the accuracy of the ROM. We discuss both challenges in the subsequent subsections.

A.6.1 Imposing the non-homogeneous time-dependent boundary conditions in the ROM

As described in Section A.3, the average velocity determined at the single junction of the STH sub-domain at coupling interface 1 is implemented as a uniform inlet velocity profile on the inlet boundary of the CFD sub-domain. Boundary conditions of the CFD sub-domain at the coupling interfaces of a coupled RELAP5/CFD system are controlled every time step as described in Section A.2. However, the boundary conditions are not explicitly present in

the reduced momentum equations (Equation A.21). Therefore, they cannot be controlled directly [161]. The selected approach in this work for handling the non-homogeneous Dirichlet BCs is the penalty method [93]. The aim of the penalty method is to enforce the BCs in the ROM with a penalty factor τ [234] as described in Section 4.3.2. The velocity boundary value determined at the single junction of the STH sub-domain, U_{STH} , is imposed at the reduced order level on the inlet boundary of the CFD sub-domain as follows

$$M_r \dot{\mathbf{a}} + \mathbf{a}^T C_r \mathbf{a} + \mathbf{A}_r \mathbf{a} - \nu (\mathbf{D}_r + \mathbf{D}T_r) \mathbf{a} - \mathbf{c}^T (\mathbf{E}_r + \mathbf{E}T_r) \mathbf{a} + \tau (\mathbf{O}_r \mathbf{a} - U_{STH} \mathbf{P}_r) = \mathbf{0}, \quad (\text{A.31})$$

where

$$P_{r_i} = (\varphi_i, \mathbf{\Lambda})_{L_2(\Gamma_{1,CFD})}, \quad (\text{A.32})$$

$$O_{r_{ij}} = (\varphi_i, \varphi_j)_{L_2(\Gamma_{1,CFD})}, \quad (\text{A.33})$$

with $\mathbf{\Lambda}$ a unit field and $\Gamma_{1,CFD}$ the boundary of the CFD sub-domain.

The penalty factor is usually tuned with a sensitivity analysis [93, 134, 161]. However, if τ tends to zero, the BCs are not enforced and if the factor tends to infinity the ROM becomes ill-conditioned [161].

A.6.2 Relative error

Reduced order models contain a lower number of degrees of freedom than the high fidelity models due to a truncation of the POD modes. That way, they are computationally more efficient, but have generally a lower accuracy than the high fidelity models [96, 149].

To determine the accuracy of the coupled model of which the CFD part is replaced by a ROM, the solutions need to be compared with those of the coupled RELAP5/CFD model.

In this work, the accuracy of the RELAP5/ROM coupled model is determined by calculating the relative L_2 error for each time step, t , between the RELAP5/CFD solutions, X_{CFD} , and the fields obtained by performing coupled RELAP5/ROM simulations, X_{ROM} . This so-called relative prediction error is defined as

$$\epsilon_{L_2(\Omega_{CFD})}(t) = \frac{\|X_{CFD}(t) - X_{ROM}(t)\|_{L_2(\Omega_{CFD})}}{\|X_{CFD}(t)\|_{L_2(\Omega_{CFD})}}. \quad (\text{A.34})$$

where X represents the velocity or pressure fields.

We compare the prediction errors with the basis projection error, which acts as a lower error bound for the reduced order model. The basis projection error, $\hat{\epsilon}_{L_2(\Omega_{CFD})}$, is defined as the relative L_2 error between the RELAP5/CFD solutions, X_{CFD} , and the projected fields, X_r , which are obtained by the L_2 -projection of the snapshots onto the POD bases:

$$\hat{\epsilon}_{L_2(\Omega_{CFD})}(t) = \frac{\|X_{CFD}(t) - X_r(t)\|_{L_2(\Omega_{CFD})}}{\|X_{CFD}(t)\|_{L_2(\Omega_{CFD})}}. \quad (\text{A.35})$$

In practice, the prediction error is larger than the projection error for a single parameter point.

A.7 Numerical test cases

In this section, the set-ups for three different configurations are described: the open pipe flow test, the open pipe flow reversal test and the closed pipe flow test. All tests are carried out for single-phase water flow with kinematic viscosity $\nu = 1.0 \cdot 10^{-6} \text{ m}^2/\text{s}$.

For the coupled models, the computational domain is divided into a CFD sub-domain and an STH sub-domain. For all configurations, the CFD sub-domain consists of a circular pipe of length $L_{CFD} = 0.5$ m and diameter $D = 0.1$ m. A mesh with 145945 hexahedral cells is constructed onto the three-dimensional domain, as depicted in Figure A.4. As the CFD sub-domain and mesh are kept unchanged, the coupling procedure for coupling RELAP5 with the reduced order model is the same as for the RELAP5/CFD coupled model.

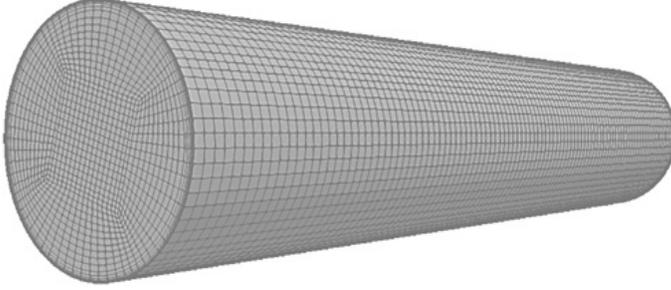


Figure A.4: Mesh of the CFD sub-domain consisting of a circular pipe of length L_{CFD} and diameter D .

The unsteady RANS equations (Equation A.14) are discretized and solved by the finite volume method with ITHACA-FV [240], which is an open source C++ library based on the finite volume solver OpenFOAM [130]. In this work, the libraries of OpenFOAM 6 are used. The spatial discretization is performed with linear interpolation schemes and the temporal discretization is treated using a first order implicit differencing scheme. The calculation of the POD modes, the Galerkin projection of the RANS solutions on the reduced subspace and the ROM simulations are also carried out with ITHACA-FV. For more details on the code, the reader is referred to [239, 240, 241].

All stand-alone STH models of the computational domains are constructed with RELAP5-MOD3.3. The reduced order CFD models are constructed according to the methodology described in section A.5. The velocity and pressure snapshots needed for the creation of the reduced subspaces are collected by performing a coupled RELAP5/CFD simulation for a certain parameter set. This is required for each of the three flow configurations. Moreover, the coupled RELAP5/CFD model is first evaluated against the corresponding STH stand alone model in order to evaluate the implicit coupling methodology. Thereafter, the coupled RELAP5/ROM models are tested and compared with the coupled RELAP5/CFD models.

All simulations are run on a single Intel[®] Xeon[®] GOLD 5118 @ 2.30GHz core.

A.7.1 Open pipe flow test

The simple open pipe configuration consists of a circular straight pipe with length $L = 8.5$ m and internal diameter $D = 0.1$ m. The pipe is split in three parts; the beginning and the ending parts have both have a length $L_{STH-1} = L_{STH-2} = 4.0$ m and the middle part has length $L_{CFD} = 0.5$ m. Figure A.5 shows a sketch of the set-up for an STH stand-alone simulation at the top. In the same figure, the set-up of the coupled RELAP5/CFD model is shown at the bottom. The beginning and ending parts of the STH domain are divided in 10 volumes of 0.4 m. The middle is either modeled with a finer mesh of 20 equally sized

volumes of 0.0025 m for the STH stand-alone simulations or assigned to the CFD code for the coupled simulations.

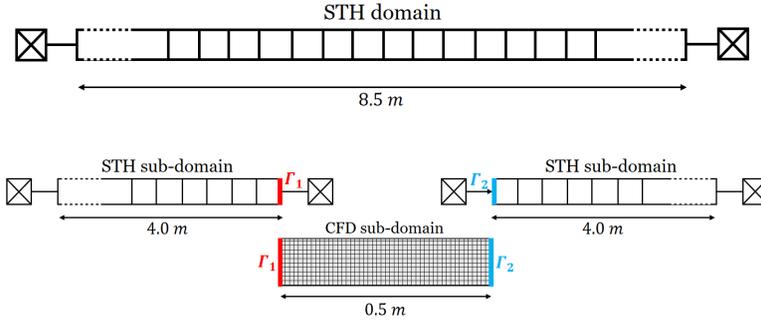


Figure A.5: RELAP5 stand-alone nodalization of the open pipe flow configuration (top) and the domain decomposition of the open pipe flow configuration for the coupled RELAP5/CFD model (bottom). The legend of Figure A.2 applies.

The fluid is initially at rest and driven by an abrupt pressure difference, ΔP , of 0.20 bar applied over the whole pipe at $t = 0$ s. The total time of simulation is 10 s. The inlet and outlet boundary conditions of the STH sub-domains are set by time-dependent volumes. A previous study by Toti et al. [260] showed that accurate results for velocity and pressure are obtained for a time step of 0.1 s in the case of implicit coupling. Therefore, the coupled simulations are performed with this time step.

The coupling methodology is evaluated by comparing the time evolution of the mass flow rate and the area-averaged pressure at coupling interface Γ_2 obtained with RELAP5/CFD and the stand-alone RELAP5 simulations for the pressure drop of 0.20 bar. Snapshots of the velocity and pressure fields that are calculated at the CFD sub-domain are collected every 0.1 s during the coupled RELAP5/CFD simulation. Thus, 100 snapshots are collected in total that are used for the construction of the RELAP5/ROM model according to section A.5.

The coupled RELAP5/ROM model is then tested for the same pressure difference and for four new conditions, namely $\Delta P = 0.10, 0.15, 0.21$ and 0.23 bar. As described in Section A.6.1, the uniform velocity boundary value at the inlet is enforced in the ROM with a penalty method. In this work, the penalty factor is set to 1.0. The RELAP5/ROM results are compared with corresponding coupled RELAP5/CFD results.

A.7.2 Open pipe flow reversal test

Using the same set-up as for the open pipe flow test case, both the RELAP5/CFD and the RELAP5/ROM models are tested for sudden flow reversal. Initially, the absolute pressure at the inlet of the STH domain is set to 1.40 bar while the outlet pressure is set to 1.20 bar. Thus, the total pressure drop over the whole pipe is 0.20 bar. Between $t = 9$ s and $t = 13$ s, the pressure at the inlet is decreased linearly up to 1.0 bar and the simulation is run up to $t = 25$ s. Once the pressure at the inlet is lower than the pressure at the outlet, the fluid eventually starts flowing in the opposite direction. This is tested for STH stand alone, RELAP5/CFD and RELAP5/ROM. As done for the open pipe flow test, snapshots are collected every 0.10

s. Moreover, the coupled RELAP5/ROM model is tested for pressure drops of 0.10, 0.15, 0.21 and 0.23 bar.

Furthermore, the ROM performance for long time integration is tested with this test case. 100 velocity, pressure and eddy viscosity snapshots are collected during the first 10 seconds of simulation time. The obtained RELAP5/ROM model, after performing POD onto the snapshots, is used to simulate the whole transient up to $t = 25$ s and the results are compared with an additional RELAP5/ROM for which 250 snapshots were collected during the whole simulation time.

A.7.3 Closed pipe flow test

The last configuration consists of a closed loop. The STH configuration of the open pipe flow test is extended with a circulation pump and an expansion tank as shown in Figure A.6. Figure A.7 shows the same configuration with the domain decomposition for the coupled simulations. The two vertical legs of the loop are 1.6 m long while the horizontal legs are 8.5 m long. The top horizontal leg is split similarly to the open pipe flow test case. In this configuration, the transient is initiated by the start of the pump causing again a mass flow ramp. The pump reaches its nominal speed after 5 seconds of simulation time. The total simulation time is 10 s. A coupled RELAP5/CFD simulation is performed with the nominal speed of the rotor set to 100 rad/s and snapshots are collected every 0.10 seconds. Coupled RELAP5/ROM simulations are also performed for a nominal rotor speed of 80 rad/s, 90 rad/s and 110 rad/s.

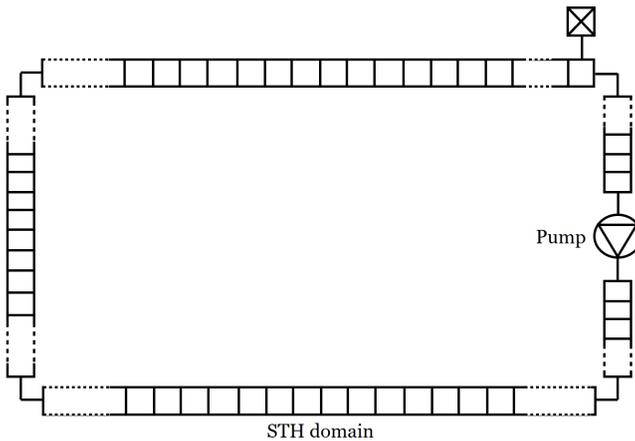


Figure A.6: Set-up of the STH stand-alone normalization of the closed pipe flow configuration. The legend of Figure A.2 applies.

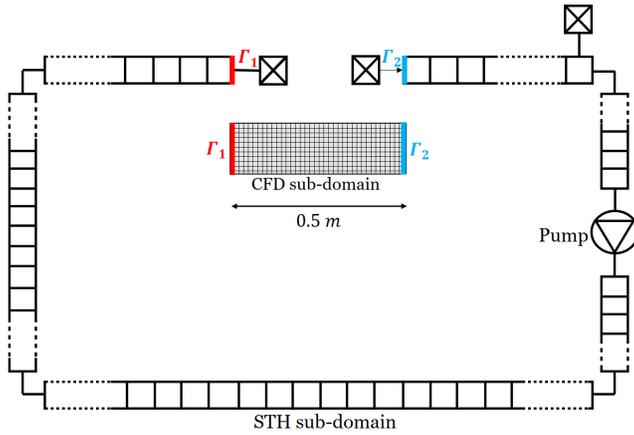


Figure A.7: Set-up of the closed pipe flow configuration for the domain decomposition coupled RELAP5/CFD model. The legend of Figure A.2 applies.

A.8 Results and analysis

For each of the flow configurations, the coupled model is first evaluated against the corresponding STH stand alone model in order to evaluate the implicit coupling methodology. Thereafter, the coupled RELAP5/ROM models are tested and compared with the coupled RELAP5/CFD models.

A.8.1 Open pipe flow test

The coupled RELAP5/CFD system is tested for a pressure drop of 0.20 bar over the open pipe. The results are compared with the results of the RELAP5 stand alone simulation. Figures A.8 and A.9 show the time evolution of the mass flow rate at interface Γ_2 and the area-averaged pressure at the same interface, respectively. Pressure oscillations are present at the beginning of the simulation, but they dissolve as the simulation time proceeds. The oscillations in pressure do not result in oscillations of the mass flow rate.

The coupled RELAP5/ROM is tested and compared with the coupled RELAP5/CFD model. Snapshots are collected every 0.1 s for a pressure drop of 0.20 bar over the open pipe. Figure A.10 shows the time-averaged basis projection error. The basis projection error is the relative error of the projected fields and the snapshots (Equation A.35). The time-averaged projection error of velocity is of the order $\mathcal{O}(10^{-2})$ and for pressure of the order $\mathcal{O}(10^{-1})$. Only a few modes are needed to represent the flow solutions due to the rapid decay of the error with increasing number of modes. Based on this, ten velocity and ten pressure modes are used to construct the reduced bases for this and all other test cases. The same number of eddy viscosity modes are used. In general, the number of POD modes should be determined based on the decay of eigenvalues.

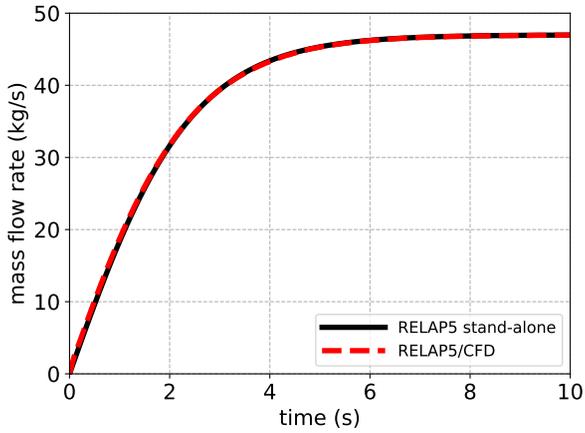


Figure A.8: Time evolution of the mass flow rate through the pipe at interface 2 in the abrupt pressure difference transient ($\Delta P = 0.20$ bar) for an open pipe flow configuration.

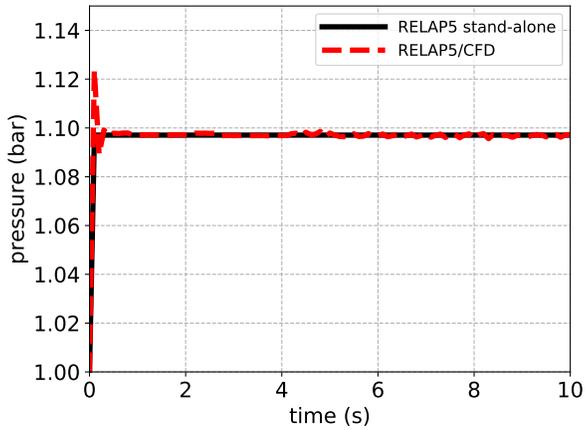


Figure A.9: Time evolution of the pressure at interface 2 in the abrupt pressure difference transient ($\Delta P = 0.20$ bar) for an open pipe flow configuration.

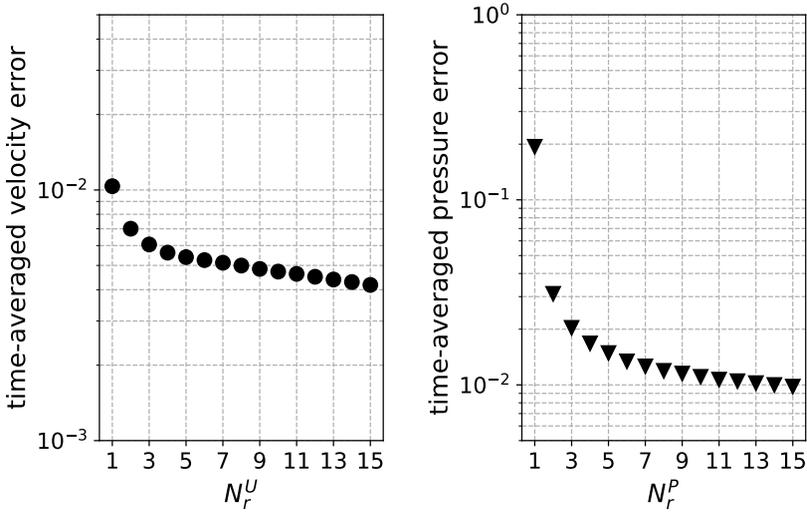
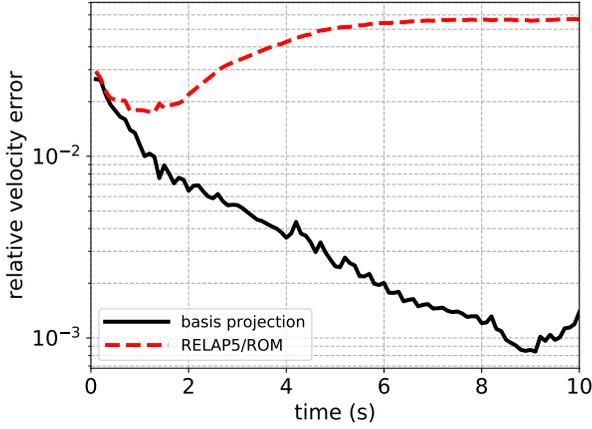


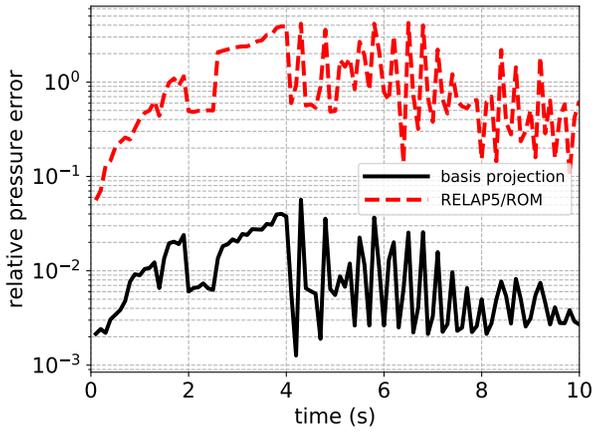
Figure A.10: The time-averaged relative basis projection error per number of modes for the open pipe flow test: (left) velocity; (right) pressure.

Furthermore, the time evolution of the relative error between the reconstructed velocity and pressure fields and the RELAP5/CFD results is plotted in Figure A.11, which are compared with the basis projection error. The relative velocity error increases over time after about 1 s of simulation time, while the basis projection error decreases over the whole simulation time. The pressure relative error is about two orders higher than the projection error. As velocity and pressure are coupled with the Pressure Poisson Equation, the velocity results are affected by the pressure results. Nevertheless, the relative velocity error stabilizes at about 5%.

We also compare the profiles of the velocity magnitude in the CFD sub-domain obtained with the RELAP5/ROM coupled model with the profiles obtained with the RELAP5/CFD model at $L_{CFD}/D = 0.5$ at $t = 1.0, 2.0$ and 10.0 s of simulation time in Figure A.12. Even though the relative L_2 error is of the order $\mathcal{O}(10^{-2})$, the velocity profiles visually overlap. Therefore, the velocity results are considered reliable for the application studied.



(a) velocity



(b) pressure

Figure A.11: Relative error and basis projection error for the CFD sub-domain in the abrupt pressure difference transient ($\Delta P = 0.20$ bar) for an open pipe flow configuration.

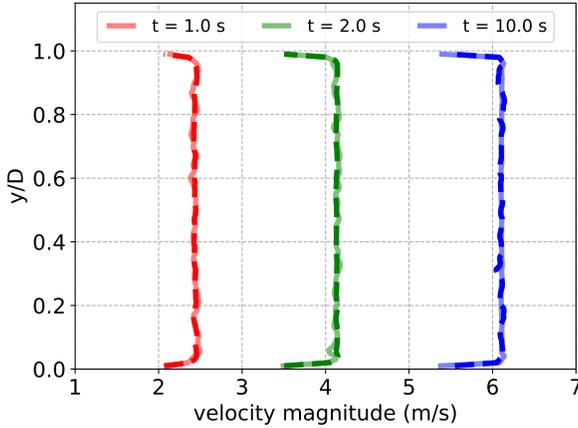


Figure A.12: Profiles of the velocity magnitude at $L_{CFD}/D = 0.5$ downstream of the inlet of the CFD sub-domain at $t = 1.0, 2.0$ and 10.0 s of simulation time in the abrupt pressure difference transient ($\Delta P = 0.20$ bar) for an open pipe flow configuration. Solid lines: RELAP5/CFD; Dashed lines: RELAP5/ROM.

RELAP5/ROM simulations are performed for pressure drops of 0.10, 0.15, 0.20, 0.21 and 0.23 bar over the whole pipe. Figures A.13 and A.14 show the time evolution of the mass flow rate at interface Γ_2 and the area-averaged pressure at the same interface, respectively. The RELAP5/ROM results overlap with the RELAP5/CFD results. Therefore, the ROM accurately predicts the CFD results for the entire parameter range even though the ROM is only constructed using the snapshots of the case with $\Delta P = 0.20$ bar. However, the ROM is only valid in a range around the parameter used for the training. As information about the flow for lower pressure drops is contained in the snapshots, the ROM can even be used to simulate a pressure drop of 0.10 bar. However, when increasing the pressure drop more than 0.23 bar, the results become unphysical as the flow characteristics are not contained in the POD modes.

The coupled RELAP5/CFD simulation of 10 seconds of simulation time takes about $2.8 \cdot 10^3$ seconds for one parameter on one Intel[®] Xeon[®] core. On the other hand, one coupled RELAP5/ROM simulation takes about $7.0 \cdot 10^2$ seconds on a single core. Therefore, the speed-up is about 4.0 times. The computational cost of the construction of the ROM (generating snapshots, calculating the POD modes and performing the Galerkin projection) is dominated by the time it takes to collect the snapshots. This cost is not taken into account in the calculation of the speed-up offered by the ROM itself.

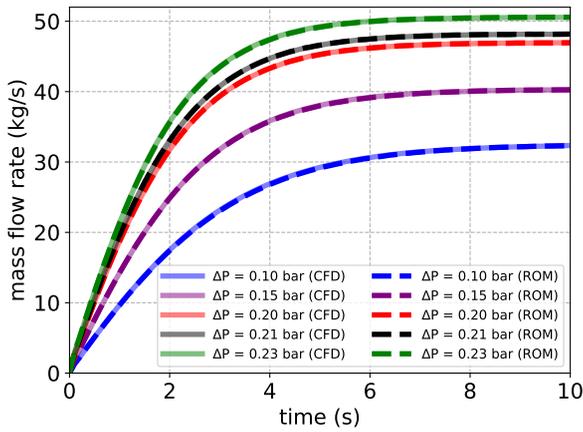


Figure A.13: Time evolution of the mass flow rate through the pipe at interface 2 in the abrupt pressure difference transient of different pressure drops for an open pipe flow configuration.

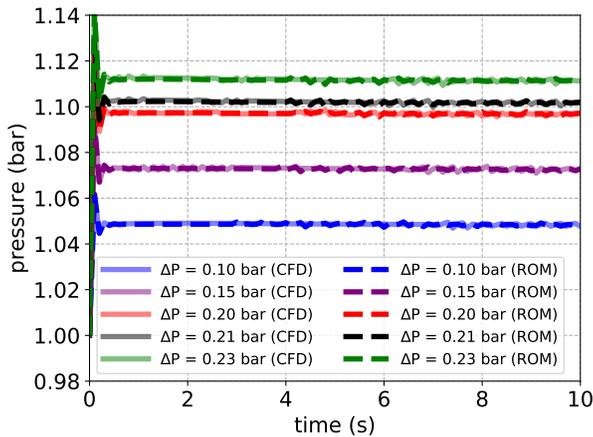


Figure A.14: Time evolution of the pressure at interface 2 in the abrupt pressure difference transient of different pressure drops for an open pipe flow configuration.

A.8.2 Open pipe flow reversal test

Coupled simulations are performed for the open pipe flow reversal test case. First, the coupled RELAP5/CFD results for a pressure drop of 0.20 bar are compared with RELAP5 stand-alone simulation as shown in Figure A.15 for the mass flow rate and Figure A.16 for pressure at interface Γ_2 . Similar to the previous test case, oscillations are present in the pressure. Especially at the beginning of the simulation, there is a spike in pressure obtained with the coupled system compared to the RELAP5 stand alone simulation. Oscillations also

occur when the drop in mass flow rate (between 12 and 13 seconds of simulations time) is the steepest. However, they disappear as the simulation time proceeds.

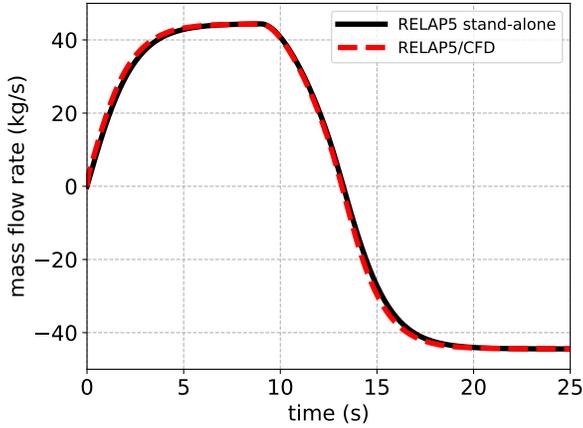


Figure A.15: Time evolution of the mass flow rate through the pipe at interface 2 in the abrupt forward and reverse pressure difference transient for a pressure drop of 0.20 bar over the open pipe.

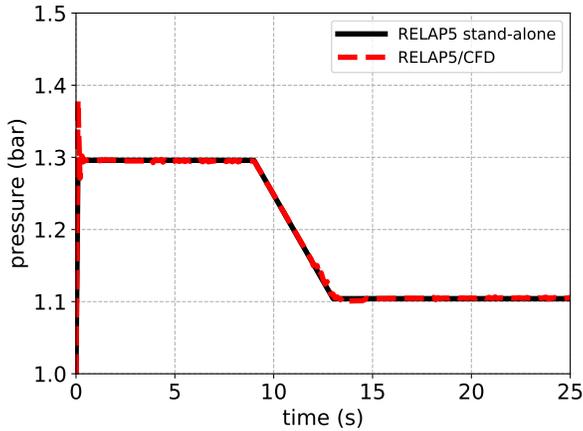


Figure A.16: Time evolution of the pressure at interface 2 in the abrupt forward and reverse pressure difference transient for a pressure drop of 0.20 bar over the open pipe.

Furthermore, the coupled RELAP5/ROM model is tested for several new values of the pressure drop over the pipe and compared to the coupled RELAP5/CFD results in Figures A.17 and A.18. The figures show that the ROM is capable of predicting the FOM results within the tested range of parameter values.

The coupled RELAP5/CFD simulation takes about $7.4 \cdot 10^3$ seconds for one parameter on a single core. On the other hand, one coupled RELAP5/ROM simulation takes about $1.6 \cdot 10^3$ seconds on a single core. Therefore, the speed-up is about 4.6 times.

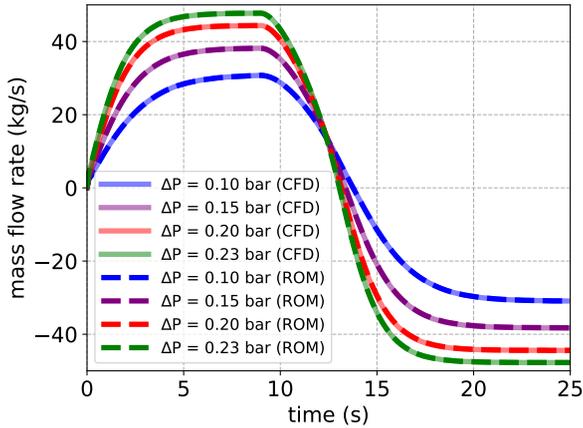


Figure A.17: Time evolution of the mass flow rate through the pipe at interface 2 in the abrupt forward and reverse pressure difference transient for different pressure drops for an open pipe flow configuration.

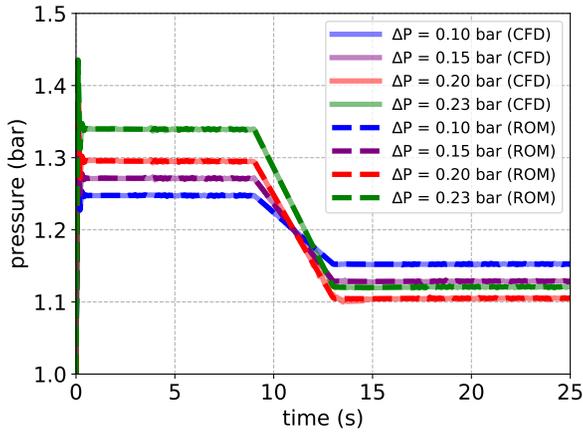


Figure A.18: Time evolution of the pressure at interface 2 in the abrupt forward and reverse pressure difference transient for different pressure drops for an open pipe flow configuration.

A.8.2.1 Long time integration

The coupled RELAP5/ROM model is also tested for long-term integration. Snapshots are collected using the RELAP5/CFD model for the reverse flow test case up to 10 seconds of simulation time. Thus, in total 100 snapshots are collected for the reduced basis construction. A RELAP5/ROM simulation is then performed up to 25 seconds of simulation time. The results for a pressure drop of 0.20 bar are shown in Figures A.19 and A.20, which are compared with the previous RELAP5/ROM model of which the reduced basis is constructed with all 250 snapshots. The RELAP5/ROM model fully predicts the behavior of the RELAP5/CFD model at coupling interface 2 as the results for the time evolution of the mass flow rate and pressure correspond to those of the RELAP5/ROM model based on 250 snapshots. It is important to note that this is true at the coupling interfaces.

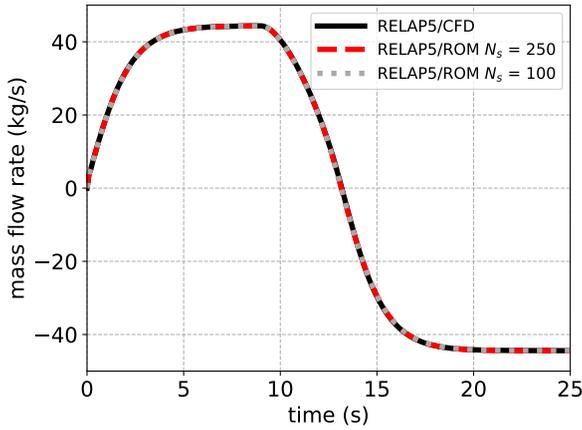


Figure A.19: Time evolution of the mass flow rate through the pipe at interface 2 in the long term integration test ($\Delta P = 0.20$ bar) for an open pipe flow configuration.

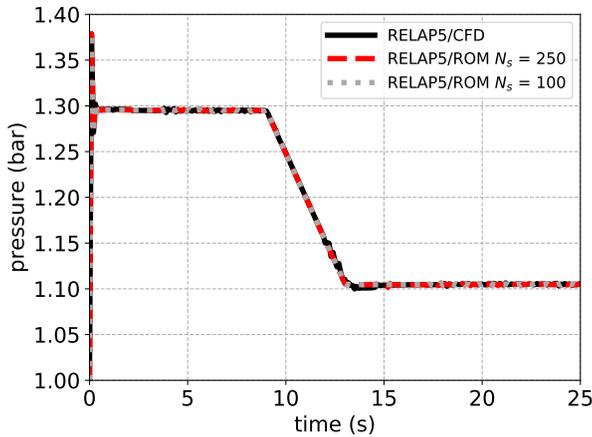
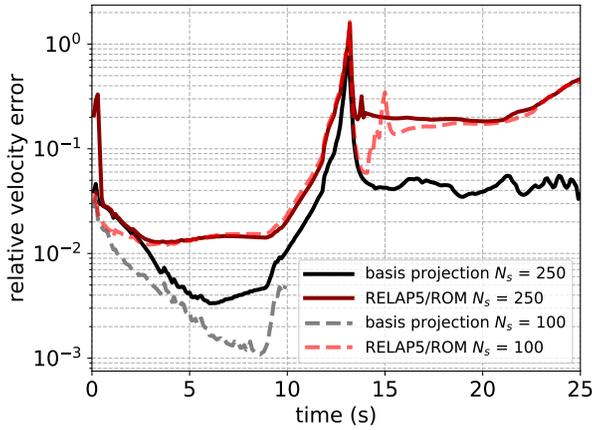


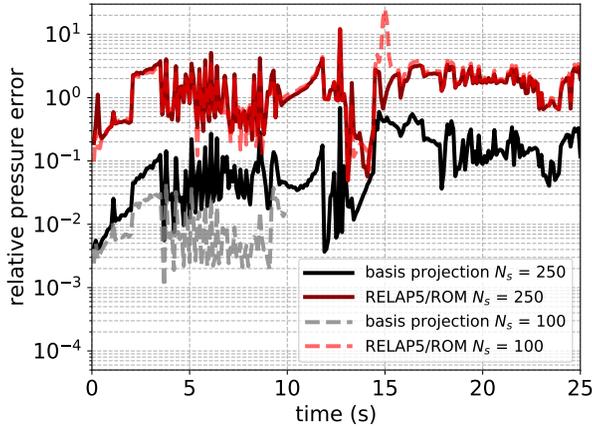
Figure A.20: Time evolution of the pressure at interface 2 in the long term integration test ($\Delta P = 0.20$ bar) for an open pipe flow configuration.

The relative error of the solution in the CFD sub-domain for the coupled RELAP5/ROM and coupled RELAP5/ROM on the long term integration are plotted in Figure A.21 for the velocity and pressure. For all models, the relative error spikes at about 13 seconds of simulation time. Around this time the decrease in mass flow rate is the steepest.

Even though only the first 100 snapshots, obtained till 10 seconds of simulation time, are included in the reduced order model in the case of long time integration, the RELAP5/ROM model is capable of reproducing the RELAP5/CFD results. This means that the overall mass flow through the pipe is conserved. Also the relative pressure errors are of the same order.



(a) velocity



(b) pressure

Figure A.21: Relative error and basis projection error in the CFD sub-domain for the long term integration test ($\Delta p = 0.20$ bar) for an open pipe flow configuration.

Figure A.22 shows the profiles of the velocity magnitude obtained with the RELAP5/CFD and RELAP5/ROM trained with 250 snapshots at $L_{CFD}/D = 0.5$ downstream of the inlet of the CFD sub-domain at $t = 1.0$, 10.0 and 25.0 s of simulation time. The results are visually overlapping, except at the final simulation time ($t = 25.0$ s). This corresponds to the relative velocity error plotted in Figure A.21a, which is about one order higher at $t = 25.0$ s compared to $t = 1.0$ and 10.0 s of simulation time.

Finally, the RELAP5/ROM model accurately determines the mass flow rate and pressure at coupling interface 2 during the flow reversal test and does not exhibit instabilities even outside the time domain in which snapshots were collected.

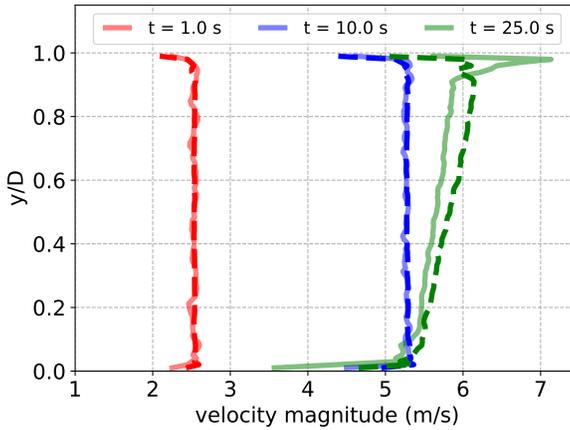


Figure A.22: Profiles of the velocity magnitude at $L_{CFD}/D = 0.5$ downstream of the inlet of the CFD sub-domain at $t = 1.0, 10.0$ and 25.0 s of simulation time of the reverse pressure difference transient for a pressure drop of 0.20 bar over the open pipe. Solid lines: RELAP5/CFD; Dashed lines: RELAP5/ROM.

A.8.3 Closed pipe flow test

The coupling methodology is also tested on the closed pipe flow test case. Snapshots are collected for a maximum pump rotor rotational speed of 100 rad/s and the results for the mass flow rate and pressure at coupling interface 2 are compared with a RELAP5 stand alone simulation in Figures A.23 and A.24, respectively.

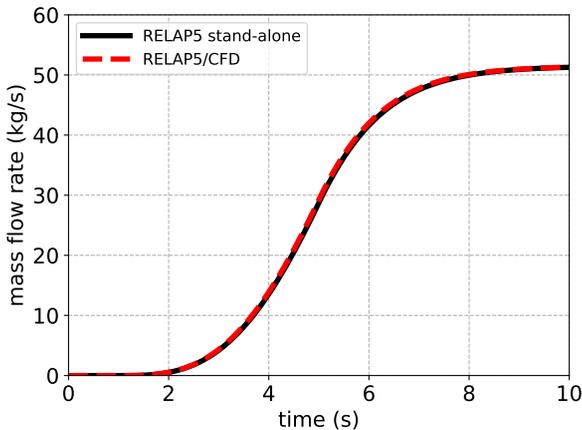


Figure A.23: Time evolution of the mass flow rate through the pipe at interface 2 for the closed pipe flow test with a maximum pump rotor rotational speed of 100 rad/s.

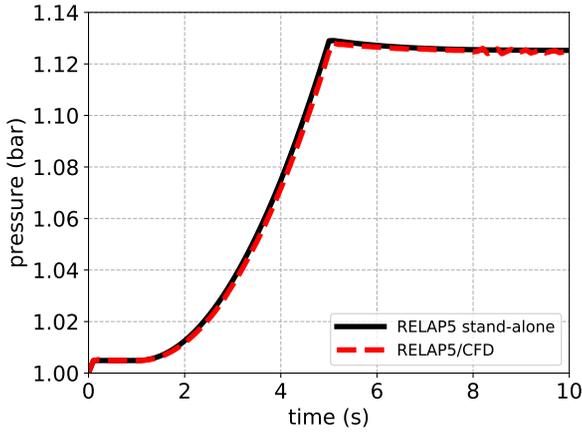


Figure A.24: Time evolution of the pressure at interface 2 for the closed pipe flow test with a maximum pump rotor rotational speed of 100 rad/s.

The closed loop test case is also used to test the coupled RELAP5/ROM model, of which the reduced basis is constructed with snapshots obtained for $\omega = 100$ rad/s, on a parametric problem. Figures A.25 and A.26 show the mass flow rate and the pressure evaluated at interface Γ_2 for the maximum pump rotor rotational speed of 80, 90, 100 and 110 rad/s, respectively. The results are compared with those obtained with the RELAP5/CFD model for the same rotational speeds.

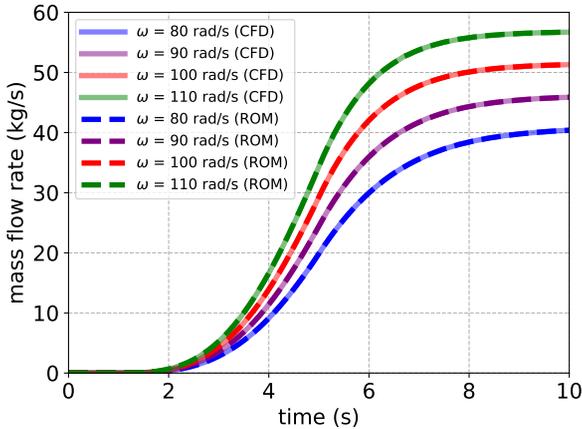


Figure A.25: Time evolution of the mass flow rate through the pipe at interface 2 for the closed pipe flow test for different maximum pump rotor rotational speeds.

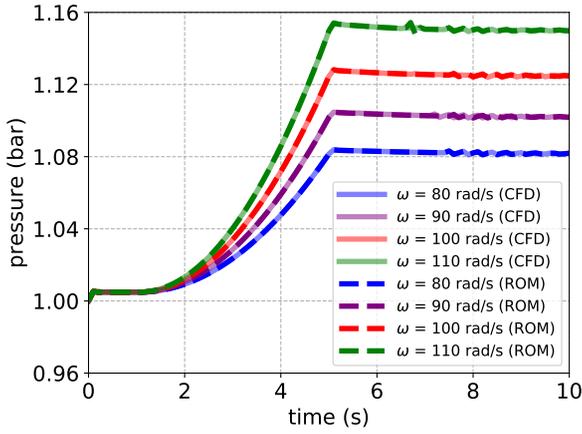
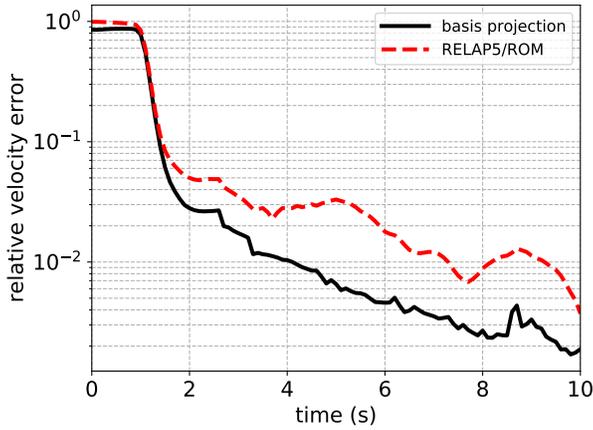
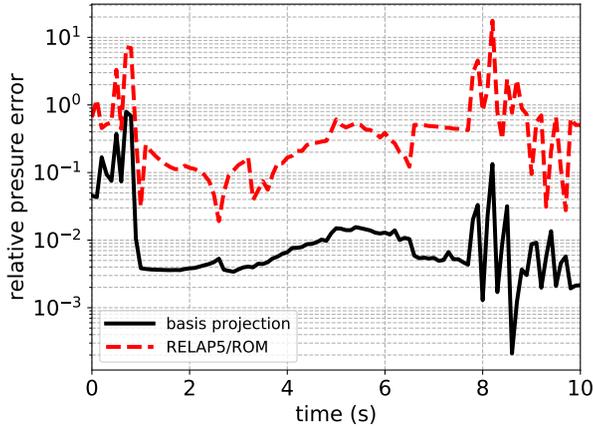


Figure A.26: Time evolution of the pressure at interface 2 for the closed pipe flow test for different maximum pump rotor rotational speeds.

The relative velocity and pressure errors are plotted in Figure A.27. The velocity relative error is the largest at the beginning of the simulation. As the flow is initially at rest, a small error in the reconstructed flow field results in large relative error. As soon as the mass flow rate increases, the relative error drops. This also indicates that the closed pipe flow test case is numerically more stable than the open pipe flow test, where the relative velocity error increased as function of time as shown in Figure A.11a. The relative pressure error is about two orders larger than the projection error.



(a) velocity



(b) pressure

Figure A.27: Relative error and basis projection error in the CFD sub-domain for the closed pipe flow test with a maximum pump rotor rotational speed of 100 rad/s.

Figure A.28 shows the profiles of the velocity magnitude in the CFD sub-domain obtained with the RELAP5/CFD and RELAP5/ROM coupled models at $L_{CFD}/D = 0.5$ at $t = 1.0, 5.0$ and 10.0 s of simulation time. The RELAP5/ROM velocity profiles are in good agreement with the RELAP5/CFD profiles. At $t = 5.0$ s the profiles are slightly deviating from each other, while they are almost fully overlapping at $t = 10.0$ s. This corresponds to Figure A.27a, which shows that the velocity relative error decreases towards the final simulation time.

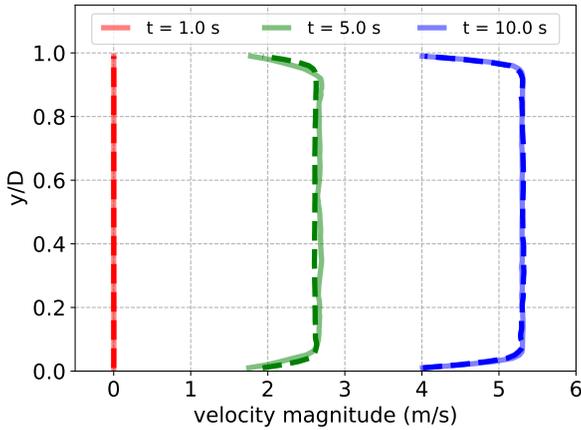
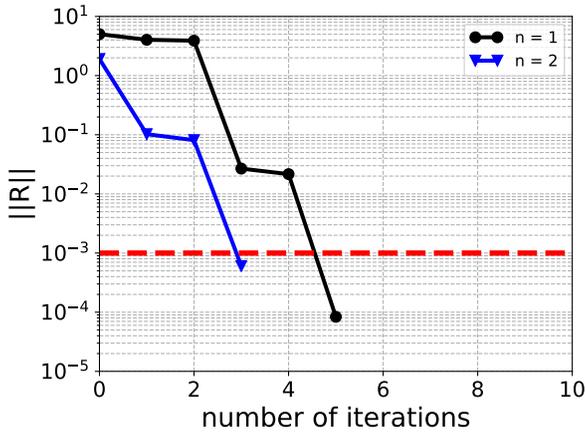


Figure A.28: Profiles of the velocity magnitude at $L_{CFD}/D = 0.5$ downstream of the inlet of the CFD sub-domain at $t = 1.0, 5.0$ and 10.0 s of simulation time of the closed pipe flow test with a maximum pump rotor rotational speed of 100 rad/s. Solid lines: RELAP5/CFD; Dashed lines: RELAP5/ROM.

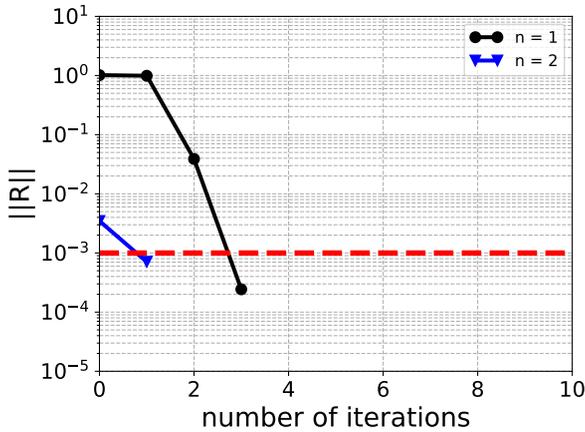
Finally, one coupled RELAP5/CFD simulation for the closed pipe flow test takes $3.2 \cdot 10^3$ seconds, while a coupled RELAP5/ROM simulation takes $6.5 \cdot 10^2$ seconds to complete. Thus, the obtained speed-up is about 3.5 times.

A.8.4 Convergence history

The performance of the implicit coupling algorithm is analyzed by checking the number of iterations needed to reach a residual norm below 10^{-3} . Figure A.29 shows the convergence history for the first two coupling time steps for the open and closed pipe flow tests. In the case of the open pipe flow test case, the first time step requires five iterations for the residual to drop below the threshold and the second time step requires three iterations. For the closed loop test case, the number of iterations is reduced to three for the first time step and only one for the second time step. Therefore, the convergence rate is higher for the closed pipe flow test than the open pipe flow test. Previous results of the relative error (Figure A.11) also showed that this test case is numerically more stable than the open pipe flow test. For both cases the Jacobian is not available in the first iteration of the first time step, which explains the difference in number of iterations for the first and second time step. In the case of the closed pipe flow test, the Jacobian computed in the first time step is also used in the second time step as Equation A.11 is satisfied [260]. Therefore, the convergence criterion is met with only one additional iteration.



(a) open pipe flow test case



(b) closed pipe flow test case

Figure A.29: Interface convergence history for the first two time steps.

A.9 Discussion

The RELAP5/CFD models exhibit numerical perturbations in the form of oscillations throughout the coupled simulations. As concluded in previous works on coupled CFD codes with 1D system codes [102, 260], these perturbations are caused by the overestimation of the mass flow rate at the coupling interfaces in the first few time steps of the simulations. A change in pressure drop will immediately affect the whole solution domain as the fluid density does not change with pressure unlike in compressible fluids. These small oscillations are also present in the results of the RELAP5/ROM models. Nevertheless, the ROMs remain stable, i.e. the ROM results do not blow up over time, even though small perturbations in reduced order models can lead to unphysical results [149]. Toti et al. [258] concluded that

reducing the time step of the coupled simulations affects how rapidly numerical oscillations are damped during the transients. However, the maximum pressure oscillation amplitude is independent from the time step.

For all test cases, only one coupled RELAP5/CFD simulation for a single parameter value has been performed for collecting the snapshots. The ROMs are capable of predicting solutions for new parameter values and for long time integration as (part of) the flow features for these new cases are similar to the snapshots. However, to be able to analyze and to improve the accuracy of the ROM for a larger parameter range, snapshots from multiple simulations for different parameter values are required to construct the reduced basis spaces. When several sets of snapshots are required for the POD, one can optimize the POD procedure by using a nested POD approach [87].

In this work, a uniform velocity profile is implemented at the inlet of the CFD sub-domain. Toti [260] et al. considered in their work for open and closed pipe flow test that the error introduced by the uniform profile is small as the velocity distribution across the section of the pipe is fairly flat in the case of fully developed turbulent pipe flow. However, as one of the main purposes of developing coupled models that accurately quantify the flow fields in specific parts of the computational domain, it is better to take into account the curvature of the velocity profiles in the case of complex flow problems.

The relative velocity error is about two orders lower at the final simulation time in the case of the closed pipe flow test compared to the open pipe flow test. The closed loop is less prone to numerical instabilities due to the absence of interruptions in the STH domain. Therefore, perturbations are transported throughout the whole STH domain within a single coupling iteration, while several coupling iterations are required in the case of the open pipe test as the upstream and downstream parts of STH sub-domain are hydraulically decoupled. Because of that, the convergence rate is generally higher for the closed pipe than for the open pipe test. Reduced order models are sensitive to these numerical instabilities what explains the difference in relative errors for these two test cases. As system analysis of nuclear installations, like MYRRHA, are dealing mainly with closed cooling loops, it is advantageous that numerical instabilities are less prone for closed loop systems.

Furthermore, the results for pressure fields calculated in the CFD sub-domain are about two orders worse than those for the velocity fields. This has been observed in the previous work of Stabile et al. [241]. Other pressure stabilization methods, rather than using the PPE method, can be used to improve the pressure fields.

A.10 Conclusions and outlook

The best-estimate system thermal-hydraulics code RELAP5 is coupled with the finite volume CFD solver OpenFOAM and its reduced order model. The codes are coupled implicitly by a domain decomposition coupling algorithm in which the hydraulics variables are exchanged between the sub-domains at the coupling boundary interfaces.

The ROM is constructed with a finite volume based POD-Galerkin projection method. The average velocity determined at the single junction of the STH sub-domain at coupling interface 1 is imposed at the inlet boundary of the reduced order model with a boundary control method, namely a penalty method.

Academic tests are carried out on open and closed pipe flow configurations. The coupled RELAP5/ROM models accurately predict the time evolution of the mass flow rate and pressure results of the coupled RELAP5/CFD models at one of the coupling interfaces. Also for new conditions, the RELAP5/ROM models are capable of reproducing the behavior of

the RELAP5/CFD models at the coupling interface. In addition, the RELAP5/ROM model for reversed flow in a closed loop flow test case performs well for long time integration. The effect of the number of POD modes on the accuracy of the RELAP5/ROM results is not investigated in this work.

The pressure results exhibit numerical perturbations in the form of oscillations throughout the coupled simulations, both with the CFD solver and the reduced order model. Nevertheless, they do not lead to a blow up of the RELAP5/ROM results, even though small perturbations in reduced order models can, generally, lead to unphysical results.

Finally, the coupled RELAP5/ROM simulations are about 3 to 5 times faster than the coupled RELAP5/CFD simulations performed on a single Intel[®] Xeon[®] core. Therefore, it is shown that the computational cost of coupled STH/CFD models can be reduced by replacing the CFD solver by a reduced order model. Furthermore, the coupled RELAP5/ROM model can be used to study a number of different conditions at a lower computational cost compared to the coupled RELAP5/CFD model.

In future work, the methodology needs to be extended to reduced order models for turbulent buoyancy driven flows, for which the discretized momentum and energy equations are coupled in a two-way manner [244, 267]. In addition, the models need to be adjusted for low-Prandtl number fluid flows [249], such as LBE. The coupled models also need to be validated against experimental results, such as a loss of flow due to a pump trip transient in the TALL-3D experimental facility [259]. Moreover, the coupling could be extended to parallel computing to speed up the simulations.

List of Figures

1.1	Conceptual scheme of the MYRRHA accelerator-driven system. (Figure adopted from [62, 257].)	2
1.2	Sketch of the MYRRHA reactor design revision 1.6 [76].	4
1.3	Schematic cross-section of the primary system of the MYRRHA reactor. (Figure adopted from [260].)	4
1.4	A pyramid representing the hierarchy of modeling approaches based on the computational cost and the amounts of resolved versus modeled physics. (Figure adopted from [119, 292].)	6
1.5	Schematic representation of the reduced order modeling approach as a compromise between the level of accuracy and the model size. (Figure adopted from [161, 231].)	8
2.1	Two-dimensional collocated grid with the location of the unknowns at the center of a cell volume P and neighboring cell volume N . Δx and Δy the cell lengths and the arrows indicate the mass fluxes through the cell faces e , w , s and n of cell P .	18
2.2	Schematic representation of the linear interpolation of velocity and pressure from the cell center to the face of the cell. \mathbf{d} is the distance vector between the cell centers P and N .	20
3.1	A sketch of the geometry of the heated lid-driven cavity problem including boundary conditions.	46
3.2	(Left) the background velocity field in m/s and (right) the corresponding pressure field in m^2/s^2 for the lid-driven cavity simulation at $t = 50$ s.	48
3.3	Evolution of the temperature with $T_{lid}^{FOM} = 1$ (base case) inside the cavity for time instances $t = 5, 50$ and 100 s.	48
3.4	Normalized POD eigenvalues of the base case with time-independent boundary conditions.	49
3.5	Cross-section of the temperature field for $x = 0.5$ m at $t = 100$ s. $T_{lid}^{FOM} = 1$ is the base case and the ROM is tested for $T_{lid}^{ROM} = 0.5, 1, 2$ and 10 with the penalty method.	50
3.6	Time evolution of the basis projection error and the relative error in the ROM with respect to the FOM for $T_{lid}^{ROM} = 0.5, 1, 2$ and 10 .	51
3.7	Cross-section of the temperature field at $x = 0.5$ m at $t = 80$ s. The time-dependent BCs are listed in the legend with $f = 0.01$ Hz.	52
3.8	Cross-section of the temperature field at $y = 0.5$ m at $t = 80$ s. The time-dependent BCs are listed in the legend with $f = 0.01$ Hz.	52

3.9	Time evolution of the basis projection error and the relative error in the ROM with respect to the FOM. The time-dependent BCs are listed in the legend with $f = 0.01$ Hz.	53
3.10	Cross-section of the temperature field at $x = 0.5$ m at $t = 80$ s. The time-dependent BCs are listed in the legend with $f = 0.01$ Hz.	54
3.11	Cross-section of the temperature field at $y = 0.5$ m at $t = 80$ s. The time-dependent BCs are listed in the legend with $f = 0.01$ Hz.	55
3.12	Time evolution of the basis projection error and the relative error in the ROM with respect to the FOM for different frequencies.	55
3.13	Comparison of the temperature field for the FOM (left) and ROM solutions (middle) at $t = 100$ s for parameter set 1 - 4 (from top to bottom). The absolute difference between the ROM and corresponding FOM is plotted on the right. All ROMs are obtained with 31 modes.	56
4.1	Sketch of the geometry of the 2D square cavity with moving top lid including boundary conditions.	71
4.2	(Left) sketch of the geometry and mesh of the Y-junction test case including boundary conditions. (Right) close up of the mesh in different zones. . . .	73
4.3	Inlet velocity boundary conditions for the full order model for the Y-junction test case.	73
4.4	Inlet velocity boundary conditions for the reduced order model for the Y-junction test case.	74
4.5	The lifting functions for velocity for the Y-junction test case.	75
4.6	Time evolution of the basis projection error and the relative L_2 error of velocity between the FOM and ROM with the lifting function and penalty method for the lid-driven cavity problem.	77
4.7	Time evolution of the basis projection error and the relative L_2 error of pressure between the FOM and ROM with the lifting function and penalty method for the lid-driven cavity problem.	77
4.8	Time evolution of the relative L_2 error of kinetic energy between the FOM and ROM with the lifting function and penalty method for the lid-driven cavity problem.	78
4.9	Comparison of the full order velocity magnitude fields (1st column), the ROM fields obtained with the lifting function method (2nd column) and penalty method (4th column) and the difference between the FOM and ROM fields obtained with the lifting function method (3rd column) and penalty method (5th column) at $t = 0.2, 1, 5$ and 10 s (from top to bottom) for the lid-driven cavity problem.	79
4.10	Comparison of the full order pressure fields (1st column), the ROM fields obtained with the lifting function method (2nd column) and penalty method (4th column) and the difference between the FOM and ROM fields obtained with the lifting function method (3rd column) and penalty method (5th column) at $t = 0.2, 1, 5$ and 10 s (from top to bottom) for the lid-driven cavity problem.	80
4.11	First 5 POD modes for (top) velocity, (middle) velocity with homogeneous BCs and (bottom) pressure for the Y-junction flow problem.	82
4.12	The time-averaged relative L_2 error per number of (left) velocity modes, N_v^u , and (right) number of pressure modes, N_p^p , for the Y-junction test case.	83

4.13	Time evolution of the relative L_2 error of velocity between the FOM and ROM with the lifting function and penalty method for the Y-junction flow problem.	84
4.14	Time evolution of the relative L_2 error of pressure between the FOM and ROM with the lifting function and penalty method for the Y-junction flow problem.	84
4.15	Time evolution of the relative L_2 error of kinetic energy between the FOM and ROM with the lifting function and penalty method for the Y-junction flow problem.	85
4.16	Comparison of the full order velocity magnitude fields (1st column), the ROM fields obtained with the lifting function method (2nd column) and penalty method (4th column) and the difference between the FOM and ROM fields obtained with the lifting function method (3rd column) and penalty method (5th column) at $t = 3, 9$ and 18 s (from top to bottom) for the Y-junction flow problem.	86
4.17	Comparison of the full order pressure fields (1st column), the ROM fields obtained with the lifting function method (2nd column) and penalty method (4th column) and the difference between the FOM and ROM fields obtained with the lifting function method (3rd column) and penalty method (5th column) at $t = 3, 9$ and 18 s (from top to bottom) for the Y-junction flow problem.	87
5.1	A sketch of the geometry of the 2D square cavity with differentially heated walls.	95
5.2	The lifting functions for temperature.	96
5.3	Cumulative eigenvalues for velocity and temperature.	97
5.4	The time-averaged relative basis projection error: (left) per number of velocity modes (N_r^u); (right) per number of temperature modes (N_r^θ).	98
5.5	First five POD modes for velocity (top) and temperature with homogeneous BCs (bottom).	98
5.6	Time evolution of the relative error between the FOM and ROM with the lifting function for all parameter sets and the basis projection error.	99
5.7	Comparison of the full order (FOM) velocity magnitude (1st column) and temperature (4th column) fields with the reduced order (ROM) fields (2nd and 5th column) and the difference between the FOM and ROM fields (3rd and 6th column) for parameter set 3 at $t = 1$ s (top), 5 s (middle) and 10 s (bottom).	101
6.1	A sketch of the geometry of the backward-facing step and the precursor domain.	115
6.2	A 2D sketch of the geometry of the backward-facing step divided into several zones and including boundaries.	115
6.3	Velocity magnitude (top), shifted kinematic pressure (middle) and temperature fields obtained with the RANS simulations for $Ri = 0.2$ (left) and $Ri = 0.4$ (right), respectively.	117
6.4	Relative basis projection error of all snapshots for different number of modes: (left) velocity relative error; (right) temperature relative error.	118

6.5	Relative prediction error for two sets of penalty factors and different Richardson numbers and $N_r^\theta = 5$. Case A: $\tau_U = 1, \tau_{\nabla\theta} = 1, \tau_\theta = 1$. Case B: $\tau_U = 10^6, \tau_{\nabla\theta} = 10^6, \tau_\theta = 10^6$. (Left) Velocity relative error; (right) temperature relative error.	119
6.6	Profiles of the normalized stream-wise velocity component at several locations downstream of the step for $Ri = 0.2$ and $Ri = 0.4$, respectively, obtained by performing RANS and ROM simulations. The legend depicts the number of temperature modes, N_r^θ , used for the ROM.	120
6.7	Profiles of the normalized wall normal velocity component at several locations downstream of the step for $Ri = 0.2$ and $Ri = 0.4$, respectively, obtained by performing RANS and ROM simulations. The legend depicts the number of temperature modes, N_r^θ , used for the ROM.	120
6.8	Normalized temperature profiles at several locations downstream of the step for $Ri = 0.2$ and $Ri = 0.4$, respectively, obtained by performing RANS and ROM simulations. The legend depicts the number of temperature modes, N_r^θ , used for the ROM.	120
6.9	Local Stanton number at the heater determined by the RANS and ROM simulations for several Richardson numbers.	121
6.10	Skin friction distribution downstream of the backward-facing step determined by the RANS and ROM simulations for several Richardson numbers.	122
6.11	Relative prediction error for all Richardson numbers with $N_r^\theta = 5$ and $N_r^\theta = 8$ temperature modes, respectively, used for the construction of the ROM: (left) velocity relative error; (right) temperature relative error.	123
6.12	Local Stanton number at the heater determined by the ROM compared with those determined by RANS simulations for several Richardson numbers.	123
6.13	Skin friction distribution downstream of the backward-facing step determined by the ROM compared with those determined by RANS simulations for several Richardson numbers.	124
6.14	Profiles of the normalized stream-wise velocity component at several locations downstream of the step for $Ri = 0.12$ and $Ri = 0.36$, respectively, obtained by performing RANS and ROM simulations. The legend depicts the number of temperature modes, N_r^θ , used for the ROM.	124
6.15	Profiles of the normalized wall normal velocity component at several locations downstream of the step for $Ri = 0.12$ and $Ri = 0.36$, respectively, obtained by performing RANS and ROM simulations. The legend depicts the number of temperature modes, N_r^θ , used for the ROM.	125
6.16	Normalized temperature profiles at several locations downstream of the step for $Ri = 0.12$ and $Ri = 0.36$, respectively, obtained by performing RANS and ROM simulations. The legend depicts the number of temperature modes, N_r^θ , used for the ROM.	125
6.17	Comparison of the ratio of eddy viscosity to kinematic viscosity at several locations downstream of the step determined by the RANS and ROM simulations for different Richardson numbers. The legend depicts the number of temperature modes, N_r^θ , used for the ROM.	126
7.1	Two-dimensional collocated grid with the location of the unknowns at the cell center and the faces of a cell volume.	133

7.2	Sketch of a two-dimensional collocated grid with the location of the cell-centered pressure (left), the cell-centered velocity (middle) and the face-centered velocity (right).	135
7.3	Two-dimensional collocated grid with the location of the velocity at the cell center and the faces of a cell volume of a cell near the boundary face of the domain.	137
7.4	Sketch of the geometry and mesh of the 2D square cavity with a moving top lid.	147
7.5	Sketch of the geometry and mesh of the 2D open cavity.	148
7.6	Velocity profiles for the lid driven cavity flow case at final simulation time: (left) normalized velocity component in the x -direction at $x/L = 0.5$; (right) normalized velocity component in the y -direction at $y/L = 0.5$	150
7.7	Normalized pressure profiles for the lid driven cavity flow case at final simulation time: (left) at $x/L = 0.5$; (right) at $y/L = 0.5$	150
7.8	Eigenvalues as function of the number of modes for the lid driven cavity flow case.	151
7.9	Relative cell-centered velocity error as a function of time for different number of modes for the lid driven cavity flow case. Dashed lines: basis projection error (projecting snapshots onto truncated basis).	152
7.10	Relative pressure error as a function of time for different number of modes for the lid driven cavity flow case. Dashed lines: basis projection error (projecting snapshots onto truncated basis).	153
7.11	Time-averaged relative basis projection and prediction errors of the lid driven cavity flow problem.	154
7.12	Computational times in seconds as function of number of modes for the lid driven cavity flow case.	155
7.13	Speedup in computational time of the ROM compared to the FOM in seconds as function of number of modes for the lid driven cavity flow case.	156
7.14	From top to bottom: Snapshots obtained at $t = 0; 0.5; 1.0; 2.0$ s with the consistent flux method: (left) cell-centered velocity magnitude in m/s; (right) pressure in Pa.	157
7.15	Eigenvalues for the open cavity flow case.	158
7.16	Relative cell-centered velocity error as a function of time for different number of modes for the open cavity flow case. Dashed lines: basis projection error (projecting snapshots onto truncated basis).	159
7.17	Relative pressure error as a function of time for different number of modes for the open cavity flow case. Dashed lines: basis projection error (projecting snapshots onto truncated basis).	160
7.18	Time-averaged relative basis projection and prediction errors of the open cavity flow problem.	161
7.19	Computational times in seconds as function of number of modes the open cavity flow case.	162
7.20	Speedup in computational time of the ROM compared to the FOM in seconds as function of number of modes for the open cavity flow case.	163
A.1	Flowchart of the generic implicit coupling numerical algorithm with the interface Quasi-Newton algorithm. $\epsilon > 0$ is the given tolerance, t denotes time and t_{end} is the final simulation time.	202

A.2	Variable exchanges over the coupling interfaces between the STH sub-domains and the CFD sub-domain.	204
A.3	Pressure extrapolation at coupling interface 2. The legend of Figure A.2 applies.	204
A.4	Mesh of the CFD sub-domain consisting of a circular pipe of length L_{CFD} and diameter D	210
A.5	RELAP5 stand-alone nodalization of the open pipe flow configuration (top) and the domain decomposition of the open pipe flow configuration for the coupled RELAP5/CFD model (bottom). The legend of Figure A.2 applies.	211
A.6	Set-up of the STH stand-alone normalization of the closed pipe flow configuration. The legend of Figure A.2 applies.	212
A.7	Set-up of the closed pipe flow configuration for the domain decomposition coupled RELAP5/CFD model. The legend of Figure A.2 applies.	213
A.8	Time evolution of the mass flow rate through the pipe at interface 2 in the abrupt pressure difference transient ($\Delta P = 0.20$ bar) for an open pipe flow configuration.	214
A.9	Time evolution of the pressure at interface 2 in the abrupt pressure difference transient ($\Delta P = 0.20$ bar) for an open pipe flow configuration.	214
A.10	The time-averaged relative basis projection error per number of modes for the open pipe flow test: (left) velocity; (right) pressure.	215
A.11	Relative error and basis projection error for the CFD sub-domain in the abrupt pressure difference transient ($\Delta P = 0.20$ bar) for an open pipe flow configuration.	216
A.12	Profiles of the velocity magnitude at $L_{CFD}/D = 0.5$ downstream of the inlet of the CFD sub-domain at $t = 1.0, 2.0$ and 10.0 s of simulation time in the abrupt pressure difference transient ($\Delta P = 0.20$ bar) for an open pipe flow configuration. Solid lines: RELAP5/CFD; Dashed lines: RELAP5/ROM.	217
A.13	Time evolution of the mass flow rate through the pipe at interface 2 in the abrupt pressure difference transient of different pressure drops for an open pipe flow configuration.	218
A.14	Time evolution of the pressure at interface 2 in the abrupt pressure difference transient of different pressure drops for an open pipe flow configuration.	218
A.15	Time evolution of the mass flow rate through the pipe at interface 2 in the abrupt forward and reverse pressure difference transient for a pressure drop of 0.20 bar over the open pipe.	219
A.16	Time evolution of the pressure at interface 2 in the abrupt forward and reverse pressure difference transient for a pressure drop of 0.20 bar over the open pipe.	219
A.17	Time evolution of the mass flow rate through the pipe at interface 2 in the abrupt forward and reverse pressure difference transient for different pressure drops for an open pipe flow configuration.	220
A.18	Time evolution of the pressure at interface 2 in the abrupt forward and reverse pressure difference transient for different pressure drops for an open pipe flow configuration.	221
A.19	Time evolution of the mass flow rate through the pipe at interface 2 in the long term integration test ($\Delta P = 0.20$ bar) for an open pipe flow configuration.	222
A.20	Time evolution of the pressure at interface 2 in the long term integration test ($\Delta P = 0.20$ bar) for an open pipe flow configuration.	222

A.21 Relative error and basis projection error in the CFD sub-domain for the long term integration test ($\Delta p = 0.20$ bar) for an open pipe flow configuration. 223

A.22 Profiles of the velocity magnitude at $L_{CFD}/D = 0.5$ downstream of the inlet of the CFD sub-domain at $t = 1.0, 10.0$ and 25.0 s of simulation time of the reverse pressure difference transient for a pressure drop of 0.20 bar over the open pipe. Solid lines: RELAP5/CFD; Dashed lines: RELAP5/ROM. 224

A.23 Time evolution of the mass flow rate through the pipe at interface 2 for the closed pipe flow test with a maximum pump rotor rotational speed of 100 rad/s. 224

A.24 Time evolution of the pressure at interface 2 for the closed pipe flow test with a maximum pump rotor rotational speed of 100 rad/s. 225

A.25 Time evolution of the mass flow rate through the pipe at interface 2 for the closed pipe flow test for different maximum pump rotor rotational speeds. 225

A.26 Time evolution of the pressure at interface 2 for the closed pipe flow test for different maximum pump rotor rotational speeds. 226

A.27 Relative error and basis projection error in the CFD sub-domain for the closed pipe flow test with a maximum pump rotor rotational speed of 100 rad/s. 227

A.28 Profiles of the velocity magnitude at $L_{CFD}/D = 0.5$ downstream of the inlet of the CFD sub-domain at $t = 1.0, 5.0$ and 10.0 s of simulation time of the closed pipe flow test with a maximum pump rotor rotational speed of 100 rad/s. Solid lines: RELAP5/CFD; Dashed lines: RELAP5/ROM. 228

A.29 Interface convergence history for the first two time steps. 229

List of Tables

2.1	Overview of the reduced order modeling challenges per chapter.	37
3.1	Parameter sets for time-dependent Dirichlet BCs defined by Equation 3.27 .	47
3.2	Cumulative eigenvalues (CV) of the base case for temperature.	49
4.1	Computational time (clock time) for the FOM simulation, POD, calculating reduced matrices offline (Matrices), determining penalty factor with iterative method (Penalty) and ROM simulation.	78
4.2	The cumulative eigenvalues for the Y-junction test case. The second and third columns report the cumulative eigenvalues (total of the first 20 modes) for the velocity and pressure fields, respectively.	81
4.3	Computational time (clock time) for the FOM simulation, POD modes, calculating reduced matrices offline (Matrices), determining penalty factor with iterative method (Penalty) and ROM simulation.	85
5.1	Parameter sets for the temperature BCs.	95
6.1	Low-Reynolds Launder-Sharma k - ε model constants and source terms. . .	107
6.2	Damping coefficients with $Re_t = \frac{k^2}{\nu\varepsilon}$	107
6.3	The number of cells, N , along the horizontal (x_1 , x_2 and x_3) and vertical sides (y_1 , y_2) of each zone depicted in Figure 6.2.	115
6.4	y^+ values at the heater compared with the results of [230].	116
7.1	Computational details for the lid driven cavity and open cavity flow problems.	148

