System-Theoretical Model Reduction for Reservoir Simulation and Optimization

System-Theoretical Model Reduction for Reservoir Simulation and optimization

PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. dr. ir. J.T. Fokkema,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op

maandag 09 februari 2009 om 10:00 uur

door

Renato MARKOVINOVIĆ

elektrotechnisch ingenieur

geboren te Živinice, Bosnië en Herzegovina

Dit proefschrift is goedgekeurd door de promotor:
Prof. dr. ir. J.D. Jansen

Samenstelling promotiecommissie:

Rector Magnificus,	voorzitter
Prof. dr. ir. J.D. Jansen,	Technische Universiteit Delft and Shell International E&P, promotor
Prof. dr. ir. P.M.J. Van den Hof,	Technische Universiteit Delft
Prof. dr. ir. O.H. Bosgra,	Technische Universiteit Delft
Prof. dr. W.R. Rossen,	Technische Universiteit Delft
Prof. dr. ir. J.M.A. Scherpen,	Rijksuniversiteit Groningen
Prof. dr. ir. A.C.P.M. Backx,	Technische Universiteit Eindhoven
dr. J.C. Vink,	Shell International Exploration and Production

The research for this thesis was financially supported by Shell International Exploration and Production as part of the VALUE project which received support of the Dutch Government in the form of Senter/Novem subsidy under the "Technologische Samenwerking" programme (projectnumber TSGE3134). Moreover, the research formed part of ISAPP, a joint research project between Shell, TNO and TU Delft.

Cover design: Lejla Alić

ISBN: 978-90-8891-089-0

Keywords: petroleum, reservoir engineering, systems and control theory, model reduction, simulation, optimization, iterative numerical analysis

Printed in The Netherlands.

to LBOH

Acknowledgments

I am very grateful to anyone who contributed, actively or by supporting me, that this thesis has come into existence. My sincere thanks to:

- Cor and Jan-Dirk, for hiring me as a PhD student.
- Jan-Dirk for supervising and guiding my PhD research. Jan-Dirk, I have no words to thank you enough for your help, support and encouragement, and especially for making it possible for me to continue with finishing this thesis after my escapade of breaking my promotion for a year and a half. Thank you for your attitude and all your actions towards me all the way along.
- Roald, Joris, Ali-Vakili, Maarten, Jorn, Willem-Jan, Koen, Nikolai, Patrik, Gosia, Rouhi, etc., for providing a pleasant work environment.
- Leendert, Heijn and Jorn, for making me feel privileged to have supervised your MSc theses. Your work has been very valuable to this research, and as you can see, Jorn's effort has directly contributed to a chapter of this book.
- Joris, for exactly the same as you said about me in your acknowledgments. Mazzel.
- Ali-Vakili, for much more than helping me, on a regular basis, to get into my office after I had lost my campus card.
- Dietz-lab people for friendly and enjoyable coffee breaks and 'dartpotjes'.
- The members of my thesis committee, for the obvious reason.
- Zineta and Emil, for loving and supporting me through all phases of my life. Ćale, if you are by any means aware of these words, I would like to let you know how proud and privileged I feel to have had you and Mama as my parents.
- My lovely wife and my kids, for making my life complete. LBOH, I'm so lucky and grateful to have you in my life. Hou zoveel van jullie!

Delft, February 2009
Renato Markovinović

Contents

1 CHAPTER

Introduction

This thesis is concerned with 'system-theoretic' low-order modeling of heterogeneous reservoir systems for purpose of efficient simulation and optimization of flooding processes with multiple injection and production (smart) wells. Other important applications of low-order representations could e.g. be in inversion/regularization procedures for identification of reservoir properties and/or for optimal placement of (both monitoring and controlling) wells. This chapter gives a concise background on closed-loop reservoir management and provides a motivation for the research performed in this thesis.

1.1 Background and Problem Definition

In order for hydrocarbon production to keep up with the ever growing energy demand, it is essential to find ways to improve the recovery factor of existing oil fields. A rapidly emerging methodology to achieve this goal is the application of measurements and control techniques to improve the control of reservoir flow over the life of the reservoir. Particularly promising in this context is the use of sensors and remotely controllable valves in both wells and at surface, in combination with large-scale and uncertain reservoir models. In the oil industry, this technology is known under various names, such as 'smart fields', 'intelligent fields', 'e-field', 'digital oilfield', 'closed-loop reservoir management', or 'real-time reservoir management'.

Figure 1.1 depicts the idea. The *system* may represent reservoir(s) in any combination with wells and surface facilities (Figure 1.2), but is in this thesis restricted to the subsurface part of the whole. The main focus is on developing concepts and algorithms for efficient implementation and integrated application of *data assimilation* and *(model-based) production optimization*.

Data assimilation. Compared to the size of a typical reservoir, the information available from various sensors at surface or downhole, and from more indirect measurements such as time-lapse seismics, is very scarce (and noisy). As the subsurface is generally highly heterogeneous, the *state* of the system (pressures and

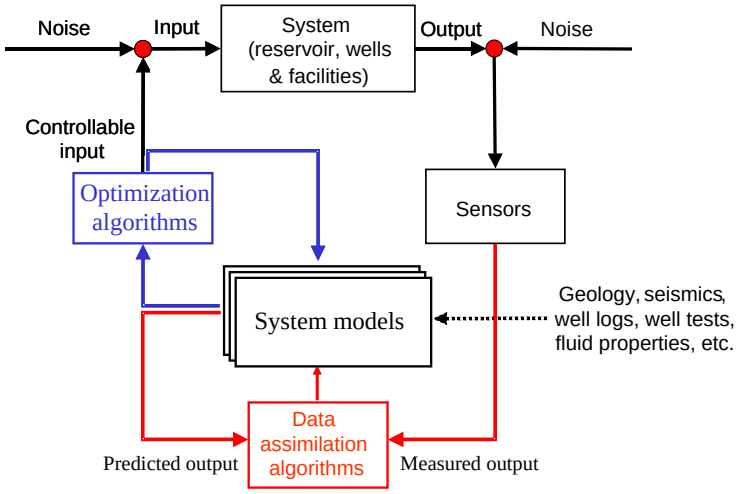


Figure 1.1: Reservoir management represented as a model-based closed-loop controlled process (after [145]).

saturations in the reservoir) as well as the reservoir parameters relevant to the flow (e.g. permeability) are known only to a very limited extent, and in the case of the parameters really only at the wells. To make matters worse, also the *input* to the system is known to a limited extent. While the situation in the wells (i.e., water injection rates or gas lift rates, bottom-hole pressures) may be roughly known, 'uncontrollable' input as, e.g., aquifer support (in this thesis assumed absent) may be a major unknown. As a result of these severe uncertainties, reservoir models are inevitably only a (very) crude approximation of reality. Their predictive capability is therefore very limited and, moreover, tends to deteriorate over time very quickly. In a process known as 'history matching' (HM) (or batch estimation), the reservoir parameters are updated using measured output from different sources over a specified estimation window. HM is frequently performed with iterative non-linear least squares algorithms based on gradient-searches [108, 172, 203, 230, 259]. Gradient-free optimization algorithms like simulated annealing or genetic algorithms have also been considered [218, 231, 247], but for large-scale applications they appear to be computationally prohibitive. As the history-matching problem is inherently notoriously ill-posed, a regularization term in the form of data-independent prior information is generally added to the mismatch function. Further regularization of the HM solution is possible by adding the model-errors[1] as a 'weak constraint' in the minimization problem, which can be solved efficiently in an iterative way by the so-called 'Representer Method' [220, 232]. Traditionally, HM has been performed on a campaign basis (e.g. once every few years), but in 'real-time' applications the estimation window

[1]That is, errors due to model approximations, such as a loss of spatial,or temporal, resolution, and the non-inclusion of all of the relevant physics.

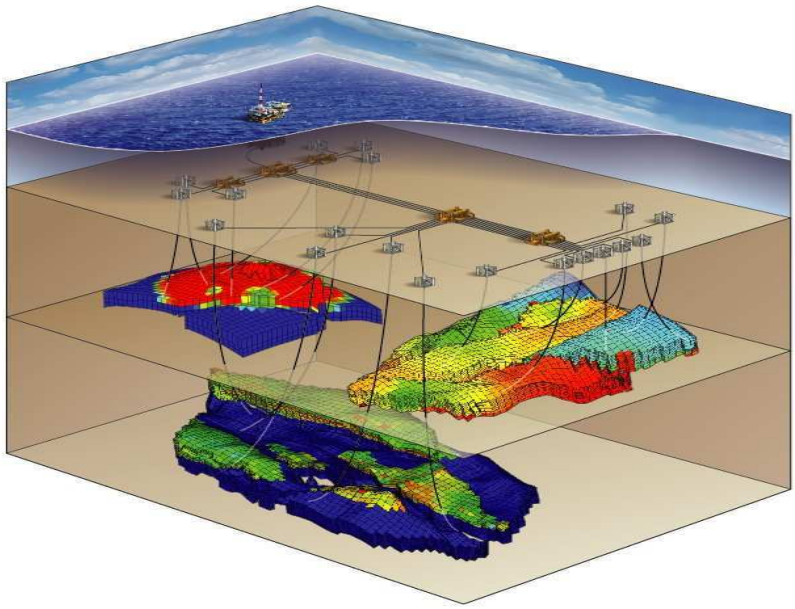


Figure 1.2: Schematic illustration of a group of hydrocarbon bearing formations connected, through a system of wells, to a network of pipelines and surface facilities.

needs to be continually (e.g., every few weeks) adjusted in order to include new observations. A recently introduced alternative to traditional history matching is the Ensemble Kalman Filter (EnKF) [83, 145, 172, 195], a Monte-Carlo filtering technique originally developed for updating of large-scale numerical models in meteorology and oceanography [9, 84, 122, 129]. Kalman filters are sequential (least-squares) state-estimators, so their natural application is continual estimation (or better, one-step ahead prediction) of reservoir pressure and saturation as new data become available. These estimates are used to update the reservoir control variables (e.g. well injection/production rates). Parameter estimation by EnKF is enabled by considering the to-be-estimated reservoir parameters, $\boldsymbol{\theta}$, as additional state-variables contaminated by measurement noise (i.e., $\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k + \boldsymbol{\epsilon}_k^{\boldsymbol{\theta}}$). A quick forecast can be obtained by integrating a single realization forward in time starting from the ensemble mean or median. A future prediction with uncertainty estimate can be provided by integrating the whole ensemble (usually between 50 and 100 realizations) forward in time. The sequential processing character makes the EnKF more convenient than the traditional (variational) HM for real-time applications. In addition to errors in the estimated parameters, the EnKF allows also for model errors. This, together with being sequential, contribute to the apparent absence of the curse of dimensionality and multiple local minima in the EnKF approach. Furthermore, EnKF can be applied with any reservoir simulator without the need of complicated coding. Moreover, rather than in the high-dimensional parameter space ($10^4 - 10^6$), the solution is sought for in the space spanned by the

ensemble members. On the other hand, the HM approach, especially if casted in a Bayesian framework [69, 305, 306], does not require the linear Gaussian approximations adopted in the EnKF.

(Model-based) production optimization. Due to the heterogeneous nature of the reservoir rock, the injected fluid (e.g. water in a waterflood production scenario) will flow with different velocities throughout the reservoir. Depending on the severity and type of heterogeneity, for a fixed well configuration the injected fluid may reach the producing wells while leaving large regions of the reservoir unflooded, thus resulting in a low recovery factor. A part of remaining oil can be extracted by drilling new wells, which is a costly operation along with the halt of production. Instead, one can try to minimize the problem by controlling the injection and production in the wells themselves. In that respect, smart-wells have opened the opportunities for going for reactive/passive (in the sequel called reactive) production scenarios [7, 15, 93, 146, 154, 249, 253, 300, 301], where no action is undertaken before some change in the well measurements is observed (e.g., change valve settings at places experiencing an unwanted water cut) to proactive/active (proactive in the sequel) production control, where well control strategies start to be applied (much) before the injected fluid breaks through at producers. Theoretically and intuitively, proactive control promises achieving a higher recovery factor than reactive control, but at the price that it requires (significantly) more knowledge about the parameters and the state of the reservoir. The amount of the required information is directly proportional to the length of the (valve-setting) optimization time horizon, which can go from a short-period of time (with the uttermost limit being just instantaneous optimization) in the 'short-term' optimization, to the entire remaining production window in a 'long-term' (or life-cycle) reservoir optimization. Optimization generally aims at maximizing the net present value (NPV) or some other economic objective. Concise overviews of the various (smart-well) optimization strategies are provided, e.g., in [46, 144, 302], whereby [144, 302] also discuss the scope for 'robust reservoir optimization', i.e., optimization in presence of (generally very large) uncertainties in reservoir models (in the first place uncertainties about the subsurface structure and the parameters that influence fluid flow). Application of moving-horizon Model Predictive Control (MPC) has also been proposed [198, 240]. These references also propose a hierarchical control structure for combining optimization and control at different time-scales.

Integrated application of data assimilation and field optimization. Various methods for combining model updating (either by history matching or EnKF) and production optimization to form a closed-loop optimization framework have been proposed in the literature; see, e.g., [5, 48, 60, 145, 194, 242, 290]. Most often, the 'combined' application consists of sequential application of optimization and data assimilation techniques of choice, whereby the latter provides, at specified times or every time new measurements become available, updated estimates of the (extended) system state. These estimates are then used by the optimization algorithm to determine new optimized operating conditions to maximize hydrocarbon production or net present value (NPV) for the remaining expected life of the reservoir. For example, in [242] an adjoint method was used for both the production opti-

mization and the model updating, whereas in [48] an adjoint method was used for optimization and the EnKF was used for model updating. In [175] the EnKF method was utilized to optimize choke settings with the gradient approximated through the sensitivity provided by the ensemble. The gradient search direction is approximated by an ensemble also in closed-loop reservoir optimization in [60]. In order to make the closed-loop solution more robust against model uncertainties, the same paper proposes (as did also [275], but in open-loop optimization using adjoints), to maximize the expected value of the NPV for the ensemble instead of maximizing the net present value based on a single reservoir model as in [48, 242, 290].

Central in the model-based (closed-loop) optimization is, of course, the *reservoir model*. Typically, and in order to incorporate 'as much physics as possible', it is a large-scale physics-based (or 'white-box') model consisting of $O(10^3 - 10^6)$ equations and parameters representing a (coupled) system of discretised PDEs (Chapter 2). Since optimization (in both production and history matching) as well as the EnKF generally require many reservoir flow simulations (forward and adjoint), whereby both the simulation time and the reliability of the solution (in parameter-estimation problems in particular) is directly impacted by the number of gridblocks and the complexity of the reservoir model, it is clear that the plain combination of reservoir model updating and optimization based on models of such size may be prohibitively time consuming and error prone (or at least cumbersome), especially when multiple realizations and/or several potential well configurations have to be investigated.

This thesis is about solving/reducing this 'curse of dimensionality' problem by including a third element in the closed-loop approach of Figure 1.1, consisting of *reduction of the high-order reservoir models to the appropriate level of detail in suitable low-order models*. In other words, we foresee that real-time reservoir management will/can show its real value only in the combined closed-loop use of optimization, model-updating and model reduction techniques as schematically depicted in Figure 1.3.

1.2 Research Objectives and Solution Directions

The ultimate objective of the research into low-order modeling would therefore be twofold: a) to develop computationally efficient algorithms for situations where multiple reservoir simulations are required, such as in optimization of flooding patterns, for instance, and b) to provide an alternative to regularization in situations were a large number of reservoir model parameters needs to be adapted using a limited amount of measured data. For these purposes we could have chosen to (try to) build on any of the existing 'coarsening' methods ranging from (conventional) *upscaling* [65, 77, 100, 224, 291], in which coarse-scale equations of a prescribed analytical form are solved[2], to various *multiscale* techniques [24, 61,

[2]The form of the equations at the coarse grid may differ from the underlying fine-scale equations, in which case it is either derived via homogenization or volume-averaging procedures, or, as is often

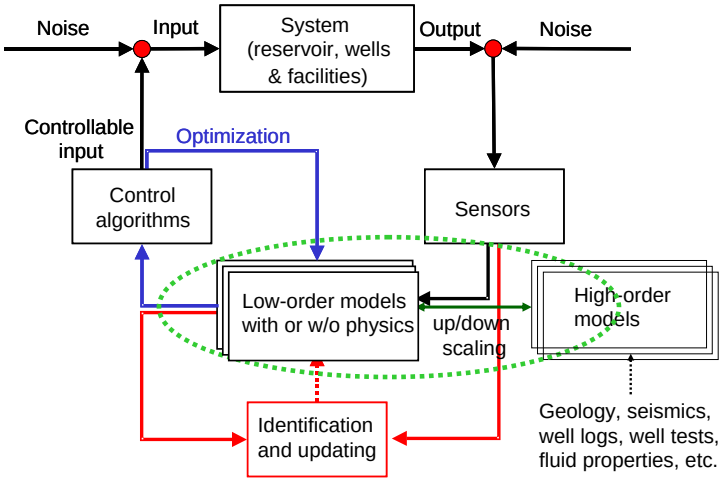


Figure 1.3: Closed-Loop Reservoir Management with Reduced Order Models.

78, 79, 99, 130, 148, 178], in which fine-scale information is carried throughout the simulation and the coarse-scale equations are not expressed analytically but rather are formed and solved numerically (generally at different grids for flow and transport).

We decided not to follow that 'obvious' route, though, for the simple reason that we believe that low-order models based on dynamically-intrinsic properties of the fine-scale system rather the type (i.e., position, form, orientation) of the coarse-grid may be more efficient and/or robust for the intended applications. For single-phase systems that can be represented as a (state-space) system of ODEs with constant coefficients, e.g. spatially discretized diffusive pressure equation for slightly compressible porous-media flow, the basic system- and control theory teaches that these properties could e.g. be the system's transfer function in the Laplace domain, the eigenstructure of the system matrix, or controllability and observability of the system[10]. While these properties are related to each other in certain senses (Chapter 4), the structural properties controllability and observability (and therewith related identifiability) are particularly interesting for our purposes. Indeed, modeling to a level of detail that can neither be observed nor controlled, is at best wasted effort, but, worse, may lead to wrong results. Numerical experiments performed in [183] in a early stage of this project with

the case in practice, simply assumed to be the same (or nearly so) as the fine-scale equations (hardly justifiable in general in multiphase flow, as applying an averaging operator to a nonlinear function is not interchangeable with the opposite sequence of the operations). The coefficients in the upscaled equations are typically computed from appropriate solutions over regions of the fine-scale model in a preprocessing step. *Static* techniques are generally used to coarse the media properties (permeabilities, porosities), while *dynamic* methods are employed to understand the effects of the small scale dynamics on the large scales.

(deterministic) 'subspace-identification' techniques [221, 276, 285] applied on a slightly compressible single-phase (synthetic) reservoir suggested that the controllable and observable subspaces of the fine-scale reservoir models are indeed of much lower dimension than the number of the grid-blocks.

For nonlinear state-space models, unsurprisingly, much less (actually really only few) systematically developed reduction techniques are available [136, 161, 225, 244]. Unfortunately, even these spare methods are not applicable to multiphase reservoir flow, due to both the size and the overall complexity of these systems (e.g., moving fluid interfaces, coupled flow and transport). Another way of extracting intrinsic information is using the system data obtained from the simulation (or, if possible, experimental) data. 'Data-driven' approaches have been successfully applied on a scala of engineering problems in various areas of science: model reduction, image processing, turbulent flow, inverse problems compression, inverse problems, to name a few. Particularly popular has been the Proper Orthogonal Decomposition (POD) technique, which guarantees the best reconstruction of the collected data in the mean square error sense [16, 22, 29, 125, 128, 152, 170, 174, 177, 205, 223, 286, 295]. In the linear case, POD turns out to have important theoretical connections to the concepts of controllability and observability, though for a particular class of input and initial-state signal forms. Depending on the area of application other names are used as well, e.g. Karhunen-Loeve (K-L) Transform, Empirical Orthogonal Functions (EOF), Principal Component Analysis (PCA) [149, 156].

The **specific objectives** in this research are therefore stated to be as follows:

Assess the usefulness and performance (in terms of computational speed and complexity) of the state-of-the-art methods and algorithms for model-order reduction when applied to linear single-phase heterogeneous reservoir systems. For two-phase waterflood flow with multiple injection and production (smart) wells, investigate the applicability and performance of the standard POD approach and, if necessary, design alternative strategies based on the same methodology. In particular, design a method to speed-up the adjoint-based waterflood optimization developed in [46].

1.3 Thesis outline/organization

The remaining part of the thesis is organized as follows. **Chapter 2** presents and shortly discusses the basis (macroscopic) equations governing the porous media flow considered in the thesis. **Chapter 3** presents the *projection-based Model Order Reduction* (MOR) framework used in all methods in this thesis to achieve model-order reduction. **Chapter 4** provides a detailed overview of modern approaches for linear MOR and assesses their performance when applied to single-phase reservoir flow problems. Considered are: *Modal Truncation*, *Singular Perturbation* (or 'residualization'), *Transfer Function Moment Matching* (explicit and implicit), and *Balanced Truncation* (exact and approximate). The applicability and performance

of the standard *Proper Orthogonal Decomposition (POD)* approach applied to two-phase waterflood flow is assessed in **Chapter 5**. The same chapter introduces the idea of a 'middle-way' approach in the course of reservoir simulation, in which the reservoir system in time is as usually solved in the original, high-dimensional space, but with an improved initial solution guess provided by a (regularly updated) POD-based reduced-order model. In **Chapter 6** an iterative method is designed to speed-up the adjoint-based waterflood optimization developed in [46]. A methodology using nested loops is employed, where the inner iterative loop makes use of a truncated basis of POD functions to calculate optimized injection and production rates. After convergence in this loop we simulate in the outer loop the original, high-order model with the optimized rates and subsequently adapt the POD basis. This new basis is used in the next inner loop to calculate new optimized injection and production rates. **Chapter 7** concludes the thesis. **Appendices A** & **B** provide auxiliary information regarding the two-phase reservoir flow equations and the general POD theory, respectively.

Additional information:

Several of the ideas and conclusions presented in this thesis were published by the author in conference or journal publications over the past years. This concerns in particular the idea to use subspace identification, POD and (empirical) balanced and modal truncation as reduced-order methods for reservoir simulation [123, 184, 185] ([123] together with T. Heijn), the use of POD in a nested loop to speed up water flooding optimization (together with J.F.M. van Doren, [272]), and the use of POD as a 'shadow simulation' to speed up iterative solution methods [182].

2 CHAPTER

Porous Media Flow: Basic Concepts and Model Formulations

This chapter presents and shortly discusses the basis (macroscopic) equations governing both single-phase and two-phase porous media flow considered in the thesis. The fluids involved are oil and water, assumed immiscible with respect to each other. The models account for compressibility and capillary effects unless explicitly stated otherwise. The porous medium itself is considered heterogeneous and isotropic, with the permeability described by a diagonal tensor. Both 2D and 3D flows are considered, with gravity effects only included in the latter as all 2D models are taken to describe horizontal flow only.

2.1 Mass Balance & Constitutive Relationships

Two basic ingredients in modelling porous media flow are mass conservation for each of the phases and an empirically determined constitutive equation relating the average mass flux of each phase to the corresponding fluid potential gradient. This equation actually accounts for conservation of momentum, and is a multiphase extension of Darcy's law. These (macroscopic) conservation laws are augmented by empirical material-dependent constitutive relationships describing saturation dependence of capillary pressure, which is the pressure across the interface between (immiscible) fluids due to interfacial forces, and relative fluid permeability, which represents a reduction in the permeability for one phase due to its interference with other phases. A general understanding is that the capillary effects are controlled by the geometry of the pore space, interfacial tension and wettability of the rock. Possible dynamic effects in the capillary pressure - fluid saturation relationship, $P_c(S_w)$, due to the hysteresis in saturation [121] are not considered here. The conservation law and the constitutive relationships together form a system of coupled nonlinear partial differential equations (PDEs), which for the two-phase flow of immiscible oil (nonwetting phase) and water (wetting

phase) in a heterogeneous porous medium can be expressed as [25, 209]:

$$\frac{\partial(\varphi S_\alpha \rho_\alpha)}{\partial t} + \bigtriangledown \cdot (\rho_\alpha \mathbf{u}_\alpha) = q_\alpha \qquad \text{Mass balance } ([\tfrac{kg}{m^3 s}]), \tag{2.1.1}$$

$$-\frac{\mathbf{K} k_{r\alpha}}{\mu_\alpha}(\bigtriangledown p_\alpha - \rho_\alpha g \bigtriangledown D) = \mathbf{u}_\alpha \qquad \text{Darcy's law } ([\tfrac{m}{s}]), \tag{2.1.2}$$

$$P_c = P_c(S_w) := p_o - p_w \quad , \quad k_{r\alpha} = k_{r\alpha}(S_w), \tag{2.1.3}$$

$$S_w + S_o = 1, \tag{2.1.4}$$

where $\varphi = \varphi(p)$ $[-]$ is the effective[1] porosity of the porous medium, and $\rho_\alpha = \rho_\alpha(p)$ $[\frac{kg}{m^3}]$ and $\mu_\alpha = \mu_\alpha(p)$ $[Pa \cdot s]$ are the pressure dependent[2] density and viscosity, respectively, of water- ($\alpha = \omega$) or oil ($\alpha = o$) phase. Vector $\mathbf{u}_\alpha$ comprises the phase velocities (in $[\frac{m}{s}]$) in the spatial directions, S_α $[-]$ is the phase saturation (i.e. the portion of the pore space occupied by phase α), $g = 9.81$ $[\frac{m}{s^2}]$ is the gravitational acceleration constant, $P_c = P_c(S_w)$ $[Pa]$ and $k_{r\alpha} = k_{r\alpha}(S_w)$ $[-]$ are the saturation dependent capillary pressure and phase relative permeability, respectively, and q_α $[\frac{kg}{m^3 s}]$ is the forcing term representing the mass added to or extracted from the system through an external source per unit time and unit volume. The convention in this thesis is that positive values of q correspond to injection, negative values to production. The intrinsic permeability[3] tensor $\mathbf{K}$ $[m^2]$ is assumed to be diagonal ($\mathbf{K} := diag(k_x, k_y, k_z)$), and D is the depth in $[m]$.

Substituting Darcy equation $(2.1.2)$ into $(2.1.1)$, the mass balance equations for oil and water can be written as:

$$\frac{\partial(\varphi S_o \rho_o)}{\partial t} - \bigtriangledown \cdot \left(\frac{\mathbf{K} k_{ro} \rho_o}{\mu_o}(\bigtriangledown p_o - \rho_o g \bigtriangledown D) \right) = q_o \tag{2.1.5}$$

$$\frac{\partial(\varphi S_w \rho_w)}{\partial t} - \bigtriangledown \cdot \left(\frac{\mathbf{K} k_{rw} \rho_w}{\mu_w}(\bigtriangledown p_w - \rho_w g \bigtriangledown D) \right) = q_w, \tag{2.1.6}$$

respectively. A system of equations as (2.1.1-2.1.4), together with initial- and boundary conditions, describes phenomena propagating forwardly in time and will throughout the thesis therefore be called a *forward model* (or a *forward system*) whenever necessary to distinguish it from an *adjoint model* (*adjoint system*) as e.g. used in the gradient-based optimal waterflooding application, introduced later in the thesis (Ch. 6).

Primary Variables - (p_o, S_w)

The two-equation system $(2.1.5\text{-}2.1.6)$ is given in terms of four system variables (two pressures and two saturations), so one needs to select two 'primary' variables out of these four in which to write $(2.1.5\text{-}2.1.6)$, in order to make the model

[1] 'Effective porosity': The interconnected void space in a rock that *contributes to fluid flow* or permeability in a reservoir. Effective porosity excludes isolated pores and pore volume occupied by water adsorbed on clay minerals or other grains. Total porosity is the total void space in the rock whether or not it contributes to fluid flow.

[2] Fluid density and viscosity generally are functions of both pressure and temperature. In the models in this thesis, constant temperature is assumed, though.

[3] Intrinsic permeability is a property only of the medium (porosity and pore structure) and is independent of the nature of the fluid(s).

well-defined [25]. This choice may be crucial for computational efficiency and robustness, but also for simplicity in evaluating the 'secondary' variables (i.e., those defined through the primary variables, like fluid mobilities, capillary pressure; see below), as well as the remaining, non-selected pressures and saturations. For instance, in a fully time-implicit formulation and a Newton based linearization approach for finding a numerical solution, the choice of primary variables may have a large effect on the conditioning of the associated Jacobian matrix and hence on the number of iterations required to solve the Jacobian system, as well as on the number of Newton iterations. Perhaps even more important is to realize that different choices of primary variables may lead to alternative pairs of differential equations equivalent to $(2.1.5\text{-}2.1.6)$ revealing the true nature of the physical problem under consideration not immediately seen from the original form of the equations. For example, the superficial resemblance of $(2.1.5\text{-}2.1.6)$ to the (nonlinear) heat conduction equation may cause one to expect two-phase problems to be essentially parabolic in nature. However, using an alternative representation in an average pressure and water saturation one obtains a representation in one pressure and one saturation equation, the former having parabolic or elliptic character (for compressible and incompressible flow, respectively) while the latter being hyperbolic in nature when capillary effects are negligible or absent, or parabolic when capillary effects play a major role [209].

The two primary variables in this thesis are always oil pressure and water saturation, the choice being principally dictated by the desire to use the same descriptions in all the different parts of the closed-loop reservoir management program (Chapter 1). Using (2.1.3-2.1.4) the system $(2.1.5\text{-}2.1.6)$ can be recast into the following $p_o - S_w$ representation of the two-phase porous media flow:

$$\frac{\partial\left(\varphi(1-S_w)\rho_o\right)}{\partial t} - \bigtriangledown \cdot \left(\frac{\mathbf{K}k_{ro}(S_w)\rho_o}{\mu_o}(\bigtriangledown p_o - \rho_o g \bigtriangledown D) \right) = q_o \quad (2.1.7)$$

$$\frac{\partial(\varphi S_w \rho_w)}{\partial t} - \bigtriangledown \cdot \left(\frac{\mathbf{K}k_{rw}(S_w)\rho_w}{\mu_w}(\bigtriangledown p_o - \frac{dP_c(S_w)}{dS_w} \bigtriangledown S_w - \rho_w g \bigtriangledown D) \right) = q_w . \quad (2.1.8)$$

Secondary Variables - Sources of Nonlinearity and Uncertainty

Equations (2.1.7-2.1.8) form a nonlinear coupled system, where the nonlinear character comes in through the saturation dependent parameters P_c and $k_{r\alpha}$ and the pressure dependent fluid properties ρ_α and μ_α. Generally speaking, and especially when the fluid properties are insensitive or only slightly sensitive to pressure variations (as is the case in all models in this thesis), the strongest nonlinearity comes from the relative permeabilities and the capillary pressure as nonlinear functions of saturation. The $k_{r\alpha}$-S_w and P_c-S_w relationships are porous media dependent and are determined by (very time-consuming) laboratory experiments on a, compared to the size of a typical reservoir, very limited number of core samples. This, together with the non-existence of a unifying method for obtaining these relationships, makes them much more uncertain compared to other properties such as density and viscosity. In this thesis, diverse $k_{r\alpha}$-S_w and P_c-S_w relationships are used, from simple first-order, i.e. straight lines (Fig. 2.1), to the more 'advanced'

relations due to *Brooks-Corey*:

$$\alpha \in \{w, o\}: \quad P_c = P_d \cdot S_{N,\alpha}^{-\frac{1}{\lambda}}, \quad S_{N,\alpha} = \frac{S_w - S_{\alpha r}}{1 - S_{wr} - S_{or}}, \quad k_{r\alpha} = k_{r\alpha}^0 \cdot S_{N,\alpha}^{n_\alpha}, \quad (2.1.9)$$

where $S_{N,\alpha}$ is the *normalized saturation*, S_{wr} and S_{or} are respectively water- and oil *residual saturations*, P_d is the *entry capillary pressure* of the rock sample, λ is the *pore size distribution index*, k_{rw} and k_{ro} are water- and oil relative permeabilities, respectively, and k_{rw}^0 and k_{ro}^0 are relative permeability end points, and n_w and n_o are empirical coefficients. Typical 'realistic' relative permeability curves used in this thesis are depicted in Fig. (2.2).

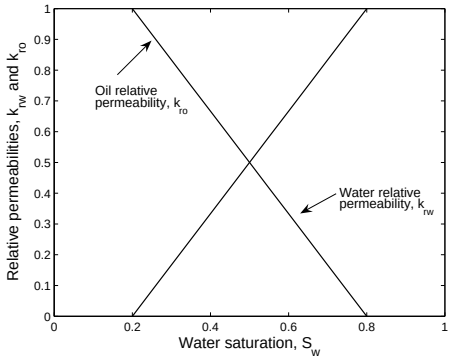


Figure 2.1: First-order relative permeability curves.

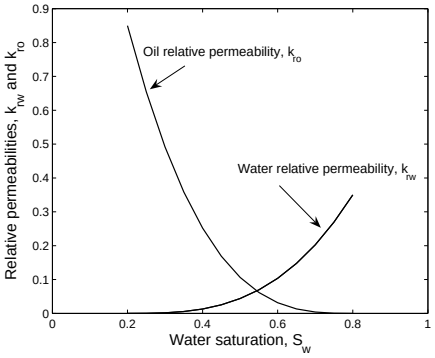


Figure 2.2: Typical relative permeability curves (water-wet rock).

Generally speaking, the shapes of water and oil relative permeability curves and the related capillary pressure curves can have a big influence on the reservoir performance as predicted by the reservoir mathematical model. In the examples in this thesis, it is assumed that capillary (diffusion) effects are almost negligible as compared to convection and *the most uncertain parameter in the model is considered to be the intrinsic or absolute permeability* **K**. For waterflooding, this assumption is generally justified.

2.2 Initial and Boundary Conditions

In all examples in this thesis the reservoir is assumed initially filled with oil. The initial water saturation is therefore always a uniformly distributed connate water saturation, i.e. $S_w(\xi, t_0) = S_{wc}$. In the absence of gravity effects, also the fluid pressure conditions in the reservoir are assumed initially constant and uniform, i.e. $p_o(\xi, t_0) = p_{o,0}$. In the above, $\xi = (x, y) \in \Omega \subset \mathbb{R}^2$ or $\xi = (x, y, z) \in \Omega \subset \mathbb{R}^3$. Regarding the boundary conditions (BCs), any reservoir in this work is assumed to lie within some closed curve or surface across which there is no flow (*no-flow condition*), with fluid injection (always water) and production (both oil and water) taking place at wells, either vertical or well segments when a (smart) horizontal

well is assumed. The enclosing curves are always assumed to be straight, i.e., the geometry of the reservoirs is always a rectangular one. The regularity of the boundaries admits representation of no-flow boundary conditions by setting the normal component of the flow vector at the boundaries to be zero, something which is numerically obviously not trivial for an arbitrary boundary curve[4].

Injection and extraction of fluids through wells can be formally represented by point sources and sinks, respectively. Unfortunately, a numerical representation of a true point source or sink, wherein the flow rate (q in $(2.1.5 - 2.1.6)$) is zero everywhere except at the wells, while infinite at the well locations, is impossible [209]. The flow rates at the wells are for this reason approximated by suitable well models. Most approximations use some (usually radial) form of the Darcy law, relating the fluid inflow and/or outflow rate to the difference between the pressure in a vicinity of the well and the pressure in the well itself [25, 26, 71, 207, 208]. A general formulation for well models is

$$q_t \quad = \quad \omega(p_{wf} - p_{gb}), \tag{2.2.1}$$

where p_{wf} is the (bottom-hole) well flowing pressure, p_{gb} is the pressure in the grid-block for phase α, and ω is a parameter containing well geometric factors and reservoir rock and fluid properties around the well. A more detailed description of the well model used in the examples in this thesis is given in Appendix A.

It is important to note that the mathematical definition of BCs at injectors and producers is subject to physical constraints that need to be taken into account, e.g. in order to prevent a (numerical) optimization procedure to deliver a physically non-feasible solution. For example, when optimizing the process of waterflooding a hydrocarbon reservoir, which can e.g. be done by determining optimal fluid injection and production strategies using optimal control theory as in [46, 47] (and in this work as well), these constraints can for instance be the conditions under which the wells are desired or forced to operate. Various types of well operating scenarios are possible, with or without explicit use of a well model. When no well model is used, the liquid injection and production rates in the individual wells can be thought as controlled directly, or equivalently, these individual flow rates can be considered as the control parameters. When a well model as $(2.2.1)$ is used, the injection flow rate can still be controlled directly, as the only injecting fluid is water and the amount of the injected fluid hence only depends on the total fluid mobility, $\lambda_t := \lambda_w + \lambda_o \equiv \frac{k_{rw}}{\mu_w} + \frac{k_{ro}}{\mu_o}$, in the reservoir directly around the well and on p_{wf}. The fluid flowing into[5] a producer, however, is generally a

[4]Usually there is a little practical interest in an accurate solution (very) close to the curved boundary, obtaining a correct flow solution in the interior of the reservoir being much more important. For this reason are reservoirs with a non-regular boundaries often represented as being embedded in a rectangular parallelopiped, and the intrinsic permeability and porosity spatially distributed functions are set zero outside the reservoir boundary curve.

[5]Or 'out of', if there it happens that the pressure in the producer becomes higher than the pressure directly around the well. Such a situation could for example occur if the well is completed through multiple reservoir sections, each operating at a different pressure regime, with as the possible result that fluids produced at some point along the well in one section flow back ("back-flow", also called "cross-flow") at some other point in another section. Including back-flow in optimization studies may cause serious numerical difficulties and its occurrence in the simulations is often made impossible by

composition of more (here two) fluids, which amounts are determined by their individual mobilities directly around the well and by p_{wf}. The instantaneous produced fluid composition cannot therefore be controlled directly by changing p_{wf} [46]. On the other side, the well flowing pressures in the individual injectors and producers can be fixed, for instance, and the individual injection and production rates controlled by the adjustment of the valve-settings (a number between 0 and 1) in the wells, in which case the well model (2.2.1) is equipped with an additional multiplication 'interval control valve' parameter α_{icv}:

$$q_t \quad = \quad \alpha_{icv}\omega(p_{wf} - p_{gb}). \tag{2.2.2}$$

In this model, the well rate can thus be 'changed' by manipulation of α_{icv}, which represents a non-physical choke model. This approach is in the literature known as "modeling a choke as a pseudodevice" [127].

2.3 Spatial Discretization and State-Space Model Formulation

Nonlinear partial differential equations like (2.1.5-2.1.6) are generally not solvable analytically, since their solutions display richer structures than can be described by analytic (distributed) functions. That means that one is generally forced to look for an approximate numerical solution at time and spatial points of interest. Construction of spatially discretized reservoir models is commonly performed using well-established techniques of finite differences (FD) (used in this thesis), finite volumes (FV), or finite elements (FE). Whatever the preferred space-discretization method might be, it will ultimately deliver a model in the following *continuous-time* nonlinear *state-space* form (Appendix A):

$$\mathbf{E}(\mathbf{x})\frac{d\mathbf{x}}{dt} \quad = \quad \mathbf{f}(\mathbf{x}, \mathbf{u}), \tag{2.3.1}$$

where $\mathbf{x} \in X \subset \mathbb{R}^n$ is an n-dimensional *state* vector, here consisting of the spatially discretized primary variables, p_o and S_w, $\mathbf{u} \in U \subset \mathbb{R}^m$ is an m-dimensional *input* (or *control*) signal, e.g. containing well flow-rates or well valve settings, and $\mathbf{E} : X \to W \subset \mathbb{R}^{n\times n}$ is a sparse state-dependent (accumulation) matrix. The exact sparsity pattern depends on the ordering of the variables in the state vector, but all patterns are directly transformable into each other by multiplication by a permutation matrix.

Note that (2.3.1) differs from the state-space description more commonly encountered in the literature:

$$\frac{d\mathbf{x}}{dt} \quad = \quad \mathbf{f}(\mathbf{x}, \mathbf{u}). \tag{2.3.2}$$

In order to distinguish between the two representations, in the sequel (2.3.2) will

appropriate choice of well model parameters.

be referred to as a *standard state-space form*, whereas (2.3.1) will be called a *generalized state-space* form.

If the accumulation matrix **E** is non-singular, the generalized state-space model can, at least formally, be brought into the standard state-space form by inverting **E**, yielding

$$\frac{d\mathbf{x}}{dt} \quad = \quad \mathbf{E}^{-1}(\mathbf{x})\mathbf{f}(\mathbf{x},\mathbf{u}) =: \widetilde{\mathbf{f}}(\mathbf{x},\mathbf{u}). \tag{2.3.3}$$

A typical situation in which **E** is singular is e.g. when both the rock and the fluids are incompressible (Appendix A). In such situations a forward model different than (2.3.1) might be more suitable or even required.

State-dependent parameter (input-affine) factored form of the system dynamics

It turns out (see Appendix A) that spatial discretization naturally delivers an **f** factorized as[6] **f**(**x**,**u**) = **A**(**x**)**x** + **g**(**x**,**u**), so that (2.3.1) hence becomes

$$\mathbf{E}(\mathbf{x})\frac{d\mathbf{x}}{dt} \quad = \quad \mathbf{A}(\mathbf{x})\mathbf{x} + \mathbf{g}(\mathbf{x},\mathbf{u}), \tag{2.3.4}$$

whereby the analytic forms of the elements of matrices **E** and **A** and vector field **g**, as well as the actual form of the latter, in a particular situation depend on the physics included (gravity, capillary pressure, ...) and the type of the control (well models used or not). Moreover, in the general case, the appearance of the state **x** in **E**, **A** and **g** is rather implicit, i.e., the elements of **E**, **A** and **g** consist of (generally non-polynomial) functions of the state variables. Formally a more correct notation would therefore be **E**(ϑ(**x**)), **A**(ϑ(**x**)) and **g**(ϑ(**x**),**u**), where ϑ is a vector comprising state-dependent parameters (compressibilities, relative permeabilities, etc.). A general *space-discrete, time-continuous generalized state-space* reservoir description may therefore be taken to be:

$$\mathbf{E}(\vartheta(\mathbf{x}))\frac{d\mathbf{x}}{dt} \quad = \quad \mathbf{A}(\vartheta(\mathbf{x}))\mathbf{x} + \mathbf{g}(\vartheta(\mathbf{x}),\mathbf{u}). \tag{2.3.5}$$

For example, the governing equations of a two-dimensional, two-phase (oil-water) reservoir flow, obtained using a finite difference discretization in space, may be expressed as

$$\mathbf{V}(\vartheta(\mathbf{x}))\mathbf{W}(\vartheta(\mathbf{x}))\frac{d\mathbf{x}}{dt} \quad = \quad \mathbf{T}(\vartheta(\mathbf{x}))\mathbf{x} + \mathbf{V}(\vartheta(\mathbf{x}))\mathbf{F}(\vartheta(\mathbf{x}))\mathbf{L}_{qu}\,\mathbf{u}, \tag{2.3.6}$$

[6] We note that a decomposition of a nonlinear n-dimensional vector field $\mathbf{h}(\mathbf{x})$ into a matrix-vector product $\mathbf{h}(\mathbf{x}) = \mathbf{A}(\mathbf{x})\mathbf{x}$ is in general *not unique*. Indeed, let $\mathbf{h}(\mathbf{x}) = \mathbf{A}(\mathbf{x})\mathbf{x}$ and let $\mathbf{K}(\mathbf{x})$ be any non-zero matrix such that $\mathbf{K}(\mathbf{x})\mathbf{x} = \mathbf{0}$. Then $\mathbf{F}(\mathbf{x}) := \mathbf{A}(\mathbf{x}) \pm \mathbf{K}(\mathbf{x})$ also satisfies $\mathbf{h}(\mathbf{x}) = \mathbf{F}(\mathbf{x})\mathbf{x}$. Actually, for $\alpha \in \mathbb{R}$, the linear combination $\mathbf{Q}(\mathbf{x},\alpha) := \alpha\mathbf{A}(\mathbf{x}) + (1-\alpha)\mathbf{F}(\mathbf{x})$ parametrizes a set of infinitely many state-dependent matrices with the same decomposing property. As an illustration of the non-uniqueness, let $n = 2$, $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ and $\mathbf{h}(\mathbf{x}) = \begin{bmatrix} x_1^2 \\ x_1 + x_2 \end{bmatrix}$. Then $\mathbf{A}_1(\mathbf{x}) = \begin{bmatrix} x_1 & 0 \\ 1 & 1 \end{bmatrix}$ and $\mathbf{A}_2(\mathbf{x}) = \begin{bmatrix} x_1 - x_2 & x_1 \\ x_1x_2 + 1 & 1 - x_1^2 \end{bmatrix}$ satisfy $\mathbf{h}(\mathbf{x}) = \mathbf{A}_1(\mathbf{x})\mathbf{x} = \mathbf{A}_2(\mathbf{x})\mathbf{x}$. In this thesis no use of this potentially/possibly useful property has been made.

where $\mathbf{x} = [p_{o,11}, S_{w,11}, \ldots, p_{o,n_x n_y}, S_{w,n_x n_y}]^T$ is the n-dimensional state vector containing oil pressures and water saturations for each of the $n_x n_y = \frac{n}{2}$ grid blocks, $\mathbf{V}$ is a diagonal mass matrix with entries that are functions of grid block volume and fluid densities, $\mathbf{W}$ is a block diagonal matrix with entries being functions of compressibility, porosity and water saturation, $\mathbf{T}$ is a block matrix containing the transmissibilities for oil and water, $\mathbf{F}$ is a (quasi[7]) block-diagonal matrix containing fractional-flow functions for water and oil, and $\mathbf{L}_{qu}$ is a selection (or location) matrix consisting of zeros and ones at appropriate places so that $\mathbf{u} \equiv \mathbf{L}_{qu}^T \mathbf{q}_t$ is the input vector containing total liquid rates (generally, $q_t = q_o + q_w$ at the producers and q_w at the injectors). Again, if a well is a producer (injector), the convention here is that its total rate is negative (positive).[8].

2.4 Discrete-Time Formulations and Methods of Solution

Time-derivative operators are in this thesis always approximated by the simple first-order finite difference, i.e. $\frac{dy}{dt} \rightarrow \frac{y(t+\Delta t)-y(t)}{\Delta t}$. Bringing (2.3.5) into a time-discrete form then requires one to choose at which time-instances the different inputs, state variables and state-dependent parameter matrices are evaluated. Various choices are possible, some of the standard ones given below.

Fully-Implicit

In this formulation all state-dependent parameters and system inputs are 'evaluated' at the next time level, yielding

$$\begin{aligned} \mathbf{0} &\equiv [\mathbf{E}(\vartheta_{k+1}) - \Delta t_k \mathbf{A}(\vartheta_{k+1})]\mathbf{x}_{k+1} - \mathbf{E}(\vartheta_{k+1})\mathbf{x}_k + \Delta t_k \mathbf{g}(\vartheta_{k+1}, \mathbf{u}_{k+1}) \\ &=: \mathbf{h}(\mathbf{x}_k, \mathbf{x}_{k+1}, \mathbf{u}_{k+1}), \end{aligned} \tag{2.4.1}$$

where k denotes the current integration time-step, $\Delta t_k := t_{k+1} - t_k$, and $\vartheta_{k+1} := \vartheta(\mathbf{x}_{k+1})$. Solving (2.4.1) generally requires an iterative treatment, usually by Newton's method [43]. The fully-implicit solution method is unconditionally stable (i.e. yields a stable solution for large timesteps, the timestep length restrictions being only those necessary to ensure convergence of the Newton iteration scheme.).

[7] A block diagonal (or diagonal block) matrix is defined as a square diagonal matrix in which the diagonal elements are square matrices. The blocks in $\mathbf{F}$ are not square matrices but two-dimensional column vectors of the form $\begin{bmatrix} f_w \\ f_o \end{bmatrix}_i$, where $f_\alpha = \frac{q_\alpha}{q_t} \overset{Pc=0}{=} \frac{\lambda_\alpha}{\lambda_t}$, $\alpha = w, o$, are the *fractional-flow* functions for water and oil, respectively, and i denotes the grid-block in question (see Appendix A for a more detailed account). Clearly, $0 \leq f_w, f_o \leq 1$ and $f_w + f_o = 1$. For an exclusively water injecting well, the fractional-flow vector is $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$. Moreover, for a grid-block j without wells the fractional-flows are not defined. Since the flow-rate for such a well is zero, $(f_w)_j$ and $(f_o)_j$ may be assigned any finite value.

[8] If a well in a grid-block i is described by a well model (e.g. (2.2.1)) with prescribed bottom-hole pressure, the resulting system of equations is similar, with $u_i = p_{bh,i}$, and $\mathbf{T}$ and $\mathbf{F}$ adjusted at appropriate places by terms involving the well-index ω_i.

Sequential Implicit

The major disadvantage of the fully implicit solution method is that at any time-step a full system of n linear equations must be solved simultaneously, where n is the total number of unknown grid-block pressures and saturations. In the sequential method, the solution of the pressure equation is separated from the saturation equation, with the first step being the solution of a set of pressure equations, followed by the solution of a set of saturation equations using the pressure solution from step one. This is similar to the IMPES method described below, with the important difference that also the saturations are treated implicitly in time [255]. The method involves reordering of the flow equations, thus implying reordering of the state-vector (in 2-D case) from $\mathbf{x} = [p_{o,11}, S_{w,11}, \ldots, p_{o,n_x n_y}, S_{w,n_x n_y}]^T$ to $\mathbf{x} = [p_{o,11}, p_{o,12}, \ldots, p_{o,n_x n_y}, S_{w,11}, S_{w,12}, \ldots, S_{w,n_x n_y}]^T$ and a change in the band structure of the system matrices. The final separation step, which combines the individual phase material balance equations by adding them, brings the thus re-ordered system into a block-triangular form which is then solved sequentially for the pressures and the saturations (each of dimension $n/2$).

As at each timestep two systems of equations are solved each being a half of the size (for 2-phase problems) of the fully-implicit formulation, for large-scale reservoir simulations solution methods based on sequential formulation often greatly outperform fully-implicit ones in terms of computational costs. Moreover, although the method is not unconditionally stable, it generally allows for time-step lengths much larger than using e.g. IMPES formulations. On the other hand, the achievable computational gain strongly depends also on the amount of iteration steps involved in the solution process, which for sequential approaches may be considerable, especially for the interblock flow terms.

IMPES (*IM*plicit *P*ressure *E*xplicit *S*aturation)

Like in the sequential method, IMPES formulations transform the coupled systems of flow equations into two separate sets of equations for the pressures and the saturations, respectively. Now, however, after solving for the pressures implicitly the saturations are determined explicitly by solving material balance equations. The relative permeabilities and the capillary pressure, as saturation dependent parameters, are assumed to be constant during a time-step. A detailed derivation of an IMPES formulation is given in Chapter 5.3.

IMPES formulations are simple to set up, and the solving process is easy to implement and requires less computer memory and CPU time compared with other methods. However, for it to be stable, this classical method may require prohibitively small time steps for the saturation (e.g. for small grid-block problems such as coning ones, or for long time integration problems).

Quasi-Implicit

Here, both pressures and saturations are considered implicitly in time, while the state-dependent parameters and inputs are evaluated at the current time level, yielding:

$$[\mathbf{E}(\vartheta_k) - \Delta t_k \mathbf{A}(\vartheta_k)]\mathbf{x}_{k+1} \quad = \quad \mathbf{E}(\vartheta_k)\mathbf{x}_k + \Delta t_k \mathbf{g}(\vartheta_k, \mathbf{u}_k). \qquad (2.4.2)$$

The reasoning here is that the state-dependent parameters are expected to vary slowly with changing states, in particular when the time-step length considered is taken rather moderate. In waterflood applications this often appears to be a valid assumption, except when high fluid velocity effects are present (e.g. water coning at production wells) as then the state-dependent phase mobilities may change rather rapidly. Insisting on considering the mobilities as constants during a timestep would in such situations imply impractically small timestep sizes in order to both ensure stability and limit the time truncation error.

AIM (*A*daptive *I*mplicit *M*ethod)

Rather than providing a fixed degree of implicitness in every gridblock at every timestep (or iteration), the AIM method assigns different levels of implicitness to the adjacent gridblocks [261, 262]. As a calculation proceeds these levels shift in space and in time as needed to maintain stability. As a switching criteria from explicit to implicit a restriction on the allowable saturation change can be used, for instance.

The mathematical procedure starts by labeling the implicit-explicit mix of unknowns and composes then the matrix problem. The matrix problem can be reduced as long as there are unknowns to be computed explicitly, therefore appropriate operations are performed to put it into a reduced form. This form is block upper-triangular with the left-upper part being diagonal and the remaining two non-zero block matrices possessing less of a band structure than the corresponding parts in the original matrix. As the degree of implicitness is allowed to change during the course of the simulation, so will accordingly also the dimensions of the matrix blocks and the associated computational costs (in terms of both the memory and the CPU time).

2.5 Single-Phase Flow

Single-phase reservoir flow is in this thesis always assumed to be slightly compressible and isothermal, which admits a model by a *linear parabolic diffusion equation with constant coefficients* for the fluid-pressure [25, 37, 209]. For instance, for a 2D, horizontal flow on a Cartesian grid, the model is:

$$\begin{aligned} \frac{\partial(\varphi\rho)}{\partial t} &= \nabla\cdot[\frac{\rho}{\mu}\mathbf{K}\nabla p] + \rho q \\ p(x,y,0) &= g(x,y), \end{aligned} \tag{2.5.1}$$

where, as usual, $\mathbf{K}$ and φ are the permeability tensor and the porosity of the medium, respectively, ρ and μ are the density and the viscosity of the fluid, and the source term $q\,[\frac{1}{s}]$ expresses a flow rate per unit volume (positive for injection, negative for production). For a slightly compressible flow (i.e., both fluid and

rock), a straightforward manipulation of (2.5.1) yields:

$$\begin{aligned} \mu\varphi c_t \frac{\partial p}{\partial t} &= \nabla \cdot \left(\frac{\rho}{\mu} \mathbf{K} \nabla p \right) + \rho q \\ p(x, y, 0) &= g(x, y), \end{aligned} \tag{2.5.2}$$

with c_t the (constant) total compressibility coefficient ($c_t = c_r + c_f$, with $c_r = \frac{1}{\varphi}(\frac{\partial \varphi}{\partial p})_T$ and $c_f \frac{1}{\rho}(\frac{\partial \rho}{\partial p})_T$ the as constant assumed rock and the fluid compressibility coefficients, respectively).

Spatial discretization of (2.5.2) will yield a model of the form:

$$\begin{aligned} \mathbf{E} \frac{d\mathbf{x}}{dt} &= \mathbf{A}\mathbf{x} + \tilde{\mathbf{q}}, \\ \mathbf{x}(0) &= \mathbf{x}_0, \end{aligned} \tag{2.5.3}$$

whereby $\mathbf{x} = \mathbf{p}$ are the grid-block pressures, and the constant diagonal matrix $\mathbf{E} = \mathbf{V}$ and the constant symmetric (sparse) matrix $\mathbf{A} = -\mathbf{T}$ are so-called 'accumulation' matrix and 'transmissibility' matrix, respectively. For instance, if a finite-difference spatial approximation is used, the matrix $\mathbf{A}$ consists of a tridiagonal part corresponding to the x-derivative, and two off-diagonal bands corresponding to the y-derivatives. Such a 'regular' structure of the matrix $\mathbf{A}$ arises from the standard *row* (or *column*) ordering of the variables (grid numbering). Different ordering schemes, like e.g. 'alternating diagonal ordering' (D4-ordering), 'red black ordering, 'nested dissection ordering', etc., which all lead to different *fill-ins* in the matrix factorizations (e.g., LU, Cholesky) of $\mathbf{A}$ (in terms of both the pattern as well as the magnitude of the non-zero elements) are not considered here. The reason is two-fold: a) orderings performing best in direct solution methods (i.e., using full factorizations) may actually perform rather worse (in terms of the number of iterations) in a 'incomplete preconditioned factorization' setting (e.g., ICCG) (see, e.g., [73] and references therein); b) the performance of the reduced-order models is, at least for linear systems, generally ordering-insensitive, due to both the small size and, as clarified in the next chapter(s), the non-sparsity of the reduced-order matrices.

3 CHAPTER

Proposed Methodology: Model Reduction by Projections

This chapter presents the framework of a methodology to achieve reservoir models dimensionality reduction based on *projection-based Model Order Reduction* (MOR). The chapter starts with a short introduction on the reasoning behind the methodology. The basic (finite-dimensional) mathematical setting of the projection-based MOR methodology is outlined in Section 3.2, followed by an overview of some of the possible choices for projection subspaces in Section 3.3.

3.1 Projection-based MOR: General Idea

Loosely speaking, projection-based MOR methods are characterized by i) 'compressing' the state's information by *projecting* the state variable onto a subspace of lower dimension than the state itself, and ii) re-writing, generally also by a projection, the system's dynamics equations in a compressed representation. The key idea hereby is that not all output of a model is evenly relevant to a particular problem. For some applications such reasoning is intuitively justified. For example, when a model is to be used for control or optimization purposes, one is generally not interested in an accurate approximation of all the variables (e.g. the states) involved in the computation, but rather in a 'good-enough' approximation of only a (very) small part of the model output. Combining the information from the high-dimensional model in a suitable way and compressing it e.g. by projecting the equations on well-defined 'dominant or important directions', reduced-order models can be obtained having a much lower dimension than the original high-order model. Similar arguments also hold for inverse problems like parameter and state estimation or prediction, where a reduced-order model may suffice if statistically accurate.

The techniques and approaches in this thesis rely highly on those developed primarily in the well-established research fields of numerical system and control theory, mechanical and structural engineering, electric circuit analysis, statistical

analysis and turbulence theory. Generally, for *single-phase* (slightly compressible) reservoir flow there are much more possibilities for efficient MOR, simply because it can be described by linear time-invariant (LTI) dynamic models for which exists a vast literature on rather systematically developed reduction approaches and stable and efficient algorithms (Chapter 4). Moreover, some characteristics inherent to multi-phase systems, like, e.g., the strongly coupled character of different physical phenomena and fluid-front interfaces moving all over the spatial domain, simply do not exist in these systems.

3.2 Mathematics of Projection-Based ROM

In the mathematical language, the projection-based class of model order reduction can be described as follows. Let the dynamic behaviour of the system under consideration be modeled by a system of equations as (2.3.5), with a given state initial condition, $\mathbf{x}(0) = \mathbf{x}_0$. In order to accommodate applications where one is interested in accurately approximating the behavior of only a small part of the model output (e.g., pressures at injectors and/or flow-rates at producers), let the "input-to-state" model (2.3.5) be augmented by an "output equation" modeling the behavior of these variables:

$$\begin{aligned} \mathbf{E}(\vartheta(\mathbf{x}))\frac{d\mathbf{x}}{dt} &= \mathbf{f}(\vartheta(\mathbf{x}), \mathbf{u}), && (3.2.1)\\ \mathbf{y} &= \mathbf{h}(\mathbf{x}, \mathbf{u}), && (3.2.2)\\ \mathbf{x}(0) &= \mathbf{x}_0. && (3.2.3) \end{aligned}$$

All projection based model reduction methods share the following feature: they determine k-dimensional ($k \ll n$) subspaces S_1 and S_2 and form a reduced system as the result of projection of the state onto S_1 and the resulting equation residual onto S_2. For arbitrary matrices $\mathbf{V}$ and $\mathbf{W}$ of size $n \times k$ spanning S_1 and S_2, respectively, the resulting reduced order model describing the time-evolution of the k-dimensional vector of the expansion coefficients $\mathbf{a}$ is given by:

$$\begin{aligned} \mathbf{E}_r(\vartheta(\mathbf{Va}))\frac{d\mathbf{a}}{dt} &= \mathbf{f}_r(\vartheta(\mathbf{Va}), \mathbf{u}), && (3.2.4)\\ \widetilde{\mathbf{y}} &= \mathbf{h}(\mathbf{Va}, \mathbf{u}),, && (3.2.5)\\ \mathbf{a}(0) &= \mathbf{V}^{\dagger}\mathbf{x}(0), && (3.2.6) \end{aligned}$$

where $\mathbf{E}_r(\vartheta(\mathbf{Va})) = \mathbf{W}^T\mathbf{E}(\vartheta(\mathbf{Va}))\mathbf{V} \in \mathbb{R}^{k\times k}$ and $\mathbf{f}_r(\vartheta(\mathbf{Va}) = \mathbf{W}^T\mathbf{f}(\vartheta(\mathbf{Va}), \mathbf{u}) \in \mathbb{R}^k$, and $\mathbf{V}^{\dagger} := (\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T$ is the pseudo-inverse of $\mathbf{V}$. If subspaces S_1 and S_2 are equal the projection is called orthogonal, otherwise it is oblique. In the sequel, matrices $\mathbf{W}$ and $\mathbf{V}$ which column spaces span S_2 and S_1, respectively, will be referred to as the *left projection matrix* and the *right projection matrix*, respectively.

When the matrix $\mathbf{E}$ is nonsingular, (3.2.1) could, at least formally, be converted into the standard state-space form (2.3.2), in which case a reduced-order model

obtained by a procedure like above could be expressed as:

$$\begin{aligned} \mathbf{W}^T\mathbf{V}\frac{d\mathbf{a}}{dt} &= \mathbf{W}^T\mathbf{E}^{-1}(\vartheta(\mathbf{Va}))\mathbf{f}(\vartheta(\mathbf{Va}),\mathbf{u}), && (3.2.7)\\ \widetilde{\mathbf{y}} &= \mathbf{h}(\mathbf{Va},\mathbf{u}), && (3.2.8)\\ \mathbf{a}(0) &= \mathbf{V}^{\dagger}\mathbf{x}(0). \end{aligned}$$

For computational reasons, projection matrices $\mathbf{W}$ and $\mathbf{V}$ are often enforced to be 'bi-orthogonal', i.e. $\mathbf{W}^T\mathbf{V} = \mathbf{I}_k$, and the state-dynamics equation would then read as: $\frac{d\mathbf{a}}{dt} = \mathbf{W}^T\mathbf{E}^{-1}(\vartheta(\mathbf{Va}))\mathbf{f}(\vartheta(\mathbf{Va}),\mathbf{u})$. Due to the matrix inversion operation, however, this form is generally not preferable for large systems. For the reduced system of equations (3.2.1) with $\mathbf{E}_r$ having full-rank this restriction would generally not hold, as the dimension of the projection subspaces in practical applications will be required to be rather small.

Figure 3.1 illustrates where the MOR as described above fits in the overall modeling process from setting up the (PDE) equations describing the physical process under consideration (here the two-phase porous-media flow) until forming a highly reduced model suitable for the intended purpose. We note that the discretized high-order reservoir flow system is formulated in the standard state-space form (2.3.2) and the particular time-discrete quasi-implicit approximation (2.4.2). This figure thus also shows that the projection-based model reduction is not restricted to the continuous-time state-space descriptions, but that it can equally well be performed on their discrete-time approximations (cq. Section 2.4). Each different discrete-time approximation will thereby generally lead to a different reduced-order model, perhaps with different computational and numerical properties.

Special case: Linear Systems

A special class of state-space systems are so-called Linear Time Invariant (LTI) state-space[1] systems, which generalized continuous-time form is:

$$\begin{aligned} \mathbf{E}\frac{d\mathbf{x}}{dt} &= \mathbf{Ax}+\mathbf{Bu}, \quad \mathbf{x}(0)=\mathbf{x}_0, && (3.2.9)\\ \mathbf{y} &= \mathbf{Cx}+\mathbf{Du}, && (3.2.10) \end{aligned}$$

where $\mathbf{E},\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times m}$ and $\mathbf{C} \in \mathbb{R}^{p\times n}$ are matrices, generally sparse for large systems, with constant coefficients, and the constant matrix $\mathbf{D} \in \mathbb{R}^{p\times m}$ models the possible instantaneous dependence of system's outputs on its inputs. The existence of a non-zero $\mathbf{D}$ depends solely on the choice of the output variables (see Example 3.2.1). On the other side, as the term $\mathbf{Du}$ does not depend on the state of the system, it therefore does not have any impact on any projection-based MOR

[1] Not every linear system can be described in a state-space form. Pure delay systems, for example, where the output of the system is just a delayed system's input, do not allow a (finite-dimensional) state-space representation. Moreover, it might, depending on the intended use of the model, also just be the modeler's choice to prefer a representation different than a state-space one (e.g., impulse-response in time domain, transfer-function in frequency/Laplace domain, etc.)

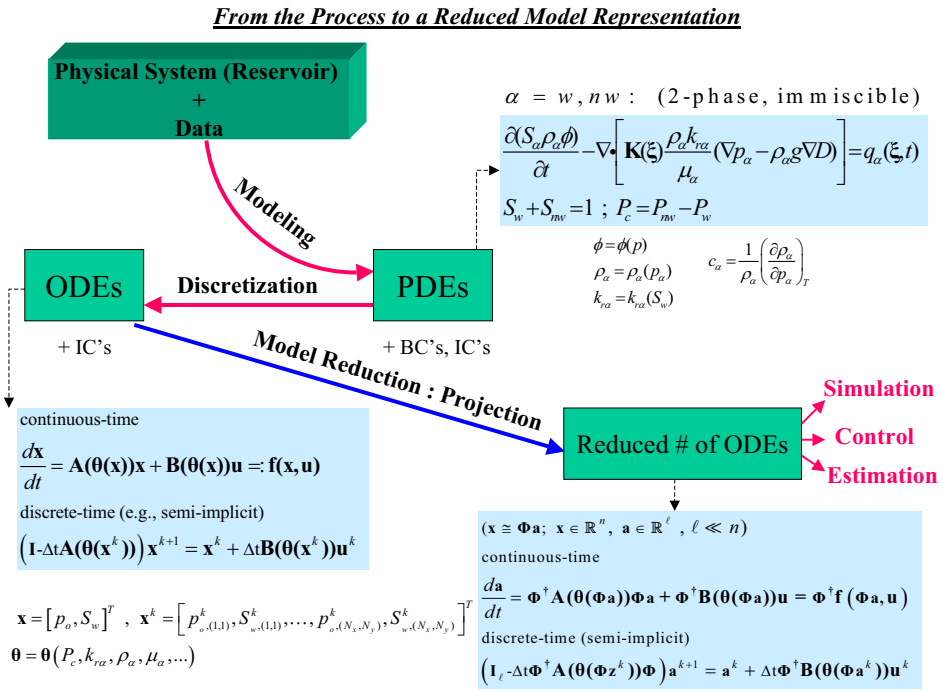


Figure 3.1: From a PDE to a low-dimensional ODE reservoir flow description using the projection MOR framework.

procedure or system properties like controllability, observability, etc. Indeed, one can always redefine the system output as $\mathbf{y} \leftarrow \mathbf{y} - \mathbf{Du} \equiv \mathbf{Cx}$. Therefore, for the purpose of presenting the theory of MOR, in the sequel of the thesis it will be frequently assumed that $\mathbf{D} \equiv \mathbf{0}$. Projection-based reduced-order models for (3.2.9-3.2.10) then have the form:

$$\begin{aligned} \mathbf{E}_r \frac{d\mathbf{a}}{dt} &= \mathbf{A}_r\mathbf{x} + \mathbf{B}_r\mathbf{u}, \quad \mathbf{a}(0) = \mathbf{V}^\dagger\mathbf{x}_0, \\ \tilde{\mathbf{y}} &= \mathbf{C}_r\mathbf{a}, \end{aligned} \tag{3.2.11}$$

with $\mathbf{E}_r = \mathbf{W}^T\mathbf{EV} \in \mathbb{R}^{k\times k}$, $\mathbf{A}_r = \mathbf{W}^T\mathbf{AV} \in \mathbb{R}^{k\times k} \in \mathbb{R}^{k\times k}$, $\mathbf{B}_r = \mathbf{W}^T\mathbf{B} \in \mathbb{R}^{k\times m}$, and $\mathbf{C}_r = \mathbf{CV} \in \mathbb{R}^{p\times k}$, and $\tilde{\mathbf{y}}$ is an approximation of $\mathbf{y}$. The reduced-order matrices can all be precalculated, hence simulation with these models is generally very fast (e.g., using direct solvers).

Example 3.2.1. (*Single phase, slightly compressible porous-media flow*)

An illustrative example of an LTI state-space model is the single-phase parabolic diffusion equation (2.5.1) describing the fluid-pressure in a horizontal reservoir on a Cartesian grid, which, for slightly compressible fluid and rock, takes the form

(2.5.2):

$$\begin{aligned} \mu\varphi c_t \frac{\partial p}{\partial t} &= \nabla \cdot \left(\frac{\rho}{\mu} \mathbf{K} \nabla p \right) + \rho q \\ p(x, y, 0) &= g(x, y), \end{aligned} \tag{3.2.12}$$

with c_t the (constant) total compressibility coefficient ($c_t = c_r + c_f$, with $c_r = \frac{1}{\varphi}(\frac{\partial \varphi}{\partial p})_T$ and $c_f \frac{1}{\rho}(\frac{\partial \rho}{\partial p})_T$ the as constant assumed rock and the fluid compressibility coefficients, respectively).

A finite-difference spatial approximation of (3.2.12) will yield a model of the form (3.2.9-3.2.10), whereby $\mathbf{x} = \mathbf{p}$ are the grid-block pressures, the diagonal matrix $\mathbf{E} = \mathbf{V}$ and the symmetric (banded) matrix $\mathbf{A} = -\mathbf{T}$ are so-called 'accumulation' matrix and 'transmissibility' matrix, respectively, and the precise form of the input-, the output- and the 'direct term' matrices $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$, respectively, depends on the choice and type of the inputs and the outputs. For example, if the output variables are chosen to form a subset of the reservoir gridblock pressures (i.e. the states) only, $\mathbf{C}$ will just be a 'selector' matrix containing only zeros except at places corresponding to the grid-blocks of interest (normally the grid-blocks connected to a well), and $\mathbf{D}$ will be zero. The case $\mathbf{D} \neq \mathbf{0}$ can e.g. occur when some or all of the output variables are chosen to be (physically more sensible and interesting) flow-rates, modeled by a well-model like e.g. (2.2.1), i.e. $q_i = \omega_i(p_{wf,i} - p_i)$. Such a choice implies that the corresponding grid-block flows are controlled by the well flowing pressures $p_{wf,i}$ at those positions (that is, $p_{wf,i}$ form a subset of the input variables). Note that in this case the diagonal element of the transmissibility matrix $\mathbf{T}$ corresponding to these grid-blocks will change by an amount of ω_i, causing $\mathbf{T}$ to become non-singular, as opposite to the singular case when all inputs are flow-rate controlled ones.

Remark 3.2.1. For a nonsingular $\mathbf{E}$, it would be possible to rewrite (3.2.9) in the standard state-space form $\frac{d\mathbf{x}}{dt} = \tilde{\mathbf{A}}\mathbf{x} + \tilde{\mathbf{B}}\mathbf{u}$, with $\tilde{\mathbf{A}} = \mathbf{E}^{-1}\mathbf{A}$ and $\tilde{\mathbf{B}} = \mathbf{E}^{-1}\mathbf{B}$, but, again, this is not preferable for large-systems due to the matrix inversion operation. Moreover, even if $\mathbf{E}$ happens to be easily invertible, using a standard state-space form so obtained may still be impractical; for instance when both $\mathbf{E}$ and $\mathbf{A}$ are sparse $\mathbf{E}^{-1}\mathbf{A}$ will generally be dense (not in the above example, though, as there $\mathbf{E} := \mathbf{V}$ is a diagonal matrix so that $\mathbf{E}^{-1}\mathbf{A} = \mathbf{V}^{-1}\mathbf{T}$ in this case is just a row-scaled version of $\mathbf{T}$).

Remark 3.2.2. For discrete-time LTI systems, that is, $\mathbf{E}\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$, $\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k$, there exist analogous MOR theories and approaches. Continuous-time LTI models can be translated into discrete-time equivalents using e.g. matrix exponential transformations or bilinear transformation of the Laplace frequency operator. In the present context the latter transformation is of particular interest. For instance, it makes it possible to solve in the continuous-time certain optimal approximation problems which are easier to formulate in the discrete-time (see, e.g., [10]), and then transform back. On the other side, due to the invariance of certain important system properties under bilinear transformation, effective iterative approximations of the related continuous-time problems can be formulated (see,

e.g., the Smith's method in Section 4.4 for computing continuous-time Gramians by transforming the corresponding Lyapunov matrix equations into an iterative sequence for solving Stein equations).

3.3 Projection subspaces and their bases

The choice for the projection matrices $\mathbf{W}$ and $\mathbf{V}$ generally depends on:

- the type of the high-dimensional model: *linear* or *nonlinear*.
- the aim of model reduction: i) *accurate approximation* of (a large part of) the whole state-space, e.g. by minimizing the error (in a suitable norm) introduced by the approximation $\mathbf{x} \approx \mathbf{Va}$, and/or the residual of the system equation, $\mathbf{E}(\vartheta(\mathbf{Va}))\mathbf{V}\frac{d\mathbf{a}}{dt} - \mathbf{f}(\vartheta(\mathbf{Va}), \mathbf{u})$, or of system's I/O behavior (for purpose of open- or closed-loop control, for instance), ii) *acceleration and/or relaxation* of solutions of optimization or inverse problems in general, etc.
- Required computational effort to compute these matrices.

In this thesis, the projection subspaces are always defined by basis functions having a (spatially) *global* character, as opposed to local basis functions used in 'finite' discretization methods (piecewise polynomials in the FEM, grid functions in the FDM, etc.). The expectation when using global spatial basis functions over local ones is that the former will be able to produce accurate, or at least 'adequate', functional representations while requiring much fewer unknowns to be resolved. When we say 'global', we mean that the basis functions are related to the flow behavior on the whole spatial domain and should therefore not be confused with 'global' basis functions used e.g. in multi-scale methods for reservoir flow [61, 79, 130, 148, 178], which employ non-local basis functions based on just a coarser grid than the underlying fine one. Moreover, 'our' basis functions for two-phase flow differ also from other 'truly' global (and grid-free) basis functions like e.g. Fourier or Chebyshev ones, in that our basis functions are determined by processing data obtained from numerical simulations of the underlying high-order model whereas the other ones are 'general' (analytic) basis functions. As general basis functions normally imply a very limited (direct) connection to the problem at hand, they may, even when mathematically convenient, be expected to be factually too general and not feasible for e.g. real-time estimation and optimization.

3.3.1 Linear Systems

Most MOR techniques are, not surprisingly, developed for approximation of the (I/O) behavior of linear time-invariant (LTI) systems, which continuous-time generalized state-space form is given by (3.2.9-3.2.10). The literature on both projection based as well as non-projection based techniques for model reduction of LTI

systems is very rich and the theory is very well-developed and relies on the well-defined and powerful system-theoretical notions of transfer function, moment matching, stability, controllability and observability, minimal state-space representations, Hankel operator theory, etc. [10–13, 86, 87, 115, 190, 260]. As for the projection-based methods, they all fall into the following major types according to the particular method of obtaining projection subspaces:

- *Krylov-subspace* (moment-matching) methods: match a certain number of moments of the system's transfer function and obtain their projection subspaces as the Krylov subspaces[2] of matrices $\mathcal{K}\{(j\omega\mathbf{E}-\mathbf{A})^{-1}, \mathbf{B}\}$ and $\mathcal{K}\{(j\omega\mathbf{E}-\mathbf{A})^{-T}, \mathbf{C}^T\}$, where $j\omega$ is the Laplace variable. All Krylov based methods are very fast but tend to create rather larger projection subspaces than necessary.
- *SVD* based methods: these obtain their projection subspaces as the 'dominant eigenspaces', e.g. of the product of so-called 'controllability' and 'observability' Gramians of the system as in the methods based on 'balanced realizations'.
- '*Snapshots*' based methods: these obtain their projection subspaces from the snapshots of the states that the system passes when simulating some training input (see also for nonlinear systems hereafter).

An interesting result that holds for this class of systems is the following.

Proposition 3.3.1. *Given a generalized LTI state-space model:*

$$\begin{aligned} \boldsymbol{E}\frac{d\boldsymbol{x}}{dt} &= \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}, \\ \boldsymbol{y} &= \boldsymbol{C}\boldsymbol{x}, \end{aligned}$$

and two k-dimensional reduced models:

$$\begin{aligned} \boldsymbol{W}_1^T\boldsymbol{E}\boldsymbol{V}_1\frac{d\boldsymbol{a}_1}{dt} &= \boldsymbol{W}_1^T\boldsymbol{A}\boldsymbol{V}_1\boldsymbol{a}_1 + \boldsymbol{W}_1^T\boldsymbol{B}\boldsymbol{u}, \\ \boldsymbol{y}_1 &= \boldsymbol{C}\boldsymbol{V}_1\boldsymbol{a}_1, \end{aligned}$$

and

$$\begin{aligned} \boldsymbol{W}_2^T\boldsymbol{E}\boldsymbol{V}_2\frac{d\boldsymbol{a}_2}{dt} &= \boldsymbol{W}_2^T\boldsymbol{A}\boldsymbol{V}_2\boldsymbol{a}_2 + \boldsymbol{W}_2^T\boldsymbol{B}\boldsymbol{u}, \\ \boldsymbol{y}_2 &= \boldsymbol{C}\boldsymbol{V}_2\boldsymbol{a}_2, \end{aligned}$$

where the column spaces of $\boldsymbol{W}_1$ and $\boldsymbol{V}_1$ are the same as those of $\boldsymbol{W}_2$ and $\boldsymbol{V}_2$, respectively. The following is true:

The two reduced order systems are equivalent in that they have the same transfer function $\boldsymbol{H}_r(s) \triangleq \boldsymbol{C}_r(s\boldsymbol{E}_r - \boldsymbol{A}_r)^{-1}\boldsymbol{B}_r$.

[2] A r-dimensional *Krylov subspace*, $\mathcal{K}_r$, of a pair $(\mathbf{A}, \mathbf{B})$ is defined as $\mathcal{K}_r = colspan\{\mathbf{B}, \mathbf{AB}, \ldots, \mathbf{A}^{r-1}\mathbf{B}\}$.

Proof. The proof is straightforward and uses the fact that there always exist $k \times k$-dimensional nonsingular matrices $\mathbf{R}$ and $\mathbf{U}$ such that

$$\begin{aligned} \mathbf{W}_1 &= \mathbf{W}_2\mathbf{R}, \\ \mathbf{V}_1 &= \mathbf{V}_2\mathbf{U}, \end{aligned}$$

so that

$$\begin{aligned} \mathbf{H}_{r1}(s) &= \mathbf{C}\mathbf{V}_1(s\mathbf{W}_1^T\mathbf{E}\mathbf{V}_1 - \mathbf{W}_1^T\mathbf{A}\mathbf{V}_1)^{-1}\mathbf{W}_1^T\mathbf{B} \\ &= \mathbf{C}\mathbf{V}_2\mathbf{U}[\mathbf{R}^T\mathbf{W}_2^T(s\mathbf{E} - \mathbf{A})\mathbf{V}_2\mathbf{U}]^{-1}\mathbf{R}^T\mathbf{W}_2^T\mathbf{B} \\ &= \mathbf{C}\mathbf{V}_2\mathbf{U}[\mathbf{R}^T(s\mathbf{W}_2^T\mathbf{E}\mathbf{V}_2 - \mathbf{W}_2^T\mathbf{A}\mathbf{V}_2)\mathbf{U}]^{-1}\mathbf{R}^T\mathbf{W}_2^T\mathbf{B} \\ &= \mathbf{C}\mathbf{V}_2\mathbf{U}\mathbf{U}^{-1}(s\mathbf{W}_2^T\mathbf{E}\mathbf{V}_2 - \mathbf{W}_2^T\mathbf{A}\mathbf{V}_2)^{-1}\mathbf{R}^{-T}\mathbf{R}^T\mathbf{W}_2^T\mathbf{B} \\ &= \mathbf{C}\mathbf{V}_2(s\mathbf{W}_2^T\mathbf{E}\mathbf{V}_2 - \mathbf{W}_2^T\mathbf{A}\mathbf{V}_2)^{-1}\mathbf{W}_2^T\mathbf{B} \\ &= \mathbf{H}_{r2}(s). \end{aligned}$$

□

This result is of fundamental importance for the development of projection-based MOR techniques and algorithms. Indeed, it states that *the choice of bases for S_1 and S_2 is actually immaterial for the I/O behavior. That is, the exact projection matrices are not important, only their column spaces are. Any two pairs of the left and the right projection matrices whose columns span S_2 and S_1, respectively, will be 'I/O equivalent'* (numerically, though, there may be considerable differences between such equivalent models: conditioning, etc.).

3.3.2 Nonlinear State-Space Systems

Model reduction of nonlinear dynamical systems is a much more difficult problem. For example, a suitable matrix projection pair in the LTI case is by many of the aforementioned reduction techniques determined using only system matrices $\mathbf{A}$, $\mathbf{B}$ and/or $\mathbf{C}$, which is in a general nonlinear case clearly impossible. While a constant basis can always be determined based on a linearization of the (smooth-enough) nonlinear model, thus using only the linearized system matrices, the validity of such a basis will in a general case clearly be restricted to a very limited range of inputs and/or small time intervals, which is simply a consequence of the fact that linearizations approximate only around the expansion points, i.e. locally [123]. To improve validity and accuracy more terms in a Taylor expansion of the nonlinear terms could be used, e.g. two as in a 'quadratic reduction method' where the nonlinear vector field $\mathbf{f}$ in $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{B}\mathbf{u}$ is approximated by the first tow-terms of the Taylor series of $\mathbf{f}$ around an expansion point in the state-space [59]. Although the approximate state dynamics obtained this way involves some of the nonlinearity of the original nonlinear model, the reduction basis (normally spanning a Krylov subspace and found by assistance of an Arnoldi process) is still based only on the linear part, i.e. the Jacobian matrix, of $\mathbf{f}$.

The same is the case also in another technique for nonlinear MOR based on linearizations, the so-called *trajectory-based piecewise linear* (TPWL) *MOR* [225–228, 263, 284], which is based on forming a quasi-piecewise-linear approximate representation of the system nonlinearity as a weighted sum of local linearizations around several 'suitably chosen' expansion points. For each of the linearized models a reduction basis can be obtained by a linear MOR technique and the thus obtained reduced bases could be aggregated and, if necessary, bi-orthogonalized (e.g., by SVD) in order to remove (almost) linearly dependent vectors present in the different bases to obtain the final projection matrices $\mathbf{V}$ and $\mathbf{W}$. Combining projection bases implies covering a larger part of the system's nonlinearity than in the other two approaches mentioned above. It also means that the final projection bases have 'time-global' character, as opposite to a time-local character of those employed in the first two approaches described above (single first-order and quadratic nonlinearity approximation). As covering the whole state-space of the high-dimensional nonlinear model with linear approximations would of course be, if feasible at all, extremely expensive, one generally uses one or more training trajectories of the high-dimensional system, corresponding to selected training input(s) (and parameters, if the system description except the states explicitly also involves parameters [44]). As exact high-dimensional non-linear simulations are generally expensive, for determining the linearization points one may instead perform cost-effective approximate simulations, e.g. using reduced order linearized systems; as long as the approximate trajectories remain 'close' to the exact ones, the collection of the linearized representations will 'cover' the suitable part of the state-space. The reduced linearized systems in such a procedure could e.g. be reduced-order representations of the subsequently determined linearized models 'as one goes along the training trajectory'; that is, at a given linearization point one may generate a linearized system, determine a reduced-order model of it, and then simulate this reduced model to reach the next linearization point [225]. Issues that need to be addressed in every particular situation when using TWPL are the validity of the nonlinearity approximations, how to chose the linearizations and how many of them, how to determine the 'optimal' weights in the approximation of the nonlinearities in the piece-wise linear MOR, what are 'suitable' training inputs and/or parameters, etc.

Data-driven Projection Subspaces

In all the above techniques it is the dynamical system which is approximated. In the 'data-driven' MOR methods, the approximation is w.r.t. the state itself, that is, snapshots of the solution of the system are created (by simulation or experiment) and a global basis is extracted form these snapshots that approximates the (created!) states well in a certain sense. Some of the most prominent choices for projection bases from this class are described below. Being based on actual system's solution data, projection bases determined by these methods are expected to generally be of lower dimensions than those obtained by TPWL for models involving severe nonlinearities. Indeed, the TWPL in this case needs a lot of linearizations to cover a suitable part of the state-space, implying a large reduced basis. On the other side, TPWL may be expected to be more robust w.r.t. changes in the inputs and parameters. Moreover, it is generally much cheaper to evaluate

as all local Jacobians are already reduced and stored. This, however, also implies that TWPL requires much more storage. Conceptually, snapshot MOR techniques can be combined with TWPL, which could e.g. be beneficial w.r.t. the cost in evaluation of the nonlinear terms in the dynamical model.

It appears that all these 'hybrid' methods are conceptually related to the so-called 'reduced-basis methods' developed in the late 1970's and early 1980's in [8, 196, 199, 200] in the context of nonlinear structural analysis and later developed more broadly in [28, 29, 32, 89, 118, 138–142, 181, 202, 215, 235], for instance. The most prominent of these early techniques for transient and/or parameter dependent problems are:

- Lagrange polynomial approach: uses as the basis basis functions solutions of the complex (here, high-dimensional) problem at several time-instances during the evolution of the system or for several values of the parameters that appear in the system's specification.
- Taylor polynomial approach: uses as the basis functions the solutions of the problem along with their derivatives w.r.t. the parameters evaluated at a reference point in the parameter space. While the solution values are obtained by evaluating the non-linear system equations, the derivative values are generally obtained by solving linear systems of matrix equations formed by differentiating the system equations w.r.t. the parameter. The linear systems for all the derivatives have the same coefficient matrix, so the computations require only backsolves (if a suitable matrix factorization is feasible, of course).
- Hermite polynomial approach: combines the above two by using as the basis functions the solutions and their derivatives for various parameter values.

In all these, adding new elements will increase the conditioning of the reduced system matrices, which in turn affects the accuracy of the reduced-basis method. To obtain a better-conditioned reduced-order system, the basis can be orthogonalized, usually by some Gram-Schmidt procedure.

An orthonormal basis is employed also in the Proper Orthogonal Decomposition (POD) approach, which is the most popular ROM approach for complex systems today. In this thesis, a member from this MOR class has been decided to employ and assess the applicability and performance of. The standard POD approach also uses basis functions that span a data set collected at several instants of time or by evaluating the computational solution for several values of the parameters appearing in the problem description or by a combination of the two, but now in a certain "optimal" fashion. As we shall see in Chapter 5, the POD basis elements are optimal in the sense that they guarantee, in the mean square error sense, the best reconstruction of the snapshots used to determine the basis among all (linear) orthonormal bases of the same size. *It is this technique which has been employed*

in this thesis for generating low-dimensional two-phase reservoir representations. The technique can equally be applied on high-dimensional linear problems. The approximation is optimal in an average sense, which may not always be the most suitable choice for multi-phase flow systems as these exhibit features as moving-fluid interfaces, shocks, etc. On the other side, these features are so characteristic that they form a real challenge for any reduction approach anyway, and it is not a priori clear which low-dimensional basis set would be the most appropriate in a particular case. To deal with some of these issues, adaptations of the standard POD approach are proposed in Chapters 5 and 5.3. Other variants on the standard POD approach worth investigating are:

- Principal Interval Decomposition (PID). The standard POD is optimal in a time-average sense and may therefore smooth out time-local information. It is therefore often not very suitable for approximating traveling structures with few modes, or more generally, for reducing models describing 'non-ergodic' phenomena/signals[3]. The (not very well known) idea of PID proposed in [132, 133] is to partition the total time interval of interest into non-overlapping time intervals and compute the dominant POD basis element(s) over each subinterval. The PID technique is further investigated and extended to reduce models over parameter ranges in [45]. If only one POD basis element is required to be determined for some or all of the subintervals, the Power Method for calculation of the largest eigenvalue and the corresponding eigenvector of a matrix is especially suitable. A disadvantage of the PID is that obtaining a good resolution of the intervals may require a large number of simulations.

- Traveling POD. For many systems flow systems exhibiting traveling structures it turns out that the optimal POD basis consists of modes which are determined completely by harmonic analysis, i.e. not from data. For systems with periodic or translational symmetry[4], for example, if one wishes to preserve symmetry in the reduced-order subspace the optimal basis can be shown to consist of Fourier modes. As a reduced-order model based on harmonic modes may require many modes to adequately capture the dynamics, (besides the abovementioned PID) several techniques have been proposed to overcome these fundamental limitations of the standard POD method for such systems. In some of those, symmetry is incorporated into the expansion [17, 105, 106, 134, 157, 234], e.g. by employing so-called 'traveling POD modes'. Consider the trivial but important example of the one-dimensional linear advection PDE $u_t + cu_x = 0$, which symmetry is characterized by translational invariance; indeed, the solution of this equation is given by $u(x,t) = g(x - ct)$ for arbitrary g. While the standard POD method would approximate the solution u as a truncated series of the form

[3]The standard POD method was originally developed for ergodic signals, however in most applications the ergodicity hypothesis is not fulfilled. Although it is still possible to employ the POD to extract spatial structures, low-dimensional models constructed using these structures are often of low quality and sometimes even useless.

[4]In physics, a symmetry corresponds to a mapping of physical state of a system which leaves the dynamics *invariant*, or unchanged (e.g., translation, rotation, reflection).

$u_{POD}(x,t) = \sum_{i=1}^{k} a_i(t)\phi_i(x)$, i.e. in a fixed spatial coordinate frame, a traveling POD basis is a 'moving one', in the sense that the solution is now approximated as $u_{travelPOD}(x,t) = \sum_{i=1}^{k} a_i(t)\phi_i(x-ct)$, which is just a spatial translation of the standard POD basis by the amount ct. In a traveling coordinate frame, the POD modes are no longer forced to be Fourier modes. As a result, usually much less modes are required to adequately capture the dynamics. In a more general case of traveling waves, the frame will move in a more complicated manner than simply ct, i.e. the solution approximation will have the form $u_{travelPOD}(x,t) = \sum_{i=1}^{k} a_i(t)\phi_i(x-c(t))$. If the traveling POD expansion is used to perform a Galerkin projection of the high-dimensional (here infinite-dimensional, as we deal with a PDE), then it is necessary to specify the evolution of the symmetry variable $c(t)$. Except for the simplest case ct, this is anything but a trivial exercise, and in the case of heterogeneous reservoir flow problems, especially multi-phase and in two or three spatial dimensions, it is probably even impossible.

- Centroidal Voronoi orthogonal decomposition (CVOD). This method combines the POD with a special type of the data clustering technique of Voronoi Tesselation[5] Just as in a POD-based setting, also here one starts with a snapshot set. However, in this case one does not determine a POD basis from the snapshot set, but rather determines the generators of a *centroidal Voronoi tessellation* (CVT) of the set, i.e., a Voronoi tesselation of the snapshot set such that the generators of the Voronoi sets are simultaneously their centers of mass [76].[6] The reduced order-basis for low-dimensional approximation of the solution of the high-dimensional system is constituted from these generators and is used just in the same way as one uses a POD-basis for that purpose [51–53, 74, 75]. CVOD can be understood as a generalization of CVT for which the snapshots set is divided into N clusters or generalized Voronoi subsets $\{\mathcal{V}_i\}_{i=1}^{N}$ having as their generators d_i-dimensional spaces each of which being spanned by the d_i-dimensional POD basis for the cluster. CVOD can also be seen as a generalization of POD for which the snapshots set is divided into N clusters and then a POD basis is separately determined for each cluster. For $N = 1$ CVOD reduces to the standard POD and for $d_i = 1$ $i = 1, \ldots, N$ it reduces to the standard CVT (see footnote). Compared to POD-based MOR approaches, CVOD is potentially less expensive; POD involves the solution of an $M \times M$ (dense) eigenproblem, where M is the number of snapshots, whereas CVOD requires the solution of several smaller eigenproblems.

[5] Divide a given set $\mathcal{S}$ into K subsets $\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_K$ such that a) no member of a set $\mathcal{S}_i$ is a member of another set $\mathcal{S}_j$ (i.e. $\mathcal{S}_i \cap \mathcal{S}_j = \emptyset, i, j \in \{1, 2, \ldots, K\}, i \neq j$) and b) every member of $\mathcal{S}$ belongs to one of the sets $\mathcal{S}_i$ (i.e. $\cup_{i=1}^{K} \mathcal{S}_i = \mathcal{S}$). The set of subsets $\{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_K\}$ is called a *tessellation of* $\mathcal{S}$. Let $\mathcal{S}$ be a set consisting of M points. Given N 'elements' $\{z_i\}_{i=1}^{N}$ ($N \leq M$) and a 'distance function' $d(z, \omega)$ for $\omega \in \mathcal{S}$, the *Voronoi subset* $\mathcal{S}_j$ is defined as $\mathcal{S}_j = \{w \in \mathcal{S} | d(w, z_j) < d(w, z_i), i = 1, \ldots, N, i \neq j\}$, that is $\mathcal{S}_j$ is the set of all elements belonging to $\mathcal{S}$ that are closer to z_j than to any of the other elements z_i. The set of Voronoi subsets $\{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_N\}$ is called a *Voronoi tessellation of* $\mathcal{S}$ or a Voronoi diagram of S and the points $\{z_i\}_{i=1}^{N}$ are the *generators* of the Voronoi tessellation.

[6] For discrete data, CVTs are closely related to the k-means and h-means clustering techniques. Just as POD bases do, CVT also possesses an optimality property, although a different one.

- "Structure-Preserving" POD. For flow problems defined on 2D and 3D spatial domains, the standard POD approach performs the analysis on the 'vectorized' data. Recently developed "Generalized Low-Rank Approximations" techniques GLRAM, 2DPCA, HOOI, etc. [160, 248, 294, 297–299] may appear valuable in overcoming this issue as they work directly on 2D (3D) solution fields, i.e. without vectorizing them first as is done in a standard POD analysis. The solutions of the associated optimal approximation problems involving non-vectorized data are determined in an iterative fashion. The convergence of the algorithms employed is generally fast.

- Sobolev-norms based POD. A good approximation of the state-variables doesn't necessarily imply a good approximation of those variables of interest which depend mostly on spatial derivatives of state-variables. Flowrates in reservoir flow systems are a typical example as they depend on pressure spatial derivative. On the other side, accurate approximation of the states are important for accurate approximation of state-dependent flow-parameters, so an improvement over the standard POD basis may, in principle, be achieved by a weighted combination of the (auto-correlations of) the snapshots of state-solutions and their spatial derivatives [137, 155, 157, 170]. More formally, in this approach all the norms are defined in terms of a (weighted) Sobolev norm rather than in the L_2 norm employed in an usual POD setting. Application of this idea to a two-phase porous media flow problem in [185] indicates that sometimes there is indeed space for improvement compared to the standard POD using such an approach. It is not clear, however, how the weights should be chosen in an automatic fashion. Indeed, it turns out to be difficult to provide a general guide on how to effectively exploit this scope in a particular situation. Namely, choosing for a 'mixed' optimality criterion based on a part of or all present spatial derivatives may actually negatively influence the approximation of the state-dependent parameters (e.g. density, relperms, etc.).

- Balanced POD, Empirical Balancing, etc.. Besides the sensitivity to the used norms, another disadvantage of the standard POD is the fact that the standard POD-modes approximate the dominant controllable subspace of the system only.[7]. The methods from this class focus on taking, by resembling the idea of balanced realizations[8], also the observability of the system into account [105, 134, 164, 165, 184, 233].

Clearly, any of the snapshots-based bases can only be as good as the snapshot sets used to generate the basis. The generation of good snapshot sets is, however, in general by no means a science, but rather an (primitive) art. The best results may actually be expected if the generation of snapshot sets is supported by design of

[7] This point is explained in detail in Chapter 5.

[8] For a systematic (local) extension of the theory of balanced realizations and BTR to (smooth, input-affine) stable finite-dimensional nonlinear systems, see [243–246]. Unfortunately, the 'exact' non-linear balancing involves solving two PDEs of the Hamilton-Jacobi type and is therefore not applicable to large non-linear systems.

experiments methodologies. Some a priori knowledge about the types of states to be simulated or optimized using the reduced-order model would be very useful in this respect.

We end this section by pointing out that generating the projection bases by any of these data-driven methods may be very costly, as they are determined by employing high-dimensional simulations. In simulation settings, that cost is justified only if the reduced-order model is subsequently used for calculations for many system's input forms and/or parameter values different than those used to generate the reduced-basis. Similarly, in control or optimization settings the cost is amortized over the many (co-)state solves performed to determine optimal controls (Chapter 6).

4 CHAPTER

MOR for Single-Phase Flow

The aim of this chapter is to present modern approaches for linear time-invariant (LTI) MOR and to assess their performance when applied to single-phase reservoir flow problems. The methods presented are: *Modal Truncation*, *Singular Perturbation* (or 'rezidualization'), *Transfer Function Moment Matching* (explicit and implicit), and *Balanced Truncation*. While conceptually and algorithmically highly sophisticated, neither of these methods appear to be (directly) applicable to nonlinear models, the ones we are ultimately interested in, except when the nonlinearity is rather weak [123]. The motivation for including the methods for MOR of LTI systems is fourfold: i) proper understanding of the MOR problem in general, ii) the fact that the method we will employ later for model reduction of two-phase flow models can be related to one of these methods, iii) the fact that single-phase simulations are often used to aid in solving the underlying nonlinear problem, and iv) there are important reservoir flow problems that can be approximated as single-phase in practice (e.g., upscaling [268, 269]).

4.1 System-Theoretic Preliminaries

Some of the presented methods are based on rigorous system-theoretical concepts of 'controllability', 'observability', Hankel operators and system norms, transfer function moments, etc. The necessary system-theoretical background is provided first. More elaborated introduction to LTI systems can be found in many textbooks [114, 254, 307], handbooks [169, 193], and many other sources.

For the sake of both simplicity and clarity of presentation, in this section the following assumptions are made:

- The (continuous-time[1]) LTI model under consideration is in the standard

[1]Except a small part in Section 4.4, Everything in this section is based on continuous-time state-space LTI models. For discrete-time (generalized) state-space LTI systems, that is, $\mathbf{E}\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$, $\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k$, analogous MOR theories and approaches exist. Continuous-time LTI models can be translated into discrete-time equivalents using e.g. bilinear transformation of the Laplace frequency

state-space form, i.e.[2]

$$\frac{d\mathbf{x}}{dt} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \quad \mathbf{x}(0) =: \mathbf{x}_0, \tag{4.1.1}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}, \tag{4.1.2}$$

with $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times m}$, $\mathbf{C} \in \mathbb{R}^{p\times n}$, and $\mathbf{D} \in \mathbb{R}^{p\times m}$. The time-solutions $\mathbf{x}(t)$ and $\mathbf{y}(t)$ are respectively given by:

$$\mathbf{x}(t) = e^{\mathbf{A}t}\mathbf{x}_0 + \int_0^t e^{\mathbf{A}(t-\tau)}\mathbf{B}\mathbf{u}(\tau)d\tau, \tag{4.1.3}$$

$$\mathbf{y}(t) = \mathbf{C}e^{\mathbf{A}t}\mathbf{x}_0 + \int_0^t \mathbf{C}e^{\mathbf{A}(t-\tau)}\mathbf{B}\mathbf{u}(\tau)d\tau + \mathbf{D}\mathbf{u}(t). \tag{4.1.4}$$

The output response in the frequency (Laplace) domain is easily shown to be given by:

$$\mathbf{Y}(s) = [\mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}]\mathbf{U}(s) + \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{x}_0 \tag{4.1.5}$$

$$= \mathbf{H}(s)\mathbf{U}(s) + \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{x}_0, \tag{4.1.6}$$

where $s =$ is the Laplace variable and $\mathbf{H}(s) := \mathbf{C}(s\mathbf{I}-\mathbf{A})^{-1}\mathbf{B}+\mathbf{D}$ is called the system's *transfer function (matrix)* and represents the response of the systems output to its input in the frequency domain for a zero initial state ($\mathbf{x}_0 = \mathbf{0}$).[3]

- **A** is (asymptotically) *stable*, i.e. all eigenvalues of **A** are confined to the left hand side (l.h.s.) of the complex plane ($\max(Re(\lambda_i)) < 0$). In the single-phase flow models in this thesis this is the case if at least one of the source flow-rates is defined as being (bottom-hole) pressure constrained. Otherwise, and unless one of the grid-block pressures is explicitly constrained to assume a value (in which case the total number of degrees would decrease by one), it is stable with a zero eigenvalue expressing the mass-conserving characteristic of the (discretized) transmissibility operator.

- A is *diagonalizable*, i.e. it is decomposable as $\mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1}$, where $\mathbf{\Lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_n)$ is the diagonal matrix of eigenvalues λ_i of **A**, and the columns $\mathbf{v}_i$ of $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n]$ are then their corresponding eigenvectors. For single-phase models in this thesis this indeed appears to be the case. Transforming the state $\mathbf{x}$ as $\mathbf{V}\mathbf{x} =: \mathbf{z}$ and premultiplying (4.1.1) by $\mathbf{V}^{-1}$ yields the

operator. The fact that certain important system properties remain invariant under the bilinear transformation is used in Section 4.4 to translate the continuous Lyapunov matrix equations for determining system's Gramians into an iterative sequence for solving Stein equations.

[2]For single-phase flow models in this thesis this is not really a restriction, as explained at the end of Sec. 3.2. For the completeness, at some places in the chapter the corresponding results for the generalized state-space systems are given as well.

[3]The non-zero initial state case is easily accommodated for by defining $\tilde{\mathbf{x}} := \mathbf{x} - \mathbf{x}_0$ and 'shifting' the output for an amount of $\mathbf{C}\mathbf{x}_0$, provided that $\mathbf{x}_0$ corresponds to a steady-state in the absence of any input signal. This is always the case in our applications. As for the MOR, it is advisable that such a shift always be applied, otherwise there will be an error component in the approximation which may be substantial unless an (otherwise unnecessarily) large projection subspace is employed.

following state-decoupled dynamic description for $\mathbf{z} = [z_1, z_2, \ldots, z_n]^T$:

$$\frac{d}{dt}\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix} = \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix}\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix} + \begin{bmatrix} \tilde{\mathbf{b}}_1^T \\ \tilde{\mathbf{b}}_2^T \\ \vdots \\ \tilde{\mathbf{b}}_n^T \end{bmatrix}\mathbf{u}, \quad (4.1.7)$$

$$\mathbf{y} = \widetilde{\mathbf{C}}\mathbf{z} + \mathbf{D}\mathbf{u} = \sum_i^n z_i\tilde{\mathbf{c}}_i + \mathbf{D}\mathbf{u}, \quad (4.1.8)$$

where $\tilde{\mathbf{b}}_i^T$ are the rows of $\widetilde{\mathbf{B}} = \mathbf{V}^{-1}\mathbf{B}$ and $\tilde{\mathbf{c}}_i$ are the columns of $\widetilde{\mathbf{C}} = \mathbf{C}\mathbf{V}$. The time-solutions for the decoupled system are given by:

$$z_i(t) = e^{\lambda_i t}z_0 + \int_0^t e^{\lambda_i(t-\tau)}\tilde{\mathbf{b}}_i^T\mathbf{u}(\tau)d\tau, \qquad i = 1, \ldots, n \quad (4.1.9)$$

$$\mathbf{y}(t) = \sum_{i=1}^n \tilde{\mathbf{c}}_i e^{\lambda_i t}z_0 + \sum_{i=1}^n \int_0^t \tilde{\mathbf{c}}_i e^{\lambda_i(t-\tau)}\tilde{\mathbf{b}}_i^T\mathbf{u}(\tau)d\tau + \mathbf{D}\mathbf{u}. \quad (4.1.10)$$

The transfer function matrix $\mathbf{H}(s)$ can be written as ($\mathbf{w}_i^T$ are the rows of $\mathbf{V}^{-1}$):

$$\begin{aligned} \mathbf{H}(s) &= \widetilde{\mathbf{C}}(s\mathbf{I} - \mathbf{\Lambda})^{-1}\widetilde{\mathbf{B}} + \mathbf{D} \\ &= \mathbf{C}\begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \ldots & \mathbf{v}_n \end{bmatrix}(s\mathbf{I} - \mathbf{\Lambda})^{-1}\begin{bmatrix} \mathbf{w}_1^T \\ \mathbf{w}_2^T \\ \vdots \\ \mathbf{w}_n^T \end{bmatrix}\mathbf{B} + \mathbf{D} \\ &= \sum_{i=1}^n [\frac{1}{s - \lambda_i}(\mathbf{C}\mathbf{v}_i)(\mathbf{w}_i^T\mathbf{B})] + \mathbf{D}, \end{aligned} \quad (4.1.11)$$

and the output response as:

$$\begin{aligned} \mathbf{Y}(s) &= \mathbf{H}(s)\mathbf{U}(s) + \widetilde{\mathbf{C}}(s\mathbf{I} - \mathbf{\Lambda})^{-1}\mathbf{z}_0 \\ &= \sum_{i=1}^n \begin{bmatrix} \frac{1}{s-\lambda_i}(\mathbf{C}\mathbf{v}_i)[(\mathbf{w}_i^T\mathbf{B})\mathbf{U}(s) + z_{0,i}] \end{bmatrix} + \mathbf{D}\mathbf{U}(s). \end{aligned} \quad (4.1.12)$$

To facilitate acquiring basic understanding of the key concepts the MOR methods in this section are based on, the introduction to these concepts will frequently be based on the assumption that the LTI system is '*SISO*', i.e. it has a single input and a single output (e.g. a single-phase reservoir flow problem with a single producer

and no injectors). In this case we have:

$$\frac{dz_i}{dt} = \lambda_i z_i + \tilde{b}_i u, \qquad i = 1, \ldots, n \tag{4.1.13}$$

$$y = \sum_{i=1}^{n} \tilde{c}_i z_i + u, \tag{4.1.14}$$

where $\tilde{b}_i$ and $\tilde{c}_i$ ($i = 1, \ldots, n$) are the entries of the vectors $\tilde{\mathbf{b}} := \mathbf{V}^{-1}\mathbf{b}$ and $\tilde{\mathbf{c}} := \mathbf{V}^T\mathbf{c}$, respectively, and $u, y, d \in \mathbb{R}$, whereas respectively the (scalar) transfer function $H(s)$ and the output response $y(s)$ are given by:

$$H(s) = \tilde{\mathbf{c}}^T(s\mathbf{I} - \mathbf{\Lambda})^{-1}\tilde{\mathbf{b}} + d = \sum_{i=1}^{n} \frac{\tilde{c}_i \tilde{b}_i}{s - \lambda_i} + d \tag{4.1.15}$$

and

$$Y(s) = \sum_{i=1}^{n} \frac{\tilde{c}_i}{s - \lambda_i}[\tilde{b}_i U(s) + z_{0,i}] + d \cdot U(s). \tag{4.1.16}$$

4.1.1 Rational Transfer Function Representation

Consider a *partial fraction expansion* of the transfer function of a n-dimensional SISO LTI model (in the sequel it will be generally assumed that the direct i/o term d is zero since $d_{red} = d \equiv \lim_{s\to\infty} H(s)$ for any projection-based MOR in this thesis. However, this 'invariance' can sometimes be exploited to refine the error in specific approximation norms, as stated in Remark 4.1.2):

$$H(s) = \sum_{i=1}^{n} \frac{c_i b_i}{s - \lambda_i} =: \sum_{i=1}^{n} \frac{r_i}{s - \lambda_i}. \tag{4.1.17}$$

Cross-multiplication of the terms in the summation yields a *rational function representation* of $H(s)$:

$$H(s) = \sum_{i=1}^{n} \frac{r_i}{s - \lambda_i} = \frac{e_0 + e_1 s + e_2 s^2 + \cdots + e_{n-1} s^{n-1}}{1 + f_1 s + f_2 s^2 + \cdots + f_n s^n}, \tag{4.1.18}$$

where the $2n$ coefficients $\{e_i\}_{i=0}^{n-1}, \{f_i\}_{i=1}^{n}$ are functions of c_i, b_i, λ_i ($i = 1, \ldots, n$).

Transfer function point- versus moment matching

Given a rational function representation (4.1.18) of the original system $H(s)$, a $k-th$ *order rational transfer function approximation* $H_r(s)$ has the form[4]:

[4]Actually, the only requirement for the reduced-order transfer function is that it must be 'proper', i.e. that the max. order of s, say ℓ, in the numerator of (4.1.19) must be smaller than that in the

$$H_r(s) \quad = \quad \frac{e_{r,0} + e_{r,1}s + e_{r,2}s^2 + \ldots e_{r,k-1}s^{k-1}}{1 + f_{r,1}s + f_{r,2}s^2 + \cdots + f_{r,k}s^k}. \tag{4.1.19}$$

Depending on what one wants to adequately approximate with H_r, there are several ways of 'choosing' the $2k$ coefficients $\{e_{r,i}\}_{i=0}^{k-1}, \{f_{r,i}\}_{i=1}^{k}$. For instance, if it were desired to match the behavior of $H(s)$ exactly in a number of frequency points s_j (Fig. 4.1), whereas the behavior in between these 'interpolation points' were not important, one could achieve that for up to $2k$ points by employing *point matching*, as shown in Section 4.3. On the other side, if one were rather interested in an excellent match of $H(s)$ and a number of its derivatives $\frac{\partial H^i}{\partial s}(s_0)$ in one or more points s_0 (Fig. 4.1), one would need to employ a *moment matching* algorithm, possibly at several frequency points.

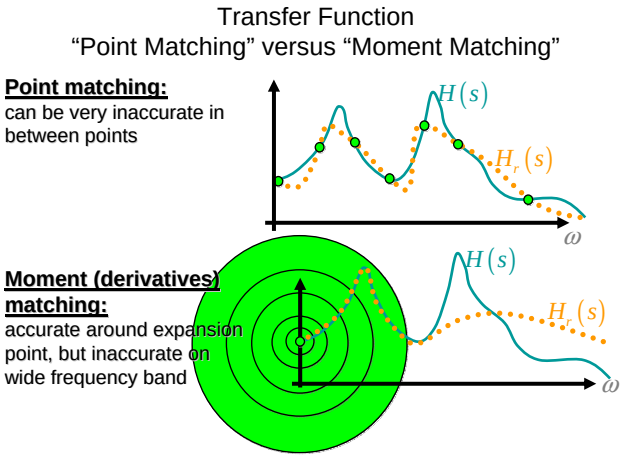


Figure 4.1: Transfer Function Point Matching versus Moment Matching.

4.1.2 Controllability, Observability, Minimal Realizations and System Operators

In (4.1.14), $\tilde{c}_j \equiv \mathbf{v}_j^T\mathbf{c} \equiv \mathbf{c}^T\mathbf{v}_j = 0$, with $\mathbf{v}_j$ the jth eigenvector of $\mathbf{A}$, would imply that no influence of $z_j(t)$ is present in the output, even though $z_j(t)$ may be influenced by the input. As at the same time the state-equations are decoupled, simulating the dynamics of the corresponding $z_j(t)$ is in this case not necessary and can thus be omitted if one is only interested in the behavior of the system's output. A mode with this property is said to be 'completely' ***unobservable***, as

denominator, k, that is, ($\ell < k$). The selection of values ℓ and k is a problem of analysis, but a natural option, e.g. due to (4.1.17), is $\ell = k - 1$.

nothing can be said about the behavior of $z_j(t)$ (in particular the initial state $z_j(0)$, as its knowledge would suffice to determine $z_j(t)$, and thus $\mathbf{z}(t)$ and ultimately $\mathbf{x}(t) = \mathbf{V}^{-1}\mathbf{z}(t)$, for all $t > 0$ for a given input) from the knowledge of $y(t)$. Similarly, $\tilde{b}_\ell \equiv \mathbf{w}_\ell^T\mathbf{b} = 0$ implies that z_ℓ is not affected by the input whatsoever. Such a mode is said to be (completely) ***uncontrollable***. It is clear that (4.1.13) does not need to be integrated in time for z_ℓ. It is, however, still required to compute $z_\ell(t) = e^{\lambda_i t}z_0$ for any t at which the output $y(t)$ needs to be determined when $z_\ell(0) \neq 0$ (unless $\ell = j$, as that would mean that the mode is simultaneously uncontrollable and unobservable).

MIMO case

Similar results hold for MIMO systems admitting a modal form as above. Considering (4.1.15) or (4.1.12), if the system (or better, the given state-space realization) is unobservable, then there is at least one λ_j which satisfies $\mathbf{C}\mathbf{v}_j = \mathbf{0}$. One refers to the pole $s = \lambda_j$ as a *hidden mode* because it does not appear in the transfer function. As for controllability, if the system is uncontrollable, then there exists at least one left eigenvector $\mathbf{w}_\ell$ satisfying $\mathbf{w}_\ell^T\mathbf{B} = \mathbf{0}^T$. Again, the pole associated with this mode does not appear in the transfer function and one refers to the pole $s = \lambda_\ell$ also as a *hidden mode*. This directly uncovers an important difference between state-space representations of a (stable) LTI system and its (unique) transfer function: whereas the poles of a transfer function are necessarily eigenvalues of the state-matrix in any state-space representation[5], the eigenvalues of a given state-space control system representation are not necessarily poles of the corresponding transfer function.

Remark 4.1.1. When a state-space system in modal form has all eigenvalues distinct, then the condition $\mathbf{w}_\ell^T\mathbf{B} = \mathbf{0}^T$ ($\mathbf{C}\mathbf{v}_\ell = \mathbf{0}$) for some $\ell \in \{1, \ldots, n\}$ is both sufficient and necessary for proclaiming the system uncontrollable (unobservable). On the other hand, when some eigenvalue is not simple, for the system to be completely controllable (observable), besides $\mathbf{w}_\ell^T\mathbf{B} \neq \mathbf{0}^T$ ($\mathbf{C}\mathbf{v}_\ell \neq \mathbf{0}$) it is also required that the rows of $\widetilde{\mathbf{B}} = \mathbf{V}^{-1}\mathbf{B}$ (columns of $\widetilde{\mathbf{C}} = \mathbf{C}\mathbf{V}$) corresponding to the repeated eigenvalue are linearly independent.

State-space dimension versus the model order: McMillan degree and minimal realizations

The above short discussion provides an easy way of reducing the dimension of the state vector of a stable LTI model: the eigenmodes which are completely unobservable and those which are completely uncontrollable can be removed from the dynamical system description without influencing the input-output behavior of the system, the behavior to be understood as the system's transfer function. In other words, the dimension of a given state-space description of an LTI system

[5] As similarity transformations of the state space do not change the (complete) controllability and observability properties of LTI systems, the same holds for the modes in any state-space representation (also called 'realization') obtained by a similarity transformation.

is not necessarily the 'intrinsic' order of the system.[6] A state-space realization of a (stable) LTI system which is both completely controllable and completely observable is said to be *minimal*. The *order* of an LTI system is thus the dimension of the state-vector in any minimal state-space representation, and is also called the *McMillan degree* of the system. The MOR in the sense of Chapter 3 assumes that the starting high-dimensional model is already in a minimal state-space form. On the other hand, the modern algorithms actually implementing MOR admit non-minimal state-space realizations as the input and remove automatically the uncontrollable and unobservable parts.

'Weak' controllability/observability

The chance that in a typical situation in practice (e.g. when the starting high-dimensional model is obtained by discretization of first-principle models) a substantial number of modes are completely unobservable and/or completely uncontrollable is not too likely. Consequently, not many (if any) of the modes in $(4.1.15)$ will satisfy $\tilde{c}_j = 0$ or $\tilde{b}_j = 0$ ($\mathbf{C}\mathbf{v}_j = \mathbf{0}$ or $\mathbf{w}_j^T\mathbf{B} = \mathbf{0}^T$ in the MIMO case $(4.1.11)$). A tempting idea is then to look at the magnitude of the non-zero $\tilde{c}_j$ and $\tilde{b}_j$ and retain only those eigenmodes corresponding to large magnitudes of these quantities. However, as a weakly observable eigenmode (i.e. small $\tilde{c}_j$) might at the same time be quite controllable (large $\tilde{b}_j$) and vice versa, it is clear that what really matters in the general case is the product of $\tilde{c}_j$ and $\tilde{b}_j$.

An important question arises here. Since there are infinitely many equivalent state-space representations, it is natural to wonder whether the mostly controllable and mostly observable modes retained in the reduced-order model should actually be the eigenmodes in the first place? Maybe there are other modes which are more controllable and more observable than the eigenmodes, thus admitting models of the same or better quality using even less modes? That appears to indeed be the case, and the BTR (Sec. 4.4) and related methods are aimed to determine state-representations that reveal such modes.

System operators, norms, and quality of approximation

Among the most popular model reduction procedures at present are those that produce (sub)optimal approximations of a given system in specific error norms. A short introduction to fundamental operators and system norms follows.

Convolution operator

For the continuous-time (stable) LTI system (4.1.1-4.1.2) one can define a *convolu-*

[6] Note that, strictly speaking, removing completely uncontrollable and completely unobservable modes cannot actually be considered as a model order reduction in the sense defined in Chapter 3, as the order of a state-space model possessing some modes uncontrollable or unobservable is intrinsically the order of the model with all the uncontrollable and unobservable modes removed.

tion operator Γ_c with kernel (impulse response matrix) $\mathbf{H}(t)$ from $\mathbf{u}(t)$ to $\mathbf{y}(t)$ as

$$\Gamma_c : \mathbf{u}(t) \to \mathbf{y}(t) \quad = \quad \mathbf{H} * \mathbf{u} = \int_{-\infty}^{\infty} \mathbf{H}(t-\tau)\mathbf{u}(\tau)d\tau. \tag{4.1.20}$$

The Laplace transform $\mathbf{H}(s)$ of the impulse response $\mathbf{H}(t) = \{\mathbf{0} : t < 0;\ \mathbf{D}\delta : t = 0;\ \mathbf{C}e^{\mathbf{A}t}\mathbf{B} : t > 0\}$) is the familiar transfer function matrix $\mathbf{H}(s) = \int_0^{\infty} \mathbf{H}(t)e^{st}dt \equiv \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D}$.

In the case of discrete-time (stable) LTI systems: $\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k, \mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k$, with the impulse response matrix $\mathbf{H}(t) = \{\mathbf{0} : t < 0;\ \mathbf{D}\delta : t = 0;\ \mathbf{C}\mathbf{A}^{t-1}\mathbf{B} : t > 0\}$) and the corresponding transfer function matrix $\mathbf{H}(z) = \mathbf{C}(z\mathbf{I} - \mathbf{A})\mathbf{B} + \mathbf{D}$ (z is the Z-transform variable), the convolution operator is defined as

$$\Gamma_d : \mathbf{u}(t) \to \mathbf{y}(t) \quad = \quad \mathbf{H} * \mathbf{u} = \sum_{k=-\infty}^{\infty} \mathbf{H}(t-k)\mathbf{u}(k). \tag{4.1.21}$$

Hankel operator

The convolution operators as defined above have infinite rank whenever the associated impulse response $\mathbf{H}$ is not identically equal to zero. However, if the domain and codomain of the convolution operator of a finite-order LTI system are restricted, finite-rank operators may be obtained possessing highly meaningful and useful properties. One such operator is the *Hankel operator*, $\mathcal{H}$, of the considered LTI system, simply described as "mapping from past (finite energy) inputs to (finite energy) future outputs". Formal definitions of the Hankel operator for the continuous-time and discrete-time LTI systems are:

Definition 4.1.1. In the *continuous-time case*, the Hankel operator $\mathbf{H}_c$ is defined as

$$\mathcal{H}_c : \mathbf{u}_-(t) \to \mathbf{y}_+(t) := \mathcal{H}_c(\mathbf{u}_-) = \int_{-\infty}^{0} \mathbf{H}(t-\tau)\mathbf{u}(\tau)d\tau, \quad t \geq 0. \tag{4.1.22}$$

In the discrete-time case, the Hankel operator $\mathbf{H}_d$ is defined as

$$\mathcal{H}_d : \mathbf{u}_-(t) \to \mathbf{y}_+(t) := \mathcal{H}_d(\mathbf{u}_-) = \sum_{k=-\infty}^{-1} \mathbf{H}(t-k)\mathbf{u}(k), \quad t \geq 0. \tag{4.1.23}$$

Hankel operators turn out to carry a number of fundamental system properties. For continuous-time systems, for instance, one can prove equality between the system order and the rank of the associated Hankel operator. Moreover, for both continuous-time or discrete-time, the order of the system (McMillan degree) is equal to the rank of the associated (block) Hankel matrix[7] characterized by Markov

[7] A *Hankel* matrix is a square matrix with constant (positive sloping, i.e. from northeast to southwest) skew-diagonals. It is an upside-down Toeplitz matrix, which is a matrix with values constant along each (northwest to southeast) diagonals.

parameter sequence (impulse response) $\{\mathbf{H}_i\}_{i=1}^{\infty}$:

$$\mathcal{H} := \begin{bmatrix} \mathbf{H}_1 & \mathbf{H}_2 & \mathbf{H}_3 & \dots \\ \mathbf{H}_2 & \mathbf{H}_3 & . & \\ \mathbf{H}_3 & . & . & \\ \vdots & & & \ddots \end{bmatrix}. \tag{4.1.24}$$

That this result is correct is easiest to 'visualize' in the discrete-time case, in which the Hankel matrix as defined above is the matrix representation of the associated Hankel operator, i.e. mapping the past inputs sequence into the future output sequence:

$$\begin{aligned} \begin{bmatrix} \mathbf{y}(0) \\ \mathbf{y}(1) \\ \mathbf{y}(2) \\ \vdots \end{bmatrix} = \mathcal{H} \begin{bmatrix} \mathbf{u}(-1) \\ \mathbf{u}(-2) \\ \mathbf{u}(-3) \\ \vdots \end{bmatrix} &= \begin{bmatrix} \mathbf{H}_1 & \mathbf{H}_2 & \mathbf{H}_3 & \dots \\ \mathbf{H}_2 & \mathbf{H}_3 & \mathbf{H}_4 & \dots \\ \mathbf{H}_3 & \mathbf{H}_4 & \mathbf{H}_5 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \mathbf{u}(-1) \\ \mathbf{u}(-2) \\ \mathbf{u}(-3) \\ \vdots \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{CB} & \mathbf{CAB} & \mathbf{CA}^2\mathbf{B} & \dots \\ \mathbf{CAB} & \mathbf{CA}^2\mathbf{B} & \mathbf{CA}^3\mathbf{B} & \dots \\ \mathbf{CA}^2\mathbf{B} & \mathbf{CA}^3\mathbf{B} & \mathbf{CA}^4\mathbf{B} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \mathbf{u}(-1) \\ \mathbf{u}(-2) \\ \mathbf{u}(-3) \\ \vdots \end{bmatrix} \\ &= \underbrace{\begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \end{bmatrix}}_{\mathcal{O}} \underbrace{\begin{bmatrix} \mathbf{B} & \mathbf{AB} & \mathbf{A}^2\mathbf{B} & \dots \end{bmatrix}}_{\mathcal{C}} \begin{bmatrix} \mathbf{u}(-1) \\ \mathbf{u}(-2) \\ \mathbf{u}(-3) \\ \vdots \end{bmatrix}. \end{aligned} \tag{4.1.25}$$

Hence, in the discrete-time case, the Hankel operator is an infinite block matrix representation that turns out to be factorizable as a product of two matrices of finite-rank. Consequently, $\mathcal{H}$ has a finite rank, too! But there is more. The matrices $\begin{bmatrix} \mathbf{B} & \mathbf{AB} & \dots & \mathbf{A}^{n-1}\mathbf{B} \end{bmatrix} := \mathcal{C}_n(\mathbf{A}, \mathbf{B})$ and $\begin{bmatrix} \mathbf{C}^T & (\mathbf{CA})^T & \dots & (\mathbf{CA}^{n-1})^T \end{bmatrix}^T =: \mathcal{O}_n(\mathbf{A}, \mathbf{C})$ are, in both continuous-time and discrete-time, called *controllability* (or reachability) *matrix* and *observability matrix*, respectively, as it can be proven (e.g., [150]) that an n-dimensional (realization of an) LTI system is fully controllable (observable) if and only if $\mathcal{C}_n(\mathbf{A}, \mathbf{B})$ ($\mathcal{O}_n(\mathbf{A}, \mathbf{C})$) is of rank n. Due to the Cayley-Hamilton theorem, it holds that $rank(\mathcal{C}_n(\mathbf{A}, \mathbf{B})) = rank(\mathcal{C})$ and $rank(\mathcal{O}_n(\mathbf{A}, \mathbf{C})) = rank(\mathcal{O})$, thus implying that the rank of $\mathcal{H} = \mathcal{O}\mathcal{C}$ can not exceed n. At the same time, this implies that the rank of the Hankel operator equals the McMillan degree of the system, i.e. it determines the order of the minimal realizations! Moreover, since the Markov parameters of an LTI system can be estimated from input-output experiments, this fact can be used to actually 'identify' a (minimal) state-space representation (***A***, ***B***, ***C***, ***D***) of the LTI system under consideration from the Hankel matrix of the i/o data, its possible factorization(s), and it 'shifted' versions. For an example of the use of these so-called *subspace identification* algorithms for

identification of single-phase reservoirs, the reader is referred to [183]. Theoretical background and algorithmic aspects of the different subspace identification methods are provided in detail in [34, 64, 147, 162, 188, 270, 276–278, 285], for instance.

Norms, Singular Values, and Quality of Approximations

Let Σ denote the LTI dynamic system under consideration, and let n be its McMillan degree. The aim of MOR techniques that concentrate on approximation of a system's input-output behavior is generally to determine a dynamical system Σ_{red} of a degree $k < n$, such that the error of approximation **e** (see Fig. 4.2, in which the systems are represented by their transfer functions) is 'small' over all inputs. What is "small" is usually determined by an appropriate norm. For example, a natural choice would be to minimize $\|\Sigma - \Sigma_{red}\|_\infty$, where the $\|\Sigma\|_\infty$ denotes the H_∞ norm of an LTI system Σ with transfer function $\mathbf{H}$:

$$\|\mathbf{\Sigma}\|_{2\cdot induced} \triangleq \sup_{\mathbf{u}\in\mathbb{L}_2} \frac{\|\mathbf{y}(t)\|_2}{\|\mathbf{u}(t)\|_2} = \sup_{\omega} \sigma_1(\mathbf{H}(j\omega)) = \|\mathbf{\Sigma}\|_\infty, \tag{4.1.26}$$

where $\mathbb{L}_2$ is the space of square-summable functions, with 2-norm (or 'energy') $\|\mathbf{z}(t)\|_2 = \sqrt{\int_\infty^\infty \|\mathbf{z}(t)\|_2^2 dt}$, and $\sigma_1(\mathbf{H}(j\omega)) = \sigma_{max}(\mathbf{H}(j\omega))$ is the largest singular value of $\mathbf{H}(j\omega)$.

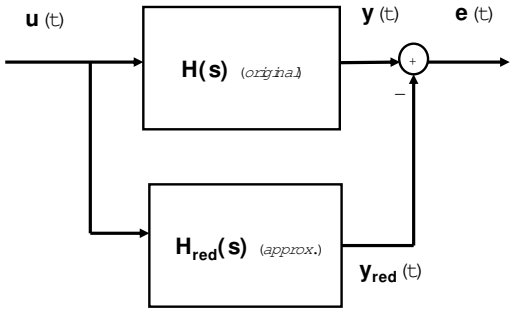


Figure 4.2: Systems comparison block scheme.

Unfortunately, there is no solution currently known for the problem of minimizing $\|\Sigma - \Sigma_{red}\|_\infty$. The big question then is "is there maybe another system norm, which does allow a solution?". The answer is affirmative and the norm in question is the *Hankel norm*, which is defined as (with $\mathbf{u} \in \mathbb{L}_2(-\infty, 0)$):

$$\|\Sigma\|_{\mathcal{H}} \triangleq \sigma_1(\mathcal{H}) = \|\mathcal{H}\|_{2\cdot induced}, \tag{4.1.27}$$

i.e., the Hankel norm of an LTI system is defined as the the maximum singular

value of its Hankel operator[8], $\sigma_1(\mathcal{H}) = \sigma_{max}(\mathcal{H})$. It can be shown that: a) the Hankel norm lower bounds the H_∞ norm, and b) twice the sum of the Hankel values upper bounds the H_∞ norm, i.e.

$$\sigma_1(\mathcal{H}) \leq \|\Sigma\|_\infty \leq 2\Sigma_{i=1}^{n}\sigma_i(\mathcal{H}). \tag{4.1.28}$$

As for the problem of approximating the system Σ by a k-th order one, Σ_{red}, using (4.1.27), (4.1.28), and the Schmidt-Mirsky inequality for matrices (linear operators): $\|\mathbf{X} - \mathbf{X}_k\|_{2\cdot induced} = \sigma_1(\mathbf{X} - \mathbf{X}_k) \geq \sigma_{k+1}(\mathbf{X})$, where k is the rank of $\mathbf{X}_k$ and $\sigma_i(\mathbf{G})$ is the i-the largest singular value of $\mathbf{G}$ in the singular value decomposition (SVD) $\mathbf{G} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, the following bounds on the Hankel norm system approximation are obtained:

$$\sigma_{k+1}(\mathcal{H}) \leq \|\Sigma - \Sigma_{red}\|_{\mathcal{H}} \leq \|\Sigma - \Sigma_{red}\|_\infty. \tag{4.1.29}$$

And here comes the important result: it can be shown that *there actually exists a lower-order system that achieves the lower bound on the Hankel norm in* (4.1.29), *i.e. for which* $\sigma_{k+1}(\mathcal{H}) = \|\Sigma - \Sigma_{red}\|_{\mathcal{H}}$! Even though the Hankel norm does not have a very natural system theoretic interpretation, this result is really remarkable. It was proven for Hankel operators by *the AAK theorem* (AAK: Adamjan, Arov and Krein) in [2]. In the prize winning paper [107], the result was extended to state-space systems. In the same paper, an entire algorithm was provided to obtain a state space representation $(\mathbf{A}_k, \mathbf{B}_k, \mathbf{C}_k, \mathbf{D}_k)$ of the optimal Hankel-norm k-th order approximant of a stable system given in state space form.

At first glance something may look strange here. Isn't it true that the minimal norm problem as the one above always has a solution, actually many numerically equivalent ones, being given by a singular value decomposition, truncated after the first k largest singular values? Of course that's true. However, a numerically optimal approximant of a Hankel matrix is in general not a Hankel matrix itself. But it needs to be, in order to be representing an LTI system. The remarkable aspect in the result above, therefore, is that a Hankel approximant can be found that approximates the system equally well.

Remark 4.1.2. Note that the 'feedthrough' (i.e. direct i/o) matrix $\mathbf{D}$ does not have any effect on the Hankel norm (an extra proof that Hankel optimal reduced models are not unique), therefore it can be modified arbitrarily. This is important as $\mathbf{D}$ does affect the H_∞ norm of the error, implying that it can be used for obtaining Hankel optimal reduced models with improved H_∞ norm error [124, 282]. As noted later in the chapter, reduced-order models obtained by either the balanced truncation technique (Section 4.4) or the optimal Hankel-norm approximation satisfy the following error bound [107]: $\|\mathbf{H}(j\omega) - \mathbf{H}_{red}(j\omega)\|_\infty \leq 2\sum_{i=k+1}^{n}$ for any order k of the reduced model. [107] shows how to calculate a $\widetilde{\mathbf{D}}$ in such a way that upper-bound on the H_∞ norm of the error is cut by half, i.e. $\|\mathbf{H}(j\omega) - \mathbf{H}_{red}(j\omega) - \widetilde{\mathbf{D}}\|_\infty \leq \sum_{i=k+1}^{n}$.

[8]It can be shown that the Hankel singular values equal the square roots of eigenvalues of the product $\mathbf{W}_c\mathbf{W}_o$, where $\mathbf{W}_c$ and $\mathbf{W}_o$ are the so-called *controllability Gramian* and *observability Gramian*, two other important system elements (see Sec. 4.4).

4.2 MOR by Modal Decomposition and Singular Perturbations

4.2.1 MOR by Eigenmodes Truncation

One of the earliest MOR approaches for linear systems was to base the reduction on the system's eigenvalues and to retain only modes corresponding to a restricted number of the eigenvalues. This technique is still of fundamental importance for many applications, especially in structural analysis. The reasoning behind this idea is easiest to explain if we consider the response of the (decoupled) modes to a constant input (here for a zero-initial state and a SISO system), $u(t) = \alpha$:

$$z_i(t) \quad = \quad \int_0^t e^{\lambda_i(t-\tau)}\tilde{b}_i u(\tau)d\tau = \frac{1}{\lambda_i}(\tilde{b}_i\alpha)(1-e^{\lambda_i t}). \tag{4.2.1}$$

For 'comparable'[9] $\tilde{b}_i$'s, modes corresponding to small $|\lambda_i|$ (called *slow modes*) will 'hang around' in the system response longer than those corresponding to large $|\lambda_i|$ (*fast modes*). The hope is then that, in this case, the number of fast decaying modes is considerable, so that the reduced-order model obtained by truncating these fast modes from the state-space would be low. By 'mode truncation' we mean the following. Let the original (stable, continuous-time, n-dimensional) state-space system under consideration in coordinates $\mathbf{x}$ have been transformed, by a suitable similarity transformation, into a new system in coordinates $\mathbf{z}$, already partitioned into a 'dominant' (here slow), $\mathbf{z}^1 \in \mathbb{R}^k$, and a 'non-dominant' (here fast), $\mathbf{z}^2 \in \mathbb{R}^{n-k}$, part:

$$\begin{aligned} \dot{\mathbf{z}} \equiv \begin{bmatrix} \dot{\mathbf{z}}^1 \\ \dot{\mathbf{z}}^2 \end{bmatrix} \quad &= \quad \begin{bmatrix} \widetilde{\mathbf{A}}_{11} & \widetilde{\mathbf{A}}_{12} \\ \widetilde{\mathbf{A}}_{21} & \widetilde{\mathbf{A}}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{z}^1 \\ \mathbf{z}^2 \end{bmatrix} + \begin{bmatrix} \widetilde{\mathbf{B}}_1 \\ \widetilde{\mathbf{B}}_2 \end{bmatrix} \mathbf{u}, \\ \mathbf{y} \quad &= \quad \begin{bmatrix} \widetilde{\mathbf{C}}_1 & \widetilde{\mathbf{C}}_2 \end{bmatrix} \begin{bmatrix} \mathbf{z}^1 \\ \mathbf{z}^2 \end{bmatrix} + \widetilde{\mathbf{D}}\mathbf{u}. \end{aligned} \tag{4.2.2}$$

Discarding the non-dominant modes yields the reduced or 'truncated' order model $\dot{\mathbf{z}}_{red} = \widetilde{\mathbf{A}}_{red}\mathbf{z}_{red} + \widetilde{\mathbf{B}}_{red}\mathbf{z}_{red}$, $\hat{\mathbf{y}}_{red} = \mathbf{C}_{red}\mathbf{z}_{red} + \mathbf{D}_{red}\mathbf{u}$, with $\widetilde{\mathbf{A}}_{red} = \widetilde{\mathbf{A}}_{11}$, $\widetilde{\mathbf{B}}_{red} = \widetilde{\mathbf{B}}_1$, $\widetilde{\mathbf{C}}_{red} = \widetilde{\mathbf{C}}_1$, $\widetilde{\mathbf{D}}_{red} = \widetilde{\mathbf{D}}$, and $\mathbf{z}_{red}(0) = \mathbf{F}_k\mathbf{z}(0)$ for some $\mathbf{F}_k \in \mathbb{R}^{k\times n}$. For example, if the performed similarity transformation was the eigendecomposition of the original system matrix into $\mathbf{E}\mathbf{\Lambda}\mathbf{E}^{-1} = [\mathbf{E}_k, \mathbf{E}_{n-k}]\ diag(\mathbf{\Lambda}_k, \mathbf{\Lambda}_{n-k})[\mathbf{V}_k, \mathbf{V}_{n-k}]^T$, with $\mathbf{V} = [\mathbf{V}_k, \mathbf{V}_{n-k}] := \mathbf{E}^{-T}$, then the matrices involved above would be $\widetilde{\mathbf{A}}_{11} = \mathbf{\Lambda}_k$, $\widetilde{\mathbf{A}}_{22} = \mathbf{\Lambda}_{n-k}$, $\widetilde{\mathbf{A}}_{12} = \mathbf{0}_{k\times(n-k)}$, $\widetilde{\mathbf{A}}_{21} = \widetilde{\mathbf{A}}_{12}^T$.

Complete truncation of some modes of a fully controllable and observable system will inevitably cause errors in the response behavior. For example, truncating the fast modes will cause the resulting reduced model to be more accurate at low- than at high frequency: as a result, the time response of the reduced model may

[9] $\tilde{c}_i\tilde{b}_i$ for the mode contribution in the output $\rightarrow$ a reason for a proper scaling of the model.

be too slow, i.e. the initial dynamics of the system might be lost. Similarly, maintaining only fast modes will have larger errors at low-frequency as a consequence. Whichever modes in a minimal representation are truncated, there will always be a truncation error at any frequency. Figure 4.3 provides an illustration of this for a 21×21 single-phase example with one water injection well and one producer placed on the diagonally opposite sides of the reservoir. Reservoir and fluid properties and well- and control data are given in Tables 4.1 and 4.2. The injection rate was varied during the simulation, so the flow-rate value in Table 4.2 concerns the initial injection rate (and, at the same time, its upper limit). The producer bottom-hole pressure was kept constant and the Peaceman-type well model was used to relate this pressure to the produced flow-rate. Due to the non-zero compressibility, the production rate does not reach a steady-state at any of the piecewise constant injection intervals. The bottom-right subfigure shows the cumulative production difference between the full-order model (441 states) and the reduced-order models of several orders ranging from 50 up to 292 obtained by modal truncation of high-frequency modes. Truncated models with up to 291 states all clearly mismatch with the full-order one for all times, which, in the case of produced flow rate as an output of interest, is something one generally prefers to avoid. Truncated models starting from order 292 do approximate the full-order behavior very well. It is noted that the truncated models overestimate the production for all times.

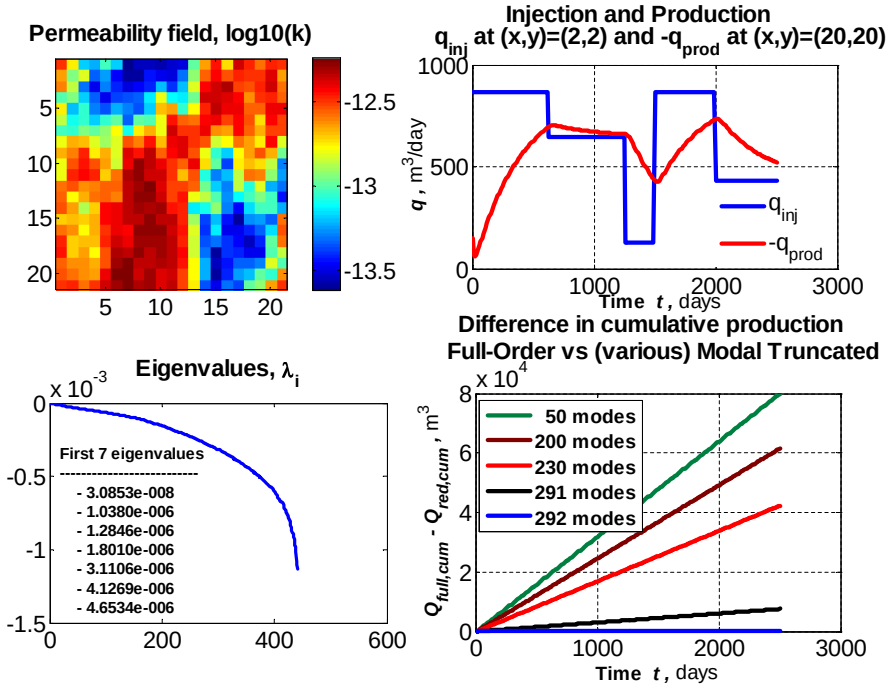


Figure 4.3: Example Single-Phase Flow MOR for a 21x21 horizontal reservoir using various orders of the reduced-order model obtained by modal truncation of the high-frequency modes.

Table 4.1: Reservoir properties and static parameters

$L_x \times L_y \times L_z$	N_x, N_y, N_z (#gridblocks)	ϕ [-]	c_ϕ, c_f [1/Pa]	μ [Pa.s]
$700[m] \times 700[m] \times 2[m]$	21,21,1	0.3	1e-8,1e-8	5e-4

Table 4.2: Well- and control data (no-flow boundaries)

$Q_{at(x,y,z)=(2,2,1)}$ $[m^3/s]$	$P_{bh,at(x,y,z)=(20,20,1)}$ [Pa]	P_{init} [Pa]	d_w (well diameter) [m]
0.01	2.7e7	3e7	0.1143

Remark 4.2.1. Starting from a full-order sparse model of order 441 and ending with a 'well-performing' dense reduced-order model with 292 states is certainly not what one aims at. The example was meant to illustrate that, using modal truncation, a) it may very well happen that the truncation error can be sufficiently decreased only with undesirably high-dimensional reduced-order models, and b) the difference in the truncation errors between the successive model orders can be substantial (e.g. between orders 291 and 292 in the example), but at the same time also quite steady for a large number of orders. As the method is based purely on the systems eigendecomposition, it may be expected that these aspects are even more present when the number of inputs and/or outputs increase. The practice learns that this indeed is the case.

As discussed next, determining in advance which order would be sufficient is not trivial for modal truncation MOR. On the other side, also discussed next, modal truncation could serve as an initial reduction step for any other projection technique, so also for those that do provide some prior error bounds but which are computationally too expensive to apply directly to the original high-dimensional model. Regarding the above example, the accurate reduced model of order 292 could be used as the 'high-dimensional' model for the so-called 'balanced truncation (BTR)', for instance, which method is the subject of Section 4.4. The results of applying this combined MOR are shown in Figure 4.4, where the order of the reduced model obtained by the BTR step is just 15.

Quality of approximation

Assuming the system to be represented in the decomposed form, the fractional expansion representation of the transfer functions for the original model and a k-dimensional reduced model yields $\mathbf{H}(s) - \mathbf{H}_{red}(s) = \sum_{i=k+1}^{n} \frac{\tilde{\mathbf{c}}_i \tilde{\mathbf{b}}_i^T}{s-\lambda_i}$, where $\tilde{\mathbf{c}}_i$ and $\tilde{\mathbf{b}}_i$ are columns of $\widetilde{\mathbf{C}}_2$ and $\widetilde{\mathbf{B}}_2$, respectively, in (4.2.2). For stable systems an estimate for $\|\mathbf{H}(s) - \mathbf{H}_{red}(s)\|_\infty$ is then given by:

$$\begin{aligned} \|\mathbf{H} - \mathbf{H}_{red}\|_\infty &\equiv \sup_{\omega\in\mathbb{R}} \|\mathbf{H}(j\omega) - \mathbf{H}_{red}(j\omega)\| = \sup_{\omega\in\mathbb{R}} \| \sum_{i=k+1}^{n} \frac{\bar{\sigma}(\tilde{\mathbf{c}}_i \tilde{\mathbf{b}}_i^T)}{j\omega - \lambda_i} \| \\ &\leq \sum_{i=k+1}^{n} \frac{\bar{\sigma}(\tilde{\mathbf{c}}_i \tilde{\mathbf{b}}_i^T)}{|\mathrm{Re}(\lambda_i)|}. \end{aligned} \tag{4.2.3}$$

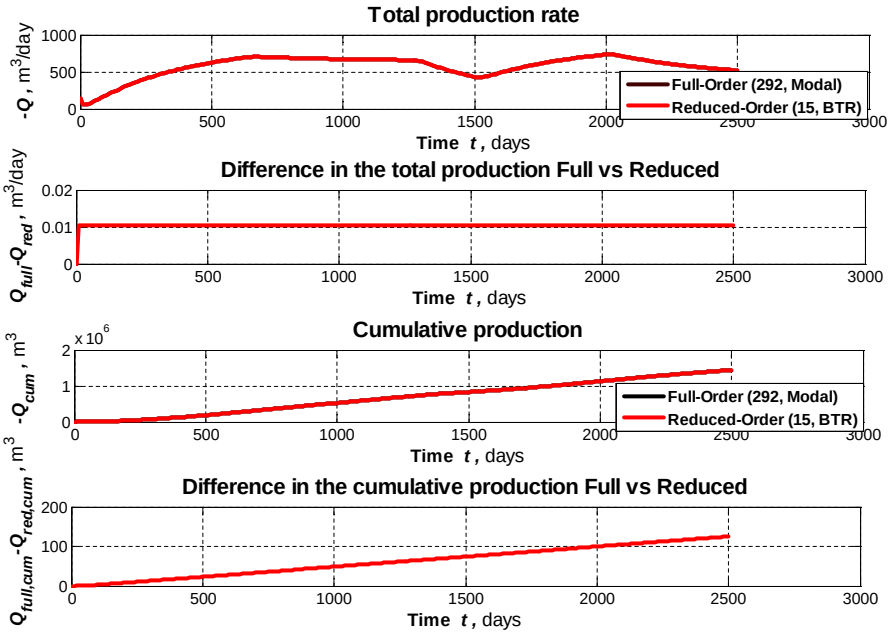


Figure 4.4: Results of combined Modal Truncation + Balanced Truncation (BTR) MOR to efficiently reduce the original 441 states high-order model from Figure 4.3. Modal Truncation was used here to obtain a highly accurate approximation (292 states) of the original high-dimensional model, which was then further reduced to 15 states by the BTR technique from Section 4.4. The BTR model matches the one with 292 states, and thus also the original model with 441 states, almost perfectly. Total simulation time: 2500 days, Timestep: 10 days, CPU-time: original model (441 states): 1.6 sec., reduced-order (15 states): 0.015 sec.

(In our models all eigenvalues, i.e. the poles, are real, so $\mathrm{Re}(\lambda_i) = \lambda_i$).

This error estimate is not an *a priori* error bound, though, as the actual error can be computed only after a choice has been made, that is, after the approximant has been determined. The lack of a reliable, general purpose method for modal dominance analysis w.r.t. the input-output behavior of the system is is a major limitation of the modal approach, despite enhancements provided e.g. in [280, 281]. This should not surprise as the modal truncation is solely based on (spectral) information contained in the system matrix $\mathbf{A}$. For a fixed input and/or output distribution (e.g., spatial distribution of the wells) this, of course, may be a serious disadvantage. Some of the truncated modes might, for instance, actually have been responsible for most of the output energy. In such cases the response will mostly likely be inaccurate.[10] However, due to its decomposed state-space form, it is relatively straightforward to combine the modal approach with tech-

[10]On the other side, just due to its input-output independence, this approach may actually appear

niques which do provide a priori error bound to select the appropriate order for an acceptable approximation error (BTR, for example, uses Hankel singular values (*HSVs*) of the system). The transfer function of a system in the modal form can be additively decomposed as $\mathbf{H} = \mathbf{H}_1 + \mathbf{H}_2$, where $\mathbf{H}_1$ and $\mathbf{H}_2$ correspond to two distinct subsets of eigenvalues (e.g. $\mathbf{H}_1(s) = \widetilde{\mathbf{C}}_1(s\mathbf{I}_k - \widetilde{\mathbf{A}}_{11})^{-1}\widetilde{\mathbf{B}}_1 + \widetilde{\mathbf{D}}$ and $\mathbf{H}_2(s) = \widetilde{\mathbf{C}}_2(s\mathbf{I}_{n-k} - \widetilde{\mathbf{A}}_{22})^{-1}\widetilde{\mathbf{B}}_2$ in (4.2.2), with $\widetilde{\mathbf{A}}_{12} = \mathbf{0}_{k\times(n-k)}$, $\widetilde{\mathbf{A}}_{21} = \widetilde{\mathbf{A}}_{12}^T$). The reduction can now be performed separately on $\mathbf{H}_1$ and $\mathbf{H}_2$. Let $\mathbf{H}_r = \mathbf{H}_{1r} + \mathbf{H}_{2r}$ be the resulting reduced model, where $\mathbf{H}_{1r}$ and $\mathbf{H}_{2r}$ are the resulting reduced subsystems computed say with the BTR method. If for the separate reduction of terms we have that $\|\mathbf{H}_i - \mathbf{H}_{ir}\| \leq \epsilon_i$, $i = 1, 2$, then $\|\mathbf{H} - \mathbf{H}_r\| \leq \epsilon_1 + \epsilon_2$. By reducing individually the terms, one can thus also control the resulting global error by choosing appropriate orders for the reduced subsystems. The technique can be readily extended to additive decompositions with more than two terms.

The real advantage of such a combined MOR approach is evident when one has to reduce very large order (sparse) models. For such models, the modal approach may be the only method which can be used for order reduction, despite the fact that even an eigen-decomposition on the full system is required (though the sparsity of the matrix involved makes the problem easier). Modal reduction is therefore often only a preliminary reduction which makes feasible further reductions using more powerful methods (in particular true when the number of inputs and/or outputs is considerable; otherwise the moment matching techniques like e.g. Krylov subspace ones from Sec. 4.3 can be even more efficient in the initial stage). Reducing a model solely by removal of unimportant system modes is rarely a good idea, especially for systems possessing clustered eigenvalues[11], as is clear from the following simple example where both modes $a - \epsilon$ and $a + \epsilon$ ($a < 0$, $|\epsilon| \ll |a|$) of the LTI system with transfer function

$$H(s) = \frac{a}{s + a + \epsilon} + \frac{a}{s + a - \epsilon} \tag{4.2.4}$$

are equally important, so none can be removed without introducing a large model reduction error, despite the fact that a very good reduced model $H_r(s)$ is given by

$$H_r(s) = \frac{2a}{s + a}. \tag{4.2.5}$$

The method of balanced truncation (BTR) of Sec. 4.4, for example, does not fail in recognizing the existence of such an one-dimensional approximator.

Finally, regarding the robustness of the basic modal approach, eigenvalues of a general matrix may be extremely sensitive to matrix perturbations (i.e., small parameter variations). In contrast, singular values of a matrix are much less sensitive

to be more robust than other methods when spatial distribution and/or the number of inputs and/or outputs do change, at least as long as the system matrix itself does not change (in our models this is the case with flow-constrained wells, as redistributing such wells or changing their numbers leaves the system matrix intact).

[11] Two other general cases where the knowledge which modes should be retained is not always clear are lightly damped high frequency modes and wide band input excitations. These are not really an issue here, though (strong, non-oscillatory damping and (quasi-)piecewise constant inputs).

to parameter variations. Due to the fact that the singular values of a symmetric matrix are the absolute values of its eigenvalues, the single-phase state matrices in this thesis are quite robust in this respect, as they are either symmetric or are similar to symmetric ones.[12]

4.2.2 Singular Perturbation MOR

The essence of the (zero-order) singular perturbation approximation (SPA), also called 'residualization', is setting the time-derivative of the 'fast part' of the system dynamics to zero, and expressing the dynamics of the slow part accordingly [88, 159, 173]. Mathematically, let the (stable, continuous-time) state-space system under consideration be already partitioned by some means in a slow ($\mathbf{z}^1$) and a fast ($\mathbf{z}^2$) part (e.g., by modal decomposition or balanced realization) as in (4.2.2). Setting $\dot{\mathbf{z}}^2 = \mathbf{0}$ yields $\mathbf{z}^2 = -\widetilde{\mathbf{A}}_{22}^{-1}(\widetilde{\mathbf{A}}_{21}\mathbf{z}_1 + \widetilde{\mathbf{B}}_2\mathbf{u})$, which, substituted in the expressions for the dynamics of $\mathbf{z}^1$ and the output $\mathbf{y}$, delivers the *singularly perturbed reduced order model*:[13]

$$\dot{\mathbf{z}}_{red} = \widetilde{\mathbf{A}}_{red}\mathbf{z}_{red} + \widetilde{\mathbf{B}}_{red}\mathbf{u}, \tag{4.2.6}$$

$$\hat{\mathbf{y}} = \widetilde{\mathbf{C}}_{red}\mathbf{z}_{red} + \widetilde{\mathbf{D}}_{red}\mathbf{u}, \tag{4.2.7}$$

where $\mathbf{z}_{red} := \mathbf{z}^1$, $\widetilde{\mathbf{A}}_{red} := \widetilde{\mathbf{A}}_{11} - \widetilde{\mathbf{A}}_{12}\widetilde{\mathbf{A}}_{22}^{-1}\widetilde{\mathbf{A}}_{21}$ (i.e. $\mathbf{A}_{red}$ is the *Schur complement* of $\mathbf{A}$), $\widetilde{\mathbf{B}}_{red} := \widetilde{\mathbf{B}}_1 - \widetilde{\mathbf{A}}_{12}\widetilde{\mathbf{A}}_{22}^{-1}\widetilde{\mathbf{B}}_2$, $\widetilde{\mathbf{C}}_{red} := \widetilde{\mathbf{C}}_1 - \widetilde{\mathbf{C}}_2\widetilde{\mathbf{A}}_{22}^{-1}\widetilde{\mathbf{A}}_{21}$ and $\widetilde{\mathbf{D}}_{red} := \widetilde{\mathbf{D}} - \widetilde{\mathbf{C}}_2\widetilde{\mathbf{A}}_{22}^{-1}\widetilde{\mathbf{B}}_2$. Since the original model is assumed stable, $\widetilde{\mathbf{A}}_{22}^{-1}$ exists.

A direct consequence of setting state-derivatives to zero is that the method retains the steady state gain of the system, as zero state-derivative obviously represents steady state. On the contrary, mode truncation retains the systems be-

[12] Consider the single-phase pressure equation in the generalized state-space form $\mathbf{V}\dot{\mathbf{p}} = -\mathbf{T}\mathbf{p} + \mathbf{q}$, with $\mathbf{T} = \mathbf{T}^T$ and $\mathbf{V}$ is diagonal with positive diagonal elements. Note that, in general, $\mathbf{V}^{-1}\mathbf{T}$ is not symmetric (it is if the grid is uniform). However, the state-transformation $\mathbf{p} = \mathbf{V}^{-\frac{1}{2}}\hat{\mathbf{p}}$ transforms the system into $\dot{\hat{\mathbf{p}}} = -\mathbf{V}^{-\frac{1}{2}}\mathbf{T}\mathbf{V}^{-\frac{1}{2}}\hat{\mathbf{p}} + \mathbf{V}^{-\frac{1}{2}}\mathbf{q}$, where $\mathbf{V}^{-\frac{1}{2}}\mathbf{T}\mathbf{V}^{-\frac{1}{2}}$ is now symmetric, with the same eigenvalues as $\mathbf{V}^{-1}\mathbf{T}$ and the eigenvectors scaled by a multiple of $\mathbf{V}^{\frac{1}{2}}$, i.e., if $\mathbf{z}_i$ and $\mathbf{v}_i$ are the eigenvectors of $\mathbf{V}^{-\frac{1}{2}}\mathbf{T}\mathbf{V}^{-\frac{1}{2}}$ and $\mathbf{V}^{-1}\mathbf{T}$, resp., corresponding to the same eigenvalue λ_i, then $\mathbf{z}_i = \alpha_i\mathbf{V}^{\frac{1}{2}}\mathbf{v}_i$, for some $\alpha_i \in \mathbb{R}$. This result may be important, e.g. when the grid is highly nonuniform, in which case $\mathbf{V}^{-1}\mathbf{T}$ is not symmetric anymore. Besides the advantage of allowing the use of 'symmetric' matrix solvers, the symmetry of the (new) state-matrix also reveals initially 'hidden' features regarding controllability and observability of single-phase problems. In particular, it makes it easier to uncover the equivalence of the two (i.e., controllability and observability) in the case of the inputs and outputs placed at same positions. After realizing this, it was then not too difficult to prove that in such cases the knowledge of one of the two properties, e.g. controllability, also in the non-uniform case was sufficient to distill the properties of the other one, i.e., observability (and vice versa). The only extra data needed was the knowledge of one single non-zero element in the input matrix $\mathbf{B}$ and its counterpart in the output matrix $\mathbf{C}$.

[13] It is noted that singular perturbation model reduction does NOT formally belong to the class of state-space projection MOR, the reason being that there is actually no connection between the states in the original and those in the reduced order model. On the other hand, since the singular perturbation approximation is related to state-space truncation by the frequency inversion transformation $s \rightarrow \frac{1}{s}$, it is frequently also itself considered to be a truncation method.

havior at infinite frequency. For this reason modal residualization, when feasible ($\widetilde{\mathbf{A}}_{22}^{-1}$ might be too expensive to compute explicitly if the system is very large and the required order of the reduced model is small), is preferred for low frequency modeling, whereas modal truncation is preferable when accuracy is required at high frequencies. While this reasoning is clearly supported by Fig. 4.5, it is emphasized, though, that reduced-order models obtained using the singular perturbation approach may not always yield good low frequency approximations, especially when there is no clear time scale separation into slow and fast subsystems. Singular perturbation is therefore much more suitable for reducing stiff systems, i.e. systems possessing distinct time-scales. However, in a simulation using step(wise) inputs the technique is expected to perform rather well in general, at least as far as the long term approximation is concerned (N.B. not necessarily in the cumulative production sense, as that one depends on the approximation along the whole simulation time). Figure 4.5 shows the comparison of this approach and the modal truncation for the same system as in Fig. 4.3 for a step input. Modal rezidualization clearly outperforms modal truncation, except in a short initial transient period related to the (total, i.e. rock and fluid) compressibility. For large compressibilities this would mean that truncation (especially balanced truncation, to be explained later) would often be a better choice than rezidualization. However, for large compressibilities, the assumption of the pressure dynamics as being (almost) linear would generally be incorrect (or at least questionable). Finally, we note that the superior performance of the singular perturbation approach for (very) low compressible systems and step(wise) inputs is all but surprising, as in the limit of zero compressibility this approach can be seen as a linear solver for $-\mathbf{T}\mathbf{p} = \mathbf{q}$ based on its Schur complement system.

4.3 Transfer Function Matching and Krylov Subspaces MOR

Transfer function moment matching MOR techniques transform the original system into a reduced one in such a way that the first (say) k moments of the reduced-order transfer function $\mathbf{H}_r(s)$ match the first k moments of the high-dimensional $\mathbf{H}(s)$. The moments are defined as the value and subsequent derivatives of the transfer function at any particular frequency point s_i. The section starts with with a short introduction to the concepts of transfer function point and moment matching, which solutions turn out to be intrinsically ill-conditioned matrix equation problems if attempted to be solved directly, that is, based on the explicit use of transfer function moments, unless the number of the moments can be kept low. As stability of the obtained reduced-order models is also not guaranteed, it is shown how to preserve it while at the same time providing a simple explicit expression for the dominant poles of the original, high-dimensional model using a subspace projection technique that uses explicit moments to define congruence transformation of the state-space. This particular method provides efficient MOR solution for the examples considered in this thesis due to sufficiently low numbers of moments required, at least when the number of inputs is not large. In

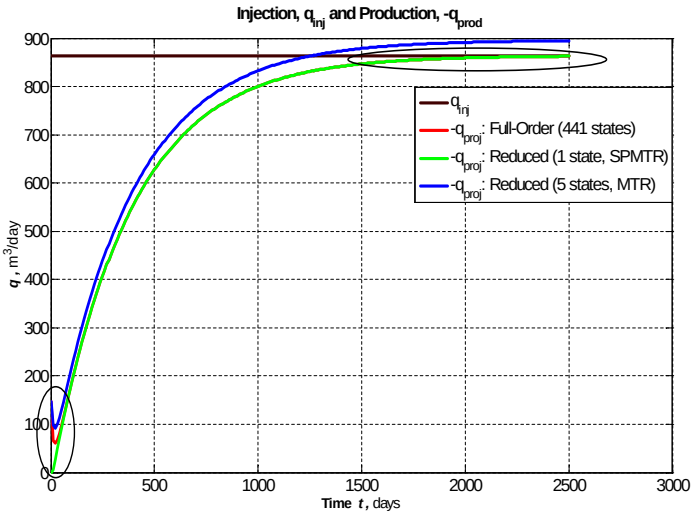


Figure 4.5: Comparison Modal Truncation (MTR) versus Singular Perturbation (SPMTR) for the slightly compressible system from Figure 4.3. Except in the initial stage, which length and thus impact is directly related to the total compressibility, the rezidualized model consisting of just 1 state clearly outperforms the truncated one with 5 modes (it actually outperforms modal truncated models of much higher orders, but the results of those would be hardly visible in this figure).

order to anticipate situations where larger numbers of moments would appear necessary (e.g. due to a particular choice of spatial-discretization, complex grids and parameter fields, large numbers of input-output pairs, etc.), the section also provides an introduction to the Krylov subspace (KS) MOR methodology, which has the remarkable characteristic that techniques employed perform transfer function *moments matching without actually computing the moments*! Instead, these methods achieve moment matching by iteratively constructing projection matrices that span adequate (generalized) Krylov subspaces. The employment of indirect moment matching techniques promotes more efficient and stable implementation of moment matching in the general case.

4.3.1 Transfer Function *Point Matching*

Let $\mathcal{S}_{2k} := \{s_j\}_{j=1}^{2k}$ be the set of $2k$ frequency points where we want the reduced-order transfer function $H_r(s)$ in (4.1.19) to match the high-order rational one, $H(s)$, defined in (4.1.18), i.e. $\forall s_i \in \mathcal{S}_{2k}$:

$$H_r(s_i) \quad = \quad \frac{e_{r,0} + e_{r,1}s_i + e_{r,2}s_i^2 + \ldots e_{r,k-1}s_i^{k-1}}{1 + f_{r,1}s_i + f_{r,2}s_i^2 + \cdots + f_{r,k}s_i^k} = H(s_i). \qquad (4.3.1)$$

Cross-multiplying and then rearranging the terms yields the following linear system of matrix equations for the coefficients $\{e_{r,i}\}_{i=0}^{k-1}, \{f_{r,i}\}_{i=1}^{k}$:

$$\begin{bmatrix} s_1 H(s_1) & s_1^2 H(s_1) & \dots & -s_1^{k-1} \\ s_2 H(s_2) & s_2^2 H(s_1) & \dots & -s_2^{k-1} \\ \vdots & \vdots & \vdots & \vdots \\ s_{2k} H(s_{2k}) & s_{2k}^2 H(s_{2k}) & \dots & -s_{2k}^{k-1} \end{bmatrix} \begin{bmatrix} e_1^r \\ e_2^r \\ \vdots \\ f_{k-1}^r \end{bmatrix} = \begin{bmatrix} H(s_1) \\ H(s_2) \\ \vdots \\ H(s_{2k}) \end{bmatrix}. \quad (4.3.2)$$

The following is important to emphasize. The columns of the 'point matching matrix' in (4.3.2) contain progressively higher powers of the 'test' frequencies. As for progressively increasing powers of s_i these columns become progressively (and very quickly) ill-conditioned, so does the matrix itself. The truth is then that, unless the order of the reduced model is kept rather low, one can not really do MOR this way without taking special care of the ill-conditioning.

4.3.2 *Explicit* Transfer Function *Moment Matching*

Let's now consider the 'moment matching' problem, i.e. matching, by a reduced-order model, the value of the transfer function of the high-dimensional model as well as a number of subsequent derivatives at a desired frequency point s. To arrive at the desired equations, let us write the (stable) transfer function of the SISO LTI system in a slightly different form:

$$\begin{aligned} H(s) = \mathbf{c}^T(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{b} &= -\mathbf{c}^T \underbrace{(\mathbf{I} - s\mathbf{A}^{-1})^{-1}}_{\text{Taylor expand w.r.t. } s\text{, for 'small' } s\mathbf{A}^{-1}} \mathbf{A}^{-1}\mathbf{b} \\ &= -\sum_{i=0}^{\infty} \mathbf{c}^T \mathbf{A}^{-(i+1)} \mathbf{b} s^i \\ &= \underbrace{-\mathbf{c}^T\mathbf{A}^{-1}\mathbf{b}}_{m_0} + \underbrace{(-\mathbf{c}^T\mathbf{A}^{-2}\mathbf{b})}_{m_1} \cdot s + \underbrace{(-\mathbf{c}^T\mathbf{A}^{-3}\mathbf{b})}_{m_2} \cdot s^2 + \dots \\ &= \sum_{i=0}^{\infty} m_i s^i. \end{aligned} \quad (4.3.3)$$

The coefficients $m_i = -\mathbf{c}^T\mathbf{A}^{-(i+1)}\mathbf{b}$, $i \geq 0$ are the *moments* we are interested in (when $d \neq 0$, then $m_0 = d - \mathbf{c}^T\mathbf{A}^{-1}\mathbf{b}$). For a steady-state match, i.e. $s = 0$, for instance, only m_0 needs to be matched[14]. The more moments are matched, the more of the transient behavior is caught. The above expressions for the moments are obtained for 'small' $s\mathbf{A}^{-1}$ and are equivalent to those obtained from the 'MacLaurin'

[14]Indeed, in the time domain we have, in a steady state, $\mathbf{0} = \mathbf{A}\mathbf{x}_{ss} + \mathbf{b}u_{ss}$, $y_{ss} = \mathbf{c}^T\mathbf{x} + du_{ss}$, which yields $y_{ss} = (d - \mathbf{c}^T\mathbf{A}^{-1}\mathbf{b})u_{ss}$.

series[15] of $H(s)$:

$$H(s) \quad = \quad H(0) + H^{'}(0)s + \frac{H^{''}(0)}{2!}s^2 + \frac{H^{'''}(0)}{3!}s^3 + \ldots. \tag{4.3.4}$$

H(s) can be expanded about any 'test' frequency s_0 for which $(s_0\mathbf{I} - \mathbf{A})$ is non-singular using the Taylor series expansion of $H(s)$, given by the infinite sum $H(s) = \sum_{i=0}^{\infty} \frac{H^{(i)}}{i!}(s_0)(s - s_0) = \sum_{i=0}^{\infty}(-1)^i \mathbf{c}^T(s_0\mathbf{I} - \mathbf{A})^{-(i+1)}\mathbf{b}(s - s_0)^i$. The expansion about $s_0 = 0$ provides the *low-frequency* moments as above, i.e. $m_i = -\mathbf{c}^T\mathbf{A}^{-i}\mathbf{b}$, $i \geq 0$, whereas expansion about $s_0 = \infty$ yields the *high-frequency* moments (so-called 'Markov parameters'), $m_i = -\mathbf{c}^T\mathbf{A}^{-(i+1)}\mathbf{b}$, $i < 0$.

Explicit (or direct) moment matching techniques, like e.g. Asymptotic Waveform Evaluation AWE [6, 62, 63, 131, 217, 260], approximate the response of the original system typically via a two-step process. First, moments m_i which correspond to frequency domain expansions of the original system transfer function are explicitly computed. In the second step, the approximate realization transfer function (4.1.19), recognized to be a *Padé* approximant of the original transfer function, is forced to correspond to $2k$ moments of the original system, whereby moments related to different expansion points (test frequencies) can be mixed, for instance j low-frequency moments and $2k - j$ high-frequency moments. This yields a system of equation as (4.3.5) for determining the denominator coefficients f_i of the reduced-order transfer function[16]:

$$-\begin{bmatrix} m_k \\ m_{k+1} \\ \vdots \\ m_{2k-1} \end{bmatrix} = \begin{bmatrix} m_0 & m_1 & \ldots & m_{k-1} \\ m_1 & m_2 & \ldots & \vdots \\ \vdots & \ldots & \ddots & m_{2k-3} \\ m_{k-1} & \ldots & m_{2k-3} & m_{2k-2} \end{bmatrix} \begin{bmatrix} f_k^r \\ f_{k-1}^r \\ \vdots \\ f_1^r \end{bmatrix}, \tag{4.3.5}$$

after which the coefficients e_i can be found by an additional matrix-vector multiplication:

$$\begin{aligned} e_0^r &= m_0 \\ e_1^r &= m_1 + f_1^r m_0 \\ &\ldots \\ e_{k-1}^r &= m_{k-1} + \sum_{i=1}^{k-1} f_i m_{k-(1+i)}. \end{aligned} \tag{4.3.6}$$

For MIMO systems with m inputs and p outputs AWE procedure is based on obtaining Pade approximations $H_{r,ij}(s)$ separately for each pair of inputs i and outputs j and then, by applying the superposition property of linear systems, group-

[15] A *MacLaurin* series is a Taylor series expansion of a function $f(x)$ about 0: $f(x) = f(0) + f^{(1)}(0)x + \frac{f^{(2)}(0)}{2!}x^2 + \frac{f^{(3)}(0)}{3!}x^3 + \ldots$.

[16] If the max. order of s, say ℓ, in the numerator of (4.1.19) is smaller than $k-1$, then the last equality in (4.3.5) reads $e_\ell^r = m_\ell \sum_{i=1}^{min(\ell,k)} f_i m_{\ell-i}$.

ing these approximations into one $p \times m$ transfer matrix function $(\mathbf{H}_r)_{ij}(s) = H_{r,ij}(s)$. The computation cost increases quadratically with the number of inputs/outputs ($O(p \times m)$).

The following is generally true about the moments:

- ***Recursive computation.*** The moments can be obtained recursively, from a series of matrix equations. For example, for a SISO LTI system with zero-initial conditions and in the generalized state-space form, the Laplace domain equations are $(s\mathbf{E} - \mathbf{A})\mathbf{X}(s) = \mathbf{b}U(s)$, $Y(s) = \mathbf{c}^T\mathbf{X}(s)$. Using $Y(s) = H(s)U(s) = \sum_{i=0}^{\infty} m_i s^i U(s)$, the state of the system can be expressed as $\mathbf{X}(s) = \sum_{i=0}^{\infty} \mathbf{r}_i s^i U(s)$ for some coefficients $\mathbf{r}_i$ (these are vectors!). Substituting this into the Laplace equation for the state yields $(s\mathbf{E} - \mathbf{A}) \sum_{i=0}^{\infty} \mathbf{r}_i s^i = \mathbf{b}$. the terms of the same power on both sides of the equation can be equated, yielding the following recursive formulae for the coefficients $\mathbf{r}_i$:

$$\begin{aligned} \mathbf{A}\mathbf{r}_0 &= -\mathbf{b}, \\ \mathbf{A}\mathbf{r}_i &= \mathbf{E}\mathbf{r}_{i-1}, \quad i = 1, \ldots, 2k. \end{aligned} \tag{4.3.7}$$

  If the dimension of the original system is moderate, the coefficients $\mathbf{r}_i$ (and hence also $m_i = \mathbf{c}^T\mathbf{r}_i$) can be determined by explicit matrix inversion. Otherwise, a single LU decomposition for all plus forward and backward substitutions for each of the equations suffices. For general sparse matrices, the complexity of an LU decomposition is $O(n^{1.4-1.7})$ instead of $O(n^3)$ for dense matrices. For single-phase problems in this thesis this complexity decreases even further, due to the simple structure of the system-, here a transmissibility, matrix $\mathbf{A}$ (symmetric, negative definite, and 5-diagonal for 2D problems and 7-diagonal for 3D problems).

- ***Ill-conditioning.*** For increasing i, the moments m_i line up more and more with the dominant eigenvector of $\mathbf{A}$, or equivalently $\mathbf{A}^{-1}$, in (4.3.3) (or $\mathbf{A}^{-1}\mathbf{E}$ in the generalized case). This is easy to see from $\mathbf{A}^{-1}\mathbf{b}$ and thinking of $\mathbf{b}$ as a weighted combination of the eigenvectors of $\mathbf{A}$, i.e.: $\mathbf{A}^{-1}\mathbf{b} = \mathbf{A}^{-1}(\alpha_1\mathbf{v}_1 + \cdots + \alpha_n\mathbf{v}_n) = \alpha_1\lambda^{-1}\mathbf{v}_1 + \cdots + \alpha_n\lambda_n^{-1}\mathbf{v}_n$. It follows that $\mathbf{A}^{-i}\mathbf{b} = \alpha_1\lambda_1^{-i}\mathbf{v}_1 + \cdots + \alpha_n\lambda_n^{-i}\mathbf{v}_n$, which approaches the term in the summation, say $\alpha_j\lambda_j^{-i}\mathbf{v}_j$, $j \in \{1, \ldots, n\}$, which dominates the sum (for instance, for a stable and symmetric $\mathbf{A}$, λ_j will be the least negative eigenvalue of $\mathbf{A}$). For increasing i this means that m_{i-1} tends to $\alpha_j\lambda_j^{-(i-1)}\mathbf{c}^T\mathbf{v}_j$ and m_i to $\alpha_j\lambda_j^{-i}\mathbf{c}^T\mathbf{v}_j$. Therefore, for 'sufficiently large' i, the subsequent moments behave according to $m_i \approx \lambda_j^{-1}m_{i-1}$. As a consequence, the (Hankel) moment matching matrix in (4.3.3) becomes rapidly ill-conditioned. In general, the 'sufficiently large' i is smaller the more separate the dominant eigenvalues are from the rest of the eigenspectrum. A discussion and an analysis of eigenspectra en moments characteristics of single-phase systems considered in this thesis is provided below.

- ***Relationship with poles (and residues) and loss of accuracy.*** Having determined the coefficients in the rational transfer function representation (4.1.19)

is not enough, as this model form can not be used for simulation. The partial expansion transfer function representation (4.1.17), on the other hand, is 'simulation-ready'[17], but it requires computation of the transfer function poles λ_i and their corresponding residues r_i. A trivial state-space representation is given by $(\mathbf{A}, \mathbf{b}, \mathbf{c}^T) = (diag(\lambda_1, \ldots, \lambda_k), [r_1, \ldots, r_k]^T, [1, \ldots, 1])$. The poles can be found by finding the roots of the polynomial expression that makes the denominator, $F(s)$, of $H_r(s)$. To compute the residues r_i there are two methods. The first one uses both numerator $E(s)$ and denominator $F(s)$ in (4.1.19). It can be shown (e.g., [260]) that the residues can be computed as $r_i = \frac{E(\lambda_i)}{\prod_{j=1, j\neq i}^{k}(\lambda_i - \lambda_j)}$, $i = 1, \ldots, k$. In the second method for computing residues r_i, the knowledge of numerator $E(s)$ is not required. Moreover, this method reveals an important relation between the moments, poles, and residues of the transfer function[18]:

$$\begin{aligned} -(\frac{r_1}{\lambda_1} + \frac{r_2}{\lambda_2} + \cdots + \frac{r_k}{\lambda_k}) &= m_0 \\ -(\frac{r_1}{\lambda_1^2} + \frac{r_2}{\lambda_2^2} + \cdots + \frac{r_k}{\lambda_k^2}) &= m_1 \\ &\vdots \\ -(\frac{r_1}{\lambda_1^{2k}} + \frac{r_2}{\lambda_2^{2k}} + \cdots + \frac{r_k}{\lambda_k^{2k}}) &= m_{2k-1}. \end{aligned} \tag{4.3.8}$$

(4.3.9)

(actually, as the number of unknown r_i is k, only k equations from (4.3.8) need to be selected to solve for r_i, $i = 1, \ldots, k$). The moment m_i is thus the power function of poles, $m_i = m_i(\lambda_1^{i-1}, \ldots, \lambda_k^{i-1})$! Consequently, the higher-order poles ('less dominant', having large magnitudes) will soon be lost numerically in the (expression for the) moments, implying that high order moments will approximate dominant (i.e. small-magnitude) poles. This practically means that explicit moment matching methods like AWE can not be reliably used to compute high-order poles and, unless the reduced model order is rather low (say, ≤ 4), unstable reduced-order models can easily be generated for stable original models when high order moments are computed. Fortunately, there exist (more) stable methods to determine dominant system poles. One of these, based on subspace-projection, is described below and will, to a certain extent, already introduce us to the idea and the

[17] Actually, when the poles and residues of a transfer function of a LTI system are known and one is interested in the output behavior only, then, for ideal input waveforms as e.g. steps and limited ramp signals, there is no simulation necessary at all, as the output can be written in a closed-form. For instance, the output response of a SISO system to unit step input $u(t)$, the Laplace of which being $\frac{1}{s}$, is easily found to be given by $y(t) = [\sum_{i=0}^{k} \frac{r_i}{\lambda_i}(e^{\lambda_i t} - 1)]u(t)$. The ability of determining the response of a dynamic system for any t using a closed-form expression is a major advantage over the transient analysis, which performance, stability and accuracy generally depend on the employed time-integration method and/or time-step sizes.

[18] The relation is easy to prove by writing the fractional expansion of $H(s)$ as $H(s) = \sum_{i=1}^{k} \frac{r_i}{s-\lambda_i} = \sum_{i=1}^{k} -\frac{r_i}{\lambda_i}(1 + \frac{s}{\lambda_i} + \frac{s^2}{\lambda_i^2} + \ldots)$, and comparing its first $2k$ terms with the corresponding terms in the moment expansion form $H(s) = m_0 + m_1 s + m_2 s^2 + \cdots + m_{2k-1} s^{2k-1}$.

setting of Krylov-subspace MOR, the subject of subsection 4.3.3. For some other methods, see e.g. [260].

- ***Shifted and multiple moment matching.*** The standard moment matching described above is w.r.t. to one single expansion point s. In the above it was $s = 0$ (as was also in the original AWE paper [216]). Approximation of the frequency response of the original system is frequently rather poor away from the expansion frequency. As a result, using expansions exclusively about $s = 0$ may produce large time domain errors near the initial time point ($t = 0$). A remedy for this was originally to use the first coefficient of the expansion about $s = \infty$ in addition to the moment information. Further improvement of the approximation near $t = 0$ can be achieved through employment of more moments in the expansion about $s = \infty$. The first attempts at expanding the system' transfer function at points other than $s = 0$ and $s = \infty$ were made in [131]. If the chosen 'shifted' frequency expansion point $s = h$ is along the positive real axis, the basic procedure for obtaining the 'shifted' moments is modified only slightly. The condition consists in considering $((s - h)\mathbf{E} + h\mathbf{E} - \mathbf{A})\mathbf{X}(s) = \mathbf{b}U(s)$ and expanding $\mathbf{X}(s)$ as a series in $s - h$. A similar recursion scheme as above between the shifted moments can be developed, with the only difference that now the matrix $h\mathbf{E} - A$ must be LU-factored, which may be advantageous when A is singular. On the other side, the sparsity, and hence the computational efficiency of the LU factorization, of $h\mathbf{E} - A$ is often reduced as compared with A, as $h\mathbf{E}$ may have zero entries there where A does not (this is not the case in this thesis). Finally, various techniques have also been developed for 'merged' AWE approximations, that is, approximations which contain information from multiple shift frequencies. Except the accuracy, employing information from expansions about multiple frequency points improves also stability of AWE approximations. A general technique for multipoint AWE moment matching is described in [6], while [63] proposes a different approach (see also [62]). For single-phase porous media equations with low-frequent input signal forms and slight compressibility it is reasonably to expect that the expansion around $s = 0$ will suffice in most of the cases (at the end, in the limit of zero compressibility, the single-phase pressure equation becomes a steady-state equation). Moreover, in these situations the number of moments required to adequately approximate the input-output transfer will, at least for systems with few or clustered wells, generally not be large.

Eigenspectra and moments of single-phase systems

Figure 4.6 depicts (a part of) eigenspectra for two reservoir systems with various well placements configurations. Figures 4.7 and 4.8 show (some of) the columns of the corresponding state block moments, whereas Figure 4.9 shows the behavior of m_i/m_{i+1} versus the dominant eigenvalue of $\mathbf{A}$ for some of the elements of the system block moments, with $m_i(k, \ell) = \mathbf{c}_k^T \mathbf{A}^{-i} \mathbf{b}_\ell$. One of the systems (parts a and b) is exactly the same as in Fig. 4.3, whereas the other system (parts c and d) is a 45x45 horizontal reservoir with two high permeable streaks and injecting

and producing wells placed on the opposite sides. In the examples injectors were always (each individually) flow-rate constrained and producers were always (also each individually) BHP constrained.

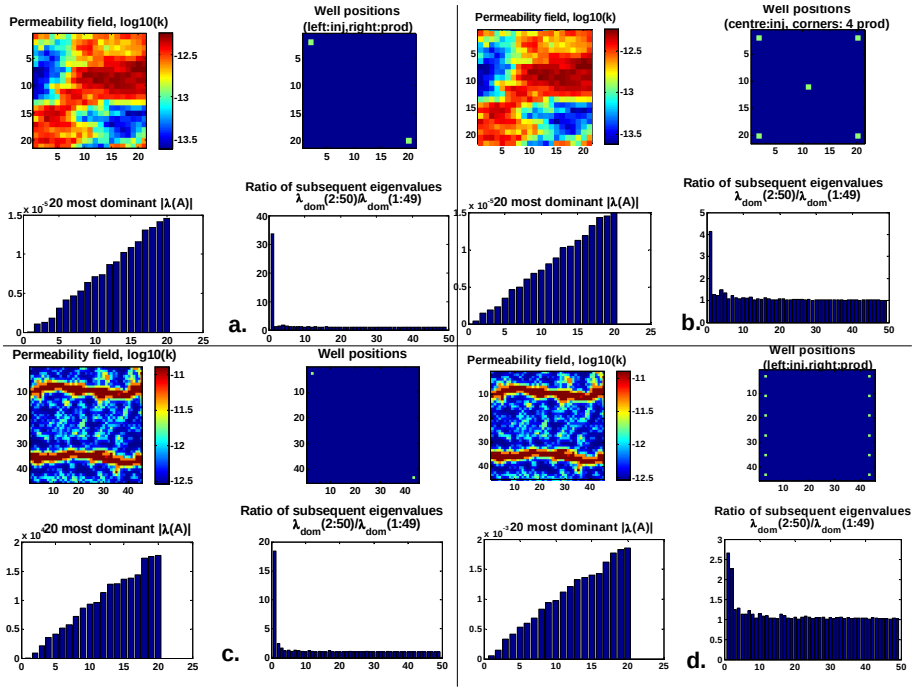


Figure 4.6: (Dominant part of) Eigenspectra of two single-phase systems, each in two different well configurations.

As for the eigenspectra, a general observation (for all models considered in this thesis) is that: a) for systems with only one BHP constrained well (connection) the separation of eigenvalues is clear, with a single close-to-zero eigenvalue dominating over the remaining, more stable part of the eigenspectrum. The separation is larger the lower the permeability of the gridblock the well is connected to[19]; b) the number of dominant eigenvalues increases with the number of wells, but it is generally still very low compared to the state-space dimension. It is important to note that flow-rate constrained wells do not introduce any new dynamics in the above sense, i.e., no change in $\mathbf{A}$ (and thus in the spectrum) is caused by the presence of such wells. However, as any new well implies a new non-zero column in the input matrix $\mathbf{B}$, controllability properties of the system do change (as do also observability properties, due to the symmetry). This change is less substantial the

[19]This is easy to understand when one recalls that without any BHP constrained well the transmissibility matrix $\mathbf{T}$ would have a single dominant zero eigenvalue and that each BHP constrained input introduces a 'Peaceman' term -α on the corresponding diagonal element of $\mathbf{T}$ (and thus $\mathbf{A}$), with α positive and depending on grid-block permeability and some geometric properties of the well. The presence of -α causes the zero eigenvalue to become negative.

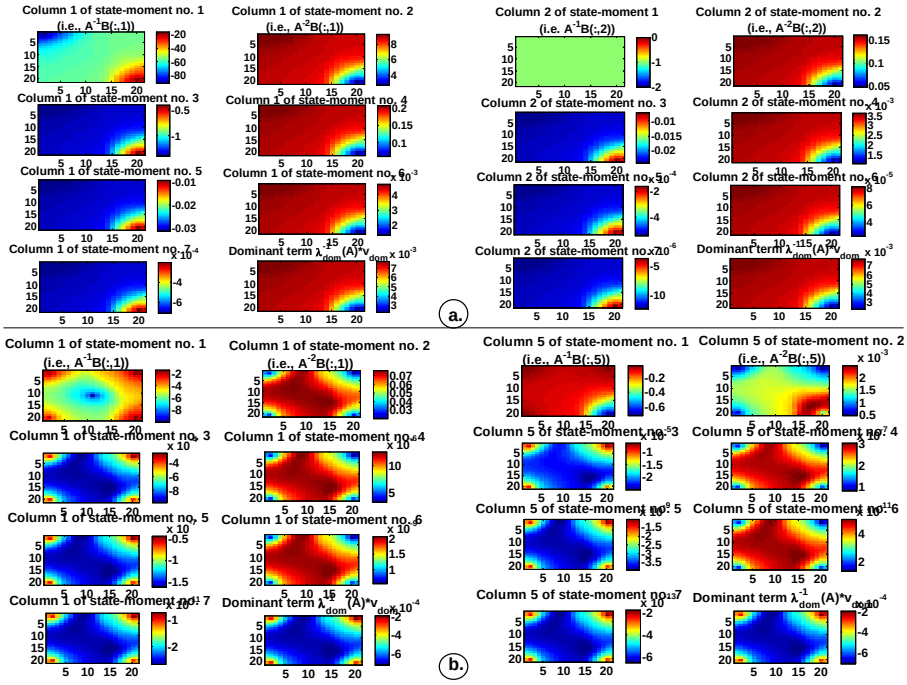


Figure 4.7: (Some of the columns of the) State Block Moments belonging to systems a. and b. from Fig. 4.6.

'more large-scale' the original problem is. This has very important consequences for applications like optimization and parameter- or state estimation. A crucial lesson one may (and should) draw is e.g. that it is totaly unrealistic to expect that without prior knowledge (e.g. constraints imposed by 'known' geological properties) or use of another types of information (e.g. seismic), one may be able to obtain fine-scale parameter estimates (of permeability, for instance) that resemble the 'underlying, real' parameter field. With such a sparse information available, there are infinitely many possible realizations of the parameter field that fit the observed data equally well. As for reservoir models, this, more or less expected conclusion, is in line with [259]. To borrow the term used in [256, 273, 274, 303], parameterizations of reservoir models obtained from spatial discretizations are not (globally) *structurally identifiable*. This conclusion is also supported by the behavior of the moments. As Figures 4.7-4.8 show, every column of the subsequent state block moments starts to line-up with the dominant eigenvector of $\mathbf{A}$ very quickly, generally already at the third or fourth moment. Consequently, as shown in Fig. 4.9, (the elements of) the system block moments rapidly line-up with their predecessors[20]. This implies that a very limited amount of information about the

[20] A remark: for configurations a. and c., the vector of the first state block moment corresponding to the BHP-constrained input equals $-[1, 1, \ldots, 1]^T$. This is generally true for all systems having a single

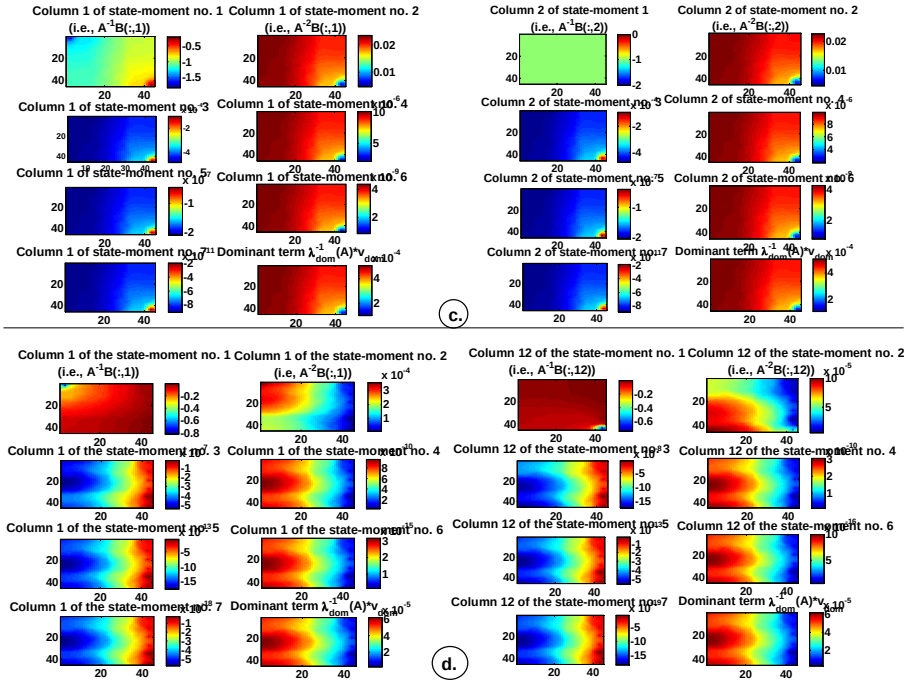


Figure 4.8: (Some of the columns of the) State Block Moments belonging to systems c. and d. from Fig. 4.6.

whole system is numerically obtainable from the moments (which together form the transfer function matrix of the system). Orthogonalization of the subspace spanned by the moments may help a bit, but the impact will generally be minimal for (very) large-scale systems. A larger number of wells (i.e. input-output pairs) may also make some difference, but the impact will be restricted to regions close to the wells if a) the wells are situated very close to each other (and thus largely correlated), as in Fig. 4.6.d (with the behavior of moments as depicted in 4.8.d and 4.9.d), and b) either the physical distances between individual wells or the number of gridblocks in the regions of interest between the wells are large (the information 'diffuses', so to speak). This is in line with [273, 274], which perform 'local structural identifiability' analysis based on observability, sensitivity and controllability properties of the system and the rank test of the so-called *(Fisher) information matrix* in order to determine 'a best identifiable, reduced-dimensional parameterization' for MIMO systems. The analysis makes use of system's Markov parameters, i.e., high-frequency moments. An excellent review and unification of linear identifiability concepts is provided in [197].

Remark 4.3.1. It is important to stress that, in general and if only (global) structural identifiability properties of a linear system are of interest, controllability and

BHP-constrained well.

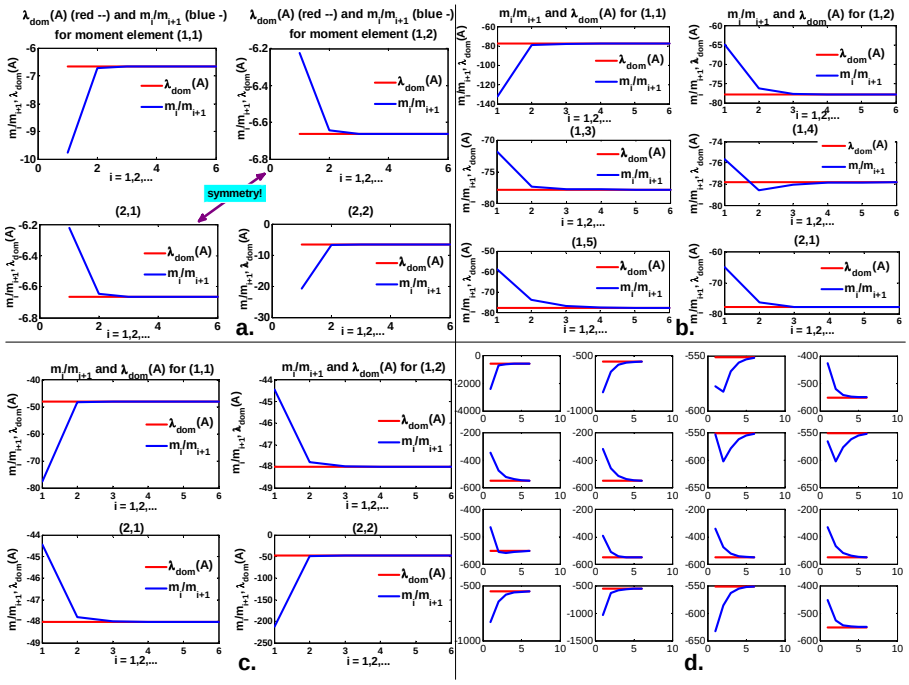


Figure 4.9: Behavior of $m_i(k,\ell)/m_{i+1}(k,\ell)$ versus λ_{dom} for some of the elements of the system block moments for the four systems from figure 4.6, with $m_i(k,\ell) = \mathbf{c}_k^T \mathbf{A}^{-i} \mathbf{b}_\ell$.

observability properties are neither necessary nor sufficient conditions for identifiability. That is, even if each unknown parameter, θ_i, is included in the part of the system which is controllable and observable, the identifiability of that parameter is not guaranteed. Moreover, the parameters in the uncontrollable or unobservable parts are not necessarily unidentifiable. [72] provides two simple examples that clarify these two points.

Pole computation and stability preservation by subspace-projection: An overture to Krylov subspace MOR

A reliable way of approximating dominant poles is by 'subspace-projection', which in SISO case basically consists of: 1) forming an $n \times k$ state moment matrix $\mathbf{R} := [\mathbf{r}_0, \mathbf{r}_1, \ldots, \mathbf{r}_{k-1}]$, where n is the dimension of the original state-space, 2) orthonormalizing $\mathbf{R}$ into an $n \times n$ orthogonal projection matrix $\mathbf{V}$, and 3) reducing, by *congruence transformation*, the original $n \times n$ system matrices $\mathbf{A}$ and $\mathbf{E}$ to $k \times k$ matrices $\mathbf{A}_{red} = \mathbf{V}^T\mathbf{A}\mathbf{V}$ and $\mathbf{E}_{red} = \mathbf{V}^T\mathbf{E}\mathbf{V}$, respectively. After this, the eigenvalues α_i of $\mathbf{A}_{red}^{-1}\mathbf{E}_{red}$ will be related to the r dominant poles λ_i we are looking for as: $\lambda_i = \frac{1}{\alpha_i}$ (as the poles of the original system are the eigenvalues of $\mathbf{E}^{-1}\mathbf{A}$, i.e. of the

inverse of $\mathbf{A}^{-1}\mathbf{E}$). Eigenvalues α_i can be easily obtained by eigendecomposition of $\mathbf{A}_{red}^{-1}\mathbf{E}_{red}$. Stability of the poles, and thus the stability (or more generally, 'passivity'[21]) of the system is guaranteed by the nature of congruence transformation. Moreover, we note that in this procedure only k moments are used to generate k poles, instead of $2k$ moments as in the traditional moment matching method described above. Orthonormalization of $\mathbf{R}$ can be simply accomplished by the standard *Gram-Schmidt* method, but it can also be performed during the moments generation process by methods such as the Arnoldi or Lanczos methods, which are numerically more stable than the standard Gram-Schmidt method.

The congruence projection matrix $\mathbf{V}$ as defined above can also be used to actually reduce the order of the original system, also in MIMO case. Noting that the state (block) moment matrix form a (block) Krylov subspace, one may consider the method as a Krylov subspace method. We, however, prefer to keep the name Krylov subspace MOR designated for methods that do not make use of explicitly computed moments and determine orthonormal bases of Krylov (block) subspaces in an iterative fashion using e.g. Arnoldi or Lanczos iterative algorithms. Implicit moment matching Krylov subspace MOR is the subject of the next subsection.

In a large majority[22] of single-phase cases considered here the method using orthogonalized matrix of state block moments performed exceptionally well using a rather low number of moments around $s = 0$. Numerically, that is quite fortunate, as (the columns of) the moments, as discussed above, all start to line-up with the dominant system's eigenvector very quickly. On the other side, as complex waveforms are not characteristic for single-phase systems considered here, the addition of only one extra moment may be expected to improve the models considerably. The symmetry of the systems also helps, as the same accuracy can be achieved using less moments than in a non-symmetric case. When the required order of the reduced model is higher than numerically achievable from the moments at $s = 0$, extending the projection subspace using information from a single or few extra expansion points may improve the matter sufficiently. Generally, the choice which expansion points to use is far from trivial, though. Due to both the nature and the diversity of possible requirements for a well performing reduced-order model, reservoir flow problems are even more complex in this respect. Actually, for this class of problems, even a priori determining whether the inclusion of extra expansion points would be of any value at all is an open issue.

[21] Loosely speaking, a system is said to be 'passive' if *it does not generate energy*. If the reduced order model looses its passivity, it may lead to unbounded responses in transient simulation, which would mean that new energy has been generated in the system. Formal definition of passivity of LTI systems includes the notions of 'positive-realness', which is not of fundamental importance in this thesis and is therefore omitted here. Readers interested in this particular subject are referred to [90, 186, 201, 260], for instance. Here we only emphasize that *the starting, high-dimensional models in this thesis are always passive*. It can be shown that congruence transforms guarantee passivity of the reduced-order models when the original, high-dimensional model is in passive form.

[22] The cases where the method had difficulties to approximate the flow production adequately concerned situations where all producing wells where situated in very low permeable zones. The flow through such wells is, as long as the total compressibility is 'slight but substantial', so low, however, that it is of no real practical significance. Anyway, even in these cases the method gave good results when used in a hybrid combination with another method, as described in Remark 4.3.2.

For example, a reduced-order model yielding a poor approximation of temporary production flow-rates (either total or individually per well) might yield excellent match of the final cumulative production.[23] However, even in such cases and as long as global flow properties are very well approximated, the reduced-order models may be very useful, e.g. for purposes of developing low-order models as in [104] (see also [242, 272] for a different technique), for 'control-relevant' upscaling [268, 269], flow prediction (e.g. Ensemble Kalman Filtering: EnKF), history matching, etcetera.

Figures 4.10 to 4.12 show simulation results for various single-phase cases using explicitly computed and orthogonalized (as a whole, using the 'orth' command in Matlab) state block moments:

a) The first case concerns the same system configuration and parameters as the system in Fig. 4.3. Besides the production flow rates for the original high-order model (441 states) and the reduced-order models with 6 states, 4 states, and 2 states, respectively, Fig. 4.10 also shows the flow-rate difference between the full-order model and each of the reduced order ones, as well as the varying injection flow rate. Here, $k = 6$, $k = 4$, and $k = 2$ are the dimensions of the projection matrix $\mathbf{V} \in \mathbb{R}^{441 \times k}$ and are equal to the rank of the state-moment matrix $\mathbf{R}$ of size 441×6 (3 state moments, 2 inputs), 441×4 (2 moments), and 441×2 (1 moment), respectively. As $\mathbf{V}$ and $\mathbf{R}$ have the same dimension for the 441×6 case, there is no (numerical) linear dependence among the (columns of the) three state-moments involved . As the figure shows, even the reduced-order model with just 4 states (2 moments) performs very well, as there is no (visual) distinction between the results obtained by it and by the high-dimensional model (the small window within the figure shows the flow-rate differences between the full-order model and the reduced models of order 4 and 6, respectively, on their natural scale). An interesting observation concerns the results for the reduced-order model with only 2 states, that is, using only the first state-moment. Whereas there clearly is an approximation error that seems to become larger the more abrupt the flow-rate change of the high-dimensional model is, it is noticeable that this error gets compensated for 'in cumulative sense', that is, the cumulative production (in this case production of a single well) remains approximated quite well. Although this behavior has not been observed in all the cases considered in this research, it has occurred frequently enough to justify further investigation.

b) The second case is a larger and more realistic system, consisting of $60 \times 220 \times 1 = 13200$ gridblocks and representing two individual horizontal slices of Model 2 from the 10th SPE Comparative Solution Project [65]. The model dimensions are 1200x2200x2 [ft] (365.76x670.56x0.6096 [m]). The two layers considered were the model's top layer, in which the permeability is smooth (Fig. 4.11.a), and the bottom layer, which is fluvial and characterized by a spaghetti of narrow high-permeability channels (Fig. 4.11.b). The heterogeneous permeability ranges over at least six orders of magnitude in both layers. Flow in the two layers is driven by imposing an injection well in the center of the reservoir and four production wells

[23] At the end, the variables being projected onto global subspaces are gridblock pressures, whereas the variables of interest in most of the cases are spatially localized flow-rates. This holds even more for multi-phase problems.

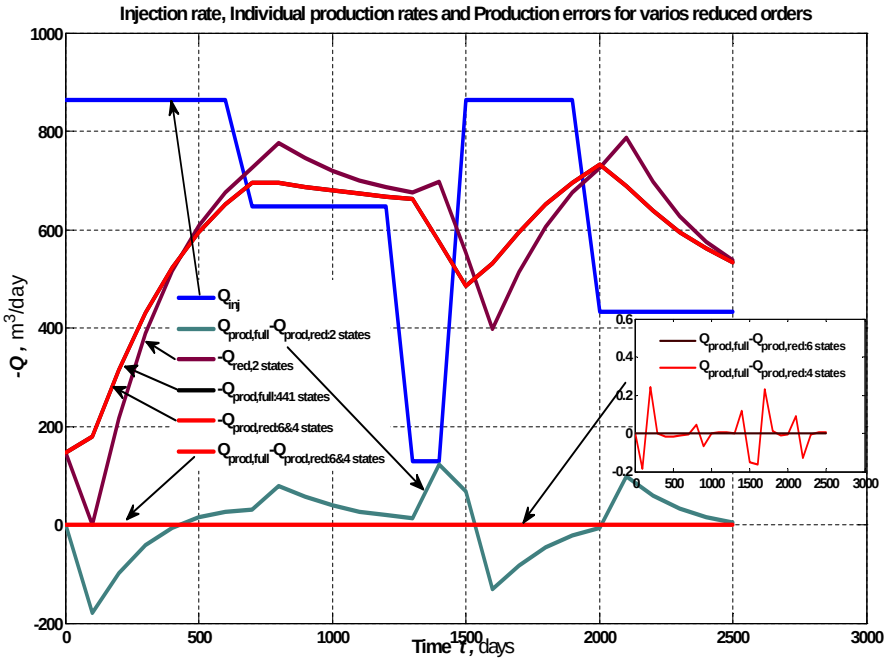


Figure 4.10: Results MOR by Projection with Explicit Moments for the same system as in Fig. 4.3. It is fair to mention that, for this particular example, singularly perturbed reduced-order models perform comparably well. Computational cost to obtain a reduced-order model is much higher, though.

in the four corners, respectively (i.e., a five-spot). According to the lower parts of Fig. 4.11, the reduced-order representation of size of just 10 performs great for both the top and the bottom layer. On the other side, (for all layers) it was observed that using more state-moments than 5 (i.e. reduced-order model of size 20) does not improve the results any more. According to the discussion above, for the input matrix $\mathbf{B} := [\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m]$, all $\mathbf{A}^{-j}\mathbf{b}_i$, $i = 1, \ldots, m$ have (numerically) lined-up with the dominant eigenvector of $\mathbf{A}$ already at $j = 5$. One can try to extend the projection space using expansions at extra points, but, as mentioned above, which points to chose is generally difficult to answer and maybe not helpful at all. Another possibility is to extend the projection subspace not by extra moments but by some other vectors, as in the method described in Remark 4.3.2. For the models regarding this case that would be unnecessary, as the approximation error (both temporary and cumulative) becomes neglectable already using three or four state moments.

c) The third case concerns the same systems as in case two, but the number of inputs and outputs is now substantially larger and equals 38 (24 injectors and 14 producers, in a five-spot configuration). For both the top and the bottom layer the

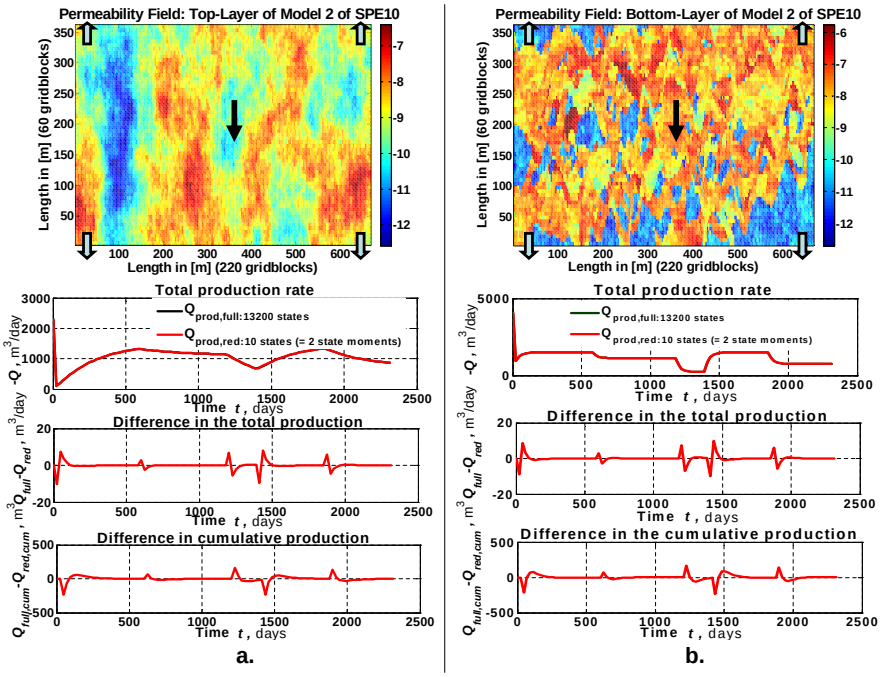
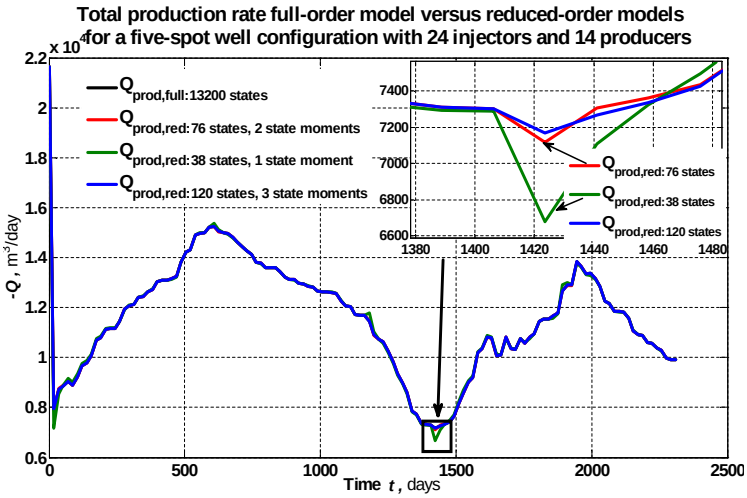


Figure 4.11: Results MOR by Projection with Explicit Moments for the top-layer (fig. a) and the bottom-layer (fig. b), respectively, of the SPE10 model 2.

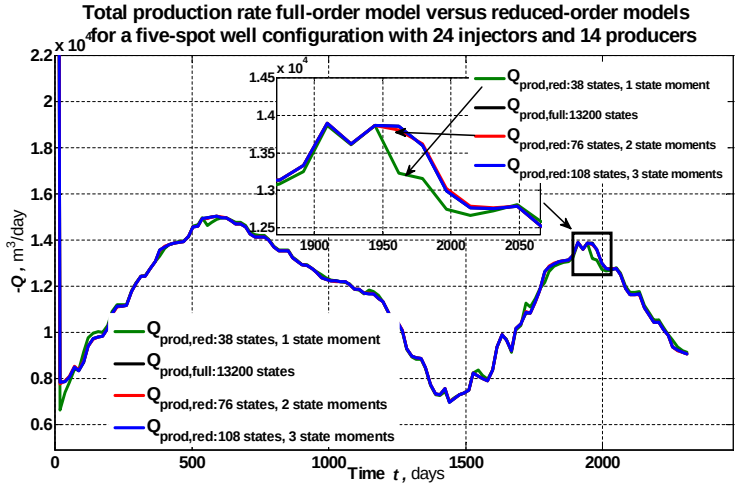
corresponding reduced-order models with 76 states (2 state block moments) were found to perform great, whereas the performance of the ones with 120 states for the top layer and 108 states for the bottom layer (3 state-moments minus some redundancy) were virtually indistinguishable from the full-order model. Fig. 4.12 shows the results. Moreover, for these two layers (and the well settings and control, the latter being piecewise constant injection rates and constant bottom-hole pressures at the producers) even just the first state block moment contained enough information to approximate the full-order model quite well.

It is important to note that, for these systems, the achievable order of the reduced model using moments is limited and can formally not go below 38. This strong and disadvantageous dependency on the number of inputs and/or outputs is characteristic to any 'full-blown'[24] moments-based model-order reduction, either explicit or implicit, and is not difficult to explain. One just needs to ob-

[24]The term 'full-blown' is adapted from [260] and is used here to distinguish model-order reduction using the standard projection framework from a model-order reduction using dominant poles approach. The latter uses the fact that all transfer functions of the original model share the same poles (!). Assuming that the dominant poles of the reduced-order transfer functions determined by an approach as above, for instance, will also (approximatively) coincide, it is sufficient to use one input-output pair to find these poles. A reduced-order model can then be determined by computing the corresponding residuals for each input-output pair. The order of this reduced model is equal to the number of the poles and therefore independent from input and output cardinality.



(a) Results for the top-layer of the SPE10 model.



(b) Results for the bottom-layer of the SPE10 model.

Figure 4.12: Total flow rate results for the two models from case 2: a five-spot well configuration with 24 (variable flow-rate constrained) injectors and 14 (constant bottom-hole pressure constrained) producers.

serve that, for state-moments based projection technique as above, each of the block state-moments $\mathbf{R}_i := \mathbf{A}^{-i}\mathbf{B}$ (or $\mathbf{R}_i := (\mathbf{A}^{-1}\mathbf{E})^{-i}\mathbf{A}^{-1}\mathbf{B}$ in the generalized case) is an $n \times m$ matrix, with m the number of inputs. Therefore, except when it is "notoriously" low, the (numerical) rank $k \leq \tilde{k}$ of $[\mathbf{R}_1, \mathbf{R}_2, \ldots, \mathbf{R}_{\tilde{k}}]$, and hence the order of the reduced-order model, will often be unnecessarily high. For the transfer function moment matching, it generally holds that the reduced model size is a multiple of the number of matched moments, whereby the multiplier is an increasing function of the number of inputs and outputs (see also Theorem 4.3.6). While the increasing cost of generating a reduced-order model will perhaps be amortized in later simulations, another, more problematic issue with a large number of inputs an/or outputs is the dense character of the projection matrices, as it directly affects simulation cost. If one desires to base the entire model order reduction on the moment matching idea, the so-called 'port-merging' and related techniques, developed in the area of VLSI[25] circuit design and analysis, could be considered, for instance. The main idea of port-merging is to make use of a potentially large degree of correlation between the various different inputs and outputs (either mutually or individually), which, based on an SVD (singular value decomposition) analysis, will yield a 'reduced-terminal' representation of the original high-dimensional state-space model (i.e., the state-dimension remains unchanged). State-space reduction can then be performed by employing the standard moment matching and congruence transformations from this section. Reference [260] provides an excellent overview of these techniques. While such a procedure will inevitably introduce some error, the reduced-order model will be more compact. Moreover, there is also a (potentially significant) practical usefulness of such an approach in the sense of reducing the necessary number of sensors and/or control units. On the other side, as state-moments are generally not sparse, SVD analysis can be computationally quite demanding for very large problems.

Remark 4.3.2. A projection matrix for MOR is not restricted to the use of one single type of vectors to determine it. That is, there is no reason why it should be formed from state (block) moments only, for instance. Realizing this, and regarding the material presented in this thesis so far, combining system's eigenmodes with state (block) moments is a natural choice. In this method, which we baptize as "Hybrid Mode Reduction" (HMR), the orthogonal state-projection matrix $\mathbf{V}$ is constructed from a given number of system's eigenvectors corresponding, for our purposes, to the eigenvalues closest to zero, and also one or more state (block) moments at $s = 0$. The rationale behind this method is that a) using the system's eigenmodes should make the reduced-order model a good approximation for slow-changes in the states, b) the inclusion of $-\mathbf{A}^{-1}\mathbf{B}$, which is the static solution for unit inputs, would make the stationary values of the state right, and c) the inclusion of higher order modes should increase the accuracy of the projection even further. We note that a similar method, is mentioned in [204]. The only, but important, difference is that, there, only $-\mathbf{A}^{-1}\mathbf{B}$ is combined with the eigenvectors.

[25] VLSI Very-Large-Scale Integration: the process of creating integrated circuits by combining millions of transistor-based circuits into a single chip.

4.3.3 *Implicit Moment Matching* using Krylov Subspaces

The basic idea behind *Krylov subspace* (KS) techniques for MOR is to *match the first few moments* of the reduced order model and the original system at one or more frequency expansion points and obtain a reduced-order state-space realization of the system *without actually computing the moments*. The insight that such an *implicit moment matching* and ('partial') state-space realization is possible comes from [110, 287].

Theorem 4.3.1. *Consider a stable n-dimensional SISO LTI system in the standard state-space form $\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x}+\boldsymbol{b}u$, $y = \boldsymbol{c}^T\boldsymbol{x}$ and a k-dimensional restriction of it: $\dot{\boldsymbol{a}} = (\boldsymbol{W}_k^T\boldsymbol{A}\boldsymbol{V}_k)\boldsymbol{a}+(\boldsymbol{W}_k^T\boldsymbol{b})u = \hat{\boldsymbol{A}}\boldsymbol{a}+\hat{\boldsymbol{b}}u$, $\hat{y} = (\boldsymbol{c}^T\boldsymbol{V}_k)\boldsymbol{a} = \hat{\boldsymbol{c}}^T\boldsymbol{a}$, and let the oblique (Krylov) projector $\pi_k = \pi_k^2 = \boldsymbol{V}_k\boldsymbol{W}_k^T$ be defined by matrices $\boldsymbol{V}_k$ and $\boldsymbol{W}_k$ such that:*

$$\begin{aligned} \text{COLSP}(\boldsymbol{V}_k) &= \mathcal{K}_k(\boldsymbol{A},\boldsymbol{b}) \\ &\equiv \text{span}\{\boldsymbol{b},\boldsymbol{A}\boldsymbol{b},\ldots,\boldsymbol{A}^{k-1}\boldsymbol{b}\}, \qquad (4.3.10) \\ \text{COLSP}(\boldsymbol{W}_k) &= \mathcal{K}_k(\boldsymbol{A}^T,\boldsymbol{c}) \\ &\equiv \text{span}\{\boldsymbol{c},\boldsymbol{A}^T\boldsymbol{c},\ldots,\boldsymbol{A}^{(k-1)^T}\boldsymbol{c}\}. \qquad (4.3.11) \end{aligned}$$

Then the first $2k$ Markov parameters of the original and reduced-order systems are identical.[26]

The reduced-order model from the above theorem matches the first $2k$ *high-frequency* moments of the original system.[27] In a completely analogous way of obtaining a reduced state-space realization which matches moments about $s_0 = \infty$ through an oblique projector corresponding to $\mathcal{K}_k(\mathbf{A},\mathbf{b})$ and $\mathcal{K}_k(\mathbf{A}^T,\mathbf{c})$, moment matching about $s_0 = 0$ can be obtained by employing a projector corresponding to $\mathcal{K}_k(\mathbf{A}^{-1},\mathbf{A}^{-1}\mathbf{b})$ and $\mathcal{K}_k(\mathbf{A}^{-T},\mathbf{A}^{-T}\mathbf{c})$, for instance. Actually, implicit moment matching using Krylov subspaces is possible at any particular frequency. Moreover, beyond the single point moment matching and standard-state space SISO setting, modern Krylov subspace algorithms produce reduced order models that match transfer function moments of MIMO systems in generalized state-space form at multiple frequency points. Projection matrices in Krylov-based methods can be generated using iterative algorithms like Arnoldi and Lancsoz that employ only inner-products and matrix vector multiplications. When the system matrices **A** and **E** are sparse, as normally is the case in large-scale applications, the computations involved are relatively cheap.

The literature on Krylov subspace MOR is rich. For excellent overviews and surveys of Krylov MOR techniques the reader is referred to [12, 13, 86, 87, 96].

[26] Note the equivalence between these subspaces and the controllable and observable subspaces of the full-order system for $k = n$.

[27] The fact that the idea of using Krylov subspaces as above is quite recent may be considered as rather surprising. Indeed, due to $\mathbf{x}(t) = \int_0^t e^{\mathbf{A}(t-\tau)}\mathbf{b}u(\tau)d\tau$, determining a good k-dimensional ($k << n$) approximation of the original system can be seen as ultimately connected to finding a pair $\{\hat{\mathbf{A}},\hat{\mathbf{b}}\}$ that approximates $e^{\mathbf{A}t}\mathbf{b}$ well. Taking this one step further, from input-output point of view what one is really interested in is the information in $e^{\mathbf{A}t}\mathbf{b}$ which is in direction of $\mathbf{c}$ (see also [97, 237] for utilization of an Arnoldi type of Krylov orthogonal projectors for approximation of $e^{\mathbf{A}t}\mathbf{b}$).

Deeper analysis of Krylov subspace methods, including issues as choosing the interpolation points (matching frequencies), estimating the modeling error, ensuring model stability, avoiding algorithm break-down, implementing parallelism, etc., can be found in [11, 115], for instance. In [104], KS model reduction has been proposed for purpose of developing low-order controllers for reservoir optimization. With the material presented in this subsection we hope to provide an adequate basic understanding of Krylov subspaces MOR, whereby the emphasize has been put on the basic concepts rather than detailed descriptions and algorithms. The presentation of theorems largely follows [239].

Degrees of freedom and invariance properties in the design of KS methods

The principal degrees of freedom in the design of KS methods are [239]:

- *Input and/or output Krylov subspace:* For each state-space system, there are two Krylov subspaces that are dual to each other: *input Krylov subspace*, defined using the system matrix **A** (or **A** and **E** in the generalized state-space case) and the input vector **b** (or input matrix **B** if MIMO), and *output Krylov subspace*, defined using the system matrices and the output vector **c** (matrix **C** if MIMO). Using only one of Krylov subspaces yields *one-sided* methods, whereas *two-sided* methods utilize both subspaces with the aim of matching more moments (except if the original system is symmetric, as in that case the two subspaces are equal), hence providing better approximations of the output **y** in many applications. On the other side, for one-sided methods with the particular choice $\mathbf{W} = \mathbf{V}$ the reduced-order models will preserve passivity (and thus stability) of certain passive original.

- *Starting vectors of subspaces:* In $\mathcal{K}_k(\mathbf{F}, \mathbf{r}) := span\ \{\mathbf{r}, \mathbf{Fr}, \ldots, \mathbf{F}^{k-1}\mathbf{r}\}$, where $\mathbf{F} \in \mathbb{R}^{n\times n}$, and $\mathbf{r} \in \mathbb{R}^n$ is called *starting vector*. In the 'block' Krylov subspace $\mathcal{K}_k(\mathbf{F}, \mathbf{R}) := colspan\{\mathbf{R}, \mathbf{FR}, \ldots, \mathbf{F}^{k-1}\mathbf{R}\}$, where $\mathbf{R} \in \mathbb{R}^{n\times m}$, starting vectors are located in the columns of **R**. By suitable choice of the starting vector(s) it is possible to simultaneously match moments at low- and high frequencies (i.e., moments and Markov parameters), without changing the total number of matched parameters. It is also possible to simultaneously match moments at arbitrary points by constructing orthonormal bases for unions of Krylov subspaces (multi-point moment matching is called *rational interpolation*). The standard algorithm for performing multi-point matching is the *Dual Rational Arnoldi* algorithm developed in [115] (see also [204]).

- *Krylov subspace bases:* When a choice for a Krylov subspace has been made, a basis of it needs to be determined which defines the projection. It can be shown that this influences the reduced-order model but not its transfer function (actually a corollary of Proposition 3.3.1).

- *Original state-space model representation and realization:* It appears that, unlike the invariance w.r.t. the bases, the reduced-order models obtained by projection methods are not necessarily invariant w.r.t. the original system

'representation' (that is, multiplication of the original state-equation by a non-singular matrix; examples: scaling, preconditioning, etc.) and *realization* (nonsingular similarity transformations of the state-space).

In the sequel these items are explained in more detail. As for the theorems, only the first one is provided with a proof. The proofs of the others follow the similar line and can be found in [239], for instance.

Theorems

The theorems are given for the general(ized) state-space systems (the term **Du** is omitted here as it is assumed to remain the same after reduction):

$$\mathbf{E}\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \quad \mathbf{y} = \mathbf{C}\mathbf{x}, \tag{4.3.12}$$

where $\mathbf{A},\mathbf{E} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times m}$ and $\mathbf{C} \in \mathbb{R}^{p\times n}$. When the system is SISO, u, y, $\mathbf{b}$ and $\mathbf{c}^T$ are used instead of $\mathbf{u}$, $\mathbf{y}$, $\mathbf{B}$ and $\mathbf{C}^T$, respectively. The reduced-order models obtained by state- and equation-error projections using $n \times k$ projection matrices $\mathbf{V}$ and $\mathbf{W}$, respectively, have the form:

$$\mathbf{E}_{red}\dot{\mathbf{x}}_{red} = \mathbf{A}_{red}\mathbf{x}_{red} + \mathbf{B}_{red}\mathbf{u}, \quad \hat{\mathbf{y}} = \mathbf{C}_{red}\mathbf{x}_{red}, \tag{4.3.13}$$

where $\mathbf{E}_{red} = \mathbf{W}^T\mathbf{E}\mathbf{V}$, $\mathbf{A}_{red} = \mathbf{W}^T\mathbf{A}\mathbf{V}$, $\mathbf{B}_{red} = \mathbf{W}^T\mathbf{B}$, $\mathbf{C}_{red} = \mathbf{C}\mathbf{V}$, and $\mathbf{x} = \mathbf{V}\mathbf{x}_{red}$. According to $(4.3.7)$, the system moments (around zero) are *scalars* $m_i = \mathbf{c}^T(\mathbf{A}^{-1}\mathbf{E})^i\mathbf{A}^{-1}\mathbf{b}$, $i = 0, 1, \ldots$ in SISO case and $p \times m$ *matrices* $\mathbf{M}_i = \mathbf{C}(\mathbf{A}^{-1}\mathbf{E})^i\mathbf{A}^{-1}\mathbf{B}$ in MIMO case.

SISO Krylov Moment Matching

Theorem 4.3.2. *Consider the Krylov subspace $\mathcal{K}_{k_1}(\boldsymbol{A}^{-1}\boldsymbol{E}, \boldsymbol{A}^{-1}\boldsymbol{b})$. If the projection matrix $\boldsymbol{V}$ is chosen as a basis of this subspace having rank k, and the matrix $\boldsymbol{W}$ is chosen arbitrary but such that $\boldsymbol{A}_{red}$ is nonsingular, then the first k moments around zero of the original system* $(4.3.12)$ *and the reduced order system* $(4.3.13)$ *match.*

Proof. The 0th moment of the reduced-order system is $m_{red,0} = \mathbf{c}_{red}^T\mathbf{A}_{red}^{-1}\mathbf{b}_{red} = \mathbf{c}^T\mathbf{V}(\mathbf{W}^T\mathbf{A}\mathbf{V})^{-1}\mathbf{W}^T\mathbf{b}$. As the vector $\mathbf{A}^{-1}\mathbf{b}$ is in the Krylov subspace, it can be written as a linear combination of the basis matrix $\mathbf{V}$, that is, there is a k-dimensional vector $\mathbf{f}_0$ such that $\mathbf{A}^{-1}\mathbf{b} = \mathbf{V}\mathbf{f}_0$. Using this, one obtains:

$$\begin{aligned} \mathbf{f}_0 &\equiv (\mathbf{W}^T\mathbf{A}\mathbf{V})^{-1}\mathbf{W}^T\mathbf{A}\mathbf{V}\mathbf{f}_0 = (\mathbf{W}^T\mathbf{A}\mathbf{V})^{-1}\mathbf{W}^T(\mathbf{A}\mathbf{A}^{-1})\mathbf{b} = (\mathbf{W}^T\mathbf{A}\mathbf{V})^{-1}\mathbf{W}^T\mathbf{b} \\ &= \mathbf{A}_{red}^{-1}\mathbf{b}_{red}. \end{aligned} \tag{4.3.14}$$

With this, $m_{red,0}$ becomes $m_{red,0} = \mathbf{c}^T\mathbf{V}\mathbf{f}_0 = \mathbf{c}^T\mathbf{A}^{-1}\mathbf{b} = m_0$.

For the next moment, $m_{red,1} = \mathbf{c}_{red}^T\mathbf{A}_{red}^{-1}\mathbf{E}_{red}\mathbf{A}_{red}^{-1}\mathbf{b}_{red}$, $(4.3.14)$ yields:

$$\underbrace{(\mathbf{W}^T\mathbf{A}\mathbf{V})^{-1}}_{\mathbf{A}_{red}^{-1}}\underbrace{\mathbf{W}^T\mathbf{E}\mathbf{V}}_{\mathbf{E}_{red}}\underbrace{(\mathbf{W}^T\mathbf{A}\mathbf{V})^{-1}}_{\mathbf{A}_{red}^{-1}}\underbrace{\mathbf{W}^T\mathbf{b}}_{\mathbf{b}_{red}} = \underbrace{(\mathbf{W}^T\mathbf{A}\mathbf{V})^{-1}}_{\mathbf{A}_{red}^{-1}}\underbrace{\mathbf{W}^T\mathbf{E}\mathbf{V}}_{\mathbf{E}_{red}}\mathbf{f}_0 = \mathbf{A}_{red}^{-1}\mathbf{W}^T\mathbf{E}\mathbf{A}^{-1}\mathbf{b}.$$

Now, using this and the fact that the vector $\mathbf{A}^{-1}\mathbf{E}\mathbf{A}^{-1}\mathbf{b}$ is also in the Krylov subspace and can therefore be written as $\mathbf{A}^{-1}\mathbf{E}\mathbf{A}^{-1}\mathbf{b} = \mathbf{V}\mathbf{f}_1$, one obtains:

$$\begin{aligned}\mathbf{A}_{red}^{-1}\mathbf{E}_{red}\mathbf{A}_{red}^{-1}\mathbf{b}_{red} &= (\mathbf{W}^T\mathbf{A}\mathbf{V})^{-1}\mathbf{W}^T(\mathbf{A}\mathbf{A}^{-1})\mathbf{E}\mathbf{A}^{-1}\mathbf{b} = (\mathbf{W}^T\mathbf{A}\mathbf{V})^{-1}\mathbf{W}^T\mathbf{A}\mathbf{V}\mathbf{f}_1 \\ &= \mathbf{f}_1. \end{aligned} \tag{4.3.15}$$

The moment $m_{red,1}$ therefore becomes $m_{red,1} = \mathbf{c}^T\mathbf{V}\mathbf{f}_1 = \mathbf{c}^T\mathbf{A}^{-1}\mathbf{E}\mathbf{A}^{-1}\mathbf{b} = m_1$.

Equivalence of the second moment is proven by using (4.3.14) and (4.3.15) and the fact that $(\mathbf{A}^{-1}\mathbf{E})^2\mathbf{A}^{-1}\mathbf{b}$ can be written as a linear combination[28] of the columns of $\mathbf{V}$. These steps can be repeated until $m_{red,(k-1)} = m_{k-1}$ and k moments match. □

If the original system is controllable, then k_1 in theorem 4.3.2 equals k. If $k < k_1$, the system is not fully controllable and all moments match (as then $(\mathbf{A}^{-1}\mathbf{E})^i\mathbf{A}^{-1}\mathbf{b}$ are in $\mathcal{K}_{k_1}$ for any $i > k_1$). A suitable iterative algorithm like e.g. Arnoldi, a basic (modified) version of which is given at the end of this section, will stop after k iterations as every next Krylov vector is in that case a linear combination of the previous ones. In MIMO case, where the Krylov sequence $\mathbf{G}, \mathbf{F}\mathbf{G}, \mathbf{F}^2\mathbf{G}, \ldots, \mathbf{F}^{i-1}\mathbf{G}$ with $\mathbf{F} = \mathbf{A}^{-1}\mathbf{E}, \mathbf{G} = \mathbf{A}^{-1}\mathbf{B}$ is considered, the situation is not so trivial. The main difficulty is that for $m \neq 1$, if the jth state block-moment $\mathbf{F}^{j-1}\mathbf{G}$ contains a column that is linearly dependent on columns on its left in the sequence $\mathbf{G}, \mathbf{F}\mathbf{G}, \ldots, \mathbf{F}^{j-1}\mathbf{G}, \ldots$, then, in general, not all the columns of $\mathbf{F}^{j-1}\mathbf{G}$ are linearly dependent on columns on their left. A good algorithm will therefore continue with constructing the basis for the subspace spanned by the sequence after a linearly dependent column has been found. The column itself and all its successive $\mathbf{A}^{-1}\mathbf{E}$- multiples are to be deleted. Formally, this can be done by scanning the columns of the block moments in the above sequence from left to right and deleting each column that is linearly dependent on earlier columns. Performing such an *exact deflation* yields a *deflated* block Krylov sequence. In an actual algorithm for constructing basis vectors for the original sequence in finite arithmetic also vectors that are in some sense "almost" linearly independent on earlier vectors needs to be deleted (*inexact deflation*) [91]. It is hereby stressed that the iterative algorithm will not compute any moment. Actually, as higher order moments contain increasingly less information, the algorithm will generally use different set of vectors/blocks.

The subspace $\mathcal{K}_{k_1}(\mathbf{A}^{-1}\mathbf{E}, \mathbf{A}^{-1}\mathbf{b})$ does not involve system's output data and is called *input Krylov subspace*. Being its dual subspace, the so-called *output Krylov subspace* $\mathcal{K}_{k_2}(\mathbf{A}^{-T}\mathbf{E}^T, \mathbf{A}^{-T}\mathbf{c})$ can also be used for model order reduction in a similar fashion. That is, if the matrix $\mathbf{W}$ in (4.3.13) is a basis of the output Krylov subspace $\mathcal{K}_{k_2}(\mathbf{A}^{-T}\mathbf{E}^T, \mathbf{A}^{-T}\mathbf{c})$ with rank k and $\mathbf{V}$ is arbitrary but such that $\mathbf{A}_{red}$ is nonsingular, then the first k moments of the original and the reduced order system will match. Methods that employ a basis of only one of the two dual subspaces are

[28] According to (4.3.7), $(\mathbf{A}^{-1}\mathbf{E})^i\mathbf{A}^{-1}\mathbf{b} := \mathbf{r}_i$ are the state-moments of the original system. From derivations in this proof we have that $\mathbf{r}_i = \mathbf{V}\mathbf{f}_i$, where $\mathbf{f}$ are the state-moments of the reduced-order model. Surprisingly, this simple and direct connection between the (first k) state-moments of the original and the reduced system seems to be overlooked in the literature.

called *one-sided* methods. Combining input and output Krylov subspaces, yielding *two-sided* algorithms, it is possible to match more moments, as expressed by the following theorem.

Theorem 4.3.3. *Let* $\boldsymbol{V}$ *and* $\boldsymbol{W}$ *in* (4.3.13) *be bases of Krylov subspaces* $\mathcal{K}_{k_1}(\boldsymbol{A}^{-1}\boldsymbol{E}, \boldsymbol{A}^{-1}\boldsymbol{b})$ *and* $\mathcal{K}_{k_2}(\boldsymbol{A}^{-T}\boldsymbol{E}^T, \boldsymbol{A}^{-T}\boldsymbol{c})$*, respectively, both with rank* k*. Then, assuming that* $\boldsymbol{A}$ *and* $\boldsymbol{A}_{red}$ *are nonsingular, the first* $2k$ *moments of the original and reduced-order system will match*[29]*.*

Due to the intrinsic symmetry present in the high-dimensional state-space system description of the single-phase models in this thesis, the one-sided projection involving only one projection matrix[30], here $\mathbf{V}$, is actually a two-sided projection, thus matching $2k$ moments. This follows directly as a corollary of the theorems above and from the fact that, in this case, input and output Krylov subspaces are the same.

Corollary 4.3.4. *For a state-space symmetric system* $\{\boldsymbol{E}, \boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c}\}$*, with* $\boldsymbol{E}$ *and* $\boldsymbol{A}$ *symmetric and* $\boldsymbol{b} = \boldsymbol{c}$*, for one-sided Krylov subspace methods with congruence transformation (i.e.,* $\boldsymbol{W} = \boldsymbol{V}$*), the reduced-order model matches* $2k$ *moments of the original transfer function.*

These basic results for moment matching around zero frequency, which yields steady-state accuracy, can be extended to matching moments around any $s_0 \neq 0$. To that, $\mathbf{A}$ is replaced by $(\mathbf{A} - s_0\mathbf{E})$ in the definition of moments and Krylov subspaces. For instance, in theorem 4.3.3 the subspaces $\mathcal{K}_{k_1}((\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{E}, (\mathbf{A} - s_0\mathbf{E})^{-1}\mathbf{b})$ and $\mathcal{K}_{k_2}((\mathbf{A} - s_0\mathbf{E})^{-T}\mathbf{E}^T, (\mathbf{A} - s_0\mathbf{E})^{-T}\mathbf{c})$ are considered. It is also possible to match moments at several frequency points simultaneously. General results in this so-called *Multipoint Rational Interpolation* context, based on unions of Krylov subspaces involving the different frequency points, are provided in [115]. Another general case of simultaneously matching moments and Markov parameters using single input and output Krylov subspaces is considered in [239]. As explained above, for single-phase porous media equations with low-frequent input signal forms and low compressibility it is reasonable to expect that the expansion around the single point $s = 0$ will suffice in most of the cases.

MIMO Krylov Moment Matching

In MIMO case with m inputs and p outputs, the system moments are $p \times m$ matrices and, therefore, per moment match $p \times m$ scalar parameters needs to be matched. The theorems 4.3.2 and 4.3.3 are generalized to MIMO systems as follows.

[29] In SISO case, if the system is minimal, i.e., both controllable and observable, then $k_1 = k_2 = k$. Otherwise all moments match and the iterative algorithm finds a minimal realization.

[30] In the literature, many contributions about one-sided methods using input Krylov subspaces suggest to choose $\mathbf{W} = \mathbf{V}$. If an Arnoldi process is used, for instance, the produced $\mathbf{V}$ is orthogonal, so $\mathbf{W} = \mathbf{V}$ yields $\mathbf{W}^T\mathbf{V} = \mathbf{I}$, which simplifies the computation and the use of the reduced order model. Another reason for this choice of $\mathbf{W}$ is that it yields an *congruence transformation* of the state-space, i.e. the reduced-order system matrices are then $\mathbf{E}_{red} = \mathbf{V}^T\mathbf{E}\mathbf{V}$, $\mathbf{A}_{red} = \mathbf{V}^T\mathbf{A}\mathbf{V}$, $\mathbf{B}_{red} = \mathbf{V}^T\mathbf{B}$,$\mathbf{C}_{red} = \mathbf{C}\mathbf{V}$, which, as explained earlier (page 62), yield reduced-order models with many desired characteristics (passivity (stability) preservation, dominant poles approximation, etc.).

Theorem 4.3.5. *Let **B** and **C** in (4.3.12) be $n \times m$ and $p \times n$ matrices, respectively, and let **A** be invertible. If the matrix **V** in (4.3.13) is a basis of Krylov subspace $\mathcal{K}_{k_1}(\boldsymbol{A}^{-1}\boldsymbol{E}, \boldsymbol{A}^{-1}\boldsymbol{B})$ with rank k (i.e. k is a multiple of m) and **W** is chosen arbitrary but such that the matrix $\boldsymbol{A}_{red}$ is nonsingular, then the first $\frac{k}{m} = \alpha$ moments of the original and reduced order system match.*

Proof. The proof is the same as in theorem 4.3.2, but the parameters $\mathbf{f}_0, \mathbf{f}_1, \ldots$ are $k \times m$ matrices. □

Theorem 4.3.6. *Let **A**,**B** and **C** be as in theorem 4.3.5. If the matrices **V** and **W** are bases of of Krylov subspaces $\mathcal{K}_{k_1}(\boldsymbol{A}^{-1}\boldsymbol{E}, \boldsymbol{A}^{-1}\boldsymbol{B})$ and $\mathcal{K}_{k_2}(\boldsymbol{A}^{-T}\boldsymbol{E}^T, \boldsymbol{A}^{-T}\boldsymbol{C}^T)$, respectively, both with rank k (k is a multiple of both m and p), then the first $\frac{k}{m} + \frac{k}{p}$ moments of the original and reduced order system match*[31].

Proof. See [239]. □

Invariance of reduced-order models w.r.t. basis change As stated above, employing *any* basis of input or output Krylov subspaces for order reduction yields moment matching. The following theorem and two corollaries state that even the input-output behavior of the reduced model does not depend on the choice of basis.

Theorem 4.3.7. *The transfer function of the reduced-order system (4.3.13) is independent of the particular choice of the bases **V** and **W** of respectively $\mathcal{K}_{k_1}(\boldsymbol{A}^{-1}\boldsymbol{E}, \boldsymbol{A}^{-1}\boldsymbol{B})$ and $\mathcal{K}_{k_2}(\boldsymbol{A}^{-T}\boldsymbol{E}^T, \boldsymbol{A}^{-T}\boldsymbol{C}^T)$.*

Proof. This theorem, as well as the two corollaries below, follow directly from Proposition 3.3.1. □

The corollaries concern one-sided methods. Define $\mathbb{S}(\mathbf{N}_f) := \{\mathbf{N} : \exists \mathbf{T}, \mathbf{T}^{-1} \text{ s.t. } \mathbf{N} = \mathbf{N}_f\mathbf{R}\}$. Then,

Corollary 4.3.8. *Let the matrix V used in (4.3.13) be a basis of input Krylov subspace with rank k and let $\boldsymbol{W} \in \mathbb{S}(\boldsymbol{W}_f)$ with a fixed matrix $\boldsymbol{W}_f$. Then the transfer function of the reduced-order system is independent of the particular choice of the bases **V** and the matrix **W**.*

Invariance w.r.t. original system representation and realization

Consider two system *representations* (i.e. one is obtained by multiplying the state-equation by a nonsingular matrix **P**):

$$\begin{array}{rcl} \mathbf{E}_1\dot{\mathbf{x}} &=& \mathbf{A}_1\mathbf{x} + \mathbf{B}_1\mathbf{u} \\ \mathbf{y} &=& \mathbf{C}_1\mathbf{x} \end{array} \quad , \quad \begin{array}{rcl} \mathbf{E}_2\dot{\mathbf{x}} &=& \mathbf{A}_2\mathbf{x} + \mathbf{B}_2\mathbf{u} \\ \mathbf{y} &=& \mathbf{C}_2\mathbf{x} \end{array} \tag{4.3.16}$$

[31] In the single-phase applications in this thesis $m = p$, so the number of matched moments with a two-sided algorithm (also with one-sided if the high-dimensional model is in a symmetric form) equals $\frac{2k}{m} = 2\alpha$ ($k = \alpha m$).

where $\mathbf{E}_2 = \mathbf{P}\mathbf{E}_1$, $\mathbf{A}_2 = \mathbf{P}\mathbf{A}_1$, $\mathbf{B}_2 = \mathbf{P}\mathbf{B}_1$, $\mathbf{C}_2 = \mathbf{C}_1$, and $\mathbf{A}_2^{-1}, \mathbf{A}_1^{-1}$ exist. The function of the nonsingular matrix $\mathbf{P}$ could e.g. be to 'precondition' the original model[32], but the two representations could also have resulted just from writing the system equations in different ways. Clearly, the transfer functions of the two models are the same. However, as stated next, depending on whether one uses a one-sided or a two-sided Krylov reduction method, their reduced-order models are or are not the same in general [239].

Theorem 4.3.9. *In order reduction based on projection in theorems 4.3.3 and 4.3.6 using two-sided method (i.e., projection matrices $\boldsymbol{V}_i$ and $\boldsymbol{W}_i$ are bases of $\mathcal{K}_k(\boldsymbol{A}_i^{-1}\boldsymbol{E}_i, \boldsymbol{A}_i^{-1}\boldsymbol{B}_i)$ and $\mathcal{K}_k(\boldsymbol{A}_i^{-T}\boldsymbol{E}_i^T, \boldsymbol{A}_i^{-T}\boldsymbol{C}_i^T)$, respectively, $i = 1, 2$), the reduced-order systems of the two representations in (4.3.16) are exactly the same.*

For one sided methods, the invariance of the reduced-order model when changing the representation does not hold in general, except in some cases using output Krylov subspace, see [239]. In application, (for non-symmetric systems) this can be an essential disadvantage as the results depend on the way the equations were written.

Similar results hold also for different *realizations* of the original system:

$$\begin{array}{rcl} \mathbf{E}\dot{\mathbf{x}} &=& \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \\ \mathbf{y} &=& \mathbf{C}\mathbf{x} \end{array} \quad , \quad \begin{array}{rcl} \mathbf{E}\mathbf{T}\dot{\mathbf{z}} &=& \mathbf{A}\mathbf{T}\mathbf{z} + \mathbf{B}\mathbf{u} \\ \mathbf{y} &=& \mathbf{C}\mathbf{T}\mathbf{z} \end{array} \tag{4.3.17}$$

Theorem 4.3.10. *Changing the realization of the original system does not change the input-output behavior of the reduced model if two-sided Krylov method is used.*

Again, for one-sided methods such a result does not exist in general, except in some cases using output Krylov subspace.

4.4 Balanced Truncation MOR

As stated in Subsection 4.1.2, the eigenmodes of an LTI system are generally not the best measure of the modes' (joint) weak controllability/observability. Due to the equivalence of minimal state-space realizations of an LTI system w.r.t. its transfer function, it is reasonable to ask if there are maybe other modes which are more controllable and more observable than the eigenmodes, thus admitting models of the same or better quality using even less modes? The theory of *balanced*

[32]The preconditioner $\mathbf{P}$ will change the matrix pencil $(s\mathbf{E} - \mathbf{A})$ to $\mathbf{P}(s\mathbf{E} - \mathbf{A})$. In the traditional application of solving linear systems of equations $\mathbf{A}\mathbf{x} = \mathbf{b}$ a (left) preconditioner $\mathbf{P}$ in $\mathbf{P}\mathbf{A}\mathbf{x} = \mathbf{P}\mathbf{b}$ would generally be chosen as to approximate $\mathbf{A}$ in some sense. In the frequency-dependent problems, however, the fixed $\mathbf{P}$ can not approximate $(s\mathbf{E} - \mathbf{A})^{-1}$ for all frequencies s in general. Therefore, the introduction of $\mathbf{P}$ in the preconditioning sense can only be useful over certain frequency ranges. In [115] this difference with the traditional fixed case is emphasized by calling $\mathbf{P}$ a *dynamic system (DS) preconditioner*.

realizations answers positively to this question, and the so-called *balanced truncation* (BTR) algorithms yield reduced-order models in which only the most controllable and observable modes, in a well-defined 'energetic' sense, are retained. The section is divided in three parts. The first part introduces the main theoretical results of this powerful MOR methodology. The second part presents the standard algorithms for obtaining balanced truncated models and shows their performance when applied on single-phase reservoir models. As the high computational complexity of the standard algorithms restricts the applicability of the 'exact' balancing to rather moderate system sizes, the last part of the section is devoted to the problem of obtaining approximate balancing solutions.

4.4.1 Energy Functions, System Gramians and Lyapunov Equations

Consider an n-dimensional, stable and minimal, time-continuous LTI system in the standard state-space formulation:

$$\frac{d\mathbf{x}}{dt} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \quad \mathbf{x}(0) = \mathbf{x}_0, \tag{4.4.1}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}, \tag{4.4.2}$$

with p inputs and m outputs. Perhaps the most natural way of explaining the reasoning behind BRs and BTR is to adopt the 'energetic point of view', in which the so-called *controllability energy function* and *observability energy function*, denoted respectively by L_c and L_o, play a central role.

Definition 4.4.1. The controllability and observability functions for a linear system are defined as

$$L_c(\mathbf{x}_0) := \min_{\substack{\mathbf{u} \in \mathbb{L}_2(-\infty, 0) \\ \mathbf{x}(-\infty) = \mathbf{0}, \mathbf{x}(0) = \mathbf{x}_0}} \frac{1}{2} \int_{\infty}^{0} \|\mathbf{u}(t)\|^2 dt, \tag{4.4.3}$$

and

$$L_o(\mathbf{x}_0) := \frac{1}{2} \int_{0}^{\infty} \|\mathbf{y}(t)\|^2 dt, \quad \mathbf{x}(0) = \mathbf{x}_0, \mathbf{u}(t) \equiv \mathbf{0}, \quad 0 \leq t < \infty, \tag{4.4.4}$$

respectively, where $\mathbb{L}_2(a, b)$ is the space of square-integrable functions on the interval (a, b).

The value of L_c at state $\mathbf{x}_0$ is thus the minimum amount of (finite) control energy required to reach the state $\mathbf{x}_0$ from $\mathbf{0}$. The value of L_o at state $\mathbf{x}_0$ is the amount of energy generated by the system's natural response to initial state $\mathbf{x}_0$.

In the stable linear case, both L_c and L_o are necessarily quadratic functions of

$\mathbf{x}_0$, and therefore we can write:

$$L_c(\mathbf{x}_0) = \frac{1}{2}\mathbf{x}_0^T\mathbf{W}_c^{-1}\mathbf{x}_0 \quad (4.4.5)$$

$$L_0(\mathbf{x}_0) = \frac{1}{2}\mathbf{x}_0^T\mathbf{W}_o\mathbf{x}_0, \quad (4.4.6)$$

for some (constant) symmetric positive definite (s.p.d.) matrices $\mathbf{W}_c$ and $\mathbf{W}_o$. It is a well known result that these matrices are the *controllability-* and *observability Gramians*[33] respectively, given by:

$$\mathbf{W}_c := \int_0^\infty e^{\mathbf{A}t}\mathbf{B}\mathbf{B}^T e^{\mathbf{A}^T t}dt, \quad (4.4.7)$$

$$\mathbf{W}_o := \int_0^\infty e^{\mathbf{A}^T t}\mathbf{C}^T\mathbf{C}e^{\mathbf{A}t}dt. \quad (4.4.8)$$

The importance of these matrices is thus that they respectively measure to what degree each state of the system, in the given state-space coordinates, can be excited by an input (note the inversion of $\mathbf{W}_c$ in the integral definition of L_c) and to what degree each state excites future outputs. Hence, for two different states $\mathbf{x}_1$ and $\mathbf{x}_2$, with $\|\mathbf{x}_1\|_2 = \|\mathbf{x}_2\|_2$ and $\mathbf{x}_1^T\mathbf{W}_c\mathbf{x}_1 > \mathbf{x}_2^T\mathbf{W}_c\mathbf{x}_2$, one says that the state $\mathbf{x}_1$ is 'more controllable' than the state $\mathbf{x}_2$, expressing the fact that a less-energetic input (in the $\mathbb{L}_2$-norm) is required to bring the system from rest to $\mathbf{x}_1$ than to $\mathbf{x}_2$. Conversely, states which produce larger output energy are called 'more observable', and in this sense are considered more dynamically important than less observable states.

It can be proven (see [150] or any other advanced book on system and control theory) that the Gramians are unique solutions of the Lyapunov equations:

$$\mathbf{A}\mathbf{W}_c + \mathbf{W}_c\mathbf{A}^T = -\mathbf{B}\mathbf{B}^T \quad (4.4.9)$$

and

$$\mathbf{A}^T\mathbf{W}_o + \mathbf{W}_o\mathbf{A} = -\mathbf{C}^T\mathbf{C}. \quad (4.4.10)$$

Remark 4.4.1. We note that an analysis of controllability and observability properties of single phase reservoir models has recently been provided in [302, 303]. In these and related contributions [273, 274], the reservoir models under analysis are discrete in time, as opposite to the continuous-time state-space reservoir descriptions treated in this thesis. We stress that the analysis in these models requires that the time-step length in these models is kept constant during the whole simulation as otherwise the system considered is not LTI anymore, but rather LTV (linear time-varying) (see, e.g., [109]). Moreover, in order to maximize the invariance of the results of the analysis with respect to the length of the time-discretisation step

[33] The Gramian matrix of a (piecewise-continuous) map $\mathbf{F}(t)$ in $\mathbb{R}^{n\times m}$ is defined as $\mathbf{W} = \int_0^\infty \mathbf{F}(t)\mathbf{F}^T(t)dt$. It is generally symmetric positive semi-definite, with non-negative and real eigenvalues $\{\mu_i\}_{i=1}^n$ and mutually orthogonal unitary eigenvectors $\{\mathbf{w}_i\}_{i=1}^n$. Due to the orthonormality of its eigenvectors, $\mathbf{F}(t)$ can be written as $\mathbf{F}(t) = \sum_{i=1}^n \mathbf{w}_i\mathbf{f}_i^T(t)$, where $\mathbf{f}_i^T(t) = \mathbf{w}_i^T\mathbf{F}(t)$, $i = 1, \ldots, n$. The following relations hold: $\int_0^\infty \mathbf{f}_i^T(t)\mathbf{f}_j(t)dt = 0, i \neq j$ and $\int_0^\infty \|\mathbf{f}_i^T(t)\|^2 dt \equiv \mathbf{w}_i^T\mathbf{W}\mathbf{w}_i = \mu_i$.

Δt (directly influences **A** and **B**) and the scaling of reservoir physical parameters by a scaling factor ϵ (influences **A**, **B**, **C** and **D**), certain assumptions must necessarily be imposed on these quantities (in particular the product $\epsilon\Delta t$). With these assumptions, results of the analysis in these contributions are virtually the same as what would be concluded based on the underlying continuous-time reservoir description.

4.4.2 Balanced Realizations, Truncation and Residualization

As mentioned earlier, the system's *input-output behavior is state-space realization invariant*, i.e. it does not change when an arbitrary non-singular state-space transformation $\mathbf{x} = \mathbf{Tz}$ is applied (e.g., scaling of reservoir physical parameters or time, but also grid numbering). *Controllability and observability properties of a system*, however, *are, each 'individually*, generally *not invariant* in this respect. Actually, (4.4.5) and (4.4.6) each individually suggest a different state-space projection matrix. This is where the idea of so-called *balanced realizations* and *balanced truncation* comes into the picture. A balanced realization of a (stable) LTI system is a realization in which the controllability and the observability Gramians are equal and diagonal, i.e. $\mathbf{W}_c = \mathbf{W}_o = \Sigma = diag(\sigma_1, \ldots, \sigma_n)$, with all σ's positive. The existence of such a realization is guaranteed by the following theorem [190]:

Theorem 4.4.1. *The eigenvalues of $\boldsymbol{W}_o\boldsymbol{W}_c$ are similarity invariants, i.e. they do not depend on the choice of the state space coordinates. There exists a state space representation where*

$$\boldsymbol{\Sigma} = \boldsymbol{W}_c = \boldsymbol{W}_o \quad = \quad \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_n \end{bmatrix}, \tag{4.4.11}$$

with $\sigma_1 \geq \sigma_2 \geq \ldots \sigma_n > 0$ the quare roots of the eigenvalues of $\boldsymbol{W}_c\boldsymbol{W}_o$. Such representations are called "balanced" and the system is in "balanced form". Furthermore, the σ_i's equal the Hankel singular values*, i.e., the singular values of the Hankel operator of the system.*

The first part of the theorem is easily proven by observing (e.g. from the energy functions expressions) that, under a nonsingular state transformation $\mathbf{x} = \mathbf{T}\hat{\mathbf{x}}$, the Gramians transform into $\hat{\mathbf{W}}_c = \mathbf{T}^{-1}\mathbf{W}_c\mathbf{T}^{-T}$ and $\hat{\mathbf{W}}_o = \mathbf{T}^T\mathbf{W}_o\mathbf{T}$. The product $\hat{\mathbf{W}}_c\hat{\mathbf{W}}_o = \cdots = \mathbf{T}^{-1}\mathbf{W}_c\mathbf{W}_o\mathbf{T}$ has the same[34] eigenvalues as $\mathbf{W}_c\mathbf{W}_o$. To prove the second part of the theorem it suffices to provide an algorithm which will bring any given system as defined above into a balanced form. An algorithm that does this is given next.

[34] We note that also the individual Gramians are invariant in this sense, provided that the state transformation $\mathbf{x} = \mathbf{Tz}$ is orthogonal, i.e., $\mathbf{T}^T = \mathbf{T}^{-1}$. This is particularly important for 'symmetric state-space systems' with $\mathbf{W}_c = \mathbf{W}_o$, as the transformation $\mathbf{T}$ that brings such a system in the balanced form as defined above is necessarily orthogonal (indeed, the matrix $\mathbf{T}$ diagonalizing the symmetric $\mathbf{W}_c$ is the orthogonal matrix of its eigenvectors).

A Basic Balanced Realization Algorithm

If the stable system under consideration is minimal, it can be brought into a balanced form (assuming $\mathbf{x} = \mathbf{T}\hat{\mathbf{x}}$) by the following algorithm [168]:

$$\begin{aligned}
\mathbf{W}_c &= \mathbf{R}\mathbf{R}^T \quad \text{(Cholesky)} && (4.4.12)\\
\mathbf{R}\mathbf{W}_o\mathbf{R}^T &= \mathbf{U}\boldsymbol{\Sigma}^2\mathbf{U}^T \quad \text{(SVD) (actually symmetric EVP)} && (4.4.13)\\
\mathbf{T} &= \mathbf{R}^T\mathbf{U}\boldsymbol{\Sigma}^{-\frac{1}{2}} && (4.4.14)\\
\Rightarrow \quad \hat{\mathbf{W}}_c &= \mathbf{T}^{-1}\mathbf{W}_c\mathbf{T}^{-T} = \cdots = \boldsymbol{\Sigma} && (4.4.15)\\
\hat{\mathbf{W}}_o &= \mathbf{T}^T\mathbf{W}_o\mathbf{T} = \cdots = \boldsymbol{\Sigma}, && (4.4.16)
\end{aligned}$$

where $\mathbf{R}$ is the positive-definite lower triangular matrix.

A similar algorithm can, of course, be set using the Cholesky decomposition of the observability Gramian and a symmetric EVP involving its controllability counterpart.

Balanced Truncation

Let $\mathbf{T}$ be a non-singular state-transformation, $\mathbf{x} = \mathbf{T}\mathbf{z}$, that brings the system into a balanced form with system matrices ($\mathbf{A}_{bal}$,$\mathbf{B}_{bal}$,$\mathbf{C}_{bal}$). Then, in the $\mathbf{z}$-coordinates, $L_c(\mathbf{z}_0)\frac{1}{2} = \mathbf{z}_0^T\Sigma^{-1}\mathbf{z}_0$ and $L_o(\mathbf{z}_0) = \frac{1}{2}\mathbf{z}_0^T\Sigma\mathbf{z}_0$, respectively. For small σ_i the required control energy to reach the state $\mathbf{z} = (0,\ldots,0,z_i,0,\ldots,o)^T$ is large while the output energy generated by this state is small. Hence, assuming z_i's are already ordered such that the singular values σ's satisfy, for all $i = 1,\ldots,n-1$, $\sigma_i \geq \sigma_{i+1} > 0$, if $\sigma_k >> \sigma_{k+1}$, then from the above energy point of view the state components z_{k+1} to z_n are much less important and may be removed to reduce the number of state components of the model.[35] If the system is partitioned correspondingly as

$$\begin{aligned}
\mathbf{A}_{bal} &= \begin{bmatrix} \mathbf{A}_{bal,11} & \mathbf{A}_{bal,12} \\ \mathbf{A}_{bal,21} & \mathbf{A}_{bal,22} \end{bmatrix}, \quad \mathbf{B}_{bal} = \begin{bmatrix} \mathbf{B}_{bal,1} \\ \mathbf{B}_{bal,2} \end{bmatrix}, \quad \mathbf{C}_{bal} = \begin{bmatrix} \mathbf{C}_{bal,1} & \mathbf{C}_{bal,2} \end{bmatrix}\\
\mathbf{z}^1 &= \begin{bmatrix} z_1 & \ldots & z_k \end{bmatrix}^T, \quad \mathbf{z}^2 = \begin{bmatrix} z_{k+1} & \ldots & z_n \end{bmatrix}^T, \quad \boldsymbol{\Sigma}_1 = \begin{bmatrix} \boldsymbol{\Sigma}_1 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_2 \end{bmatrix},
\end{aligned} \tag{4.4.17}$$

where $\boldsymbol{\Sigma}_1 = diag(\sigma_1,\ldots,\sigma_k)$ and $\boldsymbol{\Sigma}_2 = diag(\sigma_{k+1},\ldots,\sigma_n)$, then both subsystems ($\mathbf{A}_{bal,ii}$,$\mathbf{B}_{bal,i}$,$\mathbf{C}_{bal,i}$), $i = 1,2$, are again in balanced form, and their controllability and observability gramians are equal to $\boldsymbol{\Sigma}_i$, $i = 1,2$. Furthermore, if $\sigma_k \neq \sigma_{k+1}$, then both subsystems are asymptotically stable [214] (otherwise stable for almost all $\mathbf{T}$). It is important to stress that BTR generally does not preserve the poles of the original system due to non-zero $\mathbf{A}_{bal,12}$ and $\mathbf{A}_{bal,21}$ (dynamic coupling of retained and truncated subsystem).

[35] For an alternative criterion of choosing 'important' balanced modes, based on their contribution to the impulse response approximation of the original system, see [70].

The subsystem ($\mathbf{A}_{bal,11}$,$\mathbf{B}_{bal,1}$,$\mathbf{C}_{bal,1}$) can be used as an approximation of the full order system. The performance of the standard balanced truncation is characterized by the following error bound [107].

Theorem 4.4.2. *Let $\boldsymbol{H}(s)$ be the transfer function matrix of the full-order stable LTI system ($\boldsymbol{A}$,$\boldsymbol{B}$,$\boldsymbol{C}$), i.e. $\boldsymbol{H}(s) = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B}$, and $\boldsymbol{H}_{red}(s) = \boldsymbol{C}_{bal,1}(s\boldsymbol{I} - \boldsymbol{A}_{bal,11})^{-1}\boldsymbol{B}_{bal,1}$ be the transfer function matrix of the reduced-order system ($\boldsymbol{A}_{bal,11}$,$\boldsymbol{B}_{bal,1}$,$\boldsymbol{C}_{bal,1}$). Then*

$$\|\boldsymbol{H} - \boldsymbol{H}_{red}\|_{\mathcal{H}} \leq \|\boldsymbol{H} - \boldsymbol{H}_{red}\|_{\infty} \leq 2\Sigma_{i=k+1}^{n}\sigma_i(\mathcal{H}). \tag{4.4.18}$$

In practice, the actual $\infty-$norm of model reduction error is generally much smaller (i.e., the upper bound on it is frequently loose), whereas the error in the Hankel norm is often a relatively tight lower bound on the ∞-error. Finally, in some special cases BTR can be shown to actually be optimal in the Hankel norm [191, 192]). The conditions the system under consideration needs to satisfy in these cases are, at least in this thesis, too restrictive to be of practical use.

Remark 4.4.2. The Lyapunov matrix equations are a special symmetric variant of the more general Silvester equations $\mathbf{AX} + \mathbf{XF} + \mathbf{G} = \mathbf{0}$, where $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{m\times m}$, and $\mathbf{G}, \mathbf{X} \in \mathbb{R}^{n\times m}$. Setting $\mathbf{F} = \mathbf{A}^T$ yields a Lyapunov matrix equation. The so-called *Cross-Gramian* approach, a slightly modified balancing approach for a special class of systems with the same number of inputs and outputs, is based on a single Silvester equation with $\mathbf{F} = \mathbf{A}$; see, e.g., [10].

Remark 4.4.3. Due to the linearity of Lyapunov equations, $\mathbf{W}_c$ can be written as $\mathbf{W}_c = \sum_{i=1}^{m}\mathbf{W}_{c,i}$, where $\mathbf{W}_{c,i}$ is the solution of $\mathbf{AW}_{c,i} + \mathbf{W}_{c,i}\mathbf{A} = -\mathbf{b}_i\mathbf{b}_i^T$, and $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m]$. This offers an ideal parallelization on a parallel computer with m procesors, as the m Lyapunov equations with rank one right hand sides can be solved simultaneously. Moreover, this fact may be useful also on a sequential computer when the solution needs to be updated due to a change in the input matrix $\mathbf{B}$, e.g. when a new flow-rate constrained well, $m+1$, is added. Recalling that the system matrix $\mathbf{A}$ in that case does not change, the new solution can then be computed as the sum of the solution for the existing configuration and the solution of $\mathbf{AW}_{c,m+i} + \mathbf{W}_{c,m+i}\mathbf{A} = -\mathbf{b}_{m+i}\mathbf{b}_{m+i}^T$. Upscaling as proposed in [268, 269] is one such potential application.

Balanced Residualization

Instead of completely removing a part of the state-space from the system description, balanced realization can also be combined with the (zero-order) singular perturbation approach introduced in Section 4.2. For instance, in order to ensure exact steady-state gain and a good approximation at low (and often also at medium) frequencies, the reduced-order model is derived by setting the time-derivative of the balanced modes corresponding to small Hankel singular values to zero, i.e., for $\mathbf{z}^2$ in 4.4.17, we set $\frac{d\mathbf{z}^2}{dt} = \mathbf{0}$. The reduced-order model so obtained enjoys the

same error bound 4.4.18 ([173]). For details regarding this basic form of balanced residualization and its generalizations see [88, 94, 95, 173, 279].

Example 4.4.3. Superior performance of balanced truncation over modal truncation for single-phase reservoir systems with wells has already been mentioned in Remark 4.2.1. There, BTR was applied to a model of order 292 obtained by an initial model order reduction of the original model of order 441 using modal truncation. In the current example, the balanced truncation as presented above was applied directly to the 441 states of the large reservoir model. The results, showing an excellent performance of the truncated model of order 15, are presented in Figure 4.13. Actually, for this particular system any order of the reduced model higher than (say) 10 turned out to perform very satisfactory. According to the above discussion, for the low-frequency piecewise constant input well performing reduced models of even lower order can be obtained using balanced residualization. Figure shows this for the reduced model of order 2. The approximation error is mainly caused at the points where the input abruptly switches from one constant value to another.

These favorable results are in accordance with the above 'energetic point of view' expressed in the form of the HSVs of the system. Figure 4.15 shows the HSVs (on a logarithmic scale) of the original reservoir model. The rapid decline of the HSVs indicates that the forward 441^{th} order single-phase reservoir model from the input-output point of view actually behaves as a model of much lower order. We stress hereby that, for this particular example, the HSVs would show a similar decline also for finer resolutions of the spatial discretization of the underlying infinite-dimensional model, hence confirming that the 'intrinsic' dimensionality, from the input-output point of view, of the pressure dynamics of single-phase reservoirs is generally rather low. Actually, a very steep drop of the HSVs for this class of systems is 'justified' also from a physical point of view, as the system is purely diffusive[36]. Moreover, we note that the conclusion about the fast decay rate of the HSV is in line with the general analysis provided in [10], but also with the state-moment analysis provided earlier in this chapter. We note hereby that, due to the intrinsic symmetry of the systems considered in this thesis, it is sufficient to observe the behavior of the eigenvalues of any of the Gramians $\mathbf{W}_c$ or $\mathbf{W}_o$ instead of their product. Indeed, in the symmetrized case these two are equal[37], so $\sigma_i^2 = \lambda_i(\mathbf{W}_c\mathbf{W}_o) = \lambda_i(\mathbf{W}_c^2) = \lambda_i^2(\mathbf{W}_c) \Rightarrow \sigma_i = \lambda_i(\mathbf{W}_c)$. This "approximate" low-rank property of the Gramians can be exploited to find approximate solutions of the corresponding Lyapunov equations. Details on this aspect are provided next.

[36] For the same reason, any (linear) model with a dominant parabolic term may be expected to show such behavior of HSVs.

[37] In the starting model, the Gramians are related by $\mathbf{W}_o = \mathbf{V}\mathbf{W}_c\mathbf{V}$, which follows directly from the fact that the symmetrizing state transformation, as mentioned earlier, is $\mathbf{x} = \mathbf{V}^{-\frac{1}{2}}\mathbf{z}$. Hence, the HSVs of the original model can again be computed using one Gramian, e.g. $\sigma_i^2 = \lambda_i(\mathbf{W}_c\mathbf{W}_o) = \lambda_i((\mathbf{W}_c\mathbf{V})^2) = \lambda_i^2(\mathbf{W}_c\mathbf{V}) \Rightarrow \sigma_i = \lambda_i(\mathbf{W}_c\mathbf{V})$.

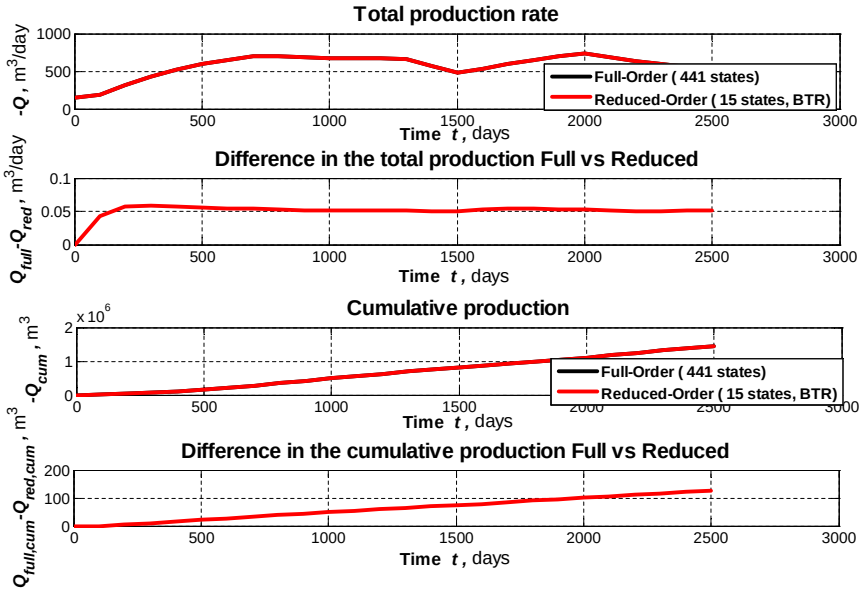


Figure 4.13: Performance of the reduced model of order 15 obtained by balanced truncation technique as presented above (i.e., keeping only the balanced modes corresponding tot the 15 largest HSVs of the original system of order 441.)

4.4.3 Algorithms for exact and approximate solutions of the system Gramians

(In the sequel of this section we will consider the controllability Gramians only. Results for its dual, the observability Gramian, are similar.)

A trivial, but computationally generally not attractive, direct approach to compute $\mathbf{W}_c$ would be by numerical integration of the infinite integral (4.4.7). Another obvious choice is to solve the corresponding Lyapunov matrix equation[38] (4.4.9): $\mathbf{A}\mathbf{W}_c + \mathbf{W}_c\mathbf{A}^T + \mathbf{B}\mathbf{B}^T = \mathbf{0}$. Early methods for exact computation of the solution of Lyapunov and related (i.e., Silvester, Riccati) matrix equations were based on Kronecker matrix products, eigenvalue decompositions [39], Jordan canonical transformation of $\mathbf{A}$, infinite series solution, etc. For more details on these methods see

[38] Lyapunov matrix equations as the ones considered here are said to be *large-scale* already for $n > 1000$, the reason being that they are mathematically equivalent to linear systems of equations with n^2 unknowns. A good summary of the theoretic properties of the Lyapunov equation is provided in [166], for instance.

[39] E.g., eigenvalue decomposition of $\mathcal{A} := (\mathbf{A}^T, \mathbf{Q}; \mathbf{0}, -\mathbf{A})$. The solution $\mathbf{W}_c$ is then given by $\mathbf{W}_c = \mathbf{M}_{11}\mathbf{M}_{21}^{-1}$; or $\mathbf{W}_c = \mathbf{M}_{11}\mathbf{M}_{21}^{-1}$ if $\mathbf{M}_{21}$ is badly conditioned, whereby $(\mathbf{M}_{11}, \mathbf{M}_{12}; \mathbf{M}_{21}, \mathbf{M}_{22})$ is the eigenvector matrix of $\mathcal{A}$. See [38], for instance.

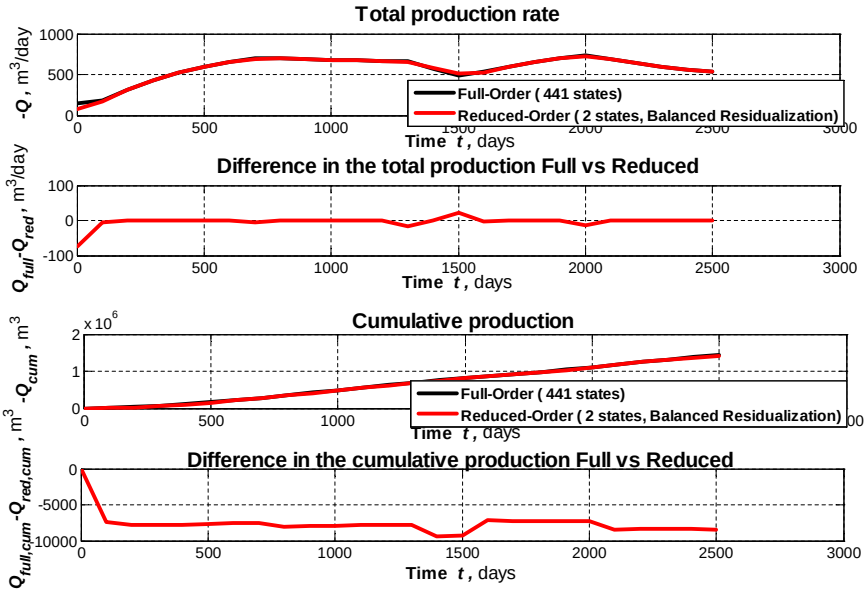


Figure 4.14: Performance of the reduced model of order 2 obtained by balanced residualization.

[31, 126] and references therein, for instance. Among all direct methods for solving Lyapunov equations perhaps the most efficient and widely used algorithm is the Bartels-Stewart (B-S) algorithm (in Matlab implemented as function *lyap*), which main idea is to apply the Schur decomposition of matrix $\mathbf{A}$ to transform the Lyapunov equation into a triangular system which can be solved efficiently by forward or backward substitutions.[40] In particular, for the controllability Lyapunov equation (4.4.9), based on transforming (by QR-factorization) the system matrix $\mathbf{A}$ into a real Schur form: $\mathbf{H} = \mathbf{U}\mathbf{A}\mathbf{U}^T$, where $\mathbf{U}$ is orthogonal and $\mathbf{H}$ is upper-triangular, and the symmetric update $\widetilde{\mathbf{B}}\widetilde{\mathbf{B}}^T \leftarrow (\mathbf{U}\mathbf{B})(\mathbf{U}\mathbf{B})^T$, 4.4.9 transforms into $\mathbf{H}\widetilde{\mathbf{W}}_c + \widetilde{\mathbf{W}}_c\mathbf{H} = -\widetilde{\mathbf{B}}\widetilde{\mathbf{B}}^T$, which is easily solved for $\widetilde{\mathbf{W}}_c$ by back-substitution. The original $\mathbf{W}_c$ and $\widetilde{\mathbf{W}}_c$ are related by $\mathbf{W}_c = \mathbf{U}^T\widetilde{\mathbf{W}}_c\mathbf{U}$. The improvement of the B-S algorithm for Lyapunov equations by Hammarling in [120] consists of solving directly for the Cholesky (or more generally, 'square-root') factor $\mathbf{R}$ of $\mathbf{W}_c = \mathbf{R}\mathbf{R}^T$ (in Matlab it is implemented as function *lyapchol*) without explicitly 'squaring' the data (e.g., multiplying $\mathbf{B}\mathbf{B}^T$). This has many advantages, in particular increased numerical stability, which is especially important for large systems where, as men-

[40] The original Bartels-Stewart algorithm [33] was developed to solve the more general Silvester equation $\mathbf{A}\mathbf{X} + \mathbf{X}\mathbf{F} = \mathbf{G}$, for $\mathbf{X}$, whereby both $\mathbf{A}$ and $\mathbf{F}$ are transformed in a real Schur form. For an extension of these algorithms to generalized Lyapunov equations, that is, those corresponding to the generalized state-space systems, see [187].

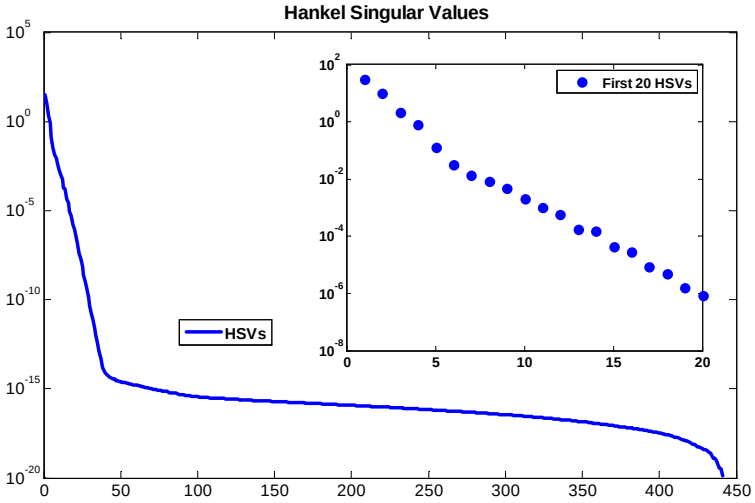


Figure 4.15: Hankel singular values (depicted on a logarithmic scale) of the 441 states single-phase example. Machine ϵ is approximately 10^{-16}.

tioned above, the Gramians (and hence also their product) are usually very badly conditioned[41]. Moreover, within a Krylov subspace or ADI method for solving Lyapunov equations typically the square-root factor of the solution and not the solution itself is needed; see, e.g., [171] (see also below).

Low-rank approximate solutions

Algorithms based on Schur decomposition have become the accepted method of solving Lyapunov equations in the small to moderate dense case. For larger dense problems, the so-called 'matrix sign function' method [40–42, 54, 98, 153, 167, 222, 229, 257], which requires only basic linear algebra operations such as matrix inversion (or solution of linear systems), is more appropriate for parallelization. However, these, and the other methods mentioned above, all have the computational and/or storage costs unacceptable for solving large and sparse Lyapunov (Silvester, Riccati, etc.) equations (although, see below for some new developments regarding the matrix sign function method). For example, a Schur decomposition results in $O(n^3)$ arithmetic operations and $O(n^2)$ storage. Furthermore, Schur decompositions of sparse matrices will be dense in general, since their orthonormal transform matrices are. Moreover, also $\mathbf{W}_c$ and $\widetilde{\mathbf{W}}_c$ will generally be dense, even

[41] For this reason BTR presented above requiring balancing the whole system followed by truncation as in the basic algorithm (4.4.12-4.4.14), is often numerically inefficient and ill-conditioned as it involves inversion of the diagonal matrix of (the square roots of the) HSVs, $\mathbf{\Sigma}^{-\frac{1}{2}}$. There are numerically better conditioned (square-root) algorithms that produce either the same truncated balanced models or truncated models similar to the balanced ones up to a specific transformation (e.g., up to a diagonal scaling or a triangular transformation). See, e.g., [10] (Ch. 7) and the references in there.

when $\mathbf{A}$ is diagonal, thus requiring excessive storage. Besides the large storage needs and CPU times, these methods also suffer from numerical instability.

Fortunately, for models of real systems, in particular diffusive as discussed above, the solutions of Lyapunov equations often appear to have a low numerical rank[42] [14, 212, 213, 266]. Since the rank of a product of matrices is equal to or less than the rank of the matrix with lowest rank in the product, this in its turn implies that the product $\mathbf{W}_c\mathbf{W}_o$ is also low-rank whenever either $\mathbf{W}_c$ or $\mathbf{W}_c$ is. On the other hand, from model reduction point of view, a rapid decay of the systems HSVs showing a rapid decay, as is the case our examples, implies that the the I/O energy coupling is dominated by just a few states, thus also implying a low numerical rank of (at least one of) the Gramians. This fact has led to development of a variety of methods for determining approximate solutions of these system quantities. The following two categories have been prevalent so far[43]:

- **Krylov subspace methods.** Any approximation (low-rank or not) of the exact solution of a Lyapunov equation will yield an 'equation residual'. In particular, if $\hat{\mathbf{W}}_c$ and $\hat{\mathbf{W}}_o$ are approximations of the exact (unique) solutions of the controllability and observability Lyapunov equations $(4.4.9)$ and $(4.4.10)$, respectively, then the following is true:

  $$\mathbf{A}\hat{\mathbf{W}}_c + \hat{\mathbf{W}}_c\mathbf{A}^T + \mathbf{B}\mathbf{B}^T = \mathbf{R}_c \neq \mathbf{0} \quad \text{and} \quad \mathbf{A}^T\hat{\mathbf{W}}_o + \hat{\mathbf{W}}_o\mathbf{A} + \mathbf{C}^T\mathbf{C} = \mathbf{R}_0 \neq \mathbf{0}.$$

  The main idea of Krylov subspace methods for approximating $\mathbf{W}_c$ and $\mathbf{W}_o$ is to force the residuals $\mathbf{R}_c$ and $\mathbf{R}_o$ to satisfy well-defined conditions, usually of Galerkin type, involving low-rank projection matrices $\mathbf{W}$ and $\mathbf{V}$ from our usual projection-based formulation of reduced-order models, i.e., for a system in the standard state-space form, $\mathbf{x}_r = (\mathbf{W}\mathbf{V})^{-1}\mathbf{W}^T\mathbf{A}\mathbf{V}\mathbf{x}_r + (\mathbf{W}\mathbf{V})^{-1}\mathbf{W}^T\mathbf{B}\mathbf{u}$, $\tilde{\mathbf{y}} = \mathbf{C}\mathbf{V}\mathbf{x}_r + \mathbf{D}\mathbf{u} = \mathbf{C}_r\mathbf{x}_r + \mathbf{D}\mathbf{u}$. The approximate Gramians $\hat{\mathbf{W}}_c$ and $\hat{\mathbf{W}}_o$ are then defined as projections, on well-defined subspaces (again) involving $\mathbf{W}$ and $\mathbf{V}$, of the exact solutions, $\mathbf{W}_{c,r}$ and $\mathbf{W}_{o,r}$, of the reduced-order Lyapunov equations $\mathbf{A}_r\mathbf{W}_{c,r} + \mathbf{W}_{c,r}\mathbf{A}_r^T = -\mathbf{B}_r\mathbf{B}_r^T$ and $\mathbf{A}_r^T\mathbf{W}_{o,r} + \mathbf{W}_{o,r}\mathbf{A}_r = -\mathbf{C}_r^T\mathbf{C}_r$, respectively. For instance, in [238], the Galerkin type conditions for the residuals $\mathbf{R}_c$ and $\mathbf{R}_o$ are chosen to be $(\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T\mathbf{R}_c\mathbf{W}(\mathbf{W}^T\mathbf{V})^{-T}$ and $\mathbf{V}^T\mathbf{R}_o\mathbf{V}$, respectively. Gramians approximations $\hat{\mathbf{W}}_c$ and $\hat{\mathbf{W}}_o$ which satisfy these conditions are respectively be defined to be $\hat{\mathbf{W}}_c := \mathbf{V}\mathbf{W}_{c,r}\mathbf{V}^T$ and $\hat{\mathbf{W}}_o := \mathbf{W}(\mathbf{W}^T\mathbf{V})^{-T}\mathbf{W}_{o,r}(\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T$. $\mathbf{V}$ and $\mathbf{W}$ are chosen as (orthonormal) bases of Krylov subspaces of $\mathcal{K}_k(\mathbf{A}^{-1}, \mathbf{A}^{-1}\mathbf{B})$ and $\mathcal{K}_k(\mathbf{A}^{-T}, \mathbf{A}^{-T}\mathbf{C}^T)$, in which, as we already know by now, the small eigenvalues of $\mathbf{A}$ are dominant (subspaces $\mathcal{K}_k(\mathbf{A}, \mathbf{B})$ and $\mathcal{K}_k(\mathbf{A}^T, \mathbf{C}^T)$ emphasizing large eigenvalues

[42] Note that this, however, might not be the case when the number of inputs (outputs) is substantial, i.e., when $\mathbf{B}$ ($\mathbf{C}$) itself is not low rank. For dynamic systems, where the relevant Gramian spans the reachable (observable) subspace, this is also intuitive: the larger rank of $\mathbf{B}$ the more independent inputs (outputs) to cover a larger portion of the corresponding subspace.

[43] For some new developments employing so-called 'hierarchical' matrices ($\mathcal{H}$-matrices) for the use in the matrix sign method see [35, 36, 111–113]. The $\mathcal{H}$-matrix format allows data-sparse approximation for a large, practically relevant class of matrices arising (for instance) from spatial discretisation methods.

of **A** are used in [143, 236], for instance). As observed in [238], this choice leads to a good approximation of the time functions in the explicit solutions of the Lyapunov equations for *large* values of t.

After obtaining the approximate Gramians, these can be used for balanced truncation, whereby the Cholesky factorization of $\hat{\mathbf{W}}_c$ and $\hat{\mathbf{W}}_o$ for square-root implementation of BTR can be accomplished by factorizing the reduced-order matrices $\mathbf{W}_{c,r}$ and $\mathbf{W}_{c,r}$; see [238] for details.

Remark 4.4.4. HSVs of the original high-order model can thus be approximated using the approximate Gramians by

$$\sigma_i^2 = \lambda_i(\mathbf{W}_c\mathbf{W}_o) \approx \lambda_i(\hat{\mathbf{W}}_c\hat{\mathbf{W}}_o) = \cdots = \lambda_i(\mathbf{V}\mathbf{W}_{c,r}\mathbf{W}_{o,r}(\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T) \quad (4.4.19)$$

By changing the sequence of matrix multiplication, eigenvalues of $\mathbf{V}\mathbf{W}_{c,r}\mathbf{W}_{o,r}$ $(\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T$ and $(\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T\mathbf{V}\mathbf{W}_{c,r}\mathbf{W}_{o,r} \equiv \mathbf{W}_{c,r}\mathbf{W}_{o,r}$ are equal. Hence, k HSVs of the original high-order system are approximated by the HSVs of the reduced-order system in $\mathbf{x}_r$. The choice for Krylov subspaces $\mathcal{K}_k(\mathbf{A}^{-1}, \mathbf{A}^{-1}\mathbf{B})$ and $\mathcal{K}_k(\mathbf{A}^{-T}, \mathbf{A}^{-T}\mathbf{C}^T)$ turns out to lead to good approximations of the largest HSVs of the original model, whereby the quality of the approximation will generally increase with increase of the number of iterations to calculate $\mathbf{V}$ and $\mathbf{W}$ (i.e. larger k). We note that this fact may be extremely useful in certain important applications, one of which being the upscaling as proposed in [268, 269]. In that methodology, the upscaled reservoir parameters are determined by minimizing cost functions involving either the differences between a certain number of individual (largest) HSVs or the difference between the total sums[44] of (squares of) all HSVs of the original high-order

[44]Regarding the total sum of (squares of) all HSVs, following Remark 6.3.3 in [10], we note the following important result for our single-phase systems with stable symmetric matrix **A**:

$$trace(\mathbf{W}_c) = -\frac{1}{2}trace(\mathbf{B}^T\mathbf{A}^{-1}\mathbf{B}) \quad , \quad trace(\mathbf{W}_o) = -\frac{1}{2}trace(\mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T). \quad (4.4.20)$$

For symmetric state-space systems, for which holds $\mathbf{W}_c = \mathbf{W}_o$, we thus have: $\sum_{i=1}^n \sigma_i = \sum_{i=1}^n \lambda_i^{\frac{1}{2}}(\mathbf{W}_c\mathbf{W}_o) = \sum_{i=1}^n \lambda_i(\mathbf{W}_c) \equiv trace(\mathbf{W}_c) = -\frac{1}{2}trace(\mathbf{C}\mathbf{A}^{-1}\mathbf{B})$ $(\equiv \frac{1}{2}trace(\mathbf{M}_0 - \mathbf{D})$, where $\mathbf{M}_0 = -\mathbf{C}\mathbf{A}^{-1}\mathbf{B} + \mathbf{D}$ is the 0^{th} system moment at $s = 0$). For our starting, 'almost symmetrized' models, with the symmetrizing transformation $\mathbf{x} = \mathbf{V}^{-\frac{1}{2}}\mathbf{z}$ ($\mathbf{V}$ being the accumulation matrix), it holds that $\sum_i^n \sigma_i = \sum_i^n \lambda_i(\mathbf{W}_c\mathbf{V}) = trace(\mathbf{W}_c\mathbf{V})$ $(\equiv trace(\mathbf{V}\mathbf{W}_c) \equiv trace(\mathbf{V}^{\frac{1}{2}}\mathbf{W}_c\mathbf{V}^{\frac{1}{2}}))$, it is not difficult to prove that $\sum_{i=1}^n \sigma_i = -\frac{1}{2}trace(\mathbf{B}^T\mathbf{V}\mathbf{A}^{-1}\mathbf{B})$ $(\equiv -\frac{1}{2}trace(\mathbf{B}^T\mathbf{V}^{\frac{1}{2}}\mathbf{A}^{-1}\mathbf{V}^{\frac{1}{2}}\mathbf{B}))$. Therefore, the computational and storage complexity of determining the sum of all HSVs of the system has decreased from solving the full Lyapunov controllability equation for $n \times n$ dense (and, most likely, ill-conditioned) $\mathbf{W}_c$ to practically a single sparse LU decomposition! Moreover, if the well-configuration is changed, the sum of the HSVs for the new full-order model is easily computed by a simple adjustment of **B** if only flow-rate constrained wells are added or removed, and, if also BHP-constrained wells are added or removed, by a low-rank update of the LU factorization (or by employing the Sherman-Morrison-Woodbury formula for a low-rank update of the inverse of **A**, if the original $\mathbf{A}^{-1}$ is available). N.B. The above expressions suggest that the accumulation matrix $\mathbf{V}$ influences the result. However, using the fact that, here, $\mathbf{A} = -\mathbf{V}^{-1}\mathbf{T}$, with $\mathbf{T}$ the transmissibility matrix (the contribution from the BHP-constrained wells incorporated), and $\mathbf{B} = \mathbf{V}^{-1}\mathbf{F}$, with $\mathbf{F}$ the input 'selector' consisting of zeros everywhere except at places corresponding to flow-rate constrained wells (where it has '1's) and BHP-constrained wells (where it has values of the corresponding well indices), the sum of HSVs is given by $\sum_{i=1}^n \sigma_i = \frac{1}{2}(\mathbf{F}^T\mathbf{T}^{-1}\mathbf{F})$ (note that this again equals a half of the transfer function matrix at $s = 0$).

model and the upscaled model (the latter initially being an existing upscaled model obtained by any conventional upscaling method).

- **Low-Rank Iterative Approximations.** The underlying idea in these methods is to consider the solution of a discrete-time system that has the same Gramians as the Gramians of the continuous-time system under investigation; this allows for computing the Gramians using a series instead of integrals. *Continuous-to-discrete-time* system transformation can e.g. be accomplished by the bilinear transformation of the Laplace variable $s = p\frac{1-z}{1+z}$, where $p < 0$ is a 'shift-parameter' to be chosen. The discrete-time system matrices $\mathbf{A}_p$ and $\mathbf{B}_p$, for instance, are then given by:

$$\mathbf{A}_p = (p\mathbf{I} + \mathbf{A})^{-1}(p\mathbf{I} - \mathbf{A}) \tag{4.4.21}$$

$$\mathbf{B}_p = \sqrt{-2p}(p\mathbf{I} + \mathbf{A})^{-1}\mathbf{B}, \tag{4.4.22}$$

and define the discrete-time Lyapunov equation[45]:

$$\mathbf{A}_p\mathbf{P}\mathbf{A}_p^T - \mathbf{P} + \mathbf{B}_p\mathbf{B}_p^T = \mathbf{0}, \tag{4.4.23}$$

which has the same solution for the Gramian as the original continuous-time Lyapunov equation (4.4.9), i.e., $\mathbf{P} = \mathbf{W}_c$. The solution is expressed as *infinite sum*:

$$\mathbf{P} \quad = \quad \sum_{j=0}^{\infty} \mathbf{A}_p^j \mathbf{B}_p \mathbf{B}_p^T \mathbf{A}_p^{Tj}. \tag{4.4.24}$$

Consider a k^{th} order approximation for $\mathbf{P}$:

$$\mathbf{P}_k \quad = \quad \sum_{j=0}^{k-1} \mathbf{A}_p^j \mathbf{B}_p \mathbf{B}_p^T \mathbf{A}_p^{Tj}. \tag{4.4.25}$$

Various methods exist to compute the approximation iteratively, the basic form of the iterative process being (**Smith method** [212, 252]):

$$\mathbf{P}_j^{Smith} \quad = \quad \mathbf{A}_p\mathbf{P}_{j-1}^{Smith}\mathbf{A}_p^T + \mathbf{B}_p\mathbf{B}_p^T; \quad \mathbf{P}_0^{Smith} = \mathbf{0}. \tag{4.4.26}$$

Since $\mathbf{A}$ is stable, all eigenvalues of $\mathbf{A}_p$ are within the unit circle and the sequence $\{\mathbf{P}_j\}_{j=0}^{\infty}$ generated by this process converges to $\mathbf{P}$.

ADI iteration (Alternating Direction-Implicit) algorithm[46] [10, 116, 288, 289] is a generalization of the Smith method by using distinct shift parameters

[45] A special case of the Stein equation $\mathbf{APB} - \mathbf{P} + \mathbf{Q} = \mathbf{0}$.

[46] The ADI iteration was first introduced in [210] as a method for solving systems arising from the discretization of elliptic and parabolic boundary value problems. In the iterative process, the solution iterates are usually generated by the solution of two linear systems with multiple right-hand sides, using different shifts $p_1, p_2, \ldots$ and usually starting from a zero initial iterate [80, 81, 92]. When applied to Lyapunov equations the two equations turn out to be mathematically equivalent to the single iteration step (4.4.27); see, e.g., [10, 116].

$p_1, p_2, \ldots, p_\ell$:

$$\mathbf{P}_j^{ADI} = \mathbf{A}_{p_j}\mathbf{P}_{j-1}^{ADI}\mathbf{A}_{p_j}^T + \mathbf{B}_{p_j}\mathbf{B}_{p_j}^T; \quad \mathbf{P}_0^{ADI} = \mathbf{0}, \tag{4.4.27}$$

where $\mathbf{A}_{p_j}, \mathbf{B}_{p_j}$ are the transformed state-space matrices with p replaced by the j^{th} shift parameter p_j (which now may be non-real; when $\mathbf{A}$ is symmetric they are in general real, though).

It was observed in [212] that, in general, a) the Smith method converges (much) slower than the ADI method, and b) a moderate increase of the number of shifts ℓ in the ADI process accelerates the convergence. However, there it was also observed that a further increase of ℓ hardly accelerates the convergence any further, which lead to the idea of **cyclic Smith(ℓ)** iteration, which basically is a special case of ADI where ℓ different shifts are used cyclically, that is, $p_{i+j\ell} = p_i$ for $j = 1, 2, \ldots$.

Low-rank iterations

In each of the above methods, full-size system Gramians $\mathbf{P}_j$ are propagated at each iteration, which makes their application to large-scale sparse systems very limited. In many cases, it is the storage requirement of $O(n^2)$ which is the limiting factor rather than the amount of computation. The remedy then is provided by the *low-rank* methods, which compute and store $n \times q$ approximate square-root factors, say $\mathbf{Z}$, of $\mathbf{P}$ instead of explicitly forming $\mathbf{P}$, where q is the numerical rank[47] of $\mathbf{P}$. The storage requirement is then reduced to $O(n \times q)$. The methods are conveniently named *Low-Rank ADI* (LR-ADI) and *Cyclic Low-Rank Smith* (LR-Smith(ℓ)). The key idea is thus to write

$$\mathbf{P}_j^{ADI} = \mathbf{Z}_j^{ADI}(\mathbf{Z}_j^{ADI})^T \quad \text{and} \quad \mathbf{P}_j^{Smith(\ell)} = \mathbf{Z}_j^{Smith(\ell)}(\mathbf{Z}_j^{Smith(\ell)})^T \tag{4.4.28}$$

and propagate the $\mathbf{Z}_j$ factors instead of $\mathbf{P}_j$. For the LR-ADI, for instance, substituting the above square-root expression for $\mathbf{P}_{j-1}^{ADI}$ into (4.4.27) yields the expression for $\mathbf{Z}_j^{ADI}$ in (4.4.28):

$$\mathbf{Z}_j^{ADI} = \left[\ \mathbf{A}_{p_j}\mathbf{Z}_{j-1}^{ADI} \quad , \quad \mathbf{B}_{p_j}\ \right], \tag{4.4.29}$$

where $\mathbf{Z}_1^{ADI} = \mathbf{B}_{p_1}$, and $\mathbf{A}_{p_i}$ and $\mathbf{B}_{p_i}$ are as defined in (4.4.21) and (4.4.22).

For the derivations of the square-root expressions of the LR-Smith(ℓ) method, which is a more efficient alternative to the LR-ADI method when the number of shift parameters is limited, and an in depth analysis of these low-rank methods, the reader is referred to [10, 39, 116, 212] and references therein. [10, 116] also propose modifications of the LR-Smith(ℓ) iteration for the cases where the r.h.s. of the Lyapunov equation is not low-rank (e.g., large number of inputs/outputs in the system) and/or slow convergence of the standard

[47] The approximate square-root factors are themselves full-rank. 'Low-rank' relates to the numerical rank of $\mathbf{P}$.

low-rank iterations.[48]

Remark 4.4.5. Clearly, a crucial issue in these iterative methods is the selection of suitable shift-parameters to accelerate the convergence. Fortunately, good heuristic algorithms for determining a (sub-)optimal set of these parameters exist [212, 288].

Remark 4.4.6. Iterative low-rank square-root Gramian approximations generally yield more accurate estimates of the dominant HSVs than Krylov subspace methods. Moreover, the latter ones tend to yield the reduced-models of unnecessarily high order. On the other side, they are simple and fast.

Approximate balanced truncation

Low-rank approximate square-root factors $\mathbf{Z}_c$ of $\mathbf{W}_c$ and $\mathbf{Z}_o$ of $\mathbf{W}_c$ can be used for approximate BTR using the algorithm given below. Again, due to $\mathbf{W}_o = \mathbf{V}\mathbf{W}_c\mathbf{V}$ for the systems in this thesis, to determine the projection matrices the knowledge of $\mathbf{Z}_c$ suffices.

Algorithm 1: Approximate Balanced Truncation (ABTR)

INPUT: Low-rank square-root factors $\mathbf{Z}_c, \mathbf{Z}_o \in \mathbb{R}^{n\times k}$ of $\mathbf{W}_c$ and $\mathbf{W}_o$, resp.

– Compute the SVD of $\mathbf{Z}_c^T\mathbf{Z}_o$:

$$\mathbf{Z}_c^T\mathbf{Z}_o = \mathbf{\Phi}\mathbf{\Sigma}\mathbf{\Psi}^T;$$

– Define:

$$\mathbf{V}_{proj} := \mathbf{Z}_c\mathbf{\Phi}\mathbf{\Sigma}^{-\frac{1}{2}}, \quad \text{and} \quad \mathbf{W}_{proj} := \mathbf{Z}_o\mathbf{\Psi}\mathbf{\Sigma}^{-\frac{1}{2}};$$

– The order-k approximated truncated balanced realization is given by:

$$\mathbf{A}_r = \mathbf{W}_{proj}^T\mathbf{A}\mathbf{V}_{proj}, \quad \mathbf{B}_r = \mathbf{W}_{proj}^T\mathbf{B}, \quad \mathbf{C}_r = \mathbf{C}\mathbf{V}_{proj}, \quad \mathbf{D}_r = \mathbf{D}.$$

Example 4.4.4. To demonstrate the performance of low-rank iteration methods in the approximation of the dominant subspace of the system Gramians, the Cyclic LR-ADI as coded in the LYAPACK[49] package was applied on the

[48] In these cases, at the i^{th} iterate, the number of columns of $\mathbf{Z}_i^{ADI}$ and $\mathbf{Z}_i^{Smith(\ell)}$ easily becomes too large from the storage point of view. For the controllability Lyapunov equation, for instance, the size of $\mathbf{Z}_i^{ADI}$ is $n \times (m \times i)$. The size of $\mathbf{Z}_i^{Smith(\ell)}$ is even larger (on the other hand, an i-step LR-Smith(ℓ) iteration requires less matrix factorizations than an i-step LR-ADI iteration). In the modified LR-Smith(ℓ) method proposed in [116], the number of columns in the low-rank square root factor does not increase unnecessarily at each iteration step. The method is based on the idea of computing, at each step, the SVD of the iterate and, given a tolerance τ, replacing the iterate with its best low rank approximation. The SVD is not recomputed, but instead updated after each step to include the new information and then truncated given the tolerance τ.

[49] LYAPACK is a MATLAB toolbox for solving certain large scale problems in control theory, which are closely related to Lyapunov equations. It employs iterative algorithms and is intended for large, sparse problems [211].

441-states single-phase model from Example 4.4.3. Figure 4.16 shows the results for the eigenvalues of the controllability Gramian. Due to the intrinsic symmetry in our examples, adequate approximation of the dominant subspace of $\mathbf{W}_c$ guarantees adequate approximation of that of $\mathbf{W}_o$, but also of the dominant HSVs as well as the BTR projection matrices. As the approximation in this case is clearly excellent, the reduced-order models obtained using the approximate square-root factors of the Gramians match those obtained by the 'exact' BTR almost perfectly, resulting in virtually indistinguishable simulation results.

For larger problems, say $n > 1200$, computing the exact HSVs was not feasible on our Pentium-IV machine with 4GB RAM, hence the only way of assessing the performance of the low-rank Gramian square-root approximation was to compare the simulation results between the full-order model and their low-rank square-root BTR approximants. Figure 4.17 shows the results for the 13200 states five-spot SPE10 models from Fig. 4.11.b.

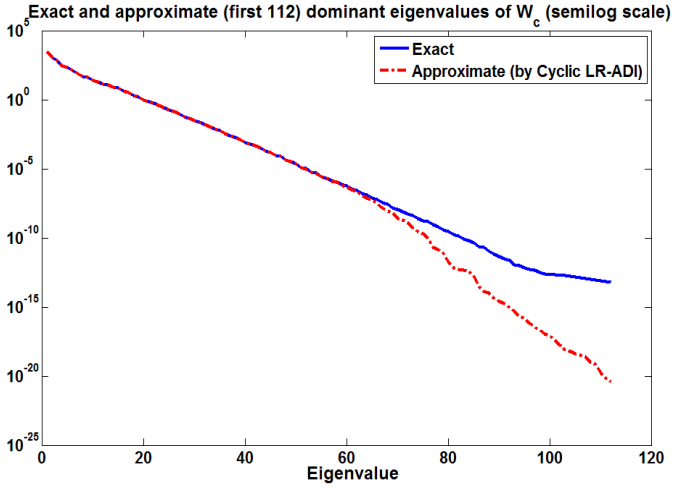


Figure 4.16: Comparison, for the 441 states system from Example 4.4.3, of the first 112 dominant eigenvalues of $\mathbf{W}_c$: Exact (blue '-') versus Cyclic LR-ADI (red '-.-').

Remark 4.4.7. It is fair to stress that for general, i.e. non-symmetric systems, accurate 'separate' low-rank dominant subspaces approximation of the controllability and observability Gramians does not guarantee an accurate approximation of the dominant subspaces of their product. Namely, the dominant eigenspaces of the controllability and observability Gramians may be very different from what is actually needed: the dominant spaces of their product. In fact, as nicely explained in [283], for instance, for some problems with innate asymmetry it may happen that the system has completely different most observable and most controllable states

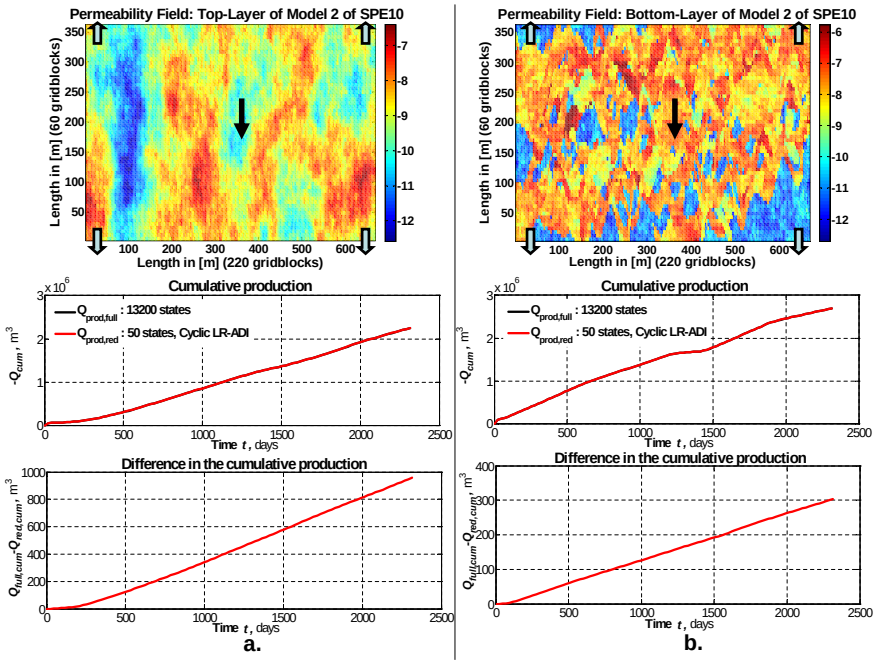


Figure 4.17: Comparison simulation results for full-order (13200 states) versus cyclic LR-ADI reduced-order (50 states) models for the five-spot SPE10 systems from Fig. 4.11.b. For both layers, the lines depicting the cumulative productions for the full- and the reduced-order model are on top of each other.

(in the most 'drastic' situation, the dominant subspaces of $\mathbf{W}_c$ and $\mathbf{W}_o$ would even be mutually exclusive). For such situations, the so-called 'Approximate Implicit Subspace Iteration with Alternate Directions' *AISIAD* method, proposed in [309], and its improvements presented in [283, 308], for instance, promise to be more reliable model reduction schemes as they approximate the dominant eigensubspaces of $\mathbf{W}_c\mathbf{W}_o$ directly. A different method (Sec. 5.1.3), which computes approximate balancing transformations directly (i.e., without separate reduction of the Gramians) using snapshots of the state solutions of the system and its (approximate) adjoint, is proposed in [233].

Remark 4.4.8. Another drawback associated with BTR using partial square-root Gramian factors is that the reduced-order model is not guaranteed to be balanced (recall that the exact square-root based BTR algorithms use full square-root (Cholesky) factors to obtain the final truncated balanced realization. When only dominant square-root factors are available, the reduced-order model may indeed not be balanced), having as a consequence that the global error bound (4.4.18) may not be strictly attained.

4.5 Summary and conclusions of the chapter

Modern system theoretic approaches for model order reduction (MOR) of linear time-invariant (LTI) state-space systems have been presented and their performance assessed when applied to the spatially discretized slightly compressible single-phase fluid-pressure equation. The techniques find their origins primarily in the well-established research fields of numerical system and control theory, mechanical and structural engineering and electric circuit analysis. In order to facilitate familiarization with the underlying theoretical and numerical aspects of the presented approaches, the introductory section of the chapter has provided a short introduction to linear systems theory, focusing on system-theoretical concepts as 'controllability', 'observability', Hankel operators and system norms, transfer function moments, etc. The methods presented in the subsequent sections were: Section 4.2: *Modal Truncation (MTR)*, *Singular Perturbation Approximation (SPA)* (or 'residualization'), Section 4.3: (explicit and implicit) *Transfer Function Moment Matching (TFMM)*, and Section 4.4: *Balanced Truncation (BTR)*.

All the methods, except the rezidualization, are projection-based, meaning that a macromodel of the original large-scale dynamical system is generated by projecting it onto some low-dimensional subspace (Chapter 3). The projection subspace may or may not be the same as the subspace the state variable itself is projected onto. In MTR, the two subspaces are generally equal and spanned by dominant eigenvectors of the system matrix. Indirect TFMM methods match a certain number of moments m_j of the system's transfer function (around one or multiple frequency points) and obtain their projection subspaces as certain Krylov subspaces involving the system matrices. It has been explained how such an indirect moment matching avoids the inherent ill-conditioning of the explicit moment matching. The conditioning of the explicit matching gets rapidly worse as the required number of the moments increases, which is a result of the fact that for increasing j, the moment m_j rapidly becomes a multiple of its predecessor m_{j-1} (hence no new information is added to the analysis); see Figure 4.9. Moment matching techniques are generally fast, but the size of the obtained reduced-order models may be unnecessarily large when the number of inputs and/or outputs (here, injector and/or producer wells) is not small. Finally, projection subspaces in BTR are obtained as the dominant eigenspaces of the product of the system's controllability and observability Gramians of the system being the solutions of the corresponding Lyapunov matrix equations. Due to the inherent (near) symmetry of the considered high-dimensional model in terms of controllability versus observability, the computational cost of the subspaces (and hence the reduced-order model) generation is factually halved. Further reduction of both the computational cost of implementing BTR as well as the order of the reduced model is allowed for by the existence of accurate low-rank (square-root) approximates of the Gramians, which (in this case) is a direct consequence of the purely diffusive nature of the high-dimensional model. Algorithms for obtaining low-rank solutions of the related Lyapunov equations using a) Krylov subspace methods and b) iterative approximations methods, have been presented in Subsection 4.4.3.

Regarding the performance of the presented methods, for piecewise constant

inputs (i.e. flow-rates and/or BHPs) with a suitable duration of the individual time segments, the shapes of the well control signals typically encountered in open-loop reservoir simulations, it has been either concluded or shown by numerical experiments that:

- reduced-order models obtained by ('zero-order') SPA, which essence is setting the time-derivative of a part of the system's state dynamics to zero and expressing the dynamics of the remaining part accordingly (Subsection 4.2.2), approximate the behavior of the high-dimensional model generally well. A direct consequence of setting state-derivatives to zero is that the method retains the steady state gain of the system, as zero state-derivative obviously represents steady state. On the contrary, mode truncation retains the systems behavior at infinite frequency. For this reason modal residualization is preferred for low frequency modeling, whereas mode truncation is preferable when accuracy is required at high frequencies.
- TFMM around the single frequency $s = 0$ is often sufficient for obtaining both accurate and very low-order models. An explicit Krylov-subspace method that uses an orthogonalized matrix of state block moments as the ('congruence') projection matrix has performed exceptionally well in a large majority of single-phase cases considered in this research using a rather low number of state block moments around $s = 0$ (page 62). Due to the absence of complex waveforms, the addition of only one extra moment improves the models considerably. Numerically, that is quite fortunate, as (the columns of) the state moments all start to line-up with the dominant system's eigenvector very quickly (Fig. 4.7 and 4.8). The symmetry of the systems also helps, as the same accuracy can be achieved using less moments than in a non-symmetric case.
- BTR has shown superior performance over MTR. The favorable performance of BTR has been attributed to being low-rank of both the controllability and the observability Gramians (and thus their product). The intrinsic dimensionality of an LTI dynamic system is directly related to the behavior of the Hankel singular values (HSVs) of the system, which are the square roots of the eigenvalues of the product of the controllability and observability Gramians. In accordance with this, the low-rank nature of the systems considered here are characterized by a rapid decline of the HSVs (see Figure 4.15; due to the abovementioned symmetry, it sufficed to consider the eigenvalues of the controllability Gramian instead of HSVs).

  For low-frequency piecewise constant inputs well performing reduced models of even lower order can be obtained using balanced SPA. The approximation error is then mainly caused at the points where the input abruptly switches from one constant value to another.

Regarding the eigenspectra of the considered high-dimensional single-phase systems, a general observation is that: a) for systems with only one BHP controlled well (connection) the separation of eigenvalues is clear, with a single closest-to-zero eigenvalue dominating over the remaining part of the (stable) eigenspectrum. The separation is larger the lower the permeability of the gridblock the

well is connected to, b) the number of dominant eigenvalues increases with the number of wells, but it is generally still very low compared to the state-space dimension (Fig. 4.6). According to Chapter 2, flow-rate controlled wells do not introduce any change in the system's dynamic matrix. Being based solely on the system matrix/pencil, MTR is therefore insensitive to both the number and the position of flow-rate controlled wells. On the other side, as any new well implies a new non-zero column in the input matrix **B**, controllability properties of the system do change (as do also observability properties, due to the symmetry). This change is less substantial the 'more large-scale' the original problem is, which has very important consequences for applications like optimization and parameter- or state estimation. In particular, it has been concluded that it is totaly unrealistic to expect that without prior knowledge (e.g. constraints imposed by 'known' geological properties) or use of another types of information (e.g. seismic), one may be able to obtain fine-scale parameter estimates (of permeability, for instance) that resemble the 'underlying, real' parameter field.

5 CHAPTER

MOR for Two-Phase Flow: Proper Orthogonal Decomposition

Section 3.3 has summarized a majority of the projection-based methods that can be found in the literature for addressing the problem of order reduction of large nonlinear models. In this thesis, because of the complexity and potentially very large size of multi-phase reservoir models, it was decided to employ data-driven projection subspaces for MOR of two-phase (waterflood) models. In these methods, the basis functions are determined by processing data obtained from numerical simulations of the underlying high-dimensional model and, as such, are expected to provide physical intuition to the actual behavior of the system.[1]

In particular, the applicability and performance of strategies based on the *Proper Orthogonal Decomposition (POD)* methodology have been assessed. A POD basis of a data set is orthonormal and it is optimal in that it guarantees the best reconstruction, in the mean square error sense, of the elements of the set among all (linear) bases of the same size. Depending on the area of application other names are used as well, e.g. Karhunen-Loeve (K-L) Transform, Empirical Orthogonal Functions (EOF), Principal Component Analysis [149, 156]: all these (and related, see [156] for an excellent overview) techniques aim for extraction of an optimal set of basis functions from a computational or experimental data base, by use of an eigenvalue analysis. Since its introduction by Karhunen in [152], and independently by Loeve in [174], as a statistical tool to analyze random process data, the POD method has been fruitfully applied to a diverse, and varied, collection of engineering problems, ranging from data analysis and compression to model order reduction. The method was first called/named POD by Lumley in [177], where it was used for study of turbulent flow by extraction and analysis of spatially coherent flow structures [58, 128, 205, 267]. It has proven to be useful also in image processing for purposes of pattern recognition, coding, classification

[1] A minor point of purely data-driven MOR techniques is the fact they do not respect the underlying model equations. Consequently, optimality of a particular basis in representing a data set used to determine the basis does not in any sense imply optimality of the reduced-order dynamic model the data has come from. For new developments involving, in a *goal-oriented* optimization framework, inclusion of model constraints in the determination of the projection bases, see [27, 50], for instance.

and compression [151, 156, 158, 170, 295], in optimization and control theory for designing reduced-order models and controllers [3, 16, 22, 23, 27, 29, 50, 125, 139–141, 180, 223], in inverse problems [30, 49, 206], and in many others. Applications to reservoir flow comprise references [57, 103, 251, 286] for deterministic flow, and [101, 102, 179, 296, 304] for stochastic porous media flow and uncertainty. Resulted from this work are references [123, 182, 184, 185, 272]. A major progress in the practical applicability of the method was made by Sirovich in [250] with the introduction of the 'method of snapshots', allowing for a considerably cheaper evaluation of the POD eigenvalue problem than the direct computation.

5.1 POD in Finite-Dimensional Setting

The mathematical theory of POD relies upon results from functional analysis, in particular the properties of Hilbert (or more generally, Sobolev) spaces, i.e. complete inner product spaces. While the theory exists for both infinite- and finite-dimensional systems, in practical applications it generally comes down to manipulation of finite-dimensional data patterns (vectors). This thesis forms no exception in this, in that the MOR implementations here also all start from a given finite-dimensional model. The focus here will therefore be on the finite-dimensional version of the POD approach. A general framework that includes both the finite- and infinite cases is provided in Appendix B.

5.1.1 Short Introduction to POD

The following few simple steps form the kernel of the finite-dimensional POD approach. Let the system be n-dimensional. First, a number of the system's n-dimensional (vector) solutions $\{\mathbf{x}_i\}_{i=1}^{M}$, called 'snapshots', are generated by numerical simulation or, when possible, by experiment, and put in a $n \times M$ data matrix

$$\mathbf{X} \;:=\; \left[\; \mathbf{x}_1 \quad \mathbf{x}_2 \quad \dots \quad \mathbf{x}_M \;\right]. \tag{5.1.1}$$

In a large-scale setting, e.g. when the finite-dimensional model is a discretized approximation of an underlying PDE, the snapshot dimension will generally be much larger than the number of snapshots, i.e. $n \gg M$. The aim of the POD is to determine ℓ ($\ell \leq M \ll n$) orthonormal eigenvectors, denoted by $\{\boldsymbol{\phi}_i\}_{j=1}^{\ell}$, such that the total square distance,

$$Q(\boldsymbol{\Phi}) \;:=\; \frac{1}{M}\sum_{i=1}^{M} \|\mathbf{x}_i - \boldsymbol{\Phi}\boldsymbol{\Phi}^T\mathbf{x}_i\|_2 \tag{5.1.2}$$

between the snapshots and its projections on the subspace defined by the $n \times \ell$ matrix

$$\boldsymbol{\Phi} \;:=\; \left[\; \boldsymbol{\phi}_1 \quad \boldsymbol{\phi}_2 \quad \dots \quad \boldsymbol{\phi}_\ell \;\right], \tag{5.1.3}$$

is minimized for any ℓ. It can be shown (Appendix B) that this optimization problem is solved by $\{\boldsymbol{\phi}_i\}_{j=1}^{\ell}$ being the eigenvectors of the $n \times n$ snapshot correlation matrix

$$\mathbf{R} \quad := \quad \frac{1}{M}\mathbf{X}\mathbf{X}^T \equiv \frac{1}{M}\sum_{i=1}^{M}\mathbf{x}_i\mathbf{x}_i^T, \tag{5.1.4}$$

corresponding to the largest eigenvalues $\{\lambda_j\}_{j=1}^{\ell}$. The number of dominant eigenvectors ℓ is usually chosen as:

$$\ell := \min_{\kappa}\left(\frac{\sum_{i=1}^{\kappa}\lambda_i}{\sum_i \lambda_i} \geq \alpha < 1\right), \tag{5.1.5}$$

i.e. the smallest number of modes required to capture desired amount of variability in the data. α is usually close to one, meaning that the first ℓ eigenvectors explain almost all of the variability contained in the snapshots set.

Sometimes, instead of $\mathbf{R}$, the snapshot covariance matrix

$$\widetilde{\mathbf{R}} \quad := \quad \frac{1}{M}\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^T \equiv \frac{1}{M}\sum_{i=1}^{M}(\mathbf{x}_i - \bar{\mathbf{x}}_i)(\mathbf{x}_i - \bar{\mathbf{x}}_i)^T, \tag{5.1.6}$$

is used, with $\bar{\mathbf{x}}$ representing the mean of the snapshots, $\bar{\mathbf{x}} = (1/M)\sum_{i=1}^{M}\mathbf{x}_i$, and $\widetilde{\mathbf{X}} := [\mathbf{x}_1 - \bar{\mathbf{x}}, \mathbf{x}_2 - \bar{\mathbf{x}}, \ldots \mathbf{x}_M - \bar{\mathbf{x}}]$. Letting $\mathbf{x} = \bar{\mathbf{x}} + \tilde{\mathbf{x}}$, the covariance matrix $\widetilde{\mathbf{R}}$ thus aims for describing fluctuations, $\tilde{\mathbf{x}}$, of data w.r.t. the mean of the collected snapshots. Geometrically, substraction of the mean of a set of points from each of the points in the set means moving the center of mass of the set to the origin of the coordinate system, as illustrated in Fig. 5.1. In specific applications as e.g. stochastic estimation, this mean correction must be included since data usually show a nonzero mean. In our deterministic model reduction applications, translation of points to make them have a zero-mean ($\tilde{\mathbf{x}} = \mathbf{0}$) is not of principal importance, although doing so may in some situations potentially increase level of detail in the reduced-order description when the same number of POD modes is retained as without mean substraction. For instance, suppose that the original snapshots are near parallel. In the not-mean-centered case, the first POD mode will lay pretty much in the direction of the average of the data. Due to the orthogonality requirement between the different POD modes, and the fact that the snapshots are almost parallel, the second mode will be able to explain very little of the 'remaining' variability in the data. The third even less than the second, etc. That means that a desired level α of 'energy' contained in the data will be captured almost exclusively by the first POD mode only, which is the best direction 'on average'. To capture α, very few 'extra' POD modes would be needed. On the other side, when the same "energy" criterion is applied to the mean-centered snapshot set, more modes would generally be retained, since the effect of the presence of the mean direction has 'disappeared'.

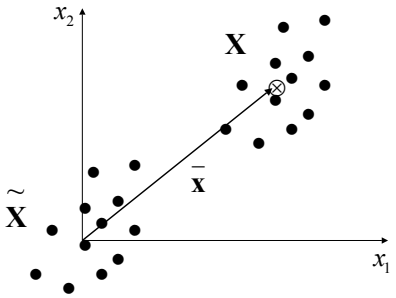


Figure 5.1: Effect of substracting the mean from the snapshots.

How are the eigensolutions computed?

Eigenvectors and eigenvalues of symmetric matrices can be solved in an iterative fashion using some form of the basic *Lanczos recursion* procedure. When the snapshot dimension is larger than the number of snapshots, i.e. $n > M$, the rank of the $n \times n$ correlation matrix $\mathbf{R}$ as defined in (5.1.4) (or the covariance matrix $\widetilde{\mathbf{R}}$) is at most M (or $M-1$, for the mean-subtracted case). Therefore, there are at most M (or $M-1$) eigenvectors $\boldsymbol{\phi}_i$ that correspond to a non-zero eigenvalue. Fortunately, it is not necessary to work with $\mathbf{X}\mathbf{X}^T$ in order to determine these eigenvectors: one can instead work with the much smaller $\mathbf{X}^T\mathbf{X}$ of size $M \times M$ and relate its eigenvectors, $\boldsymbol{v}_i$, to $\boldsymbol{\phi}_i$. Indeed, consider the eigenvalue problem

$$\mathbf{X}\mathbf{X}^T\boldsymbol{\phi}_i = \lambda_i\boldsymbol{\phi}_i, \tag{5.1.7}$$

and premultiply both sides by $c_i\mathbf{X}$ ($c_i \in \mathbb{R}\backslash\{0\}$) to obtain

$$(\mathbf{X}^T\mathbf{X})\underbrace{c_i\mathbf{X}^T\boldsymbol{\phi}_i}_{\boldsymbol{v}_i} = \lambda_i\underbrace{c_i\mathbf{X}^T\boldsymbol{\phi}_i}_{\boldsymbol{v}_i}. \tag{5.1.8}$$

Hence, $\mathbf{X}^T\mathbf{X}$ and $\mathbf{X}\mathbf{X}^T$ share the same M (or $M-1$) non-zero eigenvalues[2], and $\boldsymbol{v}_i = c_i\mathbf{X}^T\boldsymbol{\phi}_i$ are the corresponding eigenvectors. Let $\boldsymbol{v}_i$ be chosen to have unit-length, i.e. $\|\boldsymbol{v}_i\|_2^2 \equiv \boldsymbol{v}_i^T\boldsymbol{v}_i = 1$. Then,

$$\boldsymbol{v}_i^T\boldsymbol{v}_i = c_i^2\boldsymbol{\phi}_i^T\underbrace{\mathbf{X}\mathbf{X}^T\boldsymbol{\phi}_i}_{\lambda_i\boldsymbol{\phi}_i} = c_i^2\lambda_i\boldsymbol{\phi}_i^T\boldsymbol{\phi}_i = 1. \tag{5.1.9}$$

The constraint $\boldsymbol{\phi}_i^T\boldsymbol{\phi}_i \equiv \|\boldsymbol{\phi}_i\|_2^2 = 1$ requires that $c_i = \lambda_i^{-\frac{1}{2}}$, so that $\boldsymbol{v}_i = \lambda_i^{-\frac{1}{2}}\mathbf{X}^T\boldsymbol{\phi}_i$. Premultiplying with $\mathbf{X}$ and using (5.1.7), one finally obtains $\mathbf{X}\boldsymbol{v}_i = \sqrt{\lambda_i}\boldsymbol{\phi}_i$, or

[2]If X is full rank; otherwise some of the eigenvalues of $\mathbf{X}^T\mathbf{X}$ will also be zero.

equivalently

$$\boldsymbol{\phi}_i = \frac{1}{\sqrt{\lambda_i}}\mathbf{X}\boldsymbol{v}_i. \tag{5.1.10}$$

Besides being an expression that provides a more efficient computation of the POD basis when the number of snapshots is smaller than their dimension, (5.1.10) moreover reveals the important property that the *POD modes* actually *are linear combinations of the collected snapshots*, which was also the key observation of Sirovich in [250] leading to his **method of snapshots**. As such, POD modes will generally be different whenever the snapshots set is different, which is e.g. the case when different BCs (or forcing) and/or ICs are applied.

The role of SVD Instead of solving the eigenvalue problem involving products of large and, in general, dense matrices (as $\mathbf{X}\mathbf{X}^T$ or $\mathbf{X}^T\mathbf{X}$ are), one may alternatively perform a singular value decomposition (SVD) directly on the snapshot matrix $\mathbf{X}$. Indeed, let the $n \times M$ data matrix $\mathbf{X}$ ($n > M$) be decomposed as

$$\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T, \tag{5.1.11}$$

where $\mathbf{U} \in \mathbb{R}^{n\times n}$ and $\mathbf{V} \in \mathbb{R}^{M\times M}$ are orthogonal matrices ($\mathbf{U}^T\mathbf{U} = \mathbf{I}_n$ and $\mathbf{V}^T\mathbf{V} = \mathbf{I}_M$), and $\boldsymbol{\Sigma}$ is an $n \times M$ pseudo-diagonal matrix having on its leading diagonal non-negative elements arranged in a decreasing order, i.e. $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_M \geq 0$. Matrix $\mathbf{X}\mathbf{X}^T$ may now be expressed as

$$\mathbf{X}\mathbf{X}^T = \mathbf{U}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^T\mathbf{U}^T, \tag{5.1.12}$$

where $\boldsymbol{\Sigma}\boldsymbol{\Sigma}^T = diag\{\sigma_1^2, \ldots, \sigma_M^2, 0, \ldots, 0\}$, with the number of zeros equal $n - rank(\mathbf{X})$. It therefore follows that the POD modes $\boldsymbol{\phi}_i$ can be computed as "left singular vectors" (elements of $\mathbf{U}$) of $\mathbf{X}$, whereas $\lambda_i = \sigma_i^2$. Similarly, it follows that the "right singular vectors" (elements of $\mathbf{V}$) of $\mathbf{X}$ are the eigenvectors $\boldsymbol{v}_i$ of $\mathbf{X}^T\mathbf{X}$. This moreover means that the SVD (5.1.11) directly implies (5.1.10), which follows from (5.1.11) postmultiplied by $\mathbf{V}$, yielding $\mathbf{X}\mathbf{V} = \mathbf{U}\boldsymbol{\Sigma}$.

5.1.2 Standard POD scheme for MOR of ODEs

In solving problems of model order reduction, POD is just a projection-based technique as presented in Chapter 3, whereby the basis to project the governing equations onto consists of the most "energetic" modes, as defined above (the infinite-dimensional case is treated in Appendix B).

Reduced-order time-continuous model

Having determined the matrix of basis vectors $\boldsymbol{\Phi}$ as defined in (5.1.3), substitution $\mathbf{x}(t) \to \bar{\mathbf{x}} + \boldsymbol{\Phi}\mathbf{a}(t)$ into the full-order continuous-time generalized state-space

model $(3.2.1 - 3.2.3)$:

$$\mathbf{E}(\mathbf{x})\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x},\mathbf{u}) \tag{5.1.13}$$
$$\mathbf{y} = \mathbf{h}(\mathbf{x},\mathbf{u}) \tag{5.1.14}$$
$$\mathbf{x}(t_0) = \mathbf{x}_0 \tag{5.1.15}$$

yields the reduced-order model describing the time-evolution of the expansion coefficients $\mathbf{a}$:

$$\underbrace{\mathbf{\Phi}^T\mathbf{E}(\mathbf{\Phi}\mathbf{a}+\bar{\mathbf{x}})\mathbf{\Phi}}_{\mathbf{E}_{red}(\mathbf{\Phi}\mathbf{a}+\bar{\mathbf{x}})}\frac{d\mathbf{a}}{dt} = \underbrace{\mathbf{\Phi}^T\mathbf{f}(\mathbf{\Phi}\mathbf{a}+\bar{\mathbf{x}},\mathbf{u})}_{\mathbf{f}_{red}(\mathbf{\Phi}\mathbf{a}+\bar{\mathbf{x}},\mathbf{u})}, \tag{5.1.16}$$
$$\tilde{\mathbf{y}} = \mathbf{h}(\mathbf{\Phi}\mathbf{a}+\bar{\mathbf{x}},\mathbf{u}) \tag{5.1.17}$$
$$\mathbf{a}_0 = \mathbf{\Phi}^T(\mathbf{x}_0-\bar{\mathbf{x}}), \tag{5.1.18}$$

which is $(3.2.4 - 3.2.6)$ with $\mathbf{W} = \mathbf{V} = \mathbf{\Phi}$, $\bar{\mathbf{x}} = \mathbf{0}$, and the potential parameter dependence of $\mathbf{E}$ and $\mathbf{f}$ is suppressed for the sake of notation simplicity. The standard POD procedure is schematically illustrated/sumarized in Fig. 5.2.

For an (easily) invertible $\mathbf{E}$, the full-order continuous-time system-dynamics could equivalently be written as:

$$\frac{d\mathbf{x}}{dt} = \mathbf{E}(\mathbf{x})^{-1}\mathbf{f}(\mathbf{x},\mathbf{u}), \tag{5.1.19}$$

and substitution $\mathbf{x}(t) \rightarrow \bar{\mathbf{x}} + \mathbf{\Phi}\mathbf{a}(t)$ would lead to the following reduced-order model:

$$\frac{d\mathbf{a}}{dt} = \underbrace{\mathbf{\Phi}^T\mathbf{E}(\mathbf{x})^{-1}\mathbf{f}(\mathbf{\Phi}\mathbf{a}+\bar{\mathbf{x}},\mathbf{u})}_{\mathbf{f}_{red}(\mathbf{\Phi}\mathbf{a}+\bar{\mathbf{x}},\mathbf{u})}, \tag{5.1.20}$$
$$\tilde{\mathbf{y}} = \mathbf{h}(\mathbf{\Phi}\mathbf{a}+\bar{\mathbf{x}},\mathbf{u}) \tag{5.1.21}$$
$$\mathbf{a}_0 = \mathbf{\Phi}^T(\mathbf{x}_0-\bar{\mathbf{x}}). \tag{5.1.22}$$

Remark 5.1.1. It is important to emphasize that, whereas the solutions of the high-order continuous-time models are the same, the solutions of the corresponding reduced-order models are, due to their different topologies, generally not exactly so.

Discrete-time formulation(s)

The precise form of the discrete-time counterpart of the chosen reduced-order model (i.e., $(5.1.16)$ or $(5.1.20)$) will generally depend on the type of the time-discretization scheme used (explicit, implicit, etc.). For instance, if the vector function $\mathbf{f}$ is in the form $\mathbf{f}(\mathbf{x},\mathbf{u}) = \mathbf{A}(\mathbf{x})\mathbf{x} + \mathbf{B}(\mathbf{x})\mathbf{u}$ as in $(5.2.1)$, and the state-dependent (coefficient) matrices $\mathbf{E}$, $\mathbf{A}$ and $\mathbf{B}$ (as well as the input $\mathbf{u}$) are all evaluated at the current time-step, the discrete-time reduced-order dynamic equations models based

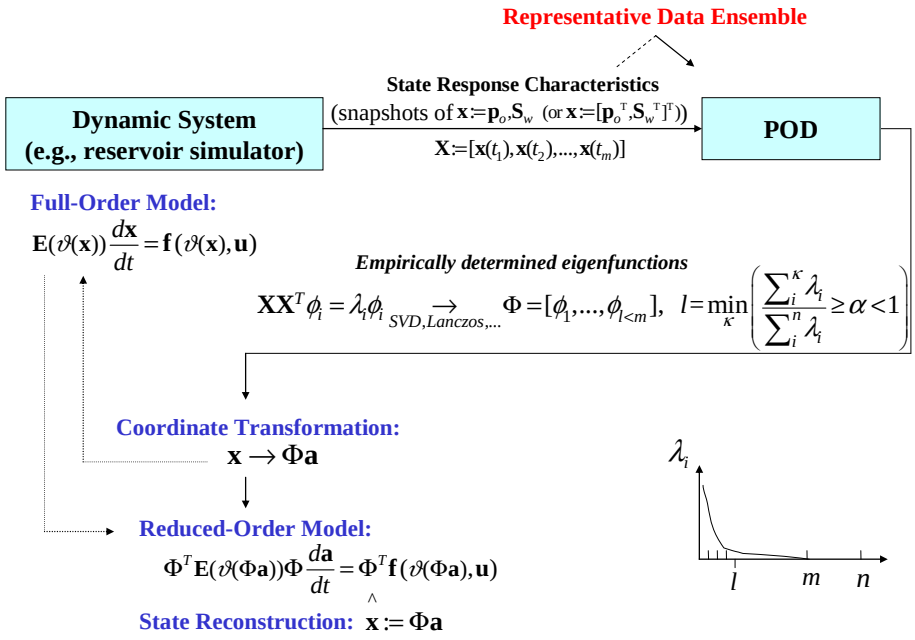


Figure 5.2: Overview of the standard POD procedure as applied to a finite-dimensional continuous-time reservoir model.

on (5.1.13) and (5.1.19)) would be $(\mathbf{E}_{red,k}-\Delta t_k\mathbf{A}_{red,k})\mathbf{a}_{k+1}=\mathbf{a}_k+\Delta t_k\mathbf{\Phi}^T\mathbf{B}_k\mathbf{u}_k$ and $(\mathbf{I}_k-\Delta t_k\mathbf{\Phi}^T\mathbf{E}_k^{-1}\mathbf{A}_k\mathbf{\Phi})\mathbf{a}_{k+1}=\mathbf{a}_k+\Delta t_k\mathbf{\Phi}^T\mathbf{E}_k^{-1}\mathbf{B}_k\mathbf{u}_k$, respectively, where $\mathbf{M}_k:=\mathbf{M}(\mathbf{\Phi a}_k+\bar{\mathbf{x}})$ and $\mathbf{M}_{red,k}:=\mathbf{\Phi}^T\mathbf{M}_k\mathbf{\Phi}$, for $\mathbf{M}=\mathbf{A},\mathbf{B},\mathbf{E}$.

The implementations of the POD approach in this chapter are based on these 'quasi-implicit' representations (cf. 2.4.2), i.e., all coefficient matrices (and the inputs) are always assumed evaluated at the current time-step, whereby the time-step is controlled by a maximum allowable saturation change.

5.1.3 POD in the LTI setting: relation to BTR

As shown below, there is an important connection of POD and balanced truncation in the (stable) LTI case. Although this fact has not been utilized in this thesis[3], it is important enough to be explained at least briefly. More information can be

[3]For linear systems this was again because of the symmetry of the systems considered. This will become clear in the sequel. The symmetry was also the reason of not considering the *Maximum Output Fraction* approach, proposed in [20, Section V] with the aim of incorporating measurements in POD basis when measurement and actuator locations are different (they are in this thesis). The rationale behind the method is that, if the non-collocated measurements are not contributing substantially to the overall spatial dynamics, approximation of measurements by the standard POD basis may fail.

found in [4, 10, 135, 164, 165, 233, 293], for instance. Recall that for a (stable) LTI system in the standard state-space formulation:

$$\frac{d\mathbf{x}}{dt} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad , \quad \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t), \tag{5.1.23}$$

with m inputs and p outputs, the so-called *controllability-* and *observability Gramians* are defined as

$$\mathbf{W}_c := \int_0^\infty e^{\mathbf{A}t}\mathbf{B}\mathbf{B}^T e^{\mathbf{A}^T t}dt \quad \text{and} \quad \mathbf{W}_o := \int_0^\infty e^{\mathbf{A}^T t}\mathbf{C}^T\mathbf{C}e^{\mathbf{A}t}dt, \tag{5.1.24}$$

respectively. The importance of these symmetric and positive-semidefinite matrices is that they respectively measure to what degree each state of the system can be excited by an input and to what degree each state excites future outputs.

Let $\mathbf{B} := [\mathbf{b}_1 \dots \mathbf{b}_m]$ and let $\mathbf{x}_{\delta_j}(t)$ denote the (zero-initial) state-response of the system to impulse input $\mathbf{u}(t) = \delta(t)\mathbf{e}_i$, where $\mathbf{e}_i \in \mathbb{R}^m$ is the canonical ith unit vector (i.e., each input channel is excited separately). The state impulse response is then $\mathbf{x}_{\delta,j}(t) = \int_0^\infty e^{\mathbf{A}(t-\tau)}\mathbf{B}\mathbf{u}(\tau)d\tau = \int_0^\infty e^{\mathbf{A}t}\mathbf{b}_j\delta(\tau)d\tau = e^{\mathbf{A}t}\mathbf{b}_j$, so that the controllability Gramian $\mathbf{W}_c$ is given by[4]

$$\begin{aligned}\mathbf{W}_c &\triangleq \int_0^\infty e^{\mathbf{A}t}\mathbf{B}\mathbf{B}^T e^{\mathbf{A}^T t}dt = \int_0^\infty e^{\mathbf{A}t}[\mathbf{b}_1 \dots \mathbf{b}_m][\mathbf{b}_1 \dots \mathbf{b}_m]^T e^{\mathbf{A}^T t}dt \\ &= \int_0^\infty [\mathbf{x}_{\delta,1}(t) \dots \mathbf{x}_{\delta,m}(t)][\mathbf{x}_{\delta,1}(t) \dots \mathbf{x}_{\delta,m}(t)]^T dt = \int_0^\infty \mathbf{X}(t)\mathbf{X}^T(t)dt.\end{aligned} \tag{5.1.25}$$

where $\mathbf{X}(t) := [\mathbf{x}_{\delta,1}(t) \dots \mathbf{x}_{\delta,m}(t)]$. The similarity between the Gramian expression (5.1.25) and the (continuous) POD is now obvious: *the (continuous) POD modes for the set of the state impulse-responses are the dominant eigenvectors of* $\mathbf{W}_c$, or, in other words, the most controllable modes of the realization. It is hereby important to note that since the Gramian matrices depend on the chosen coordinate system, so do the POD modes of the impulse dataset.

Connection between POD and the observability of a LTI system (5.1.23) can be illustrated by considering its 'dual' (or 'adjoint') system:

$$\frac{d\mathbf{z}}{dt} = \mathbf{A}^T\mathbf{z} + \mathbf{C}^T\mathbf{u}_d \quad , \quad \mathbf{y}_d(t) = \mathbf{B}^T\mathbf{z}(t), \tag{5.1.26}$$

and essentially makes use of the result that the controllability of a stable LTI system is equivalent to the observability of its dual. The dominant subspace of the observability Gramian $\mathbf{W}_o$ is thus determined by the POD modes of the (zero initial-state) adjoint system driven by the impulse inputs $\mathbf{u}_d = \delta(t)\mathbf{e}_i$.

[4] Mathematically, the same is achieved by setting $\mathbf{u} \equiv 0$ and considering $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x}$ for the initial conditions $\mathbf{x}_{0,j} = \mathbf{b}_j$ $(j = 1, \dots, m)$.

Balanced POD

According to the above, the observability Gramian of the original system (5.1.23) is the same as the controllability Gramian of its adjoint system (5.1.26). If the impulse response data is obtained by numerical simulation, it is usually given at discrete times $t_1, \ldots, t_M$. The state solutions $\{\mathbf{x}_{\delta,i}(t_k)\}_{i=1}^m$, $k = 1, \ldots, M$ can be assembled in $\mathbf{X}_{\delta,M} := [\mathbf{x}_{\delta,1}(t_1)\sqrt{\alpha_1} \ldots \mathbf{x}_{\delta,1}(t_M)\sqrt{\alpha_M} \ldots \mathbf{x}_{\delta,m}(t_1)\sqrt{\alpha_1} \ldots \mathbf{x}_{\delta,m}(t_1)\sqrt{\alpha_M}]$ such that the integral in (5.1.25) is approximated by a quadratic sum $\mathbf{X}_{\delta,M}\mathbf{X}_{\delta,M}^T =: \mathbf{W}_c^{\delta,M}$, with $\{\alpha_j\}_{j=1}^M$ appropriate quadrature coefficients.

In a completely analogous way, impulse responses of the adjoint states $\mathbf{z}$ can be numerically computed at N discrete time points and, weighted appropriately by quadrature coefficients, assembled in a data matrix $\mathbf{Y}_{\delta,M}$ such that $\mathbf{W}_o$ is approximated by the *empirical*[5] observability Gramian, $\mathbf{W}_o^{\delta,M} := \mathbf{Y}_{\delta,M}\mathbf{Y}_{\delta,M}^T$. Assuming linear independency (for numerical reasons) of both $\mathbf{X}_{\delta,M}$ and $\mathbf{Y}_{\delta,N}$, and thinking of these quantities as approximate square-roots of the corresponding Gramians, an approximate balancing transformation can be found by the Algorithm 1 in Section 4.4.3, i.e., by determining an SVD of $\mathbf{X}_{\delta,M}^T\mathbf{Y}_{\delta,N}$. The dimension of this matrix is $M \times N$, i.e., the number of primal snapshots $\times$ the number of dual snapshots, which is typically tractable.

Remark 5.1.2. As one simulation of the forward (adjoint) system is needed for each component of the input (output), the number of inputs (outputs) must not be too large in this method. Clearly, if one desires a reduced-order model while approximating the full state information adequately, i.e., $\mathbf{y} = \mathbf{x}$, the associated cost would be unmanageable. In [233], this issue is addressed by considering an alternate adjoint system: $\frac{d\mathbf{z}}{dt} = \mathbf{A}^T\mathbf{z} + \mathbf{C}^T\mathbf{\Phi}_r\mathbf{w}$, where the columns of the $p \times r$ orthogonal matrix $\mathbf{\Phi}_r$ are the first r POD modes of the the set of impulse responses of the original system (which is simply the state impulse responses for m inputs premultiplied by $\mathbf{C}$). The underlying idea of this *output projection method* is projecting the output of the original system onto a low-dimensional subspace, i.e., taking $\mathbf{y} = \mathbf{\Pi}_r\mathbf{C}\mathbf{x}$, with $\mathbf{\Pi}_r$ an orthogonal projection of rank r onto an r-dimensional subspace of the output space, such that the 2-norm of the difference between the $p \times m$ impulse response matrix, $\mathbf{H}(t)$ (the element $H_{ij}(t)$ is the output component $y_i(t)$ corresponding to the impulse input $u_j(t) = \delta(t)$), and its projection $\mathbf{\Pi}_r\mathbf{H}(t)$ is minimized. The projection $\mathbf{\Pi}_r$ achieving this can be written as $\mathbf{\Pi} = \mathbf{\Phi}_r\mathbf{\Phi}_r^T$, with $\mathbf{\Phi}_r$ as defined above. See [135, 233] for more detail.

Remark 5.1.3. If snapshots are generated using inputs signals other than impulses, 'weighted' Gramians become relevant. See [10, Section 7.6] for details on weighted BTR.

[5]The terminology *Empirical Gramians* was introduced in [164, 165], where the POD was used, in a similar fashion as here, to extend BTR to nonlinear systems.

5.2 POD applied on two-phase reservoir simulation

In this section the standard POD procedure as outlined above is applied to the finite-dimensional generalized state-space representation (2.3.6) of the the governing equations of a typical two-phase (oil-water) reservoir flow, where, for simplicity of presentation and notation, a two-dimensional horizontal reservoir only is considered:

$$\mathbf{V}(\widetilde{\mathbf{x}}(t))\mathbf{W}(\widetilde{\mathbf{x}}(t))\frac{d\mathbf{x}}{dt} = \mathbf{T}(\widetilde{\mathbf{x}}(t))\mathbf{x} + \mathbf{V}(\widetilde{\mathbf{x}}(t))\mathbf{F}(\widetilde{\mathbf{x}}(t))\mathbf{L}_{qu}\mathbf{u}, \tag{5.2.1}$$

where $\mathbf{x} = [p_{o,11}, S_{w,11}, p_{o,12}, S_{w,12}, \ldots, p_{o,n_x n_y}, S_{w,n_x n_y}]^T$ is the n-dimensional state vector containing oil pressures and water saturations for each of the $n_x n_y = \frac{n}{2}$ grid blocks, $\mathbf{V}$ is a diagonal mass matrix with entries that are functions of grid block volume and fluid densities, $\mathbf{W}$ is a block diagonal matrix with entries being functions of compressibility, porosity and water saturation, $\mathbf{T}$ is a block matrix containing the transmissibilities for oil and water, $\mathbf{F}$ is a matrix-valued function of fractional-flow functions for water and oil, and $\mathbf{L}_{qu}$ is a selection matrix consisting of zeros and ones at appropriate places so that $\mathbf{u} \equiv \mathbf{L}_{qu}^T\mathbf{q}_t$ is the input vector containing total liquid rates (generally, $q_t = q_o + q_w$ at the producers and q_w at the injectors). For a mode detailed description refer to (2.3.6). In (5.2.1), the parameter-dependence of the system matrices has been suppressed for the sake of shorter notation.

If $\mathbf{V}(\widetilde{\mathbf{x}}(t))\mathbf{W}(\widetilde{\mathbf{x}}(t))$ is invertible, which is normally the case for a non-zero compressibility, (5.2.1) may equivalently be written in the standard state-space form as

$$\frac{d\widetilde{\mathbf{x}}(t)}{dt} = \mathbf{A}_c(\widetilde{\mathbf{x}}(t))\widetilde{\mathbf{x}}(t) + \mathbf{B}_c(\widetilde{\mathbf{x}}(t))\mathbf{u}(t). \tag{5.2.2}$$

where $\mathbf{A}_c := (\mathbf{VW})^{-1}\mathbf{T}$ and $\mathbf{B}_c := \mathbf{W}^{-1}\mathbf{F}\mathbf{L}_{qu}$. As $\mathbf{V}$ and $\mathbf{W}$ are diagonal and block-diagonal, respectively, inverting $\mathbf{VW}$ is easy and stable, as it can be done analytically. The quasi-implicit time-discretization (i.e., the state-dependent coefficient matrices and the input evaluated at the current time-step, t_k) of (5.2.2) yields the form

$$[\mathbf{I}_n - \Delta t_k \mathbf{A}_c(\widetilde{\mathbf{x}}_k)]\widetilde{\mathbf{x}}_{k+1} = \widetilde{\mathbf{x}}_k + \Delta t_k \mathbf{B}_c(\widetilde{\mathbf{x}}_k)\mathbf{u}_k, \tag{5.2.3}$$

where $t_{k+1} := t_k + \Delta t_k$, and I_n denotes the $n \times n$ identity matrix. To make the analysis and the discussion in the sequel easier to follow, the alternating state-vector $\widetilde{\mathbf{x}}$ will be rearranged, using a suitable permutation matrix $\mathbf{P}$, to $\mathbf{x} := [\mathbf{p}_o^T, \mathbf{S}_w^T]^T$, where $\mathbf{p}_o = [p_{o,11}, p_{o,12}, \ldots, p_{o,n_x n_y}]^T$ and $\mathbf{S}_w = [S_{w,11}, S_{w,12}, \ldots, S_{w,n_x n_y}]^T$. Thus, $\widetilde{\mathbf{x}} = \mathbf{P}\mathbf{x}$, and using $\mathbf{P}^T\mathbf{P} = \mathbf{I}_n$, (5.2.3) may be rearranged as:

$$[\mathbf{I}_n - \Delta t_k \mathbf{P}^T\mathbf{A}_c(\mathbf{P}\mathbf{x}_k)\mathbf{P}]\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta t_k \mathbf{P}^T\mathbf{B}_c(\mathbf{P}\mathbf{x}_k)\mathbf{u}_k. \tag{5.2.4}$$

5.2.1 Derivation of reduced-order models

In a standard POD procedure, (5.2.3) would be simulated and snapshots collected at a number of time instances, not necessarily uniformly, during a time-interval [0,T], generally for various, well-designed control excitations. By 'well-designed', input signals (here, e.g., pressures) are normally meant that are consistent with the expected range of operating conditions during the intended application of the model and, moreover, so that the relevant dynamics is represented in the collected snapshots. Two-phase (or, more generally, multi-phase) reservoir flow systems, however, possess a number of properties that make a successful straightforward application of the standard implementations the POD approach generally unlikely. Therefore, adaptations of the standard POD approach seem inevitable. The adaptation in the next paragraph is related to the numerical incompatibility of the state variables that form the state vector, here saturations and pressure. More critical features, like, e.g., fluid interfaces moving all over the spatial domain, shocks, (displacing) fluid breakthrough, etc., are addressed later.

Different variability of state variables: Individual vs. Combined POD projections

A basic POD implementation would involve substitution $\mathbf{x}_k \rightarrow \mathbf{\Phi a}_k$ (or $\mathbf{x}_k \rightarrow \mathbf{\Phi a}_k + \bar{\mathbf{x}}$) and projecting (5.2.4) onto the subspace defined by a POD matrix $\mathbf{\Phi} \in \mathbb{R}^{n\times\ell}$ ($\ell \ll n$), yielding the following reduced-order model for the coefficients $\mathbf{a}_k$:

$$[\mathbf{I}_\ell - \Delta t_k \mathbf{\Phi}^T\mathbf{P}^T\mathbf{A}_c(\mathbf{P\Phi a}_k)\mathbf{P\Phi}]\mathbf{a}_{k+1} = \mathbf{a}_k + \Delta t_k \mathbf{\Phi}^T\mathbf{P}^T\mathbf{B}_c(\mathbf{P\Phi a}_k)\mathbf{u}_k, \quad (5.2.5)$$

whereby $\mathbf{\Phi}$ would 'explain' most of the variability in $\mathbf{X} = \begin{bmatrix} \mathbf{p}_{o,1} & \cdots & \mathbf{p}_{o,N} \\ \mathbf{S}_{w,1} & \cdots & \mathbf{S}_{w,N} \end{bmatrix}$.

A problem with this straightforward POD implementation is that p_0 and S_w will generally show different variability in their numerical values. Due to the property of the POD to favorize variables which numerical values show greater variability, this could consequently negatively impact the quality, or even reliability, of a reduced-order model obtained this way. If all variables were of the same type, the data could be preprocessed before the application of the POD, e.g., by normalizing all variables to have unit variance. Because of the totally different both physical and numerical nature of pressure and saturation variables, data-preprocessing in the case of multi-phase porous media equations is all but trivial. In the applications of POD in this thesis, optimal bases for the two different groups, pressures and saturations, are therefore obtained **separately**: instead of solving one eigenvalue (or SVD) problem to obtain a single dense $n \times \ell$ POD matrix $\mathbf{\Phi}$, two separate eigenvalue (SVD) problems are solved (one for $\mathbf{X}_p := [\mathbf{p}_{o,1}, \ldots, \mathbf{p}_{o,N}]$ and one for $\mathbf{X}_S := [\mathbf{S}_{w,1}, \ldots, \mathbf{S}_{w,N}]$), yielding two different POD matrices, $\mathbf{\Phi}_p \in \mathbb{R}^{\frac{n}{2}\times\ell_p}$ and $\mathbf{\Phi}_S \in \mathbb{R}^{\frac{n}{2}\times\ell_S}$, for pressures and saturations, respectively. The 'total POD matrix' to use in (5.2.5) is then formed as $\mathbf{\Phi} := \begin{bmatrix} \mathbf{\Phi}_p & \mathbf{0}_{\frac{n}{2}\times\ell_S} \\ \mathbf{0}_{\frac{n}{2}\times\ell_p} & \mathbf{\Phi}_S \end{bmatrix} \in \mathbb{R}^{n\times\ell}$. This allows

for choosing a different degree of reduction for the pressures and the saturations[6], however at expense of (possibility of) loosing some of the information about the coupling between these variables.

Using such a 'decomposed' $\mathbf{\Phi}$ and correspondingly defining $\mathbf{a} := [\mathbf{a}_p^T, \mathbf{a}_S^T]^T$, where $\mathbf{a}_p \in \mathbb{R}^{\ell_p}$ and $\mathbf{a}_p \in \mathbb{R}^{\ell_S}$, the reduced-order model (5.2.5) may be written as

$$\underbrace{\left(\hat{\mathbf{I}} - \Delta t_k \mathbf{\Phi}^T \mathbf{P}^T \mathbf{A}_c(\mathbf{P}\mathbf{\Phi}\mathbf{a}_k)\mathbf{P}\mathbf{\Phi} \right)}_{\mathbf{E}_{d,red}(\mathbf{\Phi}\mathbf{a}_k)} \underbrace{\begin{bmatrix} \mathbf{a}_{p,k+1} \\ \mathbf{a}_{S,k+1} \end{bmatrix}}_{\mathbf{a}_{k+1}} = \underbrace{\begin{bmatrix} \mathbf{a}_{p,k} \\ \mathbf{a}_{S,k} \end{bmatrix}}_{\mathbf{a}_k} - \Delta t_k \underbrace{\mathbf{\Phi}^T \mathbf{P}^T \mathbf{B}_c(\mathbf{P}\mathbf{\Phi}\mathbf{a}_k)}_{\mathbf{B}_{d,red}(\mathbf{\Phi}\mathbf{a}_k)} \mathbf{u}_k, \tag{5.2.6}$$

where $\mathbf{\Phi} = \begin{bmatrix} \mathbf{\Phi}_p & \mathbf{0}_{\frac{n}{2}\times\ell_S} \\ \mathbf{0}_{\frac{n}{2}\times\ell_p} & \mathbf{\Phi}_S \end{bmatrix}$ and $\hat{\mathbf{I}} := \begin{bmatrix} \mathbf{I}_{\ell_p} & \mathbf{0}_{\ell_p\times\ell_S} \\ \mathbf{0}_{\ell_S\times\ell_p} & \mathbf{I}_{\ell_S} \end{bmatrix} \equiv I_{\ell+S}$.

The advantage of (5.2.6) is that it 'uncouples' the pressure and the saturation expansion coefficients $\mathbf{a}_p$ and $\mathbf{a}_S$, respectively, and, moreover, that the dimensions of the different elements are immediately clear. Defining the 'permuted' POD matrix $\widetilde{\mathbf{\Phi}} := \mathbf{P}\mathbf{\Phi}$, the reduced-order model may be expressed more compactly as

$$[\mathbf{I}_\ell - \Delta t_k \widetilde{\mathbf{\Phi}}^T \mathbf{A}_c(\widetilde{\mathbf{\Phi}}\mathbf{a}_k)\widetilde{\mathbf{\Phi}}]\mathbf{a}_{k+1} \quad = \quad \mathbf{a}_k + \Delta t_k \widetilde{\mathbf{\Phi}}^T \mathbf{B}_c(\widetilde{\mathbf{\Phi}}\mathbf{a}_k)\mathbf{u}_k. \tag{5.2.7}$$

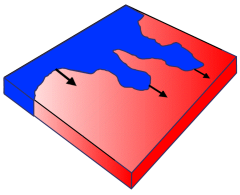
Having determined $\mathbf{a}_{k+1}$, the (approximation of the) fine-scale solution is given by $\hat{\mathbf{x}}_{k+1} := \begin{bmatrix} \hat{\mathbf{p}}_{o,k+1} \\ \hat{\mathbf{S}}_{w,k+1} \end{bmatrix} = \mathbf{\Phi}\mathbf{a}_{k+1}$. With $\mathbf{P}\mathbf{x} = \widetilde{\mathbf{x}}$ and $\mathbf{P}\mathbf{\Phi} = \widetilde{\mathbf{\Phi}}$, the alternating state vector from the original reservoir description (5.2.3) may then, if necessary, be recovered by $\hat{\widetilde{\mathbf{x}}}_k = \mathbf{P}\mathbf{\Phi}\mathbf{a}_k$.

5.2.2 On the (expected) quality of the approximation

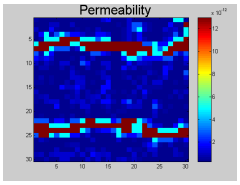
In many nonlinear problems in science a larger approximation error in the state is not necessarily a catastrophe, as long as it does not contribute substantially to the output behavior (if one is interested in just that). Multi-phase reservoir systems are fundamentally different in this respect, because of the active role of the fluid saturation(s). Indeed, it is unreasonable to expect good approximation of the production data if the saturation behavior in between the wells is approximated wrongly. Any potentially successful MOR technique must at least be able to adequately reconstruct the saturation data used to generate the reduced-order model. This requirement is valid even when all N basis functions representing the N collected linearly independent data points are employed, which is due to fact that

[6]Note that by separating the POD problems for p_o and S_w also the issue of 'physically unsensible" mathematical summation (in $\mathbf{X}\mathbf{X}^T$) of variables having different dimensionality ([Pa] vs. [-]) is circumvented. This problem could, at least conceptually, also be dealt with by nondimensionalizing the variables, in which case the choice of nondimensionalization becomes critical, however. Another possibility is to use a different inner-product definition in the POD optimization problem, as the standard inner product does not make dimensional sense if the state vector consist of different physical quantities (indeed, one can not add pressure and saturation).

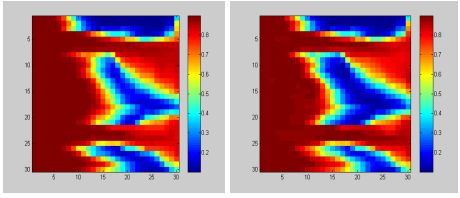
the total error in the state besides the snapshots approximation error also contains an error component coming from the projection of the vector fields involved in the governing equations onto the chosen basis. The waterflood simulation experiments performed in this research ([123, 184, 185], see Figure 5.3 for an example) and recently in [55–57] suggest that a given flow scenario can be reproduced accurately if enough snapshots have been used and 'enough' energy has been captured in the POD modes (especially saturation ones, see also below). Different sets of snapshots will of course give raise to different matrix decompositions and hence to different 'optimal' linear projections. The generation of good snapshots is, however, in general by no means a science, but rather an (primitive) art. We have not been focused on this aspect in this research much, but the simulation results in our research and in [55–57] seem to indicate that the POD approach is quite robust in this respect when the snapshots are generated by exciting the system by manipulating the injection and/or production profiles (i.e., flow-rates and/or bottom-hole pressures) consistent with the expected range of operating conditions.



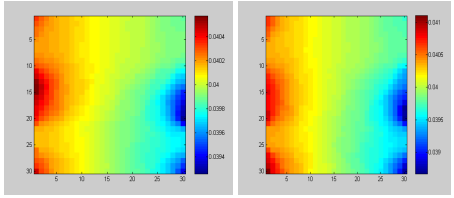
Horizontal reservoir equipped with smart well injector and producer.



Permeability field.



SATURATION (left: full order; right: reduced order)



PRESSURE (left: full order; right: reduced order)

Figure 5.3: Flow reconstruction with POD. Full-order model: 30×30 grid blocks (1800 states). Reduced-order model (9 states: 2 pressure- and 7 saturation POMs). Optimality criterion used: min. snapshots energy captured: 95% (First pressure mode accounted for 67.3 %, second mode for 32.6 %, third mode for 0.04 %, etc.; First saturation mode accounted for 51.14 %, second mode for 20.99 %, third for 12.85 %, fourth for 6.98 %, etc.). Total flow-rates kept constant. Total simulation time: 1200 days.

In order to facilitate easier recognition of possible pitfalls when applying POD to multi-phase flow, we summarize some of the distinguishing characteristic features of multi-phase reservoir systems:

- *Moving fluid-interfaces.* The traveling nature of the (steep) fluid-front interfaces forms a real challenge for any projection basis with spatially global support and is thus by no means restricted to a POD one. The latter has one extra disadvantage, which is that it is found by time-averaging. Since any time-averaging smooths out information that is only expressed over small intervals of time, correlation information in time will be lost. And so will also be the information about the time before- and after the displacing fluid breaks through at the producers. In order to adequately reproduce the collected saturation snapshots, the POD basis will generally be of an order that (greatly) overestimates the intrinsic order of the dynamics at particular time-points (or time-intervals). Intuitively, but also supported by the extensive simulation experiments performed during this research (see also Fig. 5.3), and recently also elsewhere [55, 56], it can be stated that an accurate representation of the saturation behavior requires (much) more POD modes to be retained than for the pressure.

- *Bounded saturation support.* Using basis functions having global support, it is not guaranteed that the local saturations will always and everywhere stay within their physically feasible values, i.e., within $[S_{wr}, 1 - S_{or}]$. This is true even when all N basis functions representing the N collected linearly independent data points are employed, which is due to the abovementioned contribution of the equation projection error to the total state error. Assuming that the deviation from the feasible interval is not substantial, to proceed with the simulation the grid-block in the question can be assigned the saturation value at the closest boundary of the interval (i.e., S_{wr} or $1 - S_{or}$), however at the cost of some mass balance violation. We moreover note that, because of the square-root error optimality of POD, also the POD approximants $\mathbf{\Phi}\mathbf{\Phi}^T\mathbf{x}_i$ of the training snapshots $\{\mathbf{x}_i\}_{i=1}^N$ are not guaranteedly physically admissible when less than N basis functions are used. In view of this fact, it is reasonable to question not only the usefulness of the energy criterion $(5.1.5)$ for deciding how many saturation POD modes should be retained in the standard POD basis, but also the optimization problem itself (that is, the total least squares problem of minimizing $(5.1.2)$). It seems wise to explore the possibilities for redefining the POD problem by augmenting it by a constraint expressing the necessity of staying, at least of the back-projected snapshots, within the admissible region, and/or clever clustering of the snapshots, either spatially or temporary. The latter could e.g. done in a fashion similar to the PID approach mentioned in Section 3.3.2, leading to multiple, time-local POD bases. We speculate that for a single simulation run, due to the traveling nature of the saturation fronts, clusters based on time-information and those based on spatial-information would be closely related to each other. Needless to say, the extensions/adjustments to the approach should not lead to an unreasonable increase in the time required to determine a basis.

- *Upstream mobility weighting.* In the computational schemes, the coefficients are evaluated at the interfaces between the adjacent control volumes (here grid-cells) as averages of their values within the cells. For instance, absolute permeability is commonly averaged using harmonic weighting, whereas the (strongly) saturation (and weakly pressure) dependent fluid mobilities are averaged according to some upstream weighting principle in order to ensure the physical correctness (and stability) of the solution.[7] When multiple producing and injection wells are considered, the upstream direction may be different in different regions of the reservoir. In a single-point upstream weighting (also used in our implementations), the fluid mobilities at the interface between two cells will be determined as being the mobility of in the cell under the higher fluid pressure (or potential in 3D). Expressing pressures (and saturations) in a global (and truncated) basis, there is a risk that, locally (i.e., across gridblock interfaces), the computed upstream direction (i.e., the sign of the pressure gradient) is the opposite of what it should be. Besides to incorrect results, this may (either immediately or eventually) lead to instabilities/oscillations in the flow solution.[8] Evidently, one prefers this effect not to occur, especially not in regions close to the wells.

5.2.3 On the computational complexity

The overall computational cost of a POD approach consists of two main parts: a) preprocessing cost of generating 'representative' snapshots spanning a large portion of operating conditions of interest and, subsequently, determining the POD basis, and b) the cost of solving the reduced-order model.

Pre-processing cost

The process of generating 'good' time-snapshots data ensuring a robust predictive capability of the reduced-order model may be quite costly, since it generally involves multiple simulation runs of the high-dimensional model. Due to the lack of design of experiments methodologies for flow scenario selection, a more heuristic approach is followed in which the reservoir model is excited by manipulating the injection and/or production profiles to cover operating conditions of interest. The signals (well flow-rates and/or bottom-hole pressures) need to be consistent with expected both the expected ranges of values and expected times of update). If either the number of pre-processing runs, k, is large or the time-steps in (any of) the runs were small, the POD analysis on the $n \times \sum_{i=1}^{k} N_i$ snapshots data matrices $\mathbf{X}_\xi := [\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_N] \equiv [\boldsymbol{\xi}_1^1, \ldots, \xi_{N_1}^1, \ldots, \xi_1^k, \ldots, \boldsymbol{\xi}_{N_k}^k]$, $\xi = p_o, S_w$, may be computationally too demanding[9] (recall that these matrices are dense in general).

[7] Another popular choice is to use the upstream weighting only for the relative permeabilities and capillary pressures and arithmetic weighting for the pressure-dependent coefficients as fluid viscosity.

[8] Note that neither harmonic nor arithmetic averages are sensitive in this respect.

[9] In terms of computational time it may actually not be so for (iteratively) determining a sufficient number of pressure POD basis vectors, as this number is generally not substantial. The high memory demands remain, though.

To reduce this cost one may decide to base the POD analysis (i.e. solving a SVD or an EVD problem) on: 1) 'dynamically relevant' snapshots (not trivial at all), 2) clustered[10] snapshots, and 3) 'low-rank' matrix decomposition (SVD or EVD) update algorithms (see also Section 5.3).

Cost associated with solving the reduced-order system

Solving the n-dimensional sparse linear system (5.2.4) (or (5.2.3)) has a cost of $O(n^{1.1\sim1.5})$, whereas solving the ℓ-dimensional dense reduced-order linear system (5.2.7) has a cost of $O(\ell^2)$, which is much less than the former whenever $\ell << n$. Despite this speed-up, a computational gain in determining the reduced-order solution is actually not guaranteed (and if there is a gain, it is rarely spectacular). This is due to that fact that at each time-step the matrix product $\widetilde{\mathbf{\Phi}}^T\mathbf{A}_c(\widetilde{\mathbf{\Phi}}\mathbf{a}_k)\widetilde{\mathbf{\Phi}} \in \mathbb{R}^{\ell\times\ell}$ must be evaluated, which has a cost of $O(n\times\ell^2)$.[11]

There is nothing to do about this except to try to estimate $\mathbf{a}_{k+1}$ by some other, cheaper means.[12] Some recent contributions in the POD literature (see, e.g., [19–21]) attempt to address the high computational cost of forming $\widetilde{\mathbf{\Phi}}^T\mathbf{A}_c(\widetilde{\mathbf{\Phi}}\mathbf{a}_k)\widetilde{\mathbf{\Phi}}$ by not computing the POD basis coefficients in the usual POD manner (i.e., here, by solving (5.2.5 or (5.2.7)), but rather estimating these from a partial set of the original data at a selected number of points in the spatial domain. In short, instead of defining $\mathbf{x}_k \rightarrow \mathbf{\Phi}\mathbf{a}_k \equiv \sum_{i=1}^{\ell} a_{k,i}\boldsymbol{\phi}_i$ and substituting it into the high-order model (e.g., 5.2.4) to obtain a standard POD reduced-order model (e.g., 5.2.5 or 5.2.7)) for the ℓ POD coefficients $\{a_k\}_{i=1}^{\ell}$, a G-dimensional subset (of length G) of the whole state, $\mathbf{x}_k^G \rightarrow \mathbf{\Phi}^G\mathbf{a}_k \equiv \sum_{i=1}^{\ell} a_{k,i}\boldsymbol{\phi}_i^G$, is considered comprising the elements of $\mathbf{x}_k$ at G selected spatial positions, where $\boldsymbol{\phi}_i^G$ is the subset of $\boldsymbol{\phi}_i$ comprising its elements at the same G positions (note that $\boldsymbol{\phi}_i^G$ are, in general, not orthogonal!). As the ℓ POD coefficients in $\mathbf{a}_k$ are unknown, one defines an estimate of it, denoted by $\hat{\mathbf{a}}_k$, such that $\mathbf{\Phi}^G\hat{\mathbf{a}}_k$ would approximate $\mathbf{x}_k^G$ well, that is, one writes

[10] We note reports of snapshots clustering in [55, 56] inspired by the CVT/CVOD idea mentioned in 3.3.2, wherein POD is applied to the collection of cluster centroids, where each centroid represents a particular cluster of the collected snapshots. The clustering is reported to cause an negligible computational overhead. The final number of the POD basis vectors seems to be smaller than in the normal POD analysis for the same quality of the reduced-order model and, moreover, the difference in the number of pressure and saturation POD modes seems to decrease as well.

[11] The issue is even more manifested in a fully time-implicit scheme using the Newton-Raphson solving procedure, whereby the reduced basis is used in the inner loop. There, the next timestep state solution is determined by iteratively solving (sparse) linear systems of the form $\mathbf{J}_n^i\Delta\mathbf{x}^{i+1} = \mathbf{R}_n^i$, where, n being the state dimension, $\mathbf{J}_n^i$ and $\mathbf{R}_n^i$ are the (state-dependent both the) $n\times n$ Jacobian matrix of the system and the n-dimensional equation residual vector at the current iteration i, respectively, and $\mathbf{x}^{i+1} = \mathbf{x}^i + \Delta\mathbf{x}^{i+1}$. The application of POD in this case could e.g. be iteratively solving, at each time step, reduced-order (dense) linear systems of the form $\mathbf{J}_\ell^i\Delta\mathbf{a}^{i+1} = \mathbf{R}_\ell^i$, where $\mathbf{J}_\ell^i := \mathbf{\Phi}^T\mathbf{J}_n^i\mathbf{\Phi}$ and $\mathbf{R}_\ell^i := \mathbf{\Phi}^T\mathbf{R}_\ell^i$.

[12] Actually, the reduced-order model for the expansion coefficients does not even need to be a projected-version of the high-dimensional model. Conceptually, it can be ANY model of the required dimension as long as it includes enough free parameters to optimize over by comparing its behavior with that of the high-dimensional model in a suitable sense. We note that an optimization setting would allow a simultaneous 'estimation' of the corresponding basis functions as well (i.e., by not fixing them a priori). Solving the overall optimization problem could be computationally too demanding, though. See [1, 68, 163] and references therein for some ideas in this direction.

$\mathbf{x}_k^G \approx \mathbf{\Phi}^G \hat{\mathbf{a}}_k \equiv \sum_{i=1}^{\ell} \hat{a}_{k,i} \boldsymbol{\phi}_i^G$. A G-dimensional dynamical model for $\mathbf{x}_k^G$ is set up by writing the original model in terms of $\mathbf{x}_k^G$. Using $\mathbf{x}_k^G \approx \mathbf{\Phi}^G \hat{\mathbf{a}}_k$ and the so-called *Missing Point Estimation* (MPE) expression $[(\mathbf{\Phi}^G)^T \mathbf{\Phi}^G] \hat{\mathbf{a}}_k = (\mathbf{\Phi}^G)^T \mathbf{x}_k^G$ for an optimal (spatial, static) estimation of the collected data[13] from the incomplete information, an POD-MPE ℓ-dimensional reduced-order model for $\hat{\mathbf{a}}_k$ is derived. Whereas the number to determine POD coefficients has remained the same as in the normal POD model, (much) less state-dependent coefficients need to be updated. Once the next time-step solution $\hat{\mathbf{a}}_{k+1}$ of the POD-MPE model has been found, it is used to determine the estimates of the original high-dimensional model at the remaining $n - G$ spatial points as $\hat{\mathbf{x}}_k^{n-G} = \hat{\mathbf{\Phi}}^{n-G} \hat{\mathbf{a}}_k$. Naturally, the computational gain may be expected to be bigger the smaller G is, though at the cost of a decreased performance. See [19] for detailed description of the method.

Remark 5.2.1. [57] reports the application of the MPE idea in a different, suboptimal way, namely directly to the reduced-order Jacobian term $\mathbf{\Phi}^T \mathbf{J}_n^i \mathbf{\Phi}$ in the inner loop of the Newton solver by selecting G rows of $\mathbf{\Phi}$ such that the control variables (wells) and the boundary conditions are considered mandatory and the remaining points are chosen according to a condition number criterion (generally such that $\mathbf{\Phi}^G \mathbf{\Phi} \approx \mathbf{I}_G$). The scheme is suboptimal in that the construction of both the Jacobian matrix and the residual equations is still based on the total number of gridblocks n, rather that on $G < n$ selected gridblocks as it would be the case in a full POD-MPE implementation. The reason for this choice is that a full POD-MPE implementation requires much more modifications in the existing reservoir simulator code than the suboptimal solution of applying the MPE at the linear solver level only.

5.3 POD reduced order models as solution predictors in the full-order simulation

Determining a POD basis (and, possibly, its recomputation) that is robust enough against flow scenarios different from those used to determine the basis may require many full runs of the high-dimensional model. This may be very time-consuming due to the necessity to employ an iterative solving procedure to propagate the solution in time. A good initial guess of the solution is therefore of paramount importance. As the dynamic reservoir flow problems are evolution problems, the initial guess is usually taken to be equal to the last time-step solution (constant extrapolation) or a linear combination of the solutions at the last two timesteps (linear extrapolation). While the latter may be preferable in the absence of strongly changing well terms, it may yield worse guesses at the occurrences of rapid source variations (turn off/on wells).

[13] MPE tech in [19–21], is inspired by the *Gappy POD* approach proposed in [85] for recovering missing data from a static image, and has recently also been used for the inverse problems of (aerodynamic) fluid flow reconstruction and sensing (including optimal sensor placement, i.e., how many sensors are really necessary and where to place them such that the full field can be reconstructed accurately) in [49, 292].

This section proposes a different way of using solutions from previous time-steps to obtain an improved initial guess for the current time-step solution. In particular, a number of previous time-step solutions is combined in the POD manner to deliver an orthonormal operator $\mathbf{\Phi}$ which is then used to project both the state and the equation onto. After solving the projected (and thus reduced-order) system, the initial guess for the original system is computed by backprojecting the reduced-order solution using $\mathbf{\Phi}$ (thus just as we do in our common projection based MOR). For instance, in an IMPES (incompressible) two-phase flow formulation, wherein the pressure equation to solve at each time-step is a linear algebraic system of the form $\mathbf{M}(\mathbf{S}_{w,k-1})\mathbf{p}_{o,k} = \mathbf{b}_{k-1}$, with $\mathbf{S}_{w,k}$ and $\mathbf{b}_{k-1}$ being the water saturation and the source term (wells and gravity) at the last time-step, respectively, the 'extrapolation' scheme thus per time-step: 1) solves an SVD (or EVD) problem to determine, hopefully, persistent global (spatial) correlations among the solution vectors of N_k previous time steps yielding[14] an ℓ_k-dimensional ($\ell_k \leq N_k$) POD-basis $\mathbf{\Phi}_k$, b) solves the ℓ_k-dimensional reduced-order system $\mathbf{\Phi}_k^T\mathbf{M}(\mathbf{S}_{w,k-1})\mathbf{\Phi}_k\mathbf{z}_k = \mathbf{\Phi}_k^T\mathbf{b}_{k-1}$, and c) projects the reduced-order solution $\mathbf{z}_k$ back to the original high-dimensional space to start the 'normal' iteration, i.e., $\mathbf{p}_{o,k}^{(0)} := \mathbf{\Phi}_k\mathbf{z}_k$. The solution of the reduced-order model can be interpreted as a 'shadow' running in parallel with the solution of high-order model. When the simulation run starts there will, of course, be no previous time-step solutions. The first N_1 times-steps are therefore solved in the high-dimensional space in the usual way.[15] We stress hereby that the POD basis does not need to be determined by a full SVD (or EVD) analysis when new solutions become available. It may instead be approximately updated using fast algorithms (see, e.g., [117, 119, 189]). Only occasionally, when the approximation gets deteriorated, a full analysis needs to be performed again. If N_k is small to moderate (in our examples we used max. 15 previous time-step solutions) or the number of basis functions is known a priori to be rather small, a full (iterative) analysis may be the preferable option.

It is important to note that, if the subsequently applied iterative process (Preconditioned Conjugate Gradient (PCG) method in our M-symmetric case) is considered as an exact solution method, the initial guess obtained as suggested above does not necessarily result in a reduced number of iteration steps. Advantageous effects are expected to occur in approximative iteration process, when the process is terminated once a prescribed accuracy in the decrease of the relative norms of the residual vectors is achieved (here, $\frac{\|\mathbf{M}_{k-1}\mathbf{p}_{0,k}^{(j)} - \mathbf{b}_{k-1}\|}{\|\mathbf{M}_{k-1}\mathbf{p}_{0,k}^{(0)} - \mathbf{b}_{k-1}\|} \leq \epsilon_{PCG}$). The convergence process of Krylov iteration methods is to a large extent determined by the smallest

[14] Subscript 'k' in N_k and ℓ_k emphasizes the fact that these numbers do not need to be fixed.

[15] The material in this section comes from [182]. A referee pointed us to the fact that this work is, conceptually, strongly related to that in [67]. There, the projection basis is the orthonormal basis (obtained, e.g., a Modified Gram-Schmidt process) of the Taylor-based extrapolated initial guesses from [66]. Obtaining initial guesses based on Taylor-based approximations is computationally a low-cost operation. No performance comparison has been made in this thesis between this method and our method. In any case, it is always possible to take a linear combination $\mathbf{x}^{(0)} := \alpha\mathbf{x}_1^{(0)} + (1-\alpha)\mathbf{x}_2^{(0)}$ of two initial guesses $\mathbf{x}_1^{(0)}$ and $\mathbf{x}_2^{(0)}$ for solving $\mathbf{Ax} = \mathbf{b}$ such that $\|\mathbf{r}^{(0)}\| \leq min(\|\mathbf{r}_1^{(0)}\|, \|\mathbf{r}_2^{(0)}\|)$, where $\mathbf{r}^{(0)} := \mathbf{Ax} - \mathbf{b}$ (and $\mathbf{r}_1^{(0)}$ and $\mathbf{r}_2^{(0)}$ defined accordingly). The value of α minimizing $\|\mathbf{r}^{(0)}\|^2$ is easily found to be given by $\alpha_* = -\frac{<\mathbf{r}_2^{(0)}, \mathbf{r}_1^{(0)} - \mathbf{r}_2^{(0)}>}{\|\mathbf{r}_1^{(0)} - \mathbf{r}_2^{(0)}\|^2}$.

eigenvalues of the system matrix (here $\mathbf{M}_{k-1}$) and typically shows a three-stage behavior. For an arbitrary initial guess, an iteration usually manages to ignore the effect of these troublesome eigenvalues in the early stages. After a short initial period of residual decrease, convergence slows down, as the iteration is then forced to take these troublesome eigenvalues into account. After this period of stagnation, the number of iterations shows a superlinear residual decrease. The most benefit can thus be expected in a substantially decrease of the residual before this period of superlinear convergence starts (i.e., in the stagnation period) or shortly thereafter.

5.3.1 Model problems

The models chosen to asses the performance of the proposed method all represent two-phase (oil-water) immiscible porous media flow, but differ in the number of spatial dimensions, heterogeneity, rock-fluid properties, gravity effects, as well as in the number and position of water injection and oil production wells.

Model equations: IMPES formulation

We consider implicit-pressure explicit-saturation (IMPES) scheme and investigated the scope to accelerate the iterative solution of the pressure equation, which is by far the most time-consuming part of any IMPES scheme. The equations describing the process are the equations of mass conservation for both phases $\alpha \in \{w, o\}$, where w and o refer to water and oil respectively (Chapter 2),

$$\frac{\partial \left(\phi \rho_\alpha S_\alpha\right)}{\partial t} + \nabla \cdot \left(\rho_\alpha \mathbf{u}_\alpha\right) \quad = \quad \rho_\alpha \tilde{q}_\alpha \,, \tag{5.3.1}$$

and the Darcy equations for both phases:

$$\mathbf{u}_\alpha \quad = \quad -\frac{k_{r\alpha}}{\mu_\alpha} \mathbf{K} \left(\nabla p_\alpha - \rho_\alpha \tilde{\mathbf{g}}\right), \tag{5.3.2}$$

with the fluids satisfying the usual closure relationships: $\sum_{\alpha=w,o} S_\alpha = 1$ and $P_c := P_c\left(S_w\right) = p_o - p_w$. In all examples the capillary pressure P_c is assumed to be negligible, so that $p_w \approx p_o =: p$. Moreover, the fluids are assumed to be either incompressible or slightly compressible, while the rock is considered incompressible. In the incompressible case, equations (5.3.1)–(5.3.2) can be rewritten as the following system of pressure-saturation equations:

- *Pressure equation:*

$$\nabla \cdot \mathbf{u}_T \equiv - \sum_{\alpha=w,o} \nabla \cdot \tilde{\lambda}_\alpha \mathbf{K} \left(\nabla p - \rho_\alpha \tilde{\mathbf{g}}\right) \quad = \quad \tilde{q}_T. \tag{5.3.3}$$

- *Saturation equation:*

$$\phi \frac{\partial S_w}{\partial t} \quad = \quad = -\nabla \cdot \mathbf{u}_w + \tilde{q}_w. \tag{5.3.4}$$

Here, $\mathbf{u}_T := \mathbf{u}_w + \mathbf{u}_o$ is the total velocity, $\tilde{\lambda}_\alpha = k_{r\alpha}/\mu_\alpha$ is the phase mobility, and $\tilde{q}_T := \tilde{q}_w + \tilde{q}_o$ is the total well flow rate. The relative permeability $k_{r\alpha}$ in equation (5.3.2) and therefore also the mobility $\tilde{\lambda}_\alpha$ in equation (5.3.3) are nonlinear functions of S_w, while the water velocity $\mathbf{u}_w$ in equation (5.3.3) depends on the pressure according to equation (5.3.2). Equations (5.3.3) and (5.3.4) therefore form a coupled system of nonlinear equations, but can, under the assumptions made, individually be considered as linear equations with time-varying coefficients. Numerically, the reservoirs are represented using block-centered Cartesian grids, with the continuous spatial domain uniformly discretized into n cells of dimensions $\Delta x, \Delta y$ and Δz in x, y and z direction respectively. At each time step, using a 5-point stencil in two dimensions (2D) and a 7-point stencil in 3D, the pressure p_i of each grid block $i = 1, ..., n$ is deduced from the linear matrix system:

$$\mathbf{T}\mathbf{p} \quad = \quad \mathbf{q} + \mathbf{g}, \tag{5.3.5}$$

where $\mathbf{T}$ and $\mathbf{g}$ are respectively the (symmetric) inter-block transmissibility matrix and the inter-block gravity transmissibility vector, $\mathbf{q}$ is the vector containing the well contributions, and the n-dimensional vector $\mathbf{p}$ contains the unknown grid-block pressures. The water saturation S_w is then updated in each grid block with an explicit scheme using equation (5.3.4), with the velocity field deduced from the calculated pressure solution using equation (5.3.2) and the water velocity $\mathbf{u}_w$ determined from the total velocity using:

$$\mathbf{u}_w \quad = \quad f_w \left[\mathbf{u}_T + \mathbf{K}\tilde{\lambda}_o \left(\rho_w - \rho_g \right) \tilde{\mathbf{g}} \right], \tag{5.3.6}$$

where $f_w := \tilde{\lambda}_w \Big/ \left(\tilde{\lambda}_w + \tilde{\lambda}_o \right)$ is the water fractional flow. Saturations are updated based on the orientation of the water velocity, with time steps constrained by Courant-Friedrichs-Levy conditions to ensure stability. Saturation-dependent parameters are averaged using a conventional upstream weighting [25].

The most time-consuming part of the simulation is clearly obtaining the pressure solution $\mathbf{p}$. The pressure equation (5.3.5) was therefore the model against which the simulation results were measured. The performance of the initial guess obtained by the algorithm described above, which was defined to be the average simulation runtime for each of the three simulation models, was compared with the performance of the initial guess being the last time-step solution.

Remark 5.3.1. (*Nonlinear pressure case.*) When the fluids and the reservoir rock are assumed to be slightly compressible, the pressure equation takes the following

form:

$$-\sum_{\alpha=w,o}\frac{1}{\rho_\alpha}\nabla\cdot\left[\tilde{\lambda}_\alpha\rho_\alpha\mathbf{K}\left(\nabla p-\rho_\alpha\tilde{\mathbf{g}}\right)\right] \quad = \quad \phi\left[c_o+c_\varphi+\left(c_w-c_o\right)S_w\right]\frac{\partial p}{\partial t}+\tilde{q}_T. \tag{5.3.7}$$

The spatially and temporary discretized pressure equation in the IMPES formulation is then

$$(\mathbf{T}_{k+1}-\mathbf{D}_k)\mathbf{p}_{k+1} \quad = \quad -\mathbf{D}_k\mathbf{p}_k+\mathbf{q}_k, \tag{5.3.8}$$

where k denotes the index of the current simulation time t_k, matrix $\mathbf{T}_{k+1}$ is again a transmissibility matrix containing discretized left-hand-side flux terms from equation (5.3.7) and evaluated at time $t_{k+1}=t_k+\Delta t_k$, while the 'current-time' diagonal matrix $\mathbf{D}_k$ consists of elements $(\phi_i/\Delta t_k)\left[c_o+c_\varphi+\left(c_w-c_o\right)S_w\right]_i$, with i denoting the grid-block index. Because the transmissibilities now depend also on pressure at t_{k+1}, the pressure equation (5.3.8) is non-linear, and requires an iterative treatment on its own, which is usually performed with linearization schemes as the successive-approximation method or the more advanced Newton-Raphson method. This case has not been treated in this thesis, but we expect that an acceleration algorithm for these methods could, for this model, consist of performing, at each time-step k, a few successive-approximation steps $r=0,1,\ldots$, restricted to the $\mathbf{\Phi}_k$-subspace: $\mathbf{\Phi}_k^T[\mathbf{T}(\mathbf{p}_{k+1,r}^{(0)})-\mathbf{D}_k]\mathbf{\Phi}_k\mathbf{z}_{r+1}=\mathbf{\Phi}_k^T(\mathbf{D}_k\mathbf{\Phi}_k\mathbf{z}_r+\mathbf{q}_k)$, $\mathbf{p}_{k+1,r+1}^{(0)}=\mathbf{\Phi}_k\mathbf{z}_{r+1}$, whereby the iteration is started for $r=0$ using the last time-step solution, i.e., $\mathbf{p}_{k+1,0}^{(0)}=\mathbf{p}_k$. For another, more general framework for the use of POD to compute a better initial guess for a Newton iterative method, the reader is referred to [264, 265]. [16]

5.3.2 Test cases

Numerical tests were performed on three different reservoir models, with a broad range of permeability fields as shown in Figure 5.4 below. The first model (Model A) was taken from [24], and the second one (Model B) was taken from [47]. The third model (Model C) was also taken from [24] and is an upscaled version of a widely-used test model described in [65]. The relative permeability model used in all the test cases is the conventional Corey-type relation:

$$k_{ro}=k_{ro}^0\left(1-S_w^*\right)^{n_o} \quad , \quad k_{rw}=k_{rw}^0\left(S_w^*\right)^{n_w}, \tag{5.3.9}$$

where the variable S_w^* is a normalized water saturation defined as

$$S_w^*=\frac{S_w-S_{wc}}{1-S_{or}-S_{wc}} \quad , \quad 0\leq S_w^*\leq 1 \quad , \tag{5.3.10}$$

[16] Interestingly enough, these contributions and our work [182] were published around the same time.

n_o and n_w are the Corey exponents, S_{or} is the residual oil saturation, S_{wc} is the connate (irreducible) water saturation and k^0_{ro} and k^0_{rw} are the end-point saturations. Initial conditions were taken as static equilibrium for the pressures and connate water saturation for the saturations. A short description of the main properties specific to each of the models follows.

Model A:

This model (Figure 5.4A) corresponds to a 2D case with a fine mesh composed of 30 by 30 cells. The porosity is constant and equal to 0.25, whereas the permeability field is a binary distribution of high (100 mDarcy, in white) and low (1 mDarcy, in black) values (1 mDarcy = 9.87×10^{-16} m^2). At one corner of the grid, water is injected at a constant rate of 5 m^3/day, and at the opposite corner fluids are produced at a constant pressure equal to 7.6 MPa. The grid-block dimensions are $10 \times 10 \times 1$ m. The Corey-data are: $k^0_{ro} = 1.0$, $k^0_{rw} = 0.4$, $S_{or} = S_{wc} = 0.2$, n_w=1.5, and n_o=2.0. The viscosities are $\mu_o = 0.8 \times 10^{-3}$ Pa $\cdot$ s and $\mu_w = 1.0 \times 10^{-3}$ Pa $\cdot$ s.

Model B:

This model (Figure 5.4B) is a 2D model with 2025 (45×45) uniform grid blocks. The grid-block dimensions are $10 \times 10 \times 10$ m. The left-hand side of the reservoir is equipped with a row of 45 water injection wells, one in each grid block. At the right-hand side, 45 production wells are located. The injection side is operated at a total water flow-rate of 242 m^3/day, while the producers are operated under constant pressure of 27.5 MPa. The porosity is constant and equal to 0.2. The Corey-data are: $k^0_{ro} = k^0_{rw} = 1.0$, $S_{or} = S_{wc} = 0.1$, and $n_o = n_w = 1.0$. The viscosities are taken to be equal, $\mu_o = \mu_w$=1.0 $\times$ 10^{-3} Pa $\cdot$ s. These data lead to straight-line relative permeabilities and a unit mobility ratio. In order to simulate a more non-linear waterflood situation, also quadratic relperms were used, i.e. $n_o = n_w = 2.0$.

Model C:

Model C (Figure 5.4C) is a 3D model containing 93,500 cells ($20 \times 55 \times 85$). Each cell is a $6.096 \times 3.048 \times 0.6096$ m Cartesian gridblock. Water is injected at a constant rate of 795 m^3/day in a vertical injector placed in the middle of the reservoir, while four vertical producers at the corners are operated at a constant pressure at the top of the reservoir of 27.5 MPa. The wells are connected to the grid blocks over the entire height of the reservoir. The Corey-data in this case are: $k^0_{ro} = k^0_{rw} = 1.0$, $S_{or} = S_{wc} = 0.2$, and $n_o = n_w = 2.0$. The viscosities are $\mu_o = 3.0 \times 10^{-3}$ Pa $\cdot$ s and $\mu_w = 0.3 \times 10^{-3}$. Since in this model gravity effects are included, the fluid densities are also important. These are taken to be $\rho_o = 850$ kg/m^3 and $\rho_w = 1000$ kg/m^3.

In all examples the total injection rate was controlled. In case of more than one injection well, this flow rate is distributed among the wells in proportion to the total fluid mobilities in the corresponding grid blocks. Therefore, strictly speaking, as long as the fluid mobility in the injection segments changes, so do the injection rates. However, in the following we will use the term 'varying injection rates' only to indicate varying total injection flow rates. These total flow rate variations are such that the individual well injection rates change much more abrupt than in the case of constant surface rates.

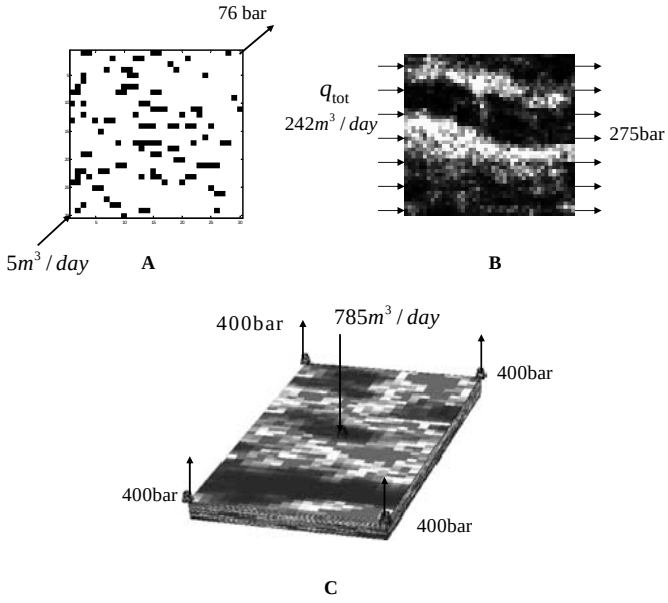


Figure 5.4: Permeability fields for the three test cases used in the study. Model A (horizontal, 30 x 30); Model B (horizontal, 45x45); Model C (3D, 20x55x85)

5.3.3 Results

The performance of our proposed method, in the sequel referred to as the 'new method', was assessed by comparing its results to those obtained with the solution of the preceding time step as the initial guess, in the sequel referred to as the 'normal method'. Because the transmissibility matrix $\mathbf{T}$ in equation (5.3.5) is symmetric positive-definite, natural candidates to solve equation (5.3.5) were conjugate-gradient-based Krylov subspace methods. Other methods tested were the minimal residual Krylov method and the fixed-point successive overrelaxation scheme. We will only discuss the results for the preconditioned conjugent gradient (PCG) cases because these are the most relevant for large-scale problems. All comparisons were made using Matlab 7 on a Pentium IV machine with 1 Gb RAM. In almost all IMPES simulations the new method performed better

than the normal method. The latter yielded slightly faster solutions when the total fluid mobility $\lambda_T = \lambda_o + \lambda_w$ and the total volumetric well rates were independent of the saturation distribution, or for a strong tolerance bound, i.e. when the stopping criterion of the iterative method was given a low value ($\leq 10^{-4}$, say). Saturation-independent total mobility occurs for a unit end-point mobility ratio ($k^0_{ro}/\mu_o = k^0_{rw}/\mu_w$) accompanied by straight-line relative permeabilities ($n_o = n_w = 1$). In that case the transmissibility matrix $\mathbf{T}$ in equation (5.3.5) does not change during the simulation and has thus to be computed only once. When at the same time also the volumetric well rates are kept constant, the pressure distribution is constant during the simulation. The preceding time step solution being equal to the 'sought' solution makes the use of any solver of course superfluous, so these special cases can not be used as benchmarks. Furthermore, particular combinations of permeability field, well configuration and injection strategy were influential as well. For instance, in Model B the row of 45 injectors and the opposite-side row of 45 producers together maintained a near-steady-state pressure field when the injection rate was constant, thus causing the number of iterations also in the normal method to be very low. In all other cases, in particular when flow-rates are varied over time, the saturation-dependent parameters and the pressure distribution must be updated frequently, and in these situations the new method often substantially accelerated the convergence of the iterative algorithms. A large gain in performance was observed in situations of: 1) no preconditioning of linear algebraic systems, 2) strong dependence of transmissibilities on saturation distribution (i.e. strongly nonlinear relative permeabilities), 3) strong heterogeneity, and 4) changing operating conditions (injection rates).

Figure 5.1 shows the results for the PCG method with a point incomplete Cholesky factorization as the preconditioner determined every 15^{th} time step, the 'optimal' drop tolerance being found to be 10^{-3}. Other preconditioned conjugate-gradient type iterative methods (bi-conjugate-gradient with point incomplete LU preconditioning, etc.) yielded similar behavior. We tested model A using steady-state and rapidly fluctuating random inputs (tests 1 and 2), and we tested three versions of model C: two truncated versions, C1 and C2, with 22,000 and 55,000 grid blocks respectively, and the full version, C3, with 93,500 grid blocks. The truncated models were reduced in height, and the injection rates were reduced proportionally. All versions of model C were tested with steady-state flow rates (tests $3-5$) and with randomly fluctuating rates (tests $6-8$). In case of steady–state rates, the sum of the injection rates was taken equal to the sum of the production rates, whereas in the case of random rates the averages of injection and production rates where equal. While the difference in simulation time was very much in favor of the new method for all cases, the largest improvement was obtained when the rates were fluctuating.

Figure 5.5 displays the typical behavior of the iteration process, expressed as the number of iterations per time step, for the new and the normal methods. This figure corresponds to test 5 in Table 5.1 (model C3, fixed injection rates) with a PCG tolerance of 10^{-5}. It required 2270 s to simulate over 300 time steps, of which 191 s (8.4 %) were required to compute the POD basis functions and another 76 s (3.3 %) to solve the reduced system of equations. This computational overhead

is more than offset by the decrease in the total number of iterations which, in this test case, resulted in a total reduction in the simulation time of 36%. In this particular case we used maximally 15 basis functions to represent the 93,500 pressure variables.

			POD-based		Normal		
Test no.	Model No. of steps	Tolerance	Simulation time (s)	No. of iterations	Simulation time (s)	No. of iterations	Improvement in time (%)
1	A	1.0E-05	18 (2 for PCG)	342	18 (4 for PCG)	1527	0
	1000	1.0E-06	19 (3 for PCG)	565	23 (9 for PCG)	4172	17
		1.0E-07	20 (5 for PCG)	850	26 (12 for PCG)	6244	23
		1.0E-08	21 (6 for PCG)	1153	28 (14 for PCG)	7648	25
2	A, random	1.0E-05	18 (2 for PCG)	270	29 (16 for PCG)	7314	38
	1000	1.0E-06	19 (2 for PCG)	417	30 (17 for PCG)	8274	37
		1.0E-07	20 (3 for PCG)	658	31 (18 for PCG)	9441	35
		1.0E-08	22 (4 for PCG)	1080	32 (19 for PCG)	10241	31
3	C1	1.0E-05	480 (132 for PCG)	1414	935 (612 for PCG)	7772	49
	300	1.0E-06	587 (248 for PCG)	3025	1066 (777 for PCG)	9958	45
		1.0E-07	750 (410 for PCG)	4895	1208 (919 for PCG)	11414	38
4	C2	1.0E-05	1720 (752 for PCG)	3595	3560 (2829 for PCG)	14036	52
	300	1.0E-06	2500 (1618 for PCG)	8007	4510 (3773 for PCG)	18649	45
5	C3	1.0E-05	2770 (1228 for PCG)	3308	4360 (3090 for PCG)	8775	36
	300	1.0E-06	4350 (2685 for PCG)	7614	7800 (6560 for PCG)	19089	44
6	C1, random	1.0E-05	475 (132 for PCG)	1455	1120 (815 for PCG)	10404	58
	300	1.0E-06	251 (590 for PCG)	3010	1195 (909 for PCG)	11521	51
		1.0E-07	710 (372 for PCG)	4638	1273 (989 for PCG)	12702	44
7	C2,random	1.0E-05	1860 (908 for PCG)	4328	5020 (4300 for PCG)	21286	63
	300	1.0E-06	2680 (1670 for PCG)	8366	5400 (4690 for PCG)	23649	50
8	C3,random	1.0E-05	3220 (1617 for PCG)	4355	9850 (8400 for PCG)	24139	67
	300	1.0E-06	4830 (3180 for PCG)	8695	10800(9450 for PCG)	26741	55

Table 5.1: Results from using the new (POD-based) method and the normal method, using PCG with preconditioning at every 15^{th} time step with incomplete Cholesky factorization (tolerance = 10^{-3}). The simulation time is the total simulation time for all time steps; the time between brackets refers to the net time used by the iterative solver. The number of iterations is the total number of iterations for all time steps. Models C1 and C2 refer to truncated versions of model C with 20x55x20 and 20x55x50 grid blocks respectively. Model C3 refers to the full version of model C with 20x55x85 grid blocks.[182]

Figure 5.6 displays the number of iterations versus the norm of the residual for the normal and the new methods, for some typical time steps. The figure illustrates that as of the third time step the new method indeed results in improved convergence. Moreover, both in the normal and the new case one can observe the three typical phases during the convergence of Krylov iteration methods: after a short initial period of residual decrease convergence slows down, to start to decrease superlinearly at some point later.

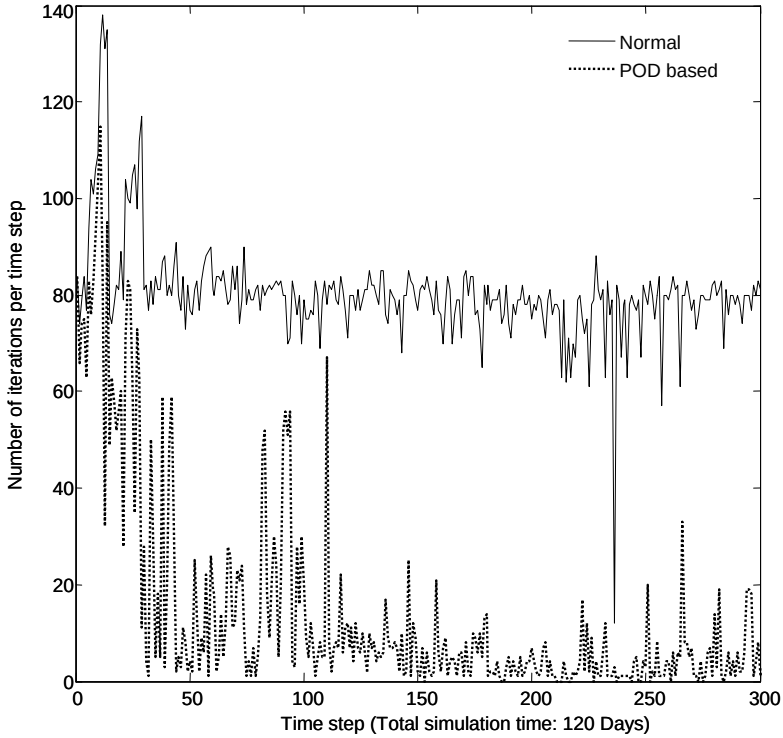


Figure 5.5: Number of iterations per time step for the normal method and the new, POD-based, method applied to model C3. Iterative solver: PCG (tolerance 10^{-5} with preconditioning using incomplete Choleski factorization (tolerance 10^{-3}) every 15^{th} timestep.

Figure 5.7 displays the results for the three versions of model C, using steady-state and randomly fluctuating injection rates (source terms), and two different tolerances for the PCG solver. It clearly illustrates that the improvement in computational performance are larger for fluctuating rates than for steady-state rates. The transient pressure fluctuations are apparently effectively captured by the low-order representation.

5.3.4 Conclusions

The proposed POD-based acceleration method for iterative solvers resulted in improved computational efficiency in almost all cases considered. Exceptions were pathological cases with steady-state inputs and near-time invariant parameters, which display near-constant behavior. Reductions in computing time up 67% were observed for test problems where preconditioning was used. The best result was achieved in the largest test problem which involved two-phase (oil-water) porous-media flow through a 3D reservoir, simulated with a 93,500 grid block finite difference model. Improvements in computational efficiency were observed

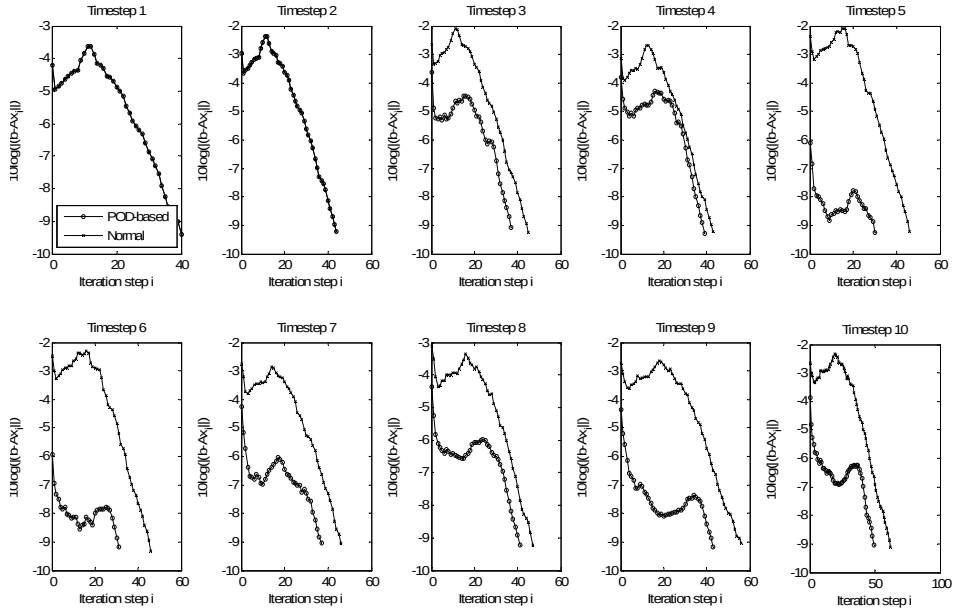


Figure 5.6: Number of iterations versus the 10-log norm of the residual for the first 10 time steps in test 7 (Model C2 with randomly fluctuating injection and production rates).

for different iterative solvers, including preconditioned conjugate gradient (PCG), minimal residual (MinRes), and successive overrelaxation (SOR) methods. The method seems to be particularly attractive for problems with time-varying parameters or time-varying source terms, and could, in theory, also be applied to accelerate the solution of nonlinear systems of equations using Newton-Raphson iteration. Further work is required to assess the efficiency of the method for such an application. Initial steps have been undertaken in [265].

5.4 Summary and conclusions of the chapter

Chapter 3 (Sec. 3.3) summarized a majority of the projection-based methods that can be found in the literature for addressing the problem of order reduction of large nonlinear models. In this thesis, because of the complexity and potentially very large size of multi-phase reservoir models, it was decided to employ data-driven projection subspaces for MOR of two-phase (waterflood) models. In these methods, the basis functions are determined by processing data obtained from numerical simulations of the underlying high-dimensional model. In particular, the applicability and performance of strategies based on the *Proper Orthogonal De-*

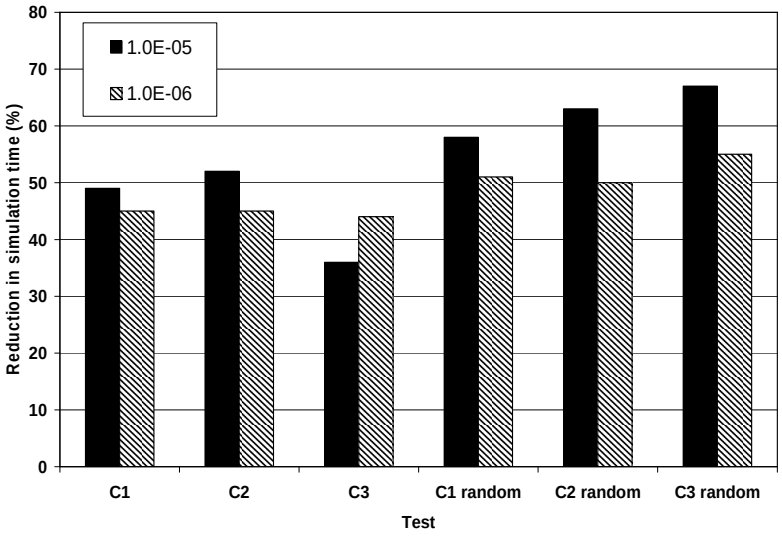


Figure 5.7: Reduction in simulation time for different (truncated) versions of model C (C1, C2 and C3 with 22000, 55000 and 93500 grid blocks respectively), different injection rates (constant or random) and different tolerances of the PCG solver (10^{-5} and 10^{-6}).

composition (POD) methodology have been assessed. A POD basis of a data set is orthonormal and it is optimal in that it guarantees the best reconstruction, in the mean square error sense, of the elements of the set among all (linear) bases of the same size. The basis functions can be determined either by a SVD analysis of the collected data or by an eigenvalue analysis of the data correlation matrix (or the covariance matrix if the mean of the data set is subtracted from the individual data prior to the analysis).

The state-vector in two-phase systems in this thesis consists of the grid-block oil-pressures p_0 and water saturations S_w. Because of the fact that POD tends to favorize variables which numerical values show greater variability, in the applications of POD in this thesis optimal bases for the two different groups, pressures and saturations, are obtained separately: instead of solving one eigenvalue (or SVD) problem to obtain a single (dense) data matrix, two separate eigenvalue (SVD) problems are solved (one for $\mathbf{X}_p := [\mathbf{p}_{o,1}, \ldots, \mathbf{p}_{o,N}]$ and one for $\mathbf{X}_S := [\mathbf{S}_{w,1}, \ldots, \mathbf{S}_{w,N}]$), yielding two separate projection matrices, $\mathbf{\Phi}_p$ and $\mathbf{\Phi}_S$, for pressures and saturations, respectively. The 'total' projection matrix is formed as $\mathbf{\Phi} := \begin{bmatrix} \mathbf{\Phi}_p & \mathbf{0} \\ \mathbf{0} & \mathbf{\Phi}_S \end{bmatrix}$. This allows for choosing a different degree of reduction for the pressures and the saturations.

Being data-driven, POD bases depend on the input signals the state-snapshots are generated with. Regarding the quality of the approximation, numerical exper-

iments in this research and elsewhere indicate that:

- a given flow scenario can be reproduced accurately if enough snapshots have been used and 'enough energy' has been captured in the POD modes.
- the POD approach is quite robust against non-training input signals if the snapshots are generated by exciting the system by manipulating the injection and/or production profiles (i.e., flow-rates and/or bottom-hole pressures) consistent with the expected range of operating conditions.
- an adequate approximation of the high-dimensional behavior generally requires (much) less POD basis functions for pressures than for saturations, which was attributed to the traveling nature of the saturation front and the time-averaging property of the (standard) POD approach.

The overall computational cost of a POD approach consists of two main parts: a) preprocessing cost of generating 'representative' snapshots spanning a large portion of operating conditions of interest and, subsequently, determining the POD basis, and b) the cost of solving the reduced-order model. As for the former, if either the number of pre-processing runs is large or the time-steps in (any of) the runs were small, the POD analysis on the complete data set may be computationally too demanding (data matrices are dense in general). To reduce this cost one may decide to base the POD analysis (i.e. solving a SVD or an EVD problem) on: 1) 'dynamically relevant' snapshots (not trivial at all), 2) 'clustered' snapshots, and 3) 'low-rank' matrix decomposition (SVD or EVD) update algorithms. Regarding the cost of solving the reduced-order model, despite a generally drastic reduction in the size of the model, there is no guarantee that the simulation of the reduced-order model will actually be faster than the simulation of the high-order model (and if there is a gain, it is rarely spectacular), which is due to that fact that at each time-step state-dependent coefficient matrices and their products needs to be evaluated. The issue is even more manifested in a fully time-implicit scheme using the Newton-Raphson solving procedure, where the coefficients and the matrix product are evaluated at each inner-loop iteration. The computational cost of forming the matrix products could potentially be reduced by not computing the POD basis coefficients in the usual POD manner, but rather estimating these from a partial set of the original data at a selected number of points in the spatial domain. Naturally, the computational gain may be expected to be bigger the smaller this number is, though at the cost of a decreased performance.

Running fine-scale simulations in order to determine a (robust) POD basis may be very time-consuming due to the necessity to employ an iterative solving procedure to propagate the solution in time. A good initial guess of the solution is therefore of paramount importance. As the dynamic reservoir flow problems are evolution problems, the initial guess is usually taken to be equal to the last time-step solution. Section 5.3 has proposed an algorithm for determining an improved initial solution guess by using solutions from previous time-steps. In particular, a number of previous time-step solutions is combined in the POD manner to deliver an orthonormal operator Φ which is then used to project both the state and the

equation onto. After solving the projected (and thus reduced-order) system, the initial guess for the original system is computed by backprojecting the reduced-order solution using $\mathbf{\Phi}$. The solution of the reduced-order model can be interpreted as a 'shadow' running in parallel with the solution of the high-order model. When the simulation run starts there will, of course, be no previous time-step solutions. First few times-steps are therefore solved in the high-dimensional space in the usual way. When applied to two-phase (incompressible) two-phase flow in an IMPES formulation (Chapter 2), the proposed POD-based acceleration method for iterative solvers resulted in improved computational efficiency in almost all cases considered. Exceptions were pathological cases with steady-state inputs and near-time invariant parameters, which display near-constant behavior. The highest reduction in computing time (67%) was achieved in the largest test problem which involved two-phase (oil-water) porous-media flow through a 3D reservoir. The method seems to be particularly attractive for problems with time-varying parameters or time-varying source terms, and could, in theory, also be applied to accelerate the solution of nonlinear systems of equations using Newton-Raphson iteration. Further work is required to assess the efficiency of the method for such an application.

Finally, an important system-theoretical connection of POD and the method of balanced truncation (BTR, Chapter 4) has been shown in Subsection 5.1.3. Essentially, when applied to LTI systems, POD modes for state impulse-responses are the dominant eigenvectors of the controllability Gramian, or, in other words, the most controllable modes of the particular state-space realization (a change in the state-space basis changes the controllability properties of the model). Connection between POD and the observability of the system can be illustrated by considering its 'dual' (or 'adjoint') system.

6 CHAPTER

Waterflood Optimization Using Reduced-Order Bases

In this chapter, which is a shorter and slightly adapted version of [272], (POD-based) reduced-order models are employed to accelerate 'lifecycle' optimization of waterflooding processes by dynamically controlling flowrates in (smart) wells.[1] The to-be-accelerated solution is based on optimal control and the adjoint equations method as developed in [46]. Optimal controls are determined iteratively by performing several forward simulations of the reservoir model and backward simulations of the corresponding adjoint system of equations. Section 6.1 provides an in-a-nutshell mathematical formulation of the optimization problem and the specific method employed to solve it. Subsection 6.1.1 describes the considered constrained optimization problem and reformulates it as a unconstrained optimization problem using the classical method of Lagrange multipliers. The modified optimization problem is solved by a steepest-ascend gradient method, with the gradients of the objective function w.r.t. the controls determined using optimal control theory (OCT) and the adjoint equations method. The elementary steps in the derivation of the full-order and the reduced-order adjoint systems are given in Subsections 6.1.2 and 6.1.3, respectively. In order to decrease the time needed to calculate optimized controls, a nested-loops iterative scheme is presented in Section 6.2, where the inner loop makes use of a truncated basis of POD functions to calculate optimized injection and production rates. In Section 6.3, the methodology is applied to a 2-dimensional, 2-phase reservoir model and compared to the full-order optimal control of [46]. The algorithm was designed and the simulation performed in the MSc thesis [271]. Section 6.4 concludes the chapter.

[1] The problem to solve in this chapter is 'uncertainty-free', meaning that the subsurface reservoir parameters and parameters that influence fluid flow ($k_{r\alpha}(S_w)$ and $P_c(S_w)$ relationships) are exact and the optimization is based on a single reservoir model. As no model updating takes place here, we refer to such optimization as 'open-loop'. For 'robust reservoir optimization', i.e., optimization in presence of uncertainties in reservoir models, see e.g. [144, 302].

6.1 The optimization problem and its solution in a nutshell

Lifecycle waterflood optimization generally aims at maximizing, over a time interval $[0, T]$, the ultimate oil recovery, in which case the quantity to maximize can be expressed in terms of water saturation at the final time T as $\sum_{j=1}^{N_{gridblocks}} (S_w(T))_j$, or at maximizing a net-present value (NPV), i.e. the sum of the incremental discounted oil production income and water injection and production costs over the life of the reservoir. In this study separate water injection costs were not considered and the NPV objective function was:

$$J = \sum_{k=0}^{N-1} \frac{\sum_{j=1}^{N_{prod}} \left[\; -r_o (q_o^*)_k^j - r_w (q_w^*)_k^j \; \right]}{(1+b)^{\tau^k}} \Delta t_k \tag{6.1.1}$$

$$= \sum_{k=0}^{N-1} J_k, \tag{6.1.2}$$

where $(q_o^*)^j \leq 0$ and $(q_w^*)^j \leq 0$ are the production rates (at surface conditions) $[\frac{m^3}{s}]$ in wells $j = 1, \ldots, N_{prod}$, $r_o > 0$ and $r_w < 0$ are the oil price $[\frac{\$}{m^3}]$ and the water cost $[\frac{\$}{m^3}]$, respectively, t_k and $\Delta t_k = t_{k+1} - t_k$ are the time and the time-step length corresponding to time step $k = 1, \ldots, N$. The term in the denominator is a discount factor representing the time-value of money (e.g., interesting rate, where b is the discount rate (cost of capital) per year [-] and $\tau^k = \sum_1^k t_k$ is the time expressed in years at time step k.

6.1.1 The optimization problem

In what follows it is more convenient to consider J as an abstract function of its arguments, which here are the state of the system, $\mathbf{x}_{k=0:N-1}$ (here grid-block pressures p_o and saturations S_w; these appear in J implicitly through the production flow rates q_o^* and q_w^* being functions of fluid mobilities), and the well control parameters $\mathbf{u}_{k=0:N-1}$ with respect to which J is to be optimized. In the applications in this chapter no well model was used and the controls $\mathbf{u}$ comprised well flow rates (Section 6.3). For a discussion of other possible types of control parameters and constraints on the controls, see [46].

Let us thus consider (the 'fixed terminal time, free terminal states' problem of maximizing):

$$J(\mathbf{u}_{0:N-1}, \mathbf{x}_{0:N-1}) = \sum_{k=0}^{N-1} J_k(\mathbf{u}_k, \mathbf{x}_k) \quad \text{subject to} \quad \mathbf{g}_{k+1}(\mathbf{u}_k, \mathbf{x}_k, \mathbf{x}_{k+1}) \equiv \mathbf{0} \; (\forall \text{k}), \tag{6.1.3}$$

where, in the numerical examples in this chapter:

$$\mathbf{g}_{k+1}(\mathbf{u}_k, \mathbf{x}_k, \mathbf{x}_{k+1}) \quad = \quad (\mathbf{I} + \Delta t_k \mathbf{A}_k)\mathbf{x}_{k+1} - \mathbf{x}_k - \Delta t_k \mathbf{B}_k \mathbf{u}_k, \tag{6.1.4}$$

which is a quasi-implicit time-discretized approximation of the reservoir PDEs (see Chapter 2). The initial state-conditions $\mathbf{x}_0$ (i.e., initial oil pressures and water saturations) are specified, whereas the final state $\mathbf{x}_N$ is free.

The 'equality' model constraints $\mathbf{g} = \mathbf{0}$ are dealt with by the aid of the classical *method of Lagrange multipliers* for solving equality constrained problems [176]. The modified objective function (*Lagrangian function*) is:

$$\begin{aligned} \bar{J}(\mathbf{u}_{0:N-1}, \mathbf{x}_{0:N}, \boldsymbol{\lambda}_{1:N}) \quad &:= \quad \sum_{k=0}^{N-1} \left[\; J_k(\mathbf{u}_k, \mathbf{x}_k) + \boldsymbol{\lambda}_{k+1}^T \mathbf{g}_{k+1}(\mathbf{u}_k, \mathbf{x}_k, \mathbf{x}_{k+1}) \; \right] \\ &= \quad \sum_{k=0}^{N-1} \bar{J}_k(\mathbf{u}_k, \mathbf{x}_k, \mathbf{x}_{k+1}, \boldsymbol{\lambda}_{k+1}). \end{aligned} \tag{6.1.5}$$

A gradient-based iterative scheme: the steepest-ascend algorithm

Optimization problems are commonly solved by *gradient-based* approaches: once the gradients have been obtained, a wide variety of gradient-based techniques is available to iterate to a (locally) optimal solution. At each iteration a gradient is needed to determine the search direction. In this study the cheap and simple steepest-ascend iteration scheme is employed, where the search direction is (positive direction of) the gradient of the (local) Lagrangian function:

$$\mathbf{u}_k^{(j+1)} \quad = \quad \mathbf{u}_k^{(j)} + \epsilon \frac{\partial \bar{J}_k}{\partial \mathbf{u}_k}, \tag{6.1.6}$$

with $k = 0, \ldots, N-1$ (i.e., in each iteration an estimation of the full control trajectory is determined, rather than a single value). As the steepest-ascent scheme performed satisfactory in our examples, with an optimum typically being found in 5-10 iterations, more advanced iteration schemes as, e.g., conjugate-gradient methods or quasi-Newton methods were not considered.

6.1.2 Obtaining the required gradients using optimal control and the adjoint method

Various methods for gradient computation have been developed, including numerical perturbation (finite-difference approximation), sensitivity equation and adjoint method. Among these, the easiest to implement is the *numerical perturbation method* (finite difference approximation). However, the computation involved is proportional to the number of unknown parameters. In our application the unknown parameters are controls of wells (or smart-well segments) and the total number of control variables equals the product of the number of wells (or

well segments) and the number of points in time at which the control variables are changed. For this reason, the numerical perturbation method is generally too expensive for our purposes. The required gradients are in this study computed by the *adjoint method*, since the gradient computation in this method is rather insensitive to the number of unknowns. An adjoint system typically follows from necessary conditions for optimality of the to-be-optimized function, here the Lagrangian $\bar{J}$ defined in (6.1.5):

$$\begin{aligned}\bar{J}(\mathbf{u}_{0:N-1}, \mathbf{x}_{0:N}, \boldsymbol{\lambda}_{1:N}) &= \sum_{k=0}^{N-1} \left[\; J_k(\mathbf{u}_k, \mathbf{x}_k) + \boldsymbol{\lambda}_{k+1}^T \mathbf{g}_{k+1}(\mathbf{u}_k, \mathbf{x}_k, \mathbf{x}_{k+1}) \; \right] \\ &= \sum_{k=0}^{N-1} \bar{J}_k(\mathbf{u}_k, \mathbf{x}_k, \mathbf{x}_{k+1}, \boldsymbol{\lambda}_{k+1}). \end{aligned} \tag{6.1.7}$$

The partial derivatives of $\bar{J}$ w.r.t. to its arguments are easily found to be given by:

$$\left[\; \frac{\partial \bar{J}}{\partial \mathbf{u}_i} \; \right]_{i=0}^{N-1} = \frac{\partial J_i(\mathbf{u}_i, \mathbf{x}_i)}{\partial \mathbf{u}_i} + \boldsymbol{\lambda}_{i+1}^T \frac{\partial \mathbf{g}_{i+1}(\mathbf{u}_i, \mathbf{x}_i, \mathbf{x}_{i+1})}{\partial \mathbf{u}_i}, \tag{6.1.8}$$

$$\left[\; \frac{\partial \bar{J}}{\partial \mathbf{x}_i} \; \right]_{i=0}^{N-1} = \frac{\partial J_i(\mathbf{u}_i, \mathbf{x}_i)}{\partial \mathbf{x}_i} + \sum_{k=i}^{i+1} \boldsymbol{\lambda}_k^T \frac{\partial \mathbf{g}_k(\mathbf{u}_{k-1}, \mathbf{x}_{k-1}, \mathbf{x}_k)}{\partial \mathbf{x}_i}, \tag{6.1.9}$$

$$\frac{\partial \bar{J}}{\partial \mathbf{x}_N} = \boldsymbol{\lambda}_N^T \frac{\partial \mathbf{g}_N}{\partial \mathbf{x}_N}, \tag{6.1.10}$$

$$\left[\; \frac{\partial \bar{J}}{\partial \boldsymbol{\lambda}_i} \; \right]_{i=1}^{N} = \mathbf{g}_i(\mathbf{u}_{i-1}, \mathbf{x}_{i-1}, \mathbf{x}_i) \equiv \mathbf{0}. \tag{6.1.11}$$

(N.B. It is stressed that (6.1.11) is a consequence of the model constraints 'only theoretically'. Numerically, the condition $\mathbf{g}_i(\mathbf{u}_{i-1}, \mathbf{x}_{i-1}, \mathbf{x}_i) = \mathbf{0}$ must be 'ensured' by solving the forward system as accurate as possible - otherwise the equivalence between the original and the modified optimization problems is lost.)

The *first order variation* $\delta\bar{J}(\mathbf{u}_{0:N-1}, \mathbf{x}_{0:N}, \boldsymbol{\lambda}_{1:N}) = \sum_{i=0}^{N-1} \frac{\partial \bar{J}}{\partial \mathbf{u}_i} \delta\mathbf{u}_i + \sum_{i=0}^{N} \frac{\partial \bar{J}}{\partial \mathbf{x}_i} \delta\mathbf{x}_i + \sum_{i=1}^{N} \frac{\partial \bar{J}}{\partial \boldsymbol{\lambda}_i} \delta\boldsymbol{\lambda}_i$ is then given by (using $\delta\mathbf{x}_0 = \mathbf{0}$):

$$\delta\bar{J} = \sum_{i=0}^{N-1} \left(\frac{\partial J_i}{\partial \mathbf{u}_i} + \boldsymbol{\lambda}_{i+1}^T \frac{\partial \mathbf{g}_{i+1}}{\partial \mathbf{u}_i}\right) \delta\mathbf{u}_i + \sum_{i=1}^{N-1} \left(\frac{\partial J_i}{\partial \mathbf{x}_i} + \boldsymbol{\lambda}_i^T \frac{\partial \mathbf{g}_i}{\partial \mathbf{x}_i} + \boldsymbol{\lambda}_{i+1}^T \frac{\partial \mathbf{g}_{i+1}}{\partial \mathbf{x}_i}\right) \delta\mathbf{x}_i + \boldsymbol{\lambda}_N^T \frac{\partial \mathbf{g}_N}{\partial \mathbf{x}_N}. \tag{6.1.12}$$

For an optimal control within its admissible set[2], (sufficiently smooth) $\delta\bar{J}$ is necessarily zero, which can be 'achieved' by 1) 'choosing' $\boldsymbol{\lambda}_N$ to be in the null-space of $(\frac{\partial \mathbf{g}_N}{\partial \mathbf{x}_N})^T$ (i.e., to satisfy the 'terminal condition' $\boldsymbol{\lambda}_N^T \frac{\partial \mathbf{g}_N}{\partial \mathbf{x}_N} = \mathbf{0}^T$) and 2) zeroing all the individual elements in the summations, or equivalently (since $\delta\mathbf{u}_{0:N-1}$ and $\delta x_{1:N-1}$ are free), by setting all the partial derivatives in $(6.1.8 - 6.1.10)$ to zero.

[2]When the control is on the constraint boundary of the admissible set, then the necessary condition for a maximum is $\delta J \leq 0$.

Then, using the fact that the final state $\mathbf{x}_N$ in $\boldsymbol{\lambda}_N^T \frac{\partial \mathbf{g}_N}{\partial \mathbf{x}_N} = \mathbf{0}^T$ is free, thus implying $\boldsymbol{\lambda}_N = \mathbf{0}$, the following **(full-order) adjoint algorithm** is obtained:

1. Let $\mathbf{x}_0$ be given. CHOOSE an *initial control trajectory* $\mathbf{u}_{0:N-1}^{(0)}$. FOR $j = 0, 1, \ldots$ DO:
2. DETERMINE the corresponding *state-trajectory* $\mathbf{x}_{1:N}^{(j)}$ by satisfying (6.1.11) for $i = 1, \ldots, N-1$ (i.e., by solving the forward model) as accurately as possible. Evaluate the objective function.
3. USING $\boldsymbol{\lambda}_N^{(j)} = \mathbf{0}$ DETERMINE, as accurately as possible, the *adjoint trajectory* $\boldsymbol{\lambda}_{1:N}^{(j)}$ by solving backwards the **adjoint system**:

$$\boldsymbol{\lambda}_i^{(j)T} \frac{\partial \mathbf{g}_i}{\partial \mathbf{x}_i} = -\boldsymbol{\lambda}_{i+1}^{(j)T} \frac{\partial \mathbf{g}_{i+1}}{\partial \mathbf{x}_i} - \frac{\partial J_i}{\partial \mathbf{x}_i}. \tag{6.1.13}$$

4. FOR $i = 0, \ldots, N-1$ COMPUTE the *gradient* $\bar{J}_i$ from (6.1.8) as

$$\frac{\partial \bar{J}_i}{\partial \mathbf{u}_i} = \frac{\partial J_i(\mathbf{u}_i^{(j)}, \mathbf{x}_i^{(j)})}{\partial \mathbf{u}_i} + \boldsymbol{\lambda}_{i+1}^{(j)T} \frac{\partial \mathbf{g}_{i+1}(\mathbf{u}_i^{(j)}, \mathbf{x}_i^{(j)}, \mathbf{x}_{i+1}^{(j)})}{\partial \mathbf{u}_i} \tag{6.1.14}$$

   and use it in the steepest-ascend algorithm (6.1.6) to compute an improved control trajectory $\mathbf{u}_{0:N-1}^{(j+1)}$.
5. REPEAT steps 2-4 until no further improvement can be found (note that, in an (exact) optimum, $\frac{\partial \bar{J}_i}{\partial \mathbf{u}_k} = \mathbf{0}^T$).

6.1.3 Reduced-order optimal control and adjoint

For the reduced-order adjoint implementation, instead of the full-order model we add a POD-based reduced-order model as the model constraint to the objective function J. That is, if $\boldsymbol{\Phi}$ is a ℓ-dimensional POD-based reduced-order basis the original n-dimensional system $\mathbf{g}$ is projected onto, then, using a set of low-order Lagrange multipliers $\boldsymbol{\mu}$ the following 'reduced' modified objective function is defined:

$$\bar{J}_{red}(\mathbf{u}_{0:N-1}, \mathbf{z}_{0:N-1}, \boldsymbol{\mu}_{1:N}) = \sum_{k=0}^{N-1} \left[\; J_k(\mathbf{u}_k, \boldsymbol{\Phi}\mathbf{z}_k) + \boldsymbol{\mu}_{k+1}^T \boldsymbol{\Phi}^T \mathbf{g}_{k+1}(\mathbf{u}_k, \boldsymbol{\Phi}\mathbf{z}_k, \boldsymbol{\Phi}\mathbf{z}_{k+1}) \; \right] \tag{6.1.15}$$

Taking the first order variation $\tilde{\bar{J}}_{red}$, and reworking the results, we obtain a reduced-order equation in terms of reduced-order Lagrange multipliers:

$$\boldsymbol{\mu}_k^T \underbrace{\left(\boldsymbol{\Phi}^T \tfrac{\partial \mathbf{g}_{k-1}}{\partial \mathbf{x}_k} \boldsymbol{\Phi} \right)}_{\ell \times \ell} = -\boldsymbol{\mu}_{k+1}^T \underbrace{\left(\boldsymbol{\Phi}^T \tfrac{\partial \mathbf{g}_k}{\partial \mathbf{x}_k} \boldsymbol{\Phi} \right)}_{\ell \times \ell} - \underbrace{\frac{\partial J_k}{\partial \mathbf{x}_k} \boldsymbol{\Phi}}_{1 \times \ell}, \tag{6.1.16}$$

Starting from the final condition $\boldsymbol{\mu}_N^T = \mathbf{0}$, the reduced-order adjoint system (6.1.16) can be integrated backward in time. Because the derivatives in (6.1.16) consist of state-dependent parameters we first calculate the full-order derivatives. They are then transformed and reduced by projecting them on the axes of the low-order model. After calculating $\boldsymbol{\mu}$ every time step we can calculate:

$$\frac{\partial \bar{J}_{k,red}}{\partial \mathbf{u}_k} = \frac{\partial J_k}{\partial \mathbf{u}_k} + \boldsymbol{\mu}_{k+1}^T \left(\boldsymbol{\Phi}^T \frac{\partial \mathbf{g}_{k+1}}{\partial \mathbf{u}_i} \right). \tag{6.1.17}$$

Improved controls using the steepest ascend update (6.1.6): $\mathbf{u}_i^{(j+1)} = \mathbf{u}_i^{(j)} + \epsilon \frac{\partial \bar{J}_{k_{red}}}{\partial \mathbf{u}_i}$. The computational advantage of using reduced-order models in OCT is that the system of equations involves ℓ unknowns, whereas the original system involved n unknowns. This decreases the simulation time considerably, especially for large systems where $\ell << n$. The gain is limited by the necessity of determining the high-order Jacobian matrices (although, in an implicit forward model formulation, these are readily available from the forward simulation) and forming the matrices $\boldsymbol{\Phi}^T \frac{\partial \mathbf{g}_{k-1}}{\partial \mathbf{x}_k} \boldsymbol{\Phi}$ and $\boldsymbol{\Phi}^T \frac{\partial \mathbf{g}_k}{\partial \mathbf{x}_k} \boldsymbol{\Phi}$. The cost of forming these products could possibly be reduced by exploiting the idea of missing point estimation (MPE) as explained in Chapter 5.

For fully-implicit simulation where more than one systems of equations have to be solved during every time step the decrease in simulation time is expected to be even higher (principally because of the absence of the necessity of matrix preconditioning in the reduced-order case).

When we simulated reduced-order reservoir models with the same controls as the original full-order models we obtained almost identical states, as long as a sufficient fraction of the relative energy of the full-order model was preserved. However, if we strongly altered the controls, and therefore the structures of the states, the states of the full-order model were less well represented by the reduced-order model. This is in line with the findings of other authors [219]. Because it is not possible to specify a priori the validity of a reduced-order model, we will use a nested approach in the development of the optimization methodology below, such that the reduced-order results are frequently validated by the full-order model.

6.2 Methodology: a nested-loop iterative scheme

We employ a methodology using nested loops, where the inner iterative loop makes use of a truncated basis of POD functions to calculate optimized injection and production rates. After convergence in this loop the original, high-order model is simulated in the outer loop with the optimized rates and subsequently the POD basis is adapted. This new basis is used in the next inner loop to calculate new optimized injection and production rates. This is depicted schematically in Figure 6.1.

More specifically, to generate a reduced-order model with POD we first simulate the dynamical behavior of the system over time interval 0 to N with an initial choice of $\mathbf{u}$ and compute the NPV. Following reference [18, 258], the initial choice of $\mathbf{u}$ reflects a flooding strategy with conventional horizontal wells with constant pressure along the well bores. Every time step we record and store a total of κ snapshots of pressures and saturations and calculate POD transformation matrices $\mathbf{\Phi}$. Now instead of using the full-order derivatives of the system we use the reduced-order derivatives for the backward calculation and calculate $\boldsymbol{\mu}$ with (6.1.16). Based on the derivatives computed with (6.1.17) we compute new controls and use them for the next reduced-order forward simulation. For this simulation we use the same transformation matrices $\mathbf{\Phi}$. This means that the computational 'overhead' of calculating $\mathbf{\Phi}$ is shared by multiple runs of the reduced-order model. To determine convergence of the inner loop we use a convergence criterion c. The inner loop has converged when the NPV of a reduced-order forward simulation is less than c times the NPV of the previous reduced-order simulation. Convergence of the inner loop may occur because a local maximum of the NPV has been reached or because the controls have changed too much to be accurately captured in the reduced system representation. Entering the outer loop again we use the improved controls in a full-order forward simulation and verify if the controls have indeed maximized the NPV. If necessary, the transformation matrices $\mathbf{\Phi}$ are replaced with new ones that reflect the altered dynamics and the inner loop is repeated. The outer loop has converged when the NPV of the full-order forward simulation is less than the NPV of the previous full-order simulation.

We implemented the methodology in a MATLAB algorithm. The advantage of the methodology is that we use reduced-order forward simulations and reduced-order optimal control, which have a shorter simulation time. A disadvantage is that an improved control of the reduced-order model is not necessarily an improved control for the full-order model. In the numerical example below we will see that in our example this is, however, not a problem. Assessment of the robustness of this approach requires further research on more realistic reservoir models.

6.3 Numerical example

The methodology was tested on a 2-dimensional model with 2025 (45×45) grid blocks, which is the same model as used in references [47, 194, 241, 242]. The dimensions of the reservoir were 450×450×10 m and the permeability field is shown in Figure 6.2. Initially the reservoir is completely saturated with oil. Liquid compressibilities of 1×10^{-10} Pa^{-1} were assigned to both water and oil. At the left side of the reservoir one horizontal water injector was introduced, divided in 45 segments by interval control valves (ICV). A horizontal producer, also divided in 45 segments, was introduced at the opposite side of the reservoir. The controls are therefore formed by the 90 injection and production rates in the smart well segments at every time step. The objective function represents a simple NPV as (6.1.1), defined as the sum of the incremental discounted oil production income and water production costs over the life of the reservoir. The wells were operated

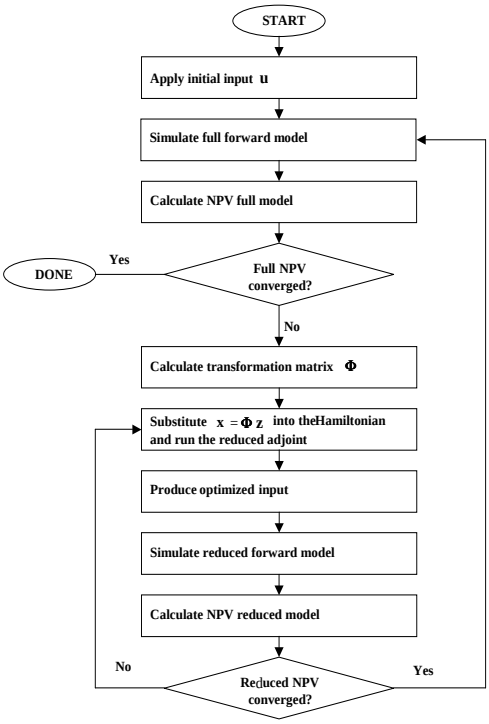


Figure 6.1: Flow chart for reduced-order OCT for water flooding.

without well model under rate-constraint: the liquid rate in an injection segment was equal to the water rate, and in a production segment the rate was equal to the sum of water rate and oil rate. The total production and injection rates were equal to each other during the entire simulation time. During optimization the flow rates were redistributed maintaining a constant total sum of the rates. In the NPV calculation an oil price $r_o = \frac{\$80}{m^3}$ and a produced water cost $r_w = \frac{\$20}{m^3}$ were used. The NPV obtained with the reduced-order and full-order optimal control algorithms was compared with the NPV of a reference case. In the reference case the injection and production rates were constant over time and a function of water and oil mobility, reflecting a conventional water flood where the wells are operated at constant bottom hole pressure. The reservoir model was simulated for 949 days with variable time step size and in this period one pore volume of liquid was injected and produced. A saturation distribution was obtained as depicted in Figure 6.3. The total NPV for the reference case is $\$10.1$ million.

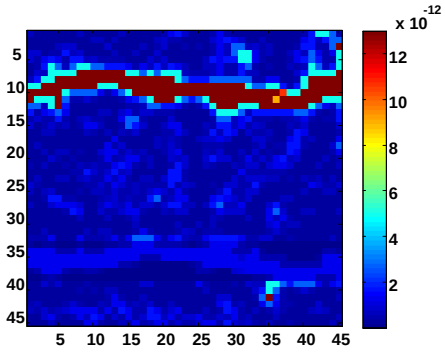


Figure 6.2: Permeability field (m^2).

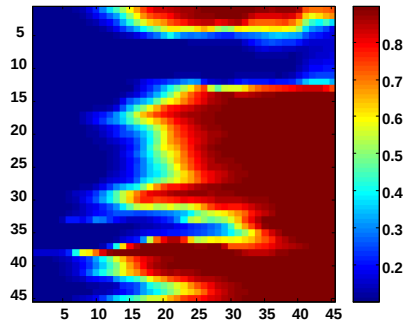


Figure 6.3: Final water saturation after 1 PV production for the reference case. Red: oil; blue: water.

6.3.1 Full-order optimal control example

The full-order control algorithm was ran starting from the reference case. With a 2.4 GHz Pentium 4 processor and 1 Gb RAM memory it took 8701 seconds (145 minutes) to run the full-order algorithm. Convergence was reached after 19 full-order forward simulations and 18 full-order backward simulations. The average simulation time for the full-order forward simulation was 144 seconds and for the full-order backward simulation 266 seconds; see the left picture in Figure 6.5. The resulting optimized rates are given in the left and the right picture of Figure 6.4, which correspond to a final oil/water saturation distribution as depicted in the middle picture of Figure 6.4. The corresponding NPV versus the number of iterations is plotted in the right picture of Figure 6.5. It can be seen that the NPV first stabilizes around $\$15$ million and later increases further to its maximum value of $\$20.3$ million. The NPV of the last iteration is slightly less than the previous one. When the algorithm was let to continue no further increase in NPV was observed. Because the optimization procedure is a local one, the maximum NPV should be regarded as a lower bound of the possible improvements.

6.3.2 Reduced-order optimal control example

For reduced-order optimal control the algorithm described in Section 6.2 was used. With an energy level of 0.999 as cut-off criterion for POD a final NPV of $\$19.3$ million was obtained, which is an increase of 87% with respect to the reference case. The maximum NPV obtained with reduced-order control approached the NPV obtained with full-order optimal control to within 95%. Convergence was reached in $5661s$, which is a reduction with 35% of the time used for the full-order optimal control. In this example we needed 10 full-order forward simulations and 13 reduced-order forward simulations. In the first iterations matrices $\mathbf{\Phi}$ were shared 2 to 4 times before the inner loop converged and matrices $\mathbf{\Phi}$ were

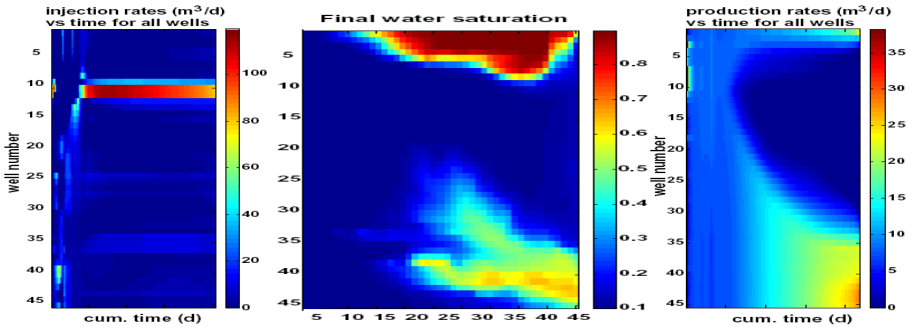


Figure 6.4: Optimized injection rates (left) and production rates (right) in $\frac{m^3}{d}$ vs. the simulation time, calculated with the full-order control algorithm. In the middle the resulting water saturation distribution. Red: oil; blue: water.

updated. The reduction in simulation time for the reduced-order forward simulation was 34% and for the reduced-order backward model was 38%. In order to maintain an energy level of 0.999 we need for the reference case in total 30 POD basis functions. The number of POD basis functions gradually increased when we used improved controls and for the optimal case we used 49 POD basis functions. This speaks in favor of our nested approach where the transformation matrix is adapted after a full-order forward simulation. Table 6.1 shows the results in more detail. The resulting optimized rates are given in the left and the right picture of Figure 6.6, which correspond to a final oil/water saturation distribution as in the right picture of Figure 6.6. The corresponding NPV versus the number of iterations is plotted in Figure 6.7. The resulting rates and the final saturation distribution obtained with reduced-order optimal control differed from the resulting rates and final saturation distribution obtained with full-order optimal control. Apparently we ended up in two different optima.

When a lower energy level of 0.99 was specified, less POD basis functions were required and the simulation time for the reduced-order simulations therefore decreased. However, in that case more simulations were needed in order to converge. Conversely, when the energy level was increased to 0.9999 a larger number of basis functions was required and the simulation time increased. The average simulation time for the full-order forward simulation was, in the latter case, almost equal to the average simulation time for the reduced-order forward model. The reduced-order backward simulations were still faster. Figure 6.8 displays the NPV versus the number of iterations in the full-order control algorithm and the reduced-order algorithm using energy levels of 0.99, 0.999 and 0.9999. The NPV obtained with reduced-order optimal control increased steeper in the beginning than the NPV obtained with full-order optimal control - an increase in NPV of 72% was reached already after two full-order forward simulations using an energy level of 0.999. After four full-order simulations the NPV slowly converged to its maximum. Changing the convergence criterion c of the inner loop from 1.01 to 1.10, we observed no significant influence on the maximum NPV; see Table 6.1.

Total time:	2h 25min
Maximum NPV:	$20.3 million
Forward runs:	19
Average time/run:	144s
Backward runs:	18
Average time/run:	266s

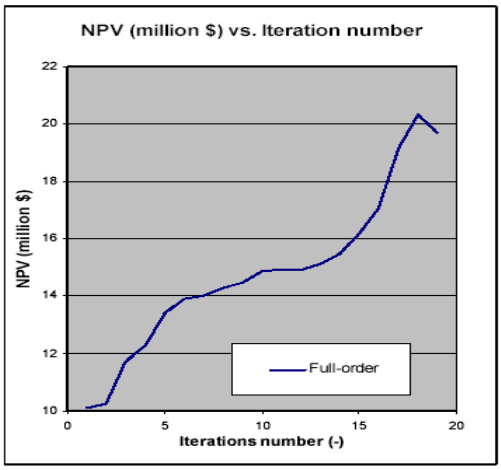


Figure 6.5: Performance of the full-order control algorithm. Left: computational effort and the final optimized NPV; Right: NPV versus number of iterations.

Also the total simulation times were almost identical. Another parameter that we varied was the number of time steps between updating the parameter-dependent matrices in the inner loop ($\mathbf{A}_k$ and $\mathbf{B}_k$ of the forward equation and the derivative matrices of the adjoint equation). Figure 6.9 displays the NPV versus the number of iterations using 1, 2, 3 and 4 time steps between the updates. Again we see a major increase of the NPV after two full-order forward simulations and a subsequent slow converge to the maximum. Not updating the matrices every time step introduces an error in calculating the improved controls. This is evident from the 7^{th} and 8^{th} columns of Table 6.1, which represent the number of retained basis functions, for a given energy level, during the first and the last iteration in the inner loop respectively. It can be observed that not updating the parameter-dependent matrices results in the need to use more basis functions in the later iterations. This effect is probably caused by spurious dynamics resulting from the more abrupt changes in the system parameters in case of less frequent updating.

6.3.3 Energy level, number of snapshots and grid size

Figure 6.10 depicts a physical interpretation of the saturation POD basis vectors for the 45×45 reservoir model. Because the vectors have a length equal to the total number of grid blocks they can be reshaped and projected on a 45×45 matrix. In the figure this is done for the first three vectors and we can recognize the spatial characteristics of the permeability field (depicted in Figure 6.2) and of the oil-water front which are both represented in the snapshots.

The three graphs in Figure 6.11 all display the number of POD basis functions ℓ versus the number of snapshots used in the calculation of Φ. For the first graph the reference case was simulated and a set of 1000 snapshots was generated

	Simulation time		Forward full		Forward reduced		Adjoint full		Adjoint reduced		# POD basis func. refer.			# POD basis func. optim.			Final NPV $
	s	h	No. of cells	s/call	No. of cells	s/call	No. of cells	s/call s	No. of cells	s/call	S_w	p	Tot.	S_w	p	Tot.	
Energy level = 0.99	5809	1.61	10	124	15	87	0	0	15	151	17	2	19	14	7	21	19,479,000
Energy level = 0.999	5661	1.57	9	151	13	100	0	0	13	164	28	2	30	28	13	41	19,315,000
Energy level = 0.9999	6746	1.87	9	129	14	131	0	0	14	200	42	2	44	58	25	83	19,371,000
Inner convergence = 1.01	5477	1.52	9	126	13	100	0	0	13	166	28	2	30	28	13	41	19,315,000
Inner convergence = 1.05	5661	1.57	9	151	13	100	0	0	13	164	28	2	30	28	13	41	19,315,000
Inner convergence = 1.10	5592	1.55	9	128	13	103	0	0	13	170	28	2	30	28	13	41	19,315,000
Constant time step = 1	5661	1.57	9	151	13	100	0	0	13	164	28	2	30	18	13	31	19,315,000
Constant time step = 2	5855	1.63	13	126	17	74	0	0	17	104	28	2	30	29	11	40	18,781,000
Constant time step = 3	4923	1.37	11	128	16	69	0	0	16	82	28	2	30	40	14	54	18,858,000
Constant time step = 4	5972	1.66	9	126	19	98	0	0	19	92	28	2	30	75	28	103	18,110,000
Full-order optimal control	8702	2.42	19	144	0	0	18	266	0	0	-	-	-	-	-	-	20,335,000
Full forward reduced backward	3774	1.05	11	130	0	0	0	0	10	165	28	2	30	28	11	39	19,139,000

Table 6.1: Results of full-order and reduced-order control algorithms for different energy levels, convergence criteria and number of time steps between updating the parameter-dependent matrices in the inner loop.

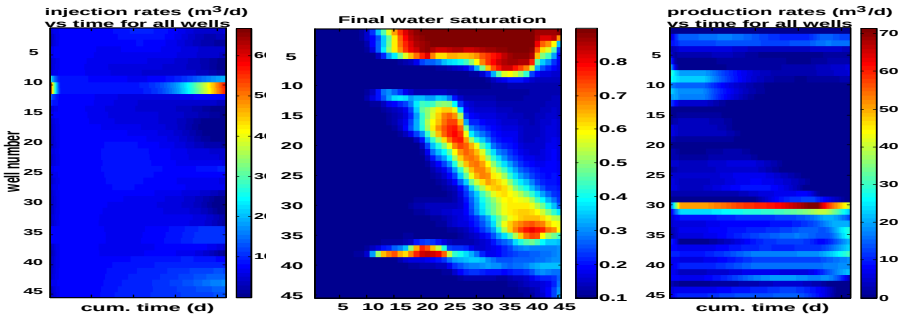


Figure 6.6: Optimized injection rates (left) and production rates (right) in $\frac{m^3}{d}$ vs. the simulation time, calculated with reduced-order control algorithm and energy level 0.999. In the middle the resulting water saturation distribution. Red: oil; blue: water.

at identical time-intervals. The same case was used also for the second graph, except that each grid block was divided in four grid blocks with identical permeabilities resulting in a 90×90 reservoir model. Also for this mode a set of 1000 snapshots was generated at identical time-intervals. For the 180×180 reservoir model the same procedure was used, which resulted in a reservoir model with 32400 grid blocks. The number of snapshots was varied by choosing from the set of 1000 snapshots a subset consisting of respectively 2, 3, 5, 9, 11, 13, 41, 51, 101, 201, 501 and 1000 snapshots at identical time-intervals. The figures illustrate how an increasing number of retained basis functions corresponds to an increasing energy level. When the number of snapshots for a given energy level was increased, an increasing number of POD basis functions was needed, which indicates that added snapshots are not a linear combination of the earlier snapshots, i.e. that they contain new information. More interestingly, we can conclude that the number of POD basis functions ℓ increases with the number of grid blocks in a non-linear fashion: when the number of grid blocks is multiplied by 16, the number of POD functions at an energy level of 99.99% for 1000 snapshots is only multiplied by 1.87. This illustrates that the reduced–order representation mainly represents the dominant structures present in the snapshots. Because we did not change the permeability field, but merely used a larger number of grid blocks to describe it, the dominant structures in the dynamics of the state variables also did not change very much. The small increase in basis functions captures some increased detail at the boundaries of the dominant structures. This implies that for a reservoir model with dominant large-scale geological features, the computational efficiency of reduced-order simulation will increase with an increasing model size. For a model that lacks clear dominant structures, e.g. one that has heterogeneities with a small correlation length, this increase in computational efficiency is less pronounced or even absent.

	Full-order	**Reduced-order**
Total time	2h 25m	**1h 22m**
Maximum NPV	$ 20.3 million	**$ 19.8 million**
Forward (time/call)	144s	**114s**
Backward (time/call)	266s	**187s**

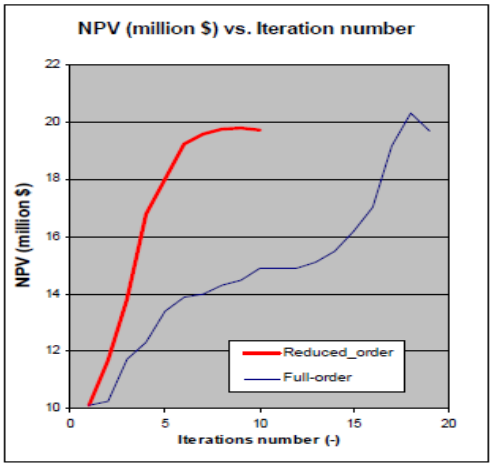


Figure 6.7: Performance of the reduced-order control algorithm (energy level 0.999) and its comparison with the full-order algorithm. Left: computational effort and the final optimized NPV; Right: NPV versus number of iterations.

6.4 Conclusion

In the example discussed it was found that reduced-order optimal control of water flooding using POD improved the NPV with respect to an uncontrolled reference case. Within a shorter simulation time, the NPV obtained by the full-order optimal control algorithm was approached closely by the NPV obtained by the reduced-order algorithm. The increase in computational efficiency was achieved by reducing the number of states in the forward and backward simulations considerably and consequently the number of equations that needed to be solved every time step. Considering a reservoir model with 4050 states (2025 pressures, 2025 saturations) and an adjoint model of 4050 states (Lagrange multipliers) reduced-order models were obtained with 20 to 100 states only. The NPV obtained by reduced-order optimal control was approached to within 95% of the NPV obtained by full-order optimal control. The resulting reduction in computing time was 35%. In general, the number of POD basis functions preserving a certain fixed level of relative energy increases during optimization, which speaks in favor of our nested reduced-order optimal control algorithm where we adapt the transformation matrix after simulating the full-order reservoir model with improved controls.

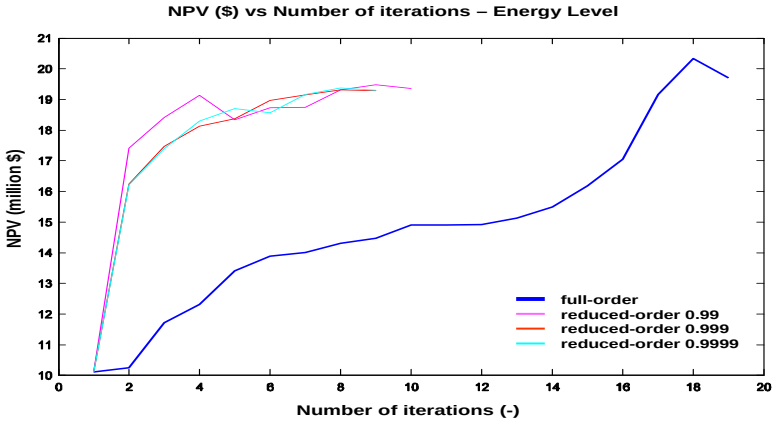


Figure 6.8: NPV versus number of iterations in the reduced-order control algorithm, for different energy levels.

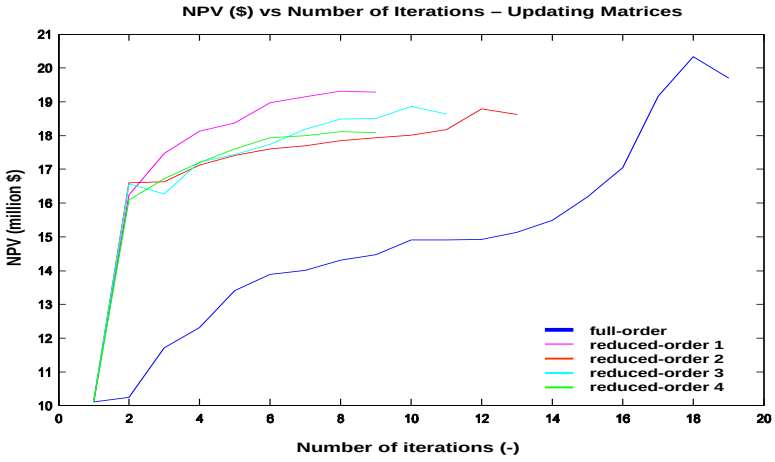


Figure 6.9: NPV versus number of iterations in the reduced-order control algorithm, for different numbers of time steps between the updating the parameter-dependent matrices.

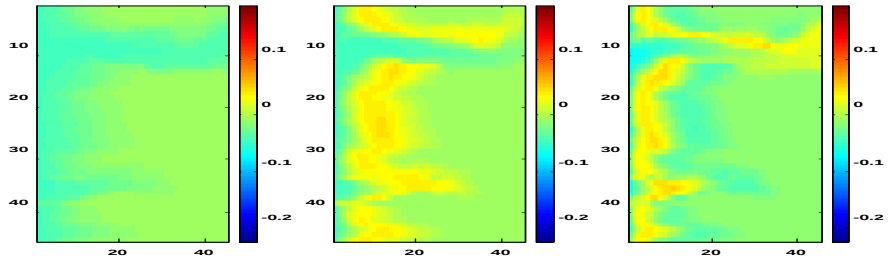


Figure 6.10: Example of the first three saturation POD basis functions projected on a 45×45 matrix.

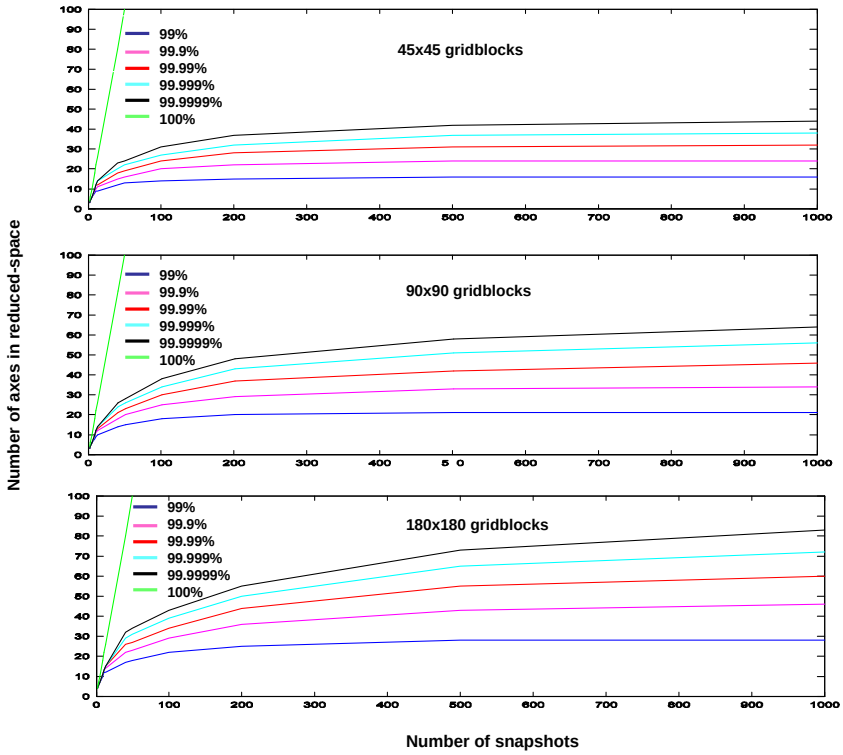


Figure 6.11: The three graphs represent the number of POD basis functions required to preserve a certain energy level, as a function of the number of snapshots, for a 45×45 reservoir model (top), a 90×90 reservoir model (middle) and a 180×180 reservoir model (bottom).

7 CHAPTER

Conclusions and Discussion

Numerical models describing subsurface fluid flow are typically large-scale, consisting of $O(10^3 - 10^6)$ equations and parameters representing a (coupled) system of discretized (nonlinear) PDEs (**Chapter 2**). Reservoir optimization and model updating based on models of such size may be prohibitively time consuming and error prone, especially when multiple reservoir model realizations and/or several potential well configurations have to be investigated.

This thesis addresses this 'curse of dimensionality' problem by reducing the high-order reservoir models to the appropriate level of detail in suitable low-order models. Traditionally, low-order reservoir model approximations are generated by "grid-coarsening" methods, which range from conventional upscaling to various multiscale techniques. This study follows a different approach, for we believe that low-order models based on 'system-theoretic' and dynamically-intrinsic properties of the fine-scale state-space system rather than the type (i.e., position, form, orientation) of the coarse-grid may be more efficient for the the intended applications. For example, for single-phase systems that can be approximated by linear systems of ODEs with constant coefficients, e.g. diffusive pressure equation for slightly compressible porous-media flow, basic system- and control theory teaches that these properties are, e.g., the system's transfer function in the Laplace domain, the eigenstructure of the system's matrix pencil, or controllability and observability of the (particular state-space realization of the) system. The structural properties controllability and observability (and therewith related identifiability) are particularly interesting, as modeling to a level of detail that can hardly be observed and controlled is at best wasted effort, but, worse, may lead to wrong results. For nonlinear state-space models, on the other side, due to both the size and the overall complexity of these systems (moving fluid interfaces, coupled flow and transport), intrinsic information needs to be sought in data obtained by simulating the fine-scale model.

For the slightly compressible single-phase flow, the performance was assessed of *Modal Truncation* (MTR), *Singular Perturbation Approximation* (SPA) (or 'rezidualization'), *Transfer Function Moment Matching* (TFMM), and *Balanced Truncation* (BTR). For two-phase (waterflood) flow, the so-called Proper Orthogonal Decom-

position (POD) technique was employed, a data-driven technique that guarantees optimal approximation of the collected data in the mean square error sense. In the linear case, POD turns out to have important theoretical connections to the concepts of controllability and observability, though for a particular class of input and initial-state signal forms. All the methods, except the residualization, are projection-based, meaning that a macromodel of the original large-scale dynamical system is generated by projecting it onto some low-dimensional subspace, whereby the projection subspace is not restricted to be the same as the subspace the state variable itself is projected onto (**Chapter 3**). In MTR, the two subspaces are generally equal and spanned by dominant eigenvectors of the system matrix/pencil. TFMM methods match a certain number of moments of the system's transfer function (around one or multiple frequency points) and obtain their projection subspaces as certain Krylov subspaces involving the system matrices. Projection subspaces in BTR are obtained as the dominant eigenspaces of the product of the system's controllability and observability Gramians of the system being the solutions of the corresponding Lyapunov matrix equations. Finally, the common projection subspace in the standard POD approach is spanned by the basis functions optimally explaining a desired percentage of the variance present in the collected data. The numerical tests conducted on several examples of heterogeneous reservoir models show that these techniques can produce accurate reduced-order models of very low order. The main observations are:

- In the slightly compressible single-phase case (**Chapter 4**), BTR consistently shows excellent performance. This is attributed to being numerically low-rank of both the controllability and the observability Gramians, as the rank of their product determines the intrinsic dimensionality of the system. Moreover, the low-rank nature of the Gramians allows for efficient use of iterative methods for approximate BTR. For low-frequency piecewise constant inputs (i.e. well flow-rates and/or BHPs), the shapes of the well control signals typically encountered in (open-loop) reservoir simulations, accurate reduced models of even lower order can be obtained by combining BTR with SPA, the essence of the latter being setting the time-derivative of a part of the system's state dynamics to zero and expressing the dynamics of the remaining part accordingly. The approximation error is mainly caused at the points where the input abruptly switches from one constant value to another. For low-frequency inputs, TFMM around the single frequency $s = 0$ is often sufficient for obtaining accurate reduced-order models. Exceptionally well performing in a majority of the examples considered was a Krylov subspace method that uses an orthogonalized matrix of state block moments around $s = 0$ as the ('congruence') projection matrix. Finally, for moment matching it generally holds that the reduced model size is a multiple of the number of matched moments, whereby the multiplier is an increasing function of the number of inputs and outputs. For this reason, for an increasing number of inputs and/or outputs the order of the reduced-order model tends to increase slower for BTR than for moment matching techniques.
- In waterflooding applications, numerical experiments (**Chapter 5**) indicate that a) a given flow scenario can be reproduced accurately if enough snap-

shots have been used and 'enough energy' has been captured in the POD modes, and b) the POD approach is quite robust against non-training input signals if the snapshots are generated by exciting the system by manipulating the injection and/or production profiles consistent with the expected range of operating conditions. As POD favorizes variables which numerical values show greater variability, projection bases for pressures and saturations were obtained separately. This allows for choosing a different degree of reduction for p and S_w. Apparently, an adequate approximation of the high-dimensional behavior generally requires (much) less basis functions for pressures than for saturations, which is attributed to the traveling nature of the saturation front and the time-averaging property of the (standard) POD approach. Regarding the cost of solving the reduced-order model, despite a generally drastic reduction in the size of the model, there is, in contrast to the single-phase case, no guarantee that the simulation of the reduced-order model will actually be faster than the simulation of the high-order model. This is due to that fact that at each time-step state-dependent coefficient matrices and their products needs to be evaluated. The issue is even more manifested in a fully time-implicit scheme using the Newton-Raphson solving procedure, where the coefficients and the matrix product are evaluated at each inner-loop iteration. The computational cost of forming the matrix products could potentially be reduced by not computing the POD basis coefficients in the usual POD manner, but rather estimating these from a partial set of the original data at a selected number of points in the spatial domain. Naturally, the computational gain may be expected to be bigger the smaller this number is, though at the cost of a decreased performance.

Running fine-scale waterflood simulations in order to determine a (robust) POD basis may be very time-consuming due to the necessity to employ an iterative procedure to propagate the solution in time. As dynamic reservoir flow problems are evolution problems, the initial guess is usually taken to be equal to the last time-step solution. The approach presented in **Chapter 5** determines an improved initial solution guess by a (regularly updated) POD-based reduced-order model. The solution of the reduced-order model can be interpreted as a 'shadow' running in parallel with the solution of high-order model. When applied to two-phase (incompressible) two-phase flow in an IMPES formulation, the proposed POD-based acceleration method for iterative solvers resulted in improved computational efficiency in almost all cases considered. Exceptions were pathological cases with steady-state inputs and near-time invariant parameters, which display near-constant behavior. The method seems to be particularly attractive for problems with time-varying parameters or time-varying source terms, and could, in theory, also be applied to accelerate the solution of nonlinear systems of equations using Newton-Raphson iteration. Further work is required to assess the efficiency of the method for such an application.

Modern approaches in reservoir optimization often require performing many reservoir flow simulations. The nested loops iterative strategy presented in **Chapter**

6 aims at speeding-up adjoint-based waterflooding optimization by employing a truncated basis of POD functions in the inner loop to calculate optimized injection and production rates. After convergence in this loop, the high-order model is simulated in the outer loop with the improved rates obtained in the inner loop. An adapted POD basis is used in the next inner loop to calculate new optimized injection and production rates, etc. In the numerical example the reduced-order optimal control of water flooding using POD improved the net present value (NPV) with respect to an uncontrolled reference case. Within a shorter simulation time, the NPV obtained by the full-order optimal control algorithm was approached closely by the NPV obtained by the reduced-order algorithm.

A APPENDIX

From PDEs to state-space models of the reservoir flow

This appendix presents in more detail the governing PDEs for two-phase (water-oil) reservoir fluid flow (under isothermal conditions) and their approximations by finite-dimensional state-space models.

A.1 PDE descriptions of two-phase reservoir flow

Consider the governing PDEs (2.1.7)-(2.1.8):

$$\frac{\partial\left(\varphi(1-S_w)\rho_o\right)}{\partial t} - \bigtriangledown\cdot\left(\frac{\mathbf{K}k_{ro}(S_w)\rho_o}{\mu_o}(\bigtriangledown p_o - \rho_o g \bigtriangledown D)\right) = q_o \quad \text{(A.1.1)}$$

$$\frac{\partial(\varphi S_w\rho_w)}{\partial t} - \bigtriangledown\cdot\left(\frac{\mathbf{K}k_{rw}(S_w)\rho_w}{\mu_w}(\bigtriangledown p_o - \bigtriangledown P_c(S_w) - \rho_w g \bigtriangledown D)\right) = q_w, \quad \text{(A.1.2)}$$

where $\bigtriangledown P_c(S_w) \equiv \frac{dP_c(S_w)}{dS_w} \bigtriangledown S_w$. Denoting with $c_\varphi := \frac{1}{\varphi}\frac{\partial\varphi}{\partial p}$ and $c_\alpha := \frac{1}{\rho_\alpha}\frac{\partial\rho_\alpha}{\partial p_\alpha}$ the rock- and the fluid (isothermal) *compressibility*, respectively, the accumulation terms on the l.h.s. of (A.1.1)-(A.1.2) may be expanded as

$$\frac{\partial(\varphi(1-S_w)\rho_o)}{\partial t} = \varphi\rho_o(1-S_w)(c_o+c_\varphi)\frac{\partial p_o}{\partial t} - \varphi\rho_o\frac{\partial S_w}{\partial t}, \quad \text{(A.1.3)}$$

$$\frac{\partial(\varphi S_w\rho_w)}{\partial t} = \varphi\rho_w S_w(c_w+c_\varphi)\frac{\partial(p_o-P_c)}{\partial t} + \varphi\rho_w\frac{\partial S_w}{\partial t}, \quad \text{(A.1.4)}$$

so that, using $\frac{\partial P_c}{\partial t} = \frac{dP_c}{dS_w}\frac{\partial S_w}{\partial t}$, (A.1.1)-(A.1.2) may be written in the following matrix-vector form:

$$\varphi \begin{bmatrix} \rho_o(1-S_w)(c_o+c_\varphi) & -\rho_o \\ \rho_w S_w(c_w+c_\varphi) & \rho_w\left(1-S_w(c_w+c_\varphi)\frac{dP_c}{dS_w}\right) \end{bmatrix} \begin{bmatrix} \frac{\partial p_o}{\partial t} \\ \frac{\partial S_w}{\partial t} \end{bmatrix}$$
$$= \begin{bmatrix} \bigtriangledown\cdot(\frac{\mathbf{K}k_{ro}\rho_o}{\mu_o}\bigtriangledown p_o) \\ \bigtriangledown\cdot(\frac{\mathbf{K}k_{rw}\rho_w}{\mu_w}\bigtriangledown p_o)-\bigtriangledown\cdot(\frac{\mathbf{K}k_{rw}\rho_w}{\mu_w}\bigtriangledown P_c) \end{bmatrix} + \begin{bmatrix} -\bigtriangledown\cdot(\frac{\mathbf{K}k_{ro}\rho_o^2}{\mu_o}g\bigtriangledown D) \\ -\bigtriangledown\cdot(\frac{\mathbf{K}k_{rw}\rho_w^2}{\mu_w}g\bigtriangledown D) \end{bmatrix}$$
$$+ \begin{bmatrix} q_o \\ q_w \end{bmatrix}. \quad \left[\frac{kg}{m^3 s}\right] \tag{A.1.5}$$

A.1.1 Various simplifications of the flow equations

System ($A.1.5$) is a *complete* PDE flow description, in that it models all major factors governing the immiscible fluid flow (3D, gravity, capillary effects). Depending on the assumptions about the physics that should be included in the model, various simplifications are possible. In *field-scale* reservoir flow studies, for instance, a generally valid assumption is $|\frac{\partial P_c}{\partial t} \ll |\frac{\partial p_o}{\partial t}|$. Moreover, in *waterflooding* applications on reservoir scale the (diffusive) effect of capillary pressure is frequently assumed to be negligible all together as compared to the (advective) effect of the pressure gradient. The flow is then described by

$$\varphi \begin{bmatrix} \rho_o(1-S_w)(c_o+c_\varphi) & -\rho_o \\ \rho_w S_w(c_w+c_\varphi) & \rho_w \end{bmatrix} \begin{bmatrix} \frac{\partial p_o}{\partial t} \\ \frac{\partial S_w}{\partial t} \end{bmatrix}$$
$$= \begin{bmatrix} \bigtriangledown\cdot(\frac{\mathbf{K}k_{ro}\rho_o}{\mu_o}\bigtriangledown p_o) \\ \bigtriangledown\cdot(\frac{\mathbf{K}k_{rw}\rho_w}{\mu_w}\bigtriangledown p_o) \end{bmatrix} + \begin{bmatrix} -\bigtriangledown\cdot(\frac{\mathbf{K}k_{ro}\rho_o^2}{\mu_o}g\bigtriangledown D) \\ -\bigtriangledown\cdot(\frac{\mathbf{K}k_{rw}\rho_w^2}{\mu_w}g\bigtriangledown D) \end{bmatrix} + \begin{bmatrix} q_o \\ q_w \end{bmatrix}. \tag{A.1.6}$$

If the flow is *two-dimensional and horizontal*, gravity effects are also missing, so that in this case the flow is governed by

$$\varphi \begin{bmatrix} \rho_o(1-S_w)(c_o+c_\varphi) & -\rho_o \\ \rho_w S_w(c_w+c_\varphi) & \rho_w\left(1-S_w(c_w+c_\varphi)\frac{dP_c}{dS_w}\right) \end{bmatrix} \begin{bmatrix} \frac{\partial p_o}{\partial t} \\ \frac{\partial S_w}{\partial t} \end{bmatrix}$$
$$= \begin{bmatrix} \bigtriangledown\cdot(\frac{\mathbf{K}k_{ro}\rho_o}{\mu_o}\bigtriangledown p_o) \\ \bigtriangledown\cdot(\frac{\mathbf{K}k_{rw}\rho_w}{\mu_w}\bigtriangledown p_o)-\bigtriangledown\cdot(\frac{\mathbf{K}k_{rw}\rho_w}{\mu_w}\bigtriangledown P_c) \end{bmatrix} + \begin{bmatrix} q_o \\ q_w \end{bmatrix}, \tag{A.1.7}$$

if capillary-effects play a role, and by

$$\varphi \begin{bmatrix} \rho_o(1-S_w)(c_o+c_\varphi) & -\rho_o \\ S_w(c_w+c_\varphi) & \rho_w \end{bmatrix} \begin{bmatrix} \frac{\partial p_o}{\partial t} \\ \frac{\partial S_w}{\partial t} \end{bmatrix}$$
$$= \begin{bmatrix} \bigtriangledown\cdot(\frac{\mathbf{K}k_{ro}\rho_o}{\mu_o}\bigtriangledown p_o) \\ \bigtriangledown\cdot(\frac{\mathbf{K}k_{rw}\rho_w}{\mu_w}\bigtriangledown p_o) \end{bmatrix} + \begin{bmatrix} q_o \\ q_w \end{bmatrix}, \tag{A.1.8}$$

if capillary-effects are neglected.

As long as at least one of the phases or the rock is compressible, the left hand side in the above equations is always invertible. In the *incompressible case*, i.e. $c_\varphi = c_w = c_o = 0$, this is not longer the case. For instance, $(A.1.8)$ then becomes:

$$\varphi \begin{bmatrix} 0 & -\rho_o \\ 0 & \rho_w \end{bmatrix} \begin{bmatrix} \frac{\partial p_o}{\partial t} \\ \frac{\partial S_w}{\partial t} \end{bmatrix} = \begin{bmatrix} \bigtriangledown \cdot (\frac{\mathbf{K} k_{ro} \rho_o}{\mu_o} \bigtriangledown p_o) \\ \bigtriangledown \cdot (\frac{\mathbf{K} k_{rw} \rho_w}{\mu_w} \bigtriangledown p_o) \end{bmatrix} + \begin{bmatrix} q_o \\ q_w \end{bmatrix}. \tag{A.1.9}$$

Actually, in this case the time-derivative of pressure disappears from the model description. Moreover, the phase densities are constant. Defining $\tilde{q}_\alpha := \frac{q_\alpha}{\rho_\alpha}$ and $\lambda_t := \lambda_w + \lambda_o = \frac{k_{rw}}{\mu_w} + \frac{k_{ro}}{\mu_o}$, $\alpha = w, o$, adding the two flow equations yields the following system:

$$\begin{aligned} -\nabla \cdot (\mathbf{K} \lambda_t \nabla p_o) &= \tilde{q}_t \\ \varphi \frac{\partial S_w}{\partial t} - \nabla \cdot (\mathbf{K} \lambda_w \nabla p_o) &= \tilde{q}_w. \end{aligned}$$

A.2 Spatial discretization

The above partial differential equations are generally not solvable analytically. Consequently, one generally looks for an approximate numerical solution at time and spatial points of interest. Construction of spatially discretized reservoir models is commonly performed using well-established techniques of finite elements (FE), finite volumes (FV), or, as in this thesis, finite differences (FD). The basic piece of the spatial domain in a FD discretization is a grid-block, and the primary variables of the system, here p_o and S_w, are defined either at grid-nodes or at grid-centers. In this thesis, a regular centered grid as schematically depicted in Fig. A.1 is always employed. At the grid level, the accumulation part of the system (i.e., the l.h.s. in the above equations, including the partial derivative which is approximated by the ordinary time-derivative operators d/dt), is evaluated using the pressure and saturation values of the corresponding grid-block. The flux terms, i.e. those the divergence operator $\nabla\cdot$ applies to, need to be evaluated at the block-interfaces. For instance, for a diagonal permeability tensor, $\mathbf{K} := diag(k_x, k_y, k_z)$, let us define, for the ijk-block, the direction-dependent phase 'transmissibilities', $T_{\alpha,\eta} := V^{ijk} \frac{k_{r\alpha} \rho_\alpha}{\mu_\alpha} k_\eta$ and $G_{\alpha,\eta} := T_{\alpha,\eta} \rho_\alpha g$, $\alpha \in \{w, o\}$, $\eta \in \{x, y, z\}$, where $V^{ijk} := \Delta x_i \Delta y_j \Delta z_k$ is the grid-block volume. In all examples in this thesis, the grid-block volume is uniform, i,e., $V^{ijk} = \Delta V$, $\forall i, j, k$. Moreover, in the 2D examples, $\Delta x_i = \Delta y_j = \Delta \xi$ is considered only. For the sake of higher generality, the derivations below uses $\Delta x_i = \Delta x$ and $\Delta y_i = \Delta y$. Setting $p := p_o$, the pressure

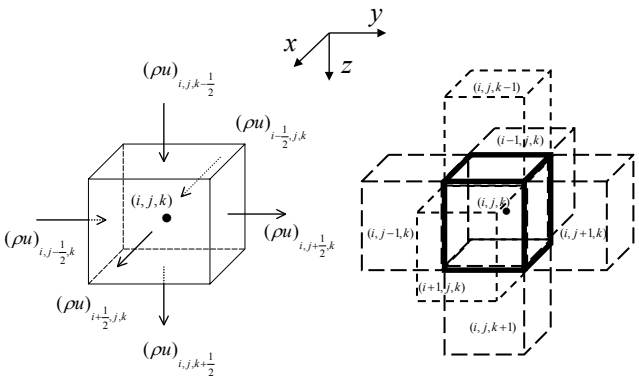


Figure A.1: Left: a grid-block, labeled by (i,j,k), with $(\rho u)_{i,j,k\pm\frac{1}{2}}$, $(\rho u)_{i,j\pm\frac{1}{2},k}$, $(\rho u)_{i\pm\frac{1}{2},j,k}$ denoting the mass fluxes across the grid-block interfaces. Right: the grid-block (i,j,k) and its six adjacent blocks arranged in a regular 7-point stencil. In 2D (3D) examples in this thesis, a regular 5-point (7-point) stencil is always used.

flux term for the grid-block ijk can then be approximated as

$$
\begin{aligned}
&\nabla \cdot (V^{ijk}\frac{\mathbf{K}k_{r\alpha}\rho_\alpha}{\mu_\alpha}\nabla p)\\
&= \frac{\partial}{\partial x}(T_{\alpha,x}\frac{\partial p}{\partial x}) + \frac{\partial}{\partial y}(T_{\alpha,y}\frac{\partial p}{\partial y}) + \frac{\partial}{\partial z}(T_{\alpha,z}\frac{\partial p}{\partial z})\\
&\approx \frac{1}{(\Delta x)^2}\left[\ (T_{\alpha,x})_{i+\frac{1}{2},j,k}\cdot(p_{i+1,j,k}-p_{i,j,k}) - (T_{\alpha,x})_{i-\frac{1}{2},j,k}\cdot(p_{i,j,k}-p_{i-1,j,k})\ \right]\\
&+ \frac{1}{(\Delta y)^2}\left[\ (T_{\alpha,y})_{i,j+\frac{1}{2},k}\cdot(p_{i,j+1,k}-p_{i,j,k}) - (T_{\alpha,y})_{i,j-\frac{1}{2},k}\cdot(p_{i,j,k}-p_{i,j-1,k})\ \right]\\
&+ \frac{1}{(\Delta z)^2}\left[\ (T_{\alpha,z})_{i,j,k+\frac{1}{2}}\cdot(p_{i,j,k+1}-p_{i,j,k}) - (T_{\alpha,z})_{i,j,k-\frac{1}{2}}\cdot(p_{i,j,k}-p_{i,j,k-1})\ \right]\\
&+ (\widetilde{T}_{\alpha,z})_{i,j,k-\frac{1}{2}}\cdot p_{i,j,k-1} + (\widetilde{T}_{\alpha,y})_{i,j-\frac{1}{2},k}\cdot p_{i,j-1,k} + (\widetilde{T}_{\alpha,x})_{i-\frac{1}{2},j,k}\cdot p_{i-1,j,k}\\
&- (\widetilde{T}_{\alpha})_{i,j,k}\cdot p_{i,j,k}\\
&+ (\widetilde{T}_{\alpha,x})_{i+\frac{1}{2},j,k}\cdot p_{i+1,j,k} + (\widetilde{T}_{\alpha,y})_{i,j+\frac{1}{2},k}\cdot p_{i,j+1,k} + (\widetilde{T}_{\alpha,z})_{i,j,k+\frac{1}{2}}\cdot p_{i,j,k+1},
\end{aligned}
$$

where $\widetilde{T}_{\alpha,\eta} := \frac{1}{(\Delta\eta)^2}T_{\alpha,\eta}$, and:

$$
\begin{aligned}
(\widetilde{T}_{\alpha})_{i,j,k} = (\widetilde{T}_{\alpha,z})_{i,j,k-\frac{1}{2}} + (\widetilde{T}_{\alpha,y})_{i,j-\frac{1}{2},k} + (\widetilde{T}_{\alpha,x})_{i-\frac{1}{2},j,k} + (\widetilde{T}_{\alpha,x})_{i+\frac{1}{2},j,k}\\
+(\widetilde{T}_{\alpha,y})_{i,j+\frac{1}{2},k} + (\widetilde{T}_{\alpha,z})_{i,j,k+\frac{1}{2}}.
\end{aligned}
$$

A similar derivation can be performed for the gravity[1] flux, $\nabla\cdot(V^{ijk}\frac{\mathbf{K}k_{r\alpha}\rho_\alpha^2}{\mu_\alpha}g\nabla D) = \sum_{\eta=x,y,z}\frac{\partial}{\partial\eta}(G_{\alpha,\eta}\frac{\partial D}{\partial\eta})$, and, if the capillary effects can not be neglected, also for the flux term $\nabla\cdot(V^{ijk}\frac{\mathbf{K}k_{r\alpha}\rho_w}{\mu_w}\frac{dP_c}{dS}\nabla S) \equiv \nabla\cdot(T_w\frac{dP_c}{dS}\nabla S) = \nabla\cdot(T_w^S\nabla S)$, where S_w is replaced by S for convenience.

A.2.1 State-space formulation

With these flux approximations, and recalling that $V^{ijk} = \Delta V$ was used here, the most general PDE system considered here, (A.1.5), can be approximated, for the grid-block ijk, as

$$\mathbf{V}_{ijk}\mathbf{W}_{ijk}\frac{d\mathbf{x}_{ijk}}{dt} = \mathbf{T}_{ijk}\begin{bmatrix}\mathbf{x}_{i,j,k-1}\\ \mathbf{x}_{i,j-1,k}\\ \mathbf{x}_{i-1,j,k}\\ \mathbf{x}_{i,j,k}\\ \mathbf{x}_{i+1,j,k}\\ \mathbf{x}_{i,j+1,k}\\ \mathbf{x}_{i,j,k+1}\end{bmatrix} + \mathbf{G}_{ijk}\begin{bmatrix}\mathbf{D}_{i,j,k-1}\\ \mathbf{D}_{i,j-1,k}\\ \mathbf{D}_{i-1,j,k}\\ \mathbf{D}_{i,j,k}\\ \mathbf{D}_{i+1,j,k}\\ \mathbf{D}_{i,j+1,k}\\ \mathbf{D}_{i,j,k+1}\end{bmatrix} + \mathbf{V}_{ijk}\tilde{\mathbf{q}}_{ijk}, \quad \left[\frac{kg}{s}\right]$$

where

$$\mathbf{x}_{ijk} = \begin{bmatrix} p_{ijk}\\ S_{ijk}\end{bmatrix} \quad , \quad \tilde{\mathbf{q}}_{ijk} := \begin{bmatrix}\frac{q_o^{ijk}}{\rho_o}\\ \frac{q_w^{ijk}}{\rho_w}\end{bmatrix}$$

$$\mathbf{V}_{ijk} = \begin{bmatrix}\Delta V\rho_o & 0\\ 0 & \Delta V\rho_w\end{bmatrix}_{ijk}$$

$$\mathbf{W}_{ijk} = \begin{bmatrix}\varphi(1-S)(c_o+c_\varphi) & -\varphi\\ \varphi S(c_w+c_\varphi) & \varphi[1-S(c_w+c_\varphi)(\frac{dP_c}{dS})]\end{bmatrix}_{ijk}$$

$$\mathbf{T}_{ijk} = \begin{bmatrix}\widetilde{\mathbf{T}}^o_{i,j,k-\frac12} & \widetilde{\mathbf{T}}^o_{i,j-\frac12,k} & \widetilde{\mathbf{T}}^o_{i-\frac12,j,k} & -\widetilde{\mathbf{T}}^o_{i,j,k} & \widetilde{\mathbf{T}}^o_{i+\frac12,j,k} & \widetilde{\mathbf{T}}^o_{i,j+\frac12,k} & \widetilde{\mathbf{T}}^o_{i,j,k+\frac12}\\ \widetilde{\mathbf{T}}^w_{i,j,k-\frac12} & \widetilde{\mathbf{T}}^w_{i,j-\frac12,k} & \widetilde{\mathbf{T}}^w_{i-\frac12,j,k} & -\widetilde{\mathbf{T}}^w_{i,j,k} & \widetilde{\mathbf{T}}^w_{i+\frac12,j,k} & \widetilde{\mathbf{T}}^w_{i,j+\frac12,k} & \widetilde{\mathbf{T}}^w_{i,j,k+\frac12}\end{bmatrix}$$

$$\widetilde{\mathbf{T}}^o_{i,j,k\pm\frac12} = \begin{bmatrix}(\widetilde{T}_{o,z})_{i,j,k\pm\frac12} & 0\end{bmatrix} \quad , \quad \widetilde{\mathbf{T}}^o_{i,j\pm\frac12,k} = \begin{bmatrix}(\widetilde{T}_{o,y})_{i,j\pm\frac12,k} & 0\end{bmatrix}$$

$$\widetilde{\mathbf{T}}^o_{i\pm\frac12,j,k} = \begin{bmatrix}(\widetilde{T}_{o,x})_{i\pm\frac12,j,k} & 0\end{bmatrix} \quad , \quad \widetilde{\mathbf{T}}^o_{i,j,k} = \begin{bmatrix}(\widetilde{T}_o)_{i,j,k} & 0\end{bmatrix}$$

[1]The approximation becomes particularly simple in the case of a horizontal 3D grid with $\Delta z_k = \Delta z$. For such a grid, the approximation for the block ijk is given by

$$\nabla\cdot(V^{ijk}\frac{\mathbf{K}k_{r\alpha}\rho_\alpha^2}{\mu_\alpha}g\nabla D) = \frac{\partial}{\partial z}(G_{\alpha,z}\frac{\partial D}{\partial z}) \overset{\Delta z_k=\Delta z}{\approx} \frac{1}{\Delta z}\left[(G_{\alpha,z})_{i,j,k+1/2} - (G_{\alpha,z})_{i,j,k-1/2}\right]$$

.

$$\widetilde{\mathbf{T}}^w_{i,j,k\pm\frac{1}{2}} = \begin{bmatrix} (\widetilde{T}_{w,z})_{i,j,k\pm\frac{1}{2}} & (\widetilde{T}^S_{w,z})_{i,j,k\pm\frac{1}{2}} \end{bmatrix}$$
$$\widetilde{\mathbf{T}}^w_{i,j\pm\frac{1}{2},k} = \begin{bmatrix} (\widetilde{T}_{w,y})_{i,j\pm\frac{1}{2},k} & (\widetilde{T}^S_{w,y})_{i,j\pm\frac{1}{2},k} \end{bmatrix}$$
$$\widetilde{\mathbf{T}}^w_{i\pm\frac{1}{2},j,k} = \begin{bmatrix} (\widetilde{T}_{w,x})_{i\pm\frac{1}{2},j,k} & (\widetilde{T}^S_{w,x})_{i\pm\frac{1}{2},j,k} \end{bmatrix}$$
$$\widetilde{\mathbf{T}}^w_{i,j,k} = \begin{bmatrix} (\widetilde{T}_w)_{i,j,k} & (\widetilde{T}^S_w)_{i,j,k} \end{bmatrix}$$
$$\mathbf{G}_{ijk} = \begin{bmatrix} \widetilde{\mathbf{G}}^o_{i,j,k-\frac{1}{2}} & \widetilde{\mathbf{G}}^o_{i,j-\frac{1}{2},k} & \widetilde{\mathbf{G}}^o_{i-\frac{1}{2},j,k} & -\widetilde{\mathbf{G}}^o_{i,j,k} & \widetilde{\mathbf{G}}^o_{i+\frac{1}{2},j,k} & \widetilde{\mathbf{G}}^o_{i,j+\frac{1}{2},k} & \widetilde{\mathbf{G}}^o_{i,j,k+\frac{1}{2}} \\ \widetilde{\mathbf{G}}^w_{i,j,k-\frac{1}{2}} & \widetilde{\mathbf{G}}^w_{i,j-\frac{1}{2},k} & \widetilde{\mathbf{G}}^w_{i-\frac{1}{2},j,k} & -\widetilde{\mathbf{G}}^w_{i,j,k} & \widetilde{\mathbf{G}}^w_{i+\frac{1}{2},j,k} & \widetilde{\mathbf{G}}^w_{i,j+\frac{1}{2},k} & \widetilde{\mathbf{G}}^w_{i,j,k+\frac{1}{2}} \end{bmatrix}$$
$$\widetilde{\mathbf{G}}^\alpha_{i,j,k\pm\frac{1}{2}} = \begin{bmatrix} (\widetilde{G}_{\alpha,z})_{i,j,k\pm\frac{1}{2}} & 0 \end{bmatrix} \quad , \quad \widetilde{\mathbf{G}}^\alpha_{i,j\pm\frac{1}{2},k} = \begin{bmatrix} (\widetilde{G}_{\alpha,y})_{i,j\pm\frac{1}{2},k} & 0 \end{bmatrix}$$
$$\widetilde{\mathbf{G}}^\alpha_{i\pm\frac{1}{2},j,k} = \begin{bmatrix} (\widetilde{G}_{\alpha,x})_{i\pm\frac{1}{2},j,k} & 0 \end{bmatrix} \quad , \quad \widetilde{\mathbf{G}}^\alpha_{i,j,k} = \begin{bmatrix} (\widetilde{G}_\alpha)_{i,j,k} & 0 \end{bmatrix}, \quad \alpha = w, o.$$

Stacking the equations $(A.2.1)$ for all the grid-blocks ijk on top of each other yields a time-continuous *(generalized) state-space* formulation:

$$\mathbf{V}\mathbf{W}\frac{d\mathbf{x}}{dt} = \mathbf{T}\mathbf{x} + \mathbf{G}\mathbf{d} + \mathbf{V}\tilde{\mathbf{q}}, \qquad \left[\frac{kg}{s}\right] \tag{A.2.1}$$

where the n-dimensional state vector now consists of oil pressures and water saturations for each of the $n_x n_y * n_z = \frac{n}{2}$ grid blocks, $\mathbf{V}$ is a diagonal mass matrix with entries that are functions of grid block volume and fluid densities, $\mathbf{W}$ is a block diagonal matrix with entries being functions of compressibility, porosity and water saturation, $\mathbf{T}$ and $\mathbf{G}$ are sparse block matrices accommodating block-interface transmissibilities for oil and water, $\mathbf{d}$ is the depth-vector, and $\tilde{\mathbf{q}}$ denotes the well flow-rates.

It deserves noting that the governing equations can be sorted arbitrarily. However, some orders of the equations will be more suitable that others from the computational point of view. The 'alternating' state form used here leads to a system with a small band width.

A.2.2 Well controls and the well model

An injector well normally injects a single phase (here water). The fluid flowing into a producer well, however, is generally a composition of more (here two) phases. Consequently, what directly can be controlled and/or measured at the producer side is generally not the individual phase flow rates, here q_o and q_w, but their total sum, $q_t = q_o + q_w$. In the flow model, the individual phase rates can be modeled through their corresponding fractional flows, $f_o := \frac{q_o}{q_t}$ and $f_w := \frac{q_w}{q_t}$. For the flow-rate controlled case (i.e., water flow-rate for injectors and the total liquid rate for the producers), the state-space model $(A.2.1)$ transforms into:

$$\mathbf{VW}\frac{d\mathbf{x}}{dt} \quad = \quad \mathbf{Tx} + \mathbf{Gd} + \mathbf{VFL}_{qu}\mathbf{u}, \tag{A.2.2}$$

where $\mathbf{F}$ is a block-diagonal matrix containing fractional-flow functions for water and oil, and $\mathbf{L}_{qu}$ is a selection (or location) matrix consisting of zeros and ones at appropriate places so that $\mathbf{u} \equiv \mathbf{L}_{qu}^T\tilde{\mathbf{q}}_t$ is the input vector containing the total liquid rates ($\tilde{q}_o + \tilde{q}_w$ at the producers and $\tilde{q}_w$ at the injectors). The blocks in $\mathbf{F}$ are two-dimensional column vectors of the form $\begin{bmatrix} f_w \\ f_o \end{bmatrix}_i$, where $f_w = \frac{q_w}{q_t}$ and $f_o = \frac{q_o}{q_t}$ are the *fractional-flow* functions for water and oil, respectively, and i denotes the grid-block in question. Clearly, $0 \leq f_w, f_o \leq 1$ and $f_w + f_o = 1$. For an exclusively water injecting well, the fractional-flow vector is $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$. Moreover, for a grid-block j without wells the fractional-flows are not defined. Since the flow-rate for such a well is zero, $(f_w)_j$ and $(f_o)_j$ may be assigned any finite value.

The well model

If a well in a grid-block i is described by a well model, $q_t = \omega(p_{wf} - p_{gb})$, with prescribed bottom-hole flowing pressure, p_{wf}, the resulting system of equations is similar, with $u_i = p_{wf,i}$, and $\mathbf{T}$ and $\mathbf{F}$ adjusted at appropriate places by terms involving the well-index ω_i. Well models used in this thesis are of the type ([82]):

$$\tilde{q}_\alpha = \frac{k_{r\alpha}}{\mu_\alpha}\frac{2\pi d_z^{well}\sqrt[3]{k_x k_y k_z}}{V_{gb}\left(ln(\frac{r_o}{r_w}) + S\right)}\left(p_{wf,\alpha} - p_{gb,\alpha}\right) = \frac{k_{r\alpha}}{\mu_\alpha}\overline{\omega}_z\left(p_{wf,\alpha} - p_{gb,\alpha}\right), \left[\frac{1}{s}\right] \tag{A.2.3}$$

where (for a vertical well):

$$\overline{\omega}_z = \frac{2\pi d_z^{well}\sqrt[3]{k_x k_y k_z}}{V_{gb}\left(ln(\frac{r_o}{r_w}) + S\right)} \quad [-] \quad \text{((const.) geometric part of the well-index)}$$

$$r_o = 0.28\frac{\left[\sqrt{k_x/k_y}(\Delta y)^2 + \sqrt{k_y/k_x}(\Delta x)^2\right]^{\frac{1}{2}}}{\sqrt[4]{k_x/k_y} + \sqrt[4]{k_y/k_x}} \quad [m] \qquad \text{(effect. well radius)}$$

d_z^{well} : the (vertical) length in $[m]$ of the well part accessible for fluid flow

It is normally assumed that there is a single bottom-hole flowing pressure, i.e., $p_{wf,o} = p_{wf,w} =: p_{wf}$. Using $p_o - p_w = P_c$ and the fact that the only phase grid-block pressure in the model is p_o, the individual phase rates can be expressed as:

$$\tilde{q}_o = \frac{k_{ro}}{\mu_o}\overline{\omega}_z\left(p_{wf} - p_{gb,o}\right) \tag{A.2.4}$$

$$\tilde{q}_w = \frac{k_{rw}}{\mu_w}\overline{\omega}_z\left(p_{wf} - p_{gb,w}\right) = \frac{k_rw}{\mu_w}\overline{\omega}_z\left(p_{wf} - p_{gb,o}\right) + \frac{k_{rw}}{\mu_w}\overline{\omega}_z P_c. \tag{A.2.5}$$

For a prescribed p_{wf}, the total flow rate can be expressed as[2]

$$\tilde{q}_t = \left(\frac{k_{rw}}{\mu_w} + \frac{k_{ro}}{\mu_o}\right)\overline{\omega}_z\left(p_{wf} - p_{gb,o}\right) + \frac{k_{rw}}{\mu_w}\overline{\omega}_z P_c. \tag{A.2.6}$$

The phase fractional flow functions can be expressed (using $\lambda_\alpha = \frac{k_{r\alpha}}{\mu_\alpha}$, $\alpha = w, o$) as:

$$f_w = \frac{\tilde{q}_o}{\tilde{q}_t} = \frac{\lambda_w \cancel{\overline{\omega}_z}\left(p_{wf} - p_{gb,o} + P_c\right)}{\lambda_w \cancel{\overline{\omega}_z}\left(p_{wf} - p_{gb,o} + P_c\right) + \lambda_o \cancel{\overline{\omega}_z}\left(p_{wf} - p_{gb,o}\right)} = \frac{\lambda_w}{\lambda_w + \lambda_o \frac{p_{wf} - p_{gb,o}}{p_{wf} - p_{gb,o} + P_c}}$$

$$f_o = \frac{\tilde{q}_o}{\tilde{q}_t} = 1 - f_w = \frac{\lambda_o}{\lambda_o + \lambda_w \frac{p_{wf} - p_{gb,o} + P_c}{p_{wf} - p_{gb,o}}}.$$

In the absence of capillary effects, i.e. when $P_c = 0$, these expressions become the familiar ones, $f_\alpha = \frac{\lambda_\alpha}{\lambda_t}$, $\alpha = w, o$.

A.2.3 Spatial weighting of parameters

The solutions of a centered grid system are grid-block quantities, therefore an averaging involving these quantities is needed in order to determine the coefficients at the grid-block interfaces. The permeability k is commonly determined by the weighted harmonic average. With constant grid block size in the spatial direction $\eta = i, j, k$, this is equal to $k_{\eta+\frac{1}{2}} = \frac{1}{\frac{1}{2k_{\eta,i}} + \frac{1}{2k_{\eta,i+1}}}$.

The weakly pressure dependent coefficients at the grid-block interfaces are generally approximated by the arithmetic mean. For viscosity μ, for instance, this means $\mu_{\eta+\frac{1}{2}} = \frac{\mu_{\eta+1} + \mu_{\eta 1}}{2}$. Besides the phase viscosities μ_α, the weighted arithmetic mean was also used for averaging the fluid densities.

The saturation dependent parameters $k_{r\alpha}$ and P_c require special treatment. For the sake of obtaining a physically relevant solution, these quantities are generally calculated according to an upstream weighting principle, whereby the upstream direction is determined by the phase potential gradient, $\nabla\Psi_\alpha := \nabla p_\alpha - \rho_\alpha g \nabla D$. In the applications in this thesis, the one-point upstream weighting is used, that is, the interface value is taken equal to the value in the grid-block from which the flow is coming. For instance, the relative permeability is averaged according to:

$$k_{r\alpha,\eta+\frac{1}{2}} = \begin{array}{ll} k_{r\alpha,\eta}, & \text{if} \quad \Psi_{\alpha,\eta} \geq \Psi_{\alpha,\eta+1} \\ k_{r\alpha,\eta+1}, & \text{if} \quad \Psi_{\alpha,\eta} < \Psi_{\alpha,\eta+1} \end{array}.$$

[2]Note that the water injection rate does not depend only on the water mobility, but rather on the total fluid mobility around the well.

B APPENDIX

Mathematics of POD

B.1 The basis POD solution

Let $u(\vec{\xi}, t)$ be the (unknown) solution of the problem one is attempting to solve, defined at any point $(\vec{\xi}, t) \in \Omega \times \mathbb{T}$ on some spatial domain Ω and time-interval $\mathbb{T} \subseteq \mathbb{R}$. No distinctions is made here between the finite- and infinite dimensional cases, i.e. the spatial and the time domain may be infinite dimensional or even unbounded. Assume further that the solutions $u(\vec{\xi}, t) : \Omega \times \mathbb{T} \rightarrow \mathbb{R}$ belong to some (separable) Hilbert space $\mathcal{H}$, e.g. $\mathcal{H} = L_2(\Omega, \mathbb{R})$, with $L_2(\Omega, \mathbb{R})$ denoting a collection of square-integrable functions[1], such as pressure profiles in the examples in this thesis, for instance. Using the fact that every (separable) Hilbert space admits an (non-unique) *orthonormal basis* which spans the whole space, denoting by $(\cdot, \cdot)$ the inner product defined on $\mathcal{H}$ the solutions $u(\vec{\xi}, t)$ may be written as

$$u(\vec{\xi}, t) \quad = \quad \sum_{i \in \mathbb{I}} a_i \phi_i = \sum_{i \in \mathbb{I}} (u, \phi_i) \phi_i, \tag{B.1.1}$$

where $\{\phi_i\}_{i \in \mathbb{I}}$ ($(\phi_i, \phi_j) = \delta_{ij}$, $i, j \in \mathbb{I}$) is a particular choice for the orthonormal basis of $\mathcal{H}$ and $\{a_i\}_{i \in \mathbb{I}} := \{(u, \phi_i)\}_{i \in \mathbb{I}}$ are the *Fourier coefficients* of u in the basis $\{\phi_i\}_{i \in \mathbb{I}}$, with $\mathbb{I} \subseteq \mathbb{Z}$ an ordered (countable) index set. In the usual POD literature, the basis functions ϕ_i in (B.1.1) are generally considered to be spatial- and the expansion coefficients a_i time functions, i.e. $\phi = \phi(\vec{\xi})$ and $a_i = a_i(t)$. The same assumption is made also in this thesis. It deserves noting, however, that, from a mathematical point of view, there is no fundamental reason why it could not be opposite, i.e., that the basis functions are time- and the coefficients spatial functions. Though interesting in its own, an investigation of this possibly useful observation is beyond the scope of this thesis. Hence, in the sequel, the basis functions are always assumed to be spatial. Define $\mathbb{I}_\ell := \{1, \ldots, \ell\}$. Denoting by

$$u_\ell(\vec{\xi}, t) \quad = \quad \sum_{i \in \mathbb{I}_\ell} a_i \phi_i = \sum_{i \in \mathbb{I}_\ell} (u, \phi_i) \phi_i, \tag{B.1.2}$$

[1] *** Definitie van square-integrable functies ***

an ℓ-order *truncated decompositions* of $u(\vec{\xi},t)$, **the POD problem** is basically to find such orthonormal basis functions $\{\phi_i\}_{i=1}^{\ell}$ which solves the following optimization problem *for any* $\ell \in \mathbb{I}$:

$$\underset{\substack{\{\phi_i\}_{i=1}^{\ell} \\ (\phi_i,\phi_j)=\delta_{ij}}}{\textbf{Minimize}} \ \langle |u(\vec{\xi},t) - u_\ell(\vec{\xi},t)|^2 \rangle \quad (\equiv \langle |u(\vec{\xi},t) - \sum_{i\in\mathbb{I}_\ell}(u,\phi_i)\phi_i|^2 \rangle), \tag{B.1.3}$$

where $\langle\cdot\rangle$ is the employed (time-)averaging operation.

Let $u_\ell(\vec{\xi},t) =: \mathcal{P}^\ell(u(\vec{\xi},t))$, where $\mathcal{P}^\ell$ is an orthogonal operator that projects u onto an ℓ-dimensional subspace. Due to the identity

$$\|u\|^2 = \|u - \mathcal{P}^\ell(u)\|^2 + \|\mathcal{P}^\ell(u)\|^2, \tag{B.1.4}$$

which holds for any orthogonal projection, the minimization problem (B.1.3) is equivalent to the maximization problem:

$$\underset{\substack{\{\phi_i\}_{i=1}^{\ell} \\ (\phi_i,\phi_j)=\delta_{ij}}}{\textbf{Maximize}} \ J(\{\phi_i\}_{i=1}^{\ell}) \quad := \quad \langle |\sum_{i\in\mathbb{I}_\ell}(u,\phi_i)\phi_i|^2 \rangle. \tag{B.1.5}$$

Straightforward manipulation:

$$\begin{aligned} |\sum_{i\in\mathbb{I}_\ell}(u,\phi_i)\phi_i|^2 &= \sum_{i\in\mathbb{I}_\ell}(u,\phi_i)\phi_i \sum_{j\in\mathbb{I}_\ell}(u,\phi_j)\phi_j = \sum_{i\in\mathbb{I}_\ell}\sum_{j\in\mathbb{I}_\ell}(u,\phi_i)(u,\phi_j)\phi_i\phi_j \\ &= \sum_{i\in\mathbb{I}_\ell}|(u,\phi_i)|^2 \quad (\equiv \sum_{i\in\mathbb{I}_\ell} a_i^2(t)) \end{aligned} \tag{B.1.6}$$

shows that the POD basis problem (B.1.5) can be written as

$$\underset{\substack{\{\phi_i\}_{i=1}^{\ell} \\ (\phi_i,\phi_j)=\delta_{ij}}}{\textbf{Maximize}} \ \langle \sum_{i\in\mathbb{I}_\ell}|(u,\phi_i)|^2 \rangle, \tag{B.1.7}$$

thus meaning that the *POD basis functions ϕ_i maximize the averaged projection of the time evolution of the flow-field u onto the subspace spanned by these basis functions*.

In Appendix *** it is shown by the use of calculus variations that (B.1.7) reduces to the eigenvalue problem

$$\int_\Omega R(\vec{\xi},\vec{\xi}')\phi_i(\vec{\xi}')d\vec{\xi}' = \lambda_i\phi_i(\vec{\xi}), \tag{B.1.8}$$

where $R(\vec{\xi},\vec{\xi}') := \langle u(\vec{\xi}), u(\vec{\xi}')\rangle$ is a time-averaged autocorrelation function. The optimal basis functions, called the proper orthogonal modes (POMs), are thus the eigenfunctions of the integral equation (B.1.8), whose kernel is the averaged correlation function R. The values λ_i corresponding to ϕ_i are named the proper orthogonal values (POVs). The energy $\mathcal{E}$ contained in the observed data is defined as the sum of the POVs, i.e., $\mathcal{E} := \sum_{i\in\mathbb{I}}\lambda_i$. The energy percentage captured by the i-th POM is then given by $\lambda_i/\mathcal{E}$. The cumulative sum of relative energies $\mathcal{E}_i$ clearly

approaches one with the increasing number of modes in the reconstruction.

In practical situations, the state of a system or a numerical model is only available at discrete spatial grid points and at discrete time instances, so that the observed data, which we call *snapshots*, that are used to approximate the autocorrelation function R are vectors rather than continuous functions. In other words, let the numbers of spatial and time points be n and N, respectively. Then, $\Omega = \{\vec{\xi}_1, \vec{\xi}_2, \ldots, \vec{\xi}_n\}$, where $\vec{\xi}_j$ is the j-th grid point, and $u(\vec{\xi}, t)$ is now represented by the *vector* $\mathbf{u}_i = [u(\vec{\xi}_1, t_i), u(\vec{\xi}_2, t_i), \ldots, u(\vec{\xi}_n, t_i)]^T$. In this case, the autocorrelation function R is replaced by the tensor product matrix $\mathbf{R}(\vec{\xi}, \vec{\xi}') = 1/N \sum_{i=1}^{N} \mathbf{u}_i \mathbf{u}_i^T$. As the kernel $\mathbf{R}$ is Hermitian, the optimal POD basis consists of the orthonormal vector solutions of the eigenvalue problem $\mathbf{R}\varphi_i = \lambda_i \varphi_i$.

We conclude this part with the observation that, since the basis functions are mutually orthonormal, the individual projections are uncorrelated. Hence it follows that the optimization problem can conceptually be tackled by first solving

$$\underset{\{\phi_1\}}{\textbf{Maximize}} \ \langle |(u, \phi_1)|^2 \rangle \quad \text{with } (\phi_1, \phi_1) = 1, \tag{B.1.9}$$

to determine ϕ_1, followed by solving

$$\underset{\{\phi_2\}}{\textbf{Maximize}} \ \langle |(u, \phi_2)|^2 \rangle \quad \text{with } (\phi_2, \phi_2) = 1 \text{ and } (\phi_1, \phi_2) = 0, \tag{B.1.10}$$

to determine ϕ_2 and so on.

B.2 POD in infinite-dimensional setting

Albeit a practical implementation of MOR techniques will generally be based on finite-dimensional data and ODEs, a POD model reduction approach can be formulated for PDEs as well. Here the basis procedure for reducing PDEs to a system of ODEs is outlined, aiming to help in understanding that the POD method is more than just a matrix decomposition algorithm as it appears in practical implementations of a POD approach.

Let a general PDE, defined on some space-time 'region' $\Omega \times \mathbb{T}$, be described by

$$u_t \quad = \quad L[u] + f(\vec{\xi}, t), \tag{B.2.1}$$

where u_t denotes the partial time-derivative of the variable of interest u, $L[\cdot]$ is some (possibly non-linear) spatial operator, and f is a source term, e.g. describing boundary conditions. Assume the possibility of representing u as an infinite-dimensional spectral decomposition as in (B.1.1), i.e. $u(\vec{\xi}, t) = \sum_{i=1}^{\infty} a_i(t)\phi_i(\vec{\xi})$, and denote by u_ℓ an ℓ-th order approximation of u as in (B.1.2), i.e. $u_\ell(\vec{\xi}, t) = \sum_{i=1}^{\ell} a_i(t)\phi_i(\vec{\xi})$. Assume the spatial basis functions ϕ_i to be orthonormal, then

(B.2.1) may be written as

$$\sum_{i=1}^{\ell} \dot{a}_i(t)\phi_i \quad = \quad L[\sum_{i=1}^{\ell} a_i(t)\phi_i] + f(\vec{\xi},t) + r(\sum_{i=1}^{\ell} a_i(t)\phi_i), \tag{B.2.2}$$

where r is a *residual* function describing the error in the equation approximation caused by the use of a truncated approximation of u itself. Note that (B.2.2) is *exact*, in that it is formally equivalent to the starting equation (B.2.1). In general, r is unknown and non-zero. The task of ROM is then to determine basis functions $\{\phi_i\}_{i=1}^{\ell}$ such that the original equation (B.2.1) are satisfied *as good as possible* by minimizing the residual r, viewed as a function on $\Omega \times \mathbb{T}$, in some well-defined sense. Due to the orthonormality of the basis functions, by projecting the equation (B.2.2) onto the span$\{\phi_i, i = 1, \dots \ell\}$, the time-coefficients $\{a_j(t)\}_{j=1}^{\ell}$ satisfy

$$\dot{a}_j(t) = \Big(\; L[\textstyle\sum_{i=1}^{\ell} a_i(t)\phi_i] + f(\vec{\xi},t), \phi_j \; \Big) + \Big(\; r(\textstyle\sum_{i=1}^{\ell} a_i(t)\phi_i), \phi_j \; \Big). \tag{B.2.3}$$

Requiring r to satisfy the **Galerkin projection** condition, $\Big(\; r(\sum_{i=1}^{\ell} a_i(t)\phi_i), \phi_j \; \Big) = 0$, w.r.t. all $\{\phi_i\}_{i=1}^{\ell}$ and for all times $t \in \mathbb{T}$, the coefficients $a_j(t)$ are determined as the solutions of the ODE system

$$\dot{a}_j(t) \quad = \quad \Big(\; L[\textstyle\sum_{i=1}^{\ell} a_i(t)\phi_i] + f(\vec{\xi},t), \phi_j \; \Big). \tag{B.2.4}$$

Conversely, solving (B.2.4) for all $a_i(t)$ exactly for all times t yields the Galerkin projection condition.

Initial conditions for reduced model

A description of a dynamic system is not complete without specified initial conditions. Let $u(\vec{\xi}, 0) = g(x)$ be the initial condition of the PDE (B.2.1). The initial condition $a_i(0)$ for the ODE system (B.2.4) is then obtained simply by projecting the PDE initial condition onto the corresponding POD mode, i.e.

$$a_i(0) \quad = \quad (g, \phi_i). \tag{B.2.5}$$

Initial conditions involving higher-order time-derivatives (i.e., when the PDE itself involves u_{tt}, u_{ttt}, etc.) are translated to initial ODE conditions in the similar way. For instance, for a PDE involving u_{tt}, an initial condition $\frac{\partial u}{\partial t}(\vec{\xi}, 0) = h(\vec{\xi})$ is transformed into the corresponding ODE initial condition by $\dot{a}_i(0) = (h, \phi_i)$.

Example B.2.1. (*Diffusion equation*: single-phase 1D compressible reservoir pressure equation)

$$\begin{aligned} \frac{\partial p}{\partial t}(x,t) &= \kappa \frac{\partial^2 p}{\partial x^2}(x,t) \\ p(x,0) &= g(x), \end{aligned} \tag{B.2.6}$$

with $\kappa = \frac{k}{\varphi \mu c_t}$ hydraulic *diffusivity* of the porous medium, k and φ the permeability and the porosity of the medium, respectively, μ the viscosity of the fluid,

and c_t the total compressibility ($c_t = c_f + c_r$, with c_f and c_r the fluid and the rock compressibility, respectively). For a pressure independent diffusivity (N.B. κ is then constant in time only, i.e. it may be nonuniform spatially), approximation $p(x,t) \simeq \sum_{i=1}^{\ell} a_i(t)\phi_i(x)$ yields the (linear) dynamics for the coefficients equations

$$\begin{aligned} \dot{a}_j(t) &= \left(\kappa \textstyle\sum_{i=1}^{\ell} a_i(t) \frac{\partial^2 \phi_i(x)}{\partial x^2}, \phi_j(x) \right). \\ a_i(0) &= (g, \phi_i). \end{aligned} \tag{B.2.7}$$

If diffusivity is a function of the pressure, i.e. $\kappa = \kappa(p)$, one should replace κ with $\hat{\kappa} := \kappa(\sum_{i=1}^{\ell} a_i(t)\phi_i(x))$, denoting the fact that the diffusivity is approximated as well. Thus, in general case one should write

$$\begin{aligned} \dot{a}_j(t) &= \left(\hat{\kappa} \textstyle\sum_{i=1}^{\ell} a_i(t) \frac{\partial^2 \phi_i(x)}{\partial x^2}, \phi_j(x) \right). \\ a_i(0) &= (g, \phi_i). \end{aligned} \tag{B.2.8}$$

Note that in the linear case, the inner product ($B.2.7$) could be evaluated *a priori*, i.e. prior to the use of the reduced order model. More specifically, one would need to compute all inner products $\langle \sum_{i=1}^{\ell} \frac{\partial^2 \phi_i}{\partial x^2}, \phi_j \rangle$ for all $i, j \in \{1, \ldots, \ell\}$, only once[2], and one would not need ϕ's any further. On the other side, when the diffusivity is pressure dependent, $\kappa(\sum_{i=1}^{\ell} a_i(t)\phi_i(x))$ need to be evaluated at every time-step during the simulation of the reduced model ($B.2.8$), so ϕ's needs to be kept and used all the time. This fact is a major obstacle in achieving a considerable reduction of CPU-time in the general non-linear case.

[2]The same would also be true in a hypothetical (nonlinear) case of a κ being a polynomial functions of pressure, i.e. $\kappa := \sum_{i=1}^{s} d_i \phi_i^{r_i}$, whereby extra inner products of the type $d_i \phi_i^{r_i} \phi_j = \{d_j \phi_j^{r_j+1}$ *if* $i = j$, *and* 0 *otherwise*$\}$ would appear.

Bibliography

[1] Achatz, U., Schmitz, G. and Greisiger, K.-M., *Principal interaction patterns in baroclinic wave life cycles*, J. Atmos. Sci. **52** (1995), 3021–3213.

[2] Adamyan, V. M. , Arov, D. Z. and Krein, M. G., *Analytic properties of Schmidt pairs of a Hankel operator and generalized Schur-Takagi problem*, Mat. Sbornik 86 (1971), 33-73.

[3] Afanasiev, K. and Hinze, M., *Adaptive control of a wake flow using proper orthogonal decomposition*, Shape optimization and optimal design (Cambridge, 1999), Lecture Notes in Pure and Appl. Math., vol. 216, Dekker, New York, 2001, pp. 317–332.

[4] Ahuja, S., Rowley, C.W., Kevrekidis, I.G., Wei, M., Colonius, T. and Tadmor, G., *Low-dimensional models for control of leading-edge vortices: Equilibria and linearized models*, AIAA Paper 2007-709, 45th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, January, 2007.

[5] Aitokhuehi, I. and Durlofsky, L.J., *Optimizing the performance of smart wells in complex reservoirs using continuously updated geological models*, Journal of Petroleum Science and Engineering, 48, 254-264, 2005.

[6] Alaybeyi, M.M., Lee, J.Y. and Rohrer, R.A., *Numerical integration and asymptotic waveform evaluation*, Technical Digest of the IEEE International Conference on Computer-Aided Design, 1992.

[7] Al-Khodhori, S.M., *Smart well technologies implementation in pdo for production & reservoir management & control*, Middle East Oil Show, 9-12 June 2003, Bahrain, 2003. SPE 81486.

[8] Almroth, B.O., Stern, P., and Brogan, F.A., *Automatic choice of global shape functions in structural analysis*, AIAA Journal **16** (1978), 525–528.

[9] Anderson, J.L., *An ensemble adjustment Kalman filter for data assimilation*, Monthly Weather Review, 129(12), 2884-2903 (2001).

[10] Antoulas, A. C., *Approximation of large-scale dynamical systems (advances in design and control)*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2005.

[11] Antoulas, A.C., *Approximation of large-scale systems*, SIAM, Philadelphia, 2005.

[12] Antoulas, A. C. and Sorensen, D. C., *Approximation of large-scale dynamical systems: an overview*, Int. J. Appl. Math. Comput. Sci. **11** (2001), no. 5, 1093–1121.

[13] Antoulas, A. C., Sorensen, D. C. and Gugercin, S., *A survey of model reduction methods for large-scale systems*, Structured matrices in mathematics, computer science, and engineering, I (Boulder, CO, 1999), Contemp. Math., vol. 280, Amer. Math. Soc., Providence, RI, 2001, pp. 193–219.

[14] Antoulas, A. C., Sorensen, D. C. and Zhou, Y., *On the decay rate of Hankel singular values and related issues*, Systems Control Lett. **46** (2002), no. 5, 323–342.

[15] Arenas, E. and Dolle, N., *Smart waterflooding tight fractured reservoirs using inflow control valves*, SPE Annual Technical Conference and Exhibition, 5-8 October 2003, Denver, Colorado, 2003. SPE 84193.

[16] Arian, E., Fahl, M. and Sachs, E.W., *Trust–region proper orthogonal decomposition for flow control*, Tech. Report 2000–25, ICASE, NASA Langley Research Center, Hampton VA 23681–2299, 2000.

[17] Armbruster, D., Stone, E. and Heiland, R., *Towards analyzing the dynamics of flames*, Fields Inst. Commun. **5** (1-17), 337–355.

[18] Asheim, H., *Maximization of water sweep efficiency by controlling production and injection rates*, Paper SPE 18365, in Proc. SPE European Petroleum Conference, London, UK, 1988.

[19] Astrid, P., *Reduction of process simulation models: a proper orthogonal decomposition approach*, PhD Dissertation, Eindhoven University of Technology, 2004.

[20] Astrid, P. and Weiland, S., *On the construction of POD models from partial observations*, Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on (2005), 2272–2277.

[21] Astrid, P., Weiland, S., Willcox, K. and Backx, T., *Missing point estimation in models described by proper orthogonal decomposition*, CDC. 43rd IEEE Conference on Decision and Control **2** (2004), 1767–1772.

[22] Atwell, J.A. and King, B.B., *Proper orthogonal decomposition for reduced basis feedback controllers for parabolic equations*, Mathematical and Computer Modelling **33(1-3)** (2001), 1–19.

[23] Atwell, J.A., Borggaard, J. and King, B.B., *Reduced order controllers for burgers' equation with a nonlinear observer*, Int. J. Appl. Math. Comput. Sci. **11(6)** (2001), 1311–1330.

[24] Audigane, P. and Blunt, M.J., *Dual mesh method for upscaling in waterflood simulation*, Transp. Porous Media **55** (2004), no. 1, 71–89.

[25] Aziz, K. and Settari, A., *Petroleum reservoir simulation*, Applied Science Publishers, London, 1979.

[26] Babu, D.K. and Odeh, A.S., *Productivity of a horizontal well*, SPE Reservoir Engineering, p. 417-421, November 1989.

[27] Bader, B., Ghattas, O., van Bloemen Waanders, B. and Willcox, K., *An optimization framework for goal oriented, model-based reduction of large-scale systems*, Conference Paper, Joint 44th IEEE CDC and ECC conference, Seville, Spain, December 2005.

[28] Balmes, E., *Parametric families of reduced finite element models: Theory and applications*, Mechanical Systems and Signal Processing **10** (1996).

[29] Banks, H., Rosario, R. and Smith, R., *Reduced-order model feedback control design: numerical implementation in a thin shell model*, IEEE Trans. Automat. Control **45** (2000), no. 7, 1312–1324.

[30] Banks, H.T., Joynern, M.L., Wincheski, B. and Winfree, W.P., *Nondestructive evaluation using a reduced-order computational methodology*, Inverse Problems **16** (2000), no. 4, 929–945.

[31] Barnett, S. and Storey, C., *Some applications of the Lyapunov matrix equation*, IMA Journal of Applied Mathematics 1968 4(1):33-42.

[32] Barrett, A. and Reddien, G., *On the reduced-basis method*, Z. Angew. Math. Mech. **75** (1995).

[33] Bartels, R.H. and Stewart, G.W., *Solution of the matrix equation ax + xb = c [f4]*, Commun. ACM **15** (1972), no. 9, 820–826.

[34] Bauer, D. and Jansson, M., *Identification of the deterministic part of mimo state space models*, Automatica, vol. 36, no. 4, pp. 497–509, Apr. 2000.

[35] Baur, U. and Benner, P., *Factorized solution of Lyapunov equations based on hierarchical matrix arithmetic*, Computing **78** (2006), no. 3, 211–234.

[36] ———, *Gramian-based model reduction for data-sparse systems*, TU Chemnitz Scientific Computing Preprints, Preprint 07-01 (2007).

[37] Bear, J., *Dynamics of fluids in porous media*, Elsevier, New York, 1972.

[38] Beavers, A.N. and Denman, E.D., *A new solution method for the Lyapunov matrix equations*, SIAM J. Appl. Math., 29:416–421, 1975.

[39] Benner, P., *Numerical linear algebra for model reduction in control and simulation*, GAMM Mitteilungen, Fak. f. Mathematik, TU Chemnitz (ISBN/ISSN: 0936-7195) **29** (2006), no. 2, 275–296.

[40] Benner, P. and Quintana-Ortí, E.S., *Solving stable generalized Lyapunov equations with the matrix sign function*, 1997.

[41] Benner, P., Byers, R., Quintana-Ortí, E.S. and Quintana-Ortí, G., *Solving algebraic Riccati equations on parallel computers using Newton's method with exact line search*, Parallel Computing **26** (2000), no. 10, 1345–1368.

[42] Benner, P., Quintana-Ortí, E.S. and Quintana-Ortí, G., *State-space truncation methods for parallel model reduction of large-scale systems*, Parallel Comput. **29** (2003), no. 11-12, 1701–1722.

[43] Blair, P.M. and Weinaug, C.F., *Solution of two phase flow problems using implicit difference equations*, paper SPE 2185 presented at the 43rd Annual Fall Meeting held in Houston, Texas, Sept. 29 - Oct. 2, 1968.

[44] Bond, B. and Daniel, L., *Parameterized model order reduction of nonlinear dynamical systems*, Proceedings of the IEEE Conference on Computer-Aided Design, November 2005.

[45] Borggaard, J., Hay, A. and Pelletier, D., *Interval-based reduced-order models for unsteady fluid flow*, International Journal of Numerical Analysis and Modeling, 4(34):353-367, (2007).

[46] Brouwer, D.R., *Dynamic water flood optimization with smart wells using optimal control theory*, PhD Dissertation, TU Delft, 2004.

[47] Brouwer, D.R. and Jansen, J.D., *Dynamic optimization of waterflooding with smart wells using optimal control theory*, SPE Journal, SPE78278-PA, Vol. 9(4), (2004), p. 391-402, December 2004.

[48] Brouwer, D.R., Nævdal, G., Jansen, J.D., Vefring, E.H. and van Kruijsdijk, C.P.J.W., *Improved reservoir management through optimal control and continuous model updating*, SPE 90149, SPE Annual Technical Conference and Exhibition, 26-29 September, Houston, Texas (2004).

[49] Bui-Thanh, T. and Damodaran, M. and Willcox, K., *Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition*, AIAA Journal **42** (2004), no. 8, 1505–1516.

[50] Bui-Thanh, T., Willcox, K., Ghattas, O. and van Bloemen Waanders, B., *Goal-oriented, model-constrained optimization for reduction of large-scale systems*, Journal of Computational Physics **224** (2007), no. 2, 880–896.

[51] Burkardt, J., Du, Q., Gunzburger, M. and Lee, H-C., *Reduced-order modeling of complex systems*, Proceedings, NA03 Dundee, 2003.

[52] Burkardt, J., Gunzburger, M. and Lee, H-C., *Centroidal voronoi tessellation-based reduced-order modeling of complex systems*, SIAM J. Sci. Comput. **28** (2006), no. 2, 459–484.

[53] _____, *POD and CVT-based reduced-order modeling of Navier-Stokes flows*, Computer Methods in Applied Mechanics and Engineering **196** (2006), 337–355.

[54] Byers, R., *Solving the algebraic Riccati equation with the matrix sign function*, Linear Algebra Appl. 85 (1987) 267-279, 1987.

[55] Cardoso, M.A., *Machine learning applied to 3-d reservoir simulation*, CS 229 Machine Learning Final Projects, Autumn 2007 (available from: http://www.stanford.edu/class/cs229/projects2007.html).

[56] Cardoso, M.A. and Durlofsky, L., *Investigation of reduced-order models for reservoir simulation*, 24^{th} SUPRI-B Industrial Affiliates Meeting, May 8-9, 2007.

[57] Cardoso, M.A., Durlofsky, L. and Sarma, P., *Development and application of reduced-order modeling procedures for subsurface flow simulation*, (to appear in) Int. J. for Numerical Methods in Engineering (IJNME); Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nme.2453, August 2008.

[58] Cazemier, W., *Proper orthogonal decomposition and low dimensional models for turbulent flow*, Ph.D. Thesis, University of Groningen, Groningen, The Netherlands, 1997.

[59] Chen, Y., *Model order reduction for nonlinear systems*, MS thesis, MIT, 1999.

[60] Chen, Y. and Oliver, D.S., *Efficient Ensemble-Based Closed-Loop Production Optimization*, SPE 112873. The 2008 SPE Improved Oil Recovery Symposium held in Tulsa, Oklahoma, U.S.A., 19-23 April 2008.

[61] Chen, Z. and Hou, T.Y., *A mixed multiscale finite element method for elliptic problems with oscillating coefficients*, Math. Comput. **72** (2003), no. 242, 541–576.

[62] Chiprout, E. and Nakhla, M., *Addressing high-speed interconnect issues in asymptotic waveform evaluation*, DAC '93: Proceedings of the 30th international conference on Design automation (New York, NY, USA), ACM, 1993, pp. 732–736.

[63] _____, *Transient waveform estimation of high-speed mcm networks using complex frequency hopping*, In Proc. of IEEE Multi-Chip Module Conference MCMC-93, 1993.

[64] Chiuso, A. and Picci G., *Some algorithmic aspects of subspace identification with inputs*, Applied Mathematics and Computer Sciences, 11, 1, pp. 55-76 (2001).

[65] Christie, M.A. and Blunt, M.J., *Tenth SPE comparative solution project: A comparison of upscaling techniques*, SPE, 72469, 2001. url: www.spe.org/csp.

[66] Clemens, M., Wilke, M. and Weiland, T., *Extrapolation strategies in numerical schemes for transient magnetic field simulations*, Magnetics, IEEE Transactions on **39** (2003), no. 3, 1171–1174.

[67] Clemens, M., Wilke, M., Schumann, R. and Weiland, T., *Subspace projection extrapolation scheme for transient field simulations*, Magnetics, IEEE Transactions on **40** (2004), no. 2, 934–937.

[68] Crommelin, D.T. and Majda, A.J., *Strategies for model reduction: Comparing different optimal bases*, J. Atmos. Sci. **61** (2004), no. 17, 2206–2217.

[69] Daoud, A.M., *Automatic history matching in bayesian framework for field-scale applications*, PhD thesis, Texas A&M University, 2004.

[70] Davidson, A.M., *Balanced systems and model reduction*, Electronics Letters (22), pp.531-532, 1986.

[71] Dietrich, J.K. and Kuo, S.S., *Predicting horizontal well productivity*, JCPT, February 29, 1996.

[72] DiStefano, J., *On the relationships between structural identifiability and the controllability, observability properties*, Automatic Control, IEEE Transactions on Automatic Control, 22(4), Aug 1977.

[73] Duff, I. S. and Meurant, G. A., *The effect of ordering on preconditioned conjugate gradients*, BIT **29** (1989), no. 4, 635–657.

[74] Du, Q. and Gunzburger, M., *Centroidal voronoi tessellation based proper orthogonal decomposition analysis*, in Proceedings of the Fluids Engineering Division Summer Meeting, FEDSM2002-31051, ASME, 2002.

[75] ———, *Model reduction by proper orthogonal decomposition coupled with centroidal voronoi tessellation*, in Proceedings of the 8th Conference on Control of Distributed Parameter Systems, Birkhauser, Basel, Switzerland, 2002, p. 137150.

[76] Du, Q., Faber, V. and Gunzburger, M., *Centroidal voronoi tessellations: Applications and algorithms*, SIAM Review **41** (1999), no. 4, 637–676.

[77] Durlofsky, L. J., *Numerical calculation of equivalent grid block permeability tensors for heterogeneous porous media*, Water Resour. Res. **27** (1991), 699–708.

[78] Durlofsky, L.J. and Efendiev, Y., *An adaptive localglobal multiscale finite volume element method for two-phase flow simulations*, Advances in Water Resources Volume 30, Issue 3, March 2007, Pages 576–588.

[79] Efendiev, Y., Ginting, V., Hou, T. Y. and Ewing, R., *Accurate multiscale finite element methods for two-phase flow simulations*, submitted to J. Comput. Phys., 2005.

[80] Ellner, N.S. and Wachspress, E.L., *New ADI model problem applications*, ACM '86: Proceedings of 1986 ACM Fall joint computer conference (Los Alamitos, CA, USA), IEEE Computer Society Press, 1986, pp. 528–534.

[81] ———, *Alternating direction implicit iteration for systems with complex spectra*, SIAM J. Numer. Anal. **28** (1991), no. 3, 859–870.

[82] Ertekin, T., Abou-Kassem, J. H. and King, G. R., *Basic applied reservoir simulation*, SPE Textbook Series 7.

[83] Evensen, G., *Data Assimilation: The Ensemble Kalman Filter*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[84] _____, *Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistics*, J. Geophys. Res., 99(C5), pp. 10,143-10,162, May 15, 1994.

[85] Everson, R. and Sirovich, L., *The Karhunen-Loève procedure for gappy data*, J. Opt. Soc. **12(8)** (1995), 1657–1664.

[86] Feldmann, P. and Freund, R.W., *Efficient linear circuit analysis by Padé approximation via the Lanczos process*, Proceedings of EURO-DAC '94 with EURO-VHDL '94, Grenoble, France, IEEE Computer Society Press, 1994, pp. 170–175.

[87] _____, *Reduced-order modeling of large linear subcircuits via a block Lanczos algorithm*, Proceedings of the 32nd Design Automation Conference, San Francisco, California, Association for Computing Machinery, Inc., 1995, pp. 474–479.

[88] Fernardo, K.V.and Nicholson H., *Singular perturbational model reduction of balanced systems*, IEEE Transactions on Automatic Control, AC-27(2), 466-468, 1982.

[89] Fink, J.P. and Rheinboldt, W.C., *On the error behavior of the reduced basis technique for nonlinear finite element approximations*, Z. Angew. Math. Mech. **63** (1983), 21–28.

[90] Freund, R., *Passive reduced-order modeling via Krylov-subspace methods*, In Proceedings of the 14th Symposium on Mathematical Theory of Networks and Systems, Perpignan, France, June 19-23, 2000.

[91] Freund, R.W., *Passive reduced-order modeling via Krylov-subspace methods*, Proceedings of the 14th Symposium on Mathematical Theory of Networks and Systems (Perpignan, France), June 19–23, 2000.

[92] Gaier, D. and Todd, J., *On the rate of convergence of optimal ADI processes*, Numerische Mathematik 9, 452–459, 1967.

[93] Gai, H., *Downhole flow control optimization in the worlds 1^{st} extended reach multilateral well at wytch farm*, SPE/IADC Drilling Conference, 27 February-1 March 2001, Amsterdam, Netherlands, February 2001. SPE 67728.

[94] Gajic Z. and Lelic M., *Refinement of balancing order reduction techniques using Kessler's formula*, American Control Conference, 2001. Proceedings of the 2001 **6** (2001), 4740–4745 vol.6.

[95] _____, *Improvement of system order reduction via balancing using the method of singular perturbations*, Automatica **37** (November 2001), 1859–1865(7).

[96] Gallivan, K., Grimme, E.J. and Van Dooren, P., *A rational Lanczos algorithm for model reduction*, Numer. Algorithms **12** (1996), no. 1-2, 33–63.

[97] Gallopoulos, E. and Saad, Y., *Efficient solution of parabolic equations by Krylov approximation methods*, SIAM J. Sci. Stat. Comput. **13** (1992), no. 5, 1236–1264.

[98] Gardiner, J.D. and Laub, A.J., *A generalization of the matrix sign function solution for algebraic Riccati equations*, Decision and Control, 1985 24th IEEE Conference on **24** (Dec. 1985), 1233–1235.

[99] Gautier, Y., Blunt, M.J. and Christie, M.A., *Nested gridding and streamline-based simulation for fast reservoir performance prediction*, Comput. Geosci. 3 (1999), pp. 295–320.

[100] Gerritsen, M. G. and Durlofsky, L.J., *Modeling Fluid Flow in Oil Reservoirs*, Annual Review of Fluid Mechanics **37** (2005), 211–238.

[101] Ghanem, R. and Dham, S., *Stochastic finite element analysis for multiphase flow in heterogeneous porous media*, Transp. Porous Media **32** (1998), no. 3, 239–262.

[102] Ghanem, R. and Spanos, P.D., *Stochastic finite elements: a spectral approach*, Springer-Verlag New York, Inc., New York, NY, USA, 1991.

[103] Gharbi, R.B., Smaoui, N. and Peters, E.J., *Prediction of unstable fluid displacements in porous media using the Karhunen-Loève decomposition*, *In Situ* 1997; **21**(4): 331-356.

[104] Gildin, E., Klie, H., Rodriguez, A., Wheeler, M.F. and Bishop, R.H., *Development of low-order controllers for high-order reservoir models and smart wells*, paper presented at the 2006 SPE Annual Technical Conference and Exhibition held in San Antonio, Texas, U.S.A., 2427 September 2006.

[105] Glavaski, S., *Robust system analysis and nonlinear system model reduction*, PhD Thesis, California Institute of Technology, May 1998.

[106] Glavaski, S., Marsden, J. E., and Murray, R. M., *Model reduction, centering, and the Karhunen-Loève expansion*, Proc. CDC **37** (1998), 2071 2076.

[107] Glover, K., *All optimal Hankel-norm approximations of linear multivariable systems and their $\mathcal{L}_\infty$ error bounds*, Int.J.Control, Vol 39, No. 6, 1984, pp. 1115-1193.

[108] Gonzalez-Rodriguez, P., Kindelan, M., Moscoso, M. and Dorn, O., *History matching problem in reservoir engineering using the propagation-backpropagation method*, Inverse Problems **21** (April 2005), 565–590.

[109] Gopal, M., *Modern control system theory*, Halsted Press, New York, NY, USA, 1984.

[110] Gragg, W.B. and Lindquist, A., *On the partial realization problem*, Linear Algebra and Applications, Vol. 50, pp. 277-319, 1983.

[111] Grasedyck, L., *Existence of a low rank or h-matrix approximant to the solution of a sylvester equation*, Numerical Linear Algebra with Applications, Vol. 22(4), pp. 371-390, 2004.

[112] Grasedyck, L. and Hackbusch, W., *A multigrid method to solve large scale sylvester equations*, SIAM J. Matrix Anal. Appl. **29** (2007), no. 3, 870–894.

[113] Grasedyck, L., Hackbusch, W. and Khoromskij, B.N., *Solution of large scale algebraic matrix Riccati equations by use of hierarchical matrices*, Computing **70** (2003), no. 2, 121–165.

[114] Green, D.W. and Limebeer, D.J.N, *Linear robust control*, Prentice-Hall, Englewood Cliffs, NJ, 1995.

[115] Grimme, E.J., *Krylov projection methods for model reduction*, Ph.D. thesis, Coordinated-Science Laboratory, University of Illinois at Urbana-Champaign, Urbana-Champaign, IL, 1997.

[116] Gugercin, S., Sorensen, D.C. and Antoulas, A.C., *A modified low-rank Smith method for large-scale Lyapunov equations*, Numer. Algorithms, (32):27-55, 2002.

[117] Gu, M. and Eisenstat, S.C., *A stable and fast algorithm for updating the singular value decomposition*, Tech. Report YALEU/DCS/RR-966, New Haven, CT, 1993.

[118] Gunzburger, M., *Finite element methods for viscous incompressible flows: a guide to theory, practice and algorithms*, Academic Press, London, 1989.

[119] Hall, P., Marshall, D. and Martin, R., *Incremental eigenanalysis for classification*, Proc. Ninth British Mashine Vision Conference (BMVC), Southampton, 1998.

[120] Hammarling, S.J., *Numerical solution of the stable, non-negative definite Lyapunov equation*, IMA J. of Numerical Analysis, Vol. 2, 1982, pp. 303-323.

[121] Hassanizadeh, S.M., Celia, M.A., and Dahle, H.K., *Dynamic effect in the capillary pressuresaturation relationship and its impacts on unsaturated flow*, Vadose Zone Journal, 1:38-57 (2002).

[122] Heemink, A., Verlaan, M. and Segers, A., *Variance reduced ensemble Kalman filtering*, Monthly Weather Review, 129:1718–1728, 2001.

[123] Heijn, T., Markovinović, R., and Jansen, J.D., *Generation of low-order reservoir models using system-theoretical concepts*, SPE Journal, June 2004, 202-218.

[124] Helmersson, A., *Model reduction using LMIs*, Proceedings of the 33rd IEEE Conference on Decision and Control, Vol. 4, Lake Buena Vista, FL, USA, pp. 3217-3222, 1994.

[125] Hinze, M. and Volkwein, S., *Proper orthogonal decomposition surrogate models for nonlinear dynamical systems: Error estimates and suboptimal control*, In Reduction of Large-Scale Systems, P. Benner, V. Mehrmann, D. C. Sorensen (eds.), Lecture Notes in Computational Science and Engineering, Vol. 45, (2005), pp.261-306.

[126] Hodel, A. and Poolla, K., *Heuristic approaches to the solution of very large sparse Lyapunov and algebraic Riccati equations*, in Proc. 27th IEEE Conf. Decis. Cont., Austin, TX, 1988, pp. 2217–2222.

[127] Holmes, J.A., *Modeling advanced wells in reservoir simulation*, Journal of Petroleum Technology, Nov. 2001, p.54-66.

[128] Holmes, P., Lumley, J.L. and Berkooz, G., *Turbulence, coherent structures, dynamical systems and symmetry*, Cambridge Monographs on Mechanics, Cambridge University Press, Cambridge, New-York, 1996.

[129] Houtekamer, P.L. and Mitchell, H.L., *A sequential ensemble Kalman filter for atmospheric data assimilation*, Monthly Weather Review, 129(1), 123-137 (2001).

[130] Hou, T.Y. and Wu, X-H, *A multiscale finite element method for elliptic problems in composite materials and porous media*, J. Comput. Phys. **134** (1997), no. 1, 169–189.

[131] Huang, X., *Padé approximation of linearized circuit responses*, PhD Thesis, Dept. of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, November, 1990.

[132] IJzerman, W.L., *Signal representation and modeling of spatial structures in fluids*, PhD Thesis, Universiteit Twente, 2000.

[133] IJzerman, W.L. and van Groesen, E., *Low-dimensional model for vortex merging in the two-dimensional temporal mixing layer*, European Journal of Mechanics B - Fluids, 20:821-840, (2001).

[134] Ilak, M. and Rowley, C.W., *Reduced-order modeling of channel flow using traveling POD and balanced POD*, 3rd AIAA Flow Control Conference, 58 June 2006, San Francisco.

[135] ———, *Reduced-order modeling of channel flow using traveling POD and balanced POD*, 3rd AIAA Flow Control Conference, 58 June 2006, San Francisco, 2006.

[136] Innocent, M., Wambacq, P., Donnay, S., *An analytic volterra-series-based model for a MEMS variable capacitor*, IEEE Trans. on CAD of Integrated Circuits and Systems **22** (2003), no. 2, 124–131.

[137] Iollo, A., Lanteri, S. and Desideri, J., *Stability properties of POD-Galerkin approximations for the compressible navier-stokes equations*, Theoret. Comput. Fluid Dynamics **13** (2000), 377–396.

[138] Ito, K. and Ravindran, S., *A reduced basis method for control problems governed by PDEs*, Tech. Report CSRC–TR97–1, Center for Research in Scientific Computation, North Carolina State University, Raleigh, 1997, `ftp://ftp.ncsu.edu/pub/ncsu/crsc/crsc-tr97-1.ps.Z`.

[139] Ito, K. and Ravindran, S.S., *Optimal control of thermally convected fluid flows*, SIAM J. on Scientific Computing **19** (1998), 1847–1869.

[140] _____, *A reduced basis method for control problems governed by PDEs*, Optimal Control of Partial Differential Equations (Basel, Boston, Berlin) (W. Desch, F. Kappel, and K. Kunisch, eds.), Int. Series of Numer. Math. Vol. 126, Birkhäuser Verlag, 1998, pp. 153–168.

[141] _____, *A reduced order method for simulation and control of fluid flows*, J. Computational Physics **143** (1998), 403–425.

[142] Ito, K., Scroggs, J.S. and Tran, H.T., *Optimal control of thermally coupled Navier–Stokes equations*, Optimal Design and Control (Basel, Boston, Berlin) (J. Borggaard, J. Burkardt, M. D. Gunzburger, and J. Peterson, eds.), Birkhäuser Verlag, 1995, pp. 199–214.

[143] Jaimoukha, I.M. and Kasenally, E.M., *Krylov subspace methods for solving large Lyapunov equations*, SIAM J. Numer. Anal. **31** (1994), no. 1, 227–251.

[144] Jansen, J.D., Bosgra, O.H. and Van den Hof, P.M.J., *Model-based control of multiphase flow in subsurface oil reservoirs*, Journal of Process Control, **18**(9), October 2008, pp. 846–855.

[145] Jansen, J.D., Brouwer, D.R., Nævdal, G. and van Kruijsdijk, C.P.J.W., *Closed-loop reservoir management*, First Break, 23:43-48 (January 2005).

[146] Jansen, J.D., Wagenvoort, A.M., Droppert, V.S., Daling, R. and Glandt, C.A., *Smart well solutions for thin oil rims: Inflow switching and the smart stinger completion*, paper SPE 77942, presented at the SPE Asia Pacific Oil and Gas Exhibition, Melbourne, Australia, 8-10 October (2002).

[147] Jansson, M. and Wahlberg, B., *On consistency of subspace system identification methods*, Automatica, Volume 34, Number 12, December 1998 , pp. 1507-1519(13).

[148] Jenny, P., Lee, S.H. and Tchelepi, H.A., *Multi-scale finite-volume method for elliptic problems in subsurface flow simulation*, J. Comput. Phys. **187** (2003), no. 1, 47–67.

[149] Jolliffe, I.T., *Principal component analysis*, Springer Verlag, 1986.

[150] Kailath, T., *Linear systems*, Prentice-Hall, 1980.

[151] Kang, M. and Katsaggelos, A., *Simultaneous multichannel image restoration and estimation of the regularization parameters*, IEEE Trans. Image Processing 6, pp. 774–778, May 1997.

[152] Karhunen, K., *Zur spektral Theorie stochasticher Prozesse*, Ann. Acad. Sci. Fennicae, Ser. A1, 34 (1946), p.1–7.

[153] Kenney, C.S. and Laub, A.J., *The matrix sign function*, IEEE Trans. Automat. Control 40(8) (1995) 1330–1348.

[154] Kharghoria, A., Zhang, F., Li, R. and Jalali, Y., *Application of distributed electrical measurements and inflow control in horizontal wells under bottomwater drive*, European Petroleum Conference, 29-31 October 2002, Aberdeen, United Kingdom, 2002. SPE 78275.

[155] Kirby, M., *Minimal dynamical systems from PDEs using Sobolev eigenfunctions*, Phys. D **57** (1992), no. 3-4, 466–475.

[156] ———, *Geometric data analysis: An empirical approach to dimensionality reduction and the study of patterns*, John Wiley & Sons, Inc., New York, NY, USA, 2000.

[157] Kirby, M. and Armbruster, D., *Reconstructing phase space from PDE simulations*, Z. Angew. Math. Phys. **43** (1992), 9991022.

[158] Kirby, M. and Sirovich, L., *Application of the Karhunen-Loève procedure for the characterization of human faces*, IEEE Trans. Pattern Anal. Mach. Intell. **12** (1990), no. 1, 103–108.

[159] Kokotovic, P.V., O'Malley, R.E. and Sannuti, P., *Singular perturbations and order reduction in control theory — an overview*, Automatica, 12:123-132, 1976.

[160] Kong, H., Wang, L., Teoh, E.K., Li, X., Wang, J-G. and Venkateswarlu, R., *Generalized 2d principal component analysis for face image representation and recognition.*, Neural Networks **18** (2005), no. 5-6, 585–594.

[161] Korenberg, M. and Hunter, I.W., *The identification of nonlinear biological systems: Volterra kernel approaches*, Annals of Biomedical Engineering, vol. 24, pp. 250–268, 1996.

[162] Kung, S.Y., *A new identification and model reduction algorithm via singular value decomposition*, Proc. 12th Asilomar Conf. on Circuits, Systems and Computers, Pacific Grove, CA, pp. 705-714, 1978.

[163] Kwasniok, F, *Low-dimensional models of the Ginzburg–Landau equation*, SIAM Journal of Applied Mathematics **61** (2001), no. 6, 2063–2079.

[164] Lall, S., Marsden, J. E., and Glavaski, S., *Empirical model reduction of controlled nonlinear systems*, In Proceedings of the IFAC World Congress, Vol. **F**, (1999), 473-478.

[165] ———, *A subspace approach to balanced truncation for model reduction of nonlinear control systems*, Int. J. Robust Nonlinear Control, 12, (2002), 519-535.

[166] Lancaster, P., *Explicit solutions of linear matrix equations*, SIAM Rev. **12** (1970), 544–566.

[167] Larin, V. B. and Aliev, F. A., *Construction of square root factor for solution of the Lyapunov matrix equation*, Syst. Control Lett. **20** (1993), no. 2, 109–112.

[168] Laub. A.J., *Computation of balancing transformations*, Proc. of the Joint Automate Control Conf., Vol. II, 1980.

[169] Levine, W.S. (editor), *The control handbook*, CRC Press, 1996.

[170] Levy, A. and Rubinstein, J., *Hilbert-space Karhunen-Loève transform with application to image analysis*, JOSA-A **16** (1999), no. 1, 28–35.

[171] Li, J-R. and White, J., *Low-rank solution of Lyapunov equations*, SIAM Rev. **46** (2004), no. 4, 693–713.

[172] Liu, N., *Automatic history matching of geologic facies*, PhD thesis, University Of Oklahoma, Graduate College, 2005.

[173] Liu, Y. and Anderson, B.D.O., *Singular perturbational approximation of balanced systems*, International Journal of Control, 50(4), 1379-1405, 1989.

[174] Loeve, M., *Functions aleatoire de second ordre*, Revue Science, 84 (1946), p.195-206.

[175] Lorentzen, R. J., Berg, A. M., Nævdal, G. and Vefring, E. H., *A new approach for dynamic optimization of waterflooding problems*, SPE-99690. In: Proceedings of the 2006 SPE intelligent energy conference and Exhibition.

[176] Luenberger, D., *Introduction to dynamic systems : theory, models, and applications*, John Wiley and Sons Inc., 1979.

[177] Lumley, J.L., *The structure of inhomogeneous turbulence*, Atmospheric turbulence and wave propagation. In A. M. Yaglom and V. I. Tatarski (eds). Nauka: Moscow, 1967; 166-178.

[178] Lunati, I. and Jenny, P., *Multiscale finite-volume method for compressible multiphase flow in porous media*, J. Comput. Phys. **216** (2006), no. 2, 616–636.

[179] Lu, Z. and D. Zhang, *Stochastic simulations for flow in nonstationary randomly heterogeneous porous media using a kl-based moment-equation approach*, SIAM Multiscale Modeling and Simulation **6(1)** (2007), 228–245.

[180] Ly, H.V. and Tran, H.T., *Proper orthogonal decomposition for flow calculations and optimal control in a horizontal CVD reactor*, Tech. Report CRSC–TR98–13, Center for Research in Scientific Computing, North Carolina State University, Raleigh, NC, 1998, `http://www2.ncsu.edu/math/CRSC/reports/reports98.html`.

[181] Maday, Y. and Rønquist, E.M., *A reduced-basis element method*, J. Sci. Comput. **17** (2002), no. 1-4, 447–459.

[182] Markovinovic, R. and Jansen, J.D., *Accelerating iterative solution methods using reduced-order models as solution predictors*, Int. J. for Numerical Methods in Engineering (IJNME) **68** (2006), 525–541.

[183] Markovinović, R., Geurtsen, E.L., and Jansen, J.D., *Subspace identification of low-order reservoir models*, XIV International Conference on Computational Methods in Water Resources, June 23-28, Delft, The Netherlands, p.281-288.

[184] Markovinović, R., Geurtsen, E.L., Heijn, T., and Jansen, J.D., *Generation of low-order reservoir models using POD, empirical gramians and subspace identification*, 8th European Conference on the Mathematics of Oil Recovery - ECMOR VIII, Freiberg, Germany, Sept 3-6 2002.

[185] Markovinović, R., Jansen, J.D. and Rommelse, J.R., *Reduced representations in reservoir simulation - extending POD to include more general optimality conditions*, In: Proc. 9th European conference on the mathematics of oil recovery (ECMOR IX), EAGE, Houten, The Netherlands, 2004, pp. B10–1–B10–6.

[186] Marques, N., Kamon, M., White, J. and Silveira, L.M., *A mixed nodal-mesh formulation for efficient extraction and passive reduced-order modeling of 3d interconnects*, Design Automation Conference, 1998, pp. 297–302.

[187] Misra, P., Syrmos, V. and Lewis, F., *A numerical algorithm for the solution of generalized Lyapunov equations*, Decision and Control, 1993., Proceedings of the 32nd IEEE Conference on (15-17 Dec 1993), 3138–3143 vol.4.

[188] Moonen,M., De Moor, B., Vandeberghe, L. and Vandewalle, J., *On- and off-line identification of linear state-space models*, International Journal of Control **49** (1989), 219–232.

[189] Moonen, M., Van Dooren, P. and Vandewalle, J., *A singular value decomposition updating algorithm for subspace tracking*, SIAM Journal on Matrix Analysis and Applications **13** (1992), no. 4, 1015–1038.

[190] Moore, B. C., *Principal component analysis in linear systems: controlability, observability, and model reduction*, **AC-26** (1981), 17–32.

[191] Mustafa, D., *Balanced truncation is Hankel-norm optimal when used to approximate a finite-difference model of a parabolic PDE*, Proceedings of the 31st IEEE Conference on Decision and Control, 1992, vol. 3.

[192] ———, *A class of systems for which balanced truncation is Hankel-norm optimal*, Proceedings of the 30th IEEE Conference on Decision and Control, 11-13 Dec 1991, vol. 2, 1991, pp. 1943–1948.

[193] Mutambara, A.G.O., *Design and analysis of control systems*, CRC Press, Boca Raton, FL, 1999.

[194] Nævdal, G., Brouwer, D.R. and Jansen, J.D., *Waterflooding using closed-loop control*, Computational Geosciences **10** (March 2006), 37–60(24).

[195] Nævdal, G., Johnsen, L.M., Aanonsen, S.I. and Vefring, E.H., *Reservoir monitoring and continuous model updating using ensemble Kalman filter*, In SPE Annual Technical Conference and Exhibition, Denver, Colorado, USA, 5 - 8 October 2003. SPE 84372.

[196] Nagy, N., *Modal representation of geometrically nonlinear behaviour by the finite element method*, Computers and Structures **10** (1979), 683–688.

[197] Nguyen, V.V. and Wood, E.F., *Review and unification of linear identifiability concepts*, SIAM Review, 24, 1982, pp. 3451.

[198] Nikolaou, M., Cullick, A.S. and Saputelli, L., *Production optimizationa moving-horizon approach*, SPE 99358-MS (2006).

[199] Noor, A.K., *Recent advances in reduction methods for nonlinear problem*, Computers and Structures **13** (1981), 31–44.

[200] Noor, A.K. and Peters, J.M., *Reduced basis technique for nonlinear analysis of structures*, AIAA Journal **18** (1980).

[201] Odabasioglu, A., Celik, M. and Pileggi, L.T., *Prima: passive reduced-order interconnect macromodeling algorithm*, ICCAD, 1997, pp. 58–65.

[202] Oliveira, I.B. and Patera, A.T., *Reduced-basis techniques for rapid reliable optimization of systems described by affinely parametrized coercive elliptic partial differential equations*, Optim. Eng. **8** (2007), no. 1, 43–65.

[203] Oliver, D.S. and Reynolds, A.C. and Liu, N., *Inverse Theory for Petroleum Reservoir Characterization and History Matching*, Cambridge University Press, 2008.

[204] Olsson, K.H.A., *Model order reduction with rational Krylov methods*, Doctoral Thesis in Numerical Analysis. Stockholm University, Sweden 2005.

[205] Orellano, A., *POD analysis of coherent structures in forced turbulent flow over a fence*, Journal of Turbulence **2** (1 January 2001), 8–8(1).

[206] Ostrowski, Z., Białecki, R. A. and Kassab, A. J., *Estimation of constant thermal conductivity by use of Proper Orthogonal Decomposition*, Computational Mechanics **37** (2005), 52–59.

[207] Peaceman, D.W., *Interpretation of well-block pressures in numerical reservoir simulation*, paper SPE 6893, presented at the SPE-AIME 52nd Annual Fall Technical Conference and Exhibition, Denver, Oct 9-12, 1977.

[208] _____, *Representation of a horizontal well in numerical simulation*, paper SPE 21217 presented at the 11th SPE symposium on Reservoir Simulation, Anaheim, California, February 17-20, 1991.

[209] _____, *Fundamentals of numerical reservoir simulation*, Developments in Petroleum Science, **6**, Elsevier, 1977.

[210] Peaceman, D.W. and Rachford, H.H., *The numerical solution of parabolic and elliptic differential equations*, J. SIAM, , Vol. 3, pages 28-41., (1955).

[211] Penzl, T., *Lyapack - a Matlab toolbox for large Lyapunov and Riccati equations, model reduction problems, and linear-quadratic optimal control problems - users' guide (version 1.0)*.

[212] _____, *A cyclic low-rank Smith method for large sparse Lyapunov equations*, SIAM Journal on Scientific Computing **21** (2000), no. 4, 1401–1418.

[213] _____, *Decay bounds for solutions of Lyapunov equations: The symmetric case*, Technical University of Chemnitz, Germany, Dec 1999.

[214] Pernebo, L. and Silverman, L. M., *Model reduction via balanced state space representations*, **AC-27** (1982), no. 2(1976), 382–387.

[215] Peterson, J.S., *The reduced basis method for incompressible viscous flow calculations*, SIAM J. Sci. Stat. Comput. **10** (1989), no. 4, 777–786.

[216] Pillage, L.T. and Rohrer, R.A., *Delay evaluation with lumped linear rlc interconnect circuit models*, Proceedings of the decennial Caltech conference on VLSI on Advanced research in VLSI (Cambridge, MA, USA), MIT Press, 1989, pp. 143–158.

[217] Pillage, L.T., Huang, X. and Rohrer, R.A., *Awesim: asymptotic waveform evaluation for timing analysis*, DAC '89: Proceedings of the 26th ACM/IEEE conference on Design automation (New York, NY, USA), ACM, 1989, pp. 634–637.

[218] Portella, R.C.M. and Prais, F., *Use of automatic history matching and geostatistical simulation to improve production forecast*, SPE 53976, SPE Lat. Am. and Carrib. Pet. Eng. Conf., Caracas, Venezuela (21-23 April 1999).

[219] Prabhu, R.D., Collis, S.S and Chang, Y., *The influence of control on proper orthogonal decomposition of wall-bounded turbulent flows*, Physics of Fluids **13** (2001), no. 2, 520–37.

[220] Przybysz-Jarnut, J., Remus, H., Jansen, J.D. and Heemink, A.W., *Application of the representer method for parameter estimation in numerical reservoir models*, Computational Geosciences **11** (March 2007), 73–85(13).

[221] Qin, S.J., *An overview of subspace identification*, Computers & Chemical Engineering, Vol. 30, Issues 10-12, 12 September 2006, Pages 1502–1513.

[222] Quintana-Ortí, E.S. and van de Geijn, R., *Specialized parallel algorithms for solving linear matrix equations in control theory*, Journal of Parallel and Distributed Computing, 61:1489–1504, 2001.

[223] Ravindran, S.S., *A reduced-order approach for optimal control of fluids using proper orthogonal decomposition*, International Journal for Numerical Methods in Fluids **34** (2000), 425–448.

[224] Renard, Ph. and de Marsily, G., *Calculating equivalent permeability: a review*, Advances in Water Resources, Vol. 20, Issues 5-6, October-December 1997, Pages 253–278.

[225] Rewieński, M., *A trajectory piecewise-linear approach to model order reduction of nonlinear dynamical systems*, PhD Dissertation, MIT, 2003.

[226] Rewieński, M. and White, J., *A trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micromachined devices*, ICCAD '01: Proceedings of the 2001 IEEE/ACM international conference on

Computer-aided design (Piscataway, NJ, USA), IEEE Press, 2001, pp. 252–257.

[227] ———, *Improving trajectory piecewise-linear approach to nonlinear model order reduction for micromachined devices using an aggregated projection basis*, in: Proceedings of the 5th International Conference on Modeling and Simulation of Microsystems, 2002.

[228] ———, *Model order reduction for nonlinear dynamical systems based on trajectory piecewise-linear approximations*, Linear Algebra and its Applications, 2005.

[229] Roberts, J.D., *Linear model reduction and solution of the algebraic Riccati equation by use of the sign function*, Internat. J. Control, 32(4):677687, 1980. (Reprint of Technical Report TR-13, CUED/B-Control, Engineering Deparment, Cambrige University, 1971).

[230] Rodrigues, J., *Calculating derivatives for automatic history matching*, Computational Geosciences **10** (2006), no. 1, 119–136.

[231] Romero, C.E., Carter, J.N., Zimmerman, R.W. and Gringarten, A.C., *Improved reservoir characterization through evolutionary computation*, SPE 62942, SPE Ann. Tech. Conf. and Exh., Dallas, Texas (1 - 4 October 2000).

[232] Rommelse, J.R., Kleptsova, O., Jansen, J.D. and Heemink, A.W., *Data assimilation in reservoir management using the representer method and the ensemble Kalman filter*, Proc. 10th European Conference on the Mathematics of Oil Recovery - ECMOR X, Amsterdam, Netherlands (2006) September 4-7.

[233] Rowley, C.W., *Model reduction for fluids, using balanced proper orthogonal decomposition*, Int. J. on Bifurcation and Chaos **15** (2005), no. 3, 997–1013.

[234] Rowley, C. W. and Marsden, J. E., *Reconstruction equations and the karhuhnen-loeve expansion for systems with symmetry*, Physica D **142** (2000), 1–19.

[235] Rozza, G., *Reduced-basis methods for elliptic equations in sub-domains with a posteriori error bounds and adaptivity*, Appl. Numer. Math. **55** (2005), no. 4, 403–424.

[236] Saad, Y., *Numerical solution of large Lyapunov equations*, Signal Processing, Scattering, Operator Theory, and Numerical Methods. Proceedings of the international symposium MTNS-89, vol III (Cambridge, MA, USA; Berlin, Germany; Basel, Switzerland) (M. A. Kaashoek, J. H. van Schuppen, and A. C. Ran, eds.), Birkhäuser, 1990, pp. 503–511.

[237] ———, *Analysis of some Krylov subspace approximations to the matrix exponential operator*, SIAM J. Numer. Anal. **29** (1992), no. 1, 209–228.

[238] Salihbahrami, B. and Lohmann, B., *Stopping criterion in order reduction of large scale systems using Krylov subspace methods*, Proceedings Appl. Math. Mech. (PAMM), Vol. 4, Issue 1, December 2004, pp. 682-683.

[239] _____, *Krylov subspace methods in linear model order reduction: Introduction and invariance properties*, Scientific Report, Univ. of Bremen, Germany, 2002.

[240] Saputelli, L., Nikolaou, M. and Economides, M.J., *Real-time reservoir management: A multiscale adaptive optimization and control approach*, Computational Geosciences **10** (2006), 61–96(36).

[241] Sarma, P., Aziz, K. and Durlofsky, L.J., *Implementation of adjoint solution for optimal control of smart wells*, Paper SPE 92864 presented at the SPE Reservoir Simulation Symposium, Houston, Jan. 31-Feb. 2. (2000).

[242] Sarma, P., Durlofsky, L., Aziz, K. and Chen, W., *Efficient real-time reservoir management using adjoint-based optimal control and model updating*, Computational Geosciences **10** (March 2006), 3–36(34).

[243] Scherpen, J.M.A., *Balancing for nonlinear systems*, Systems and Control Letters, vol.21, 143-153, 1993.

[244] _____, *Balancing for nonlinear systems*, PhD Dissertation, University of Twente, 1994.

[245] Scherpen, J.M.A., and Gray, W.S., *On singular value functions and Hankel operators for nonlinear systems*, In Proc. ACC'99, San Diego, CA, USA, pp.2360-2364.

[246] _____, *Sufficient conditions for minimality of a nonlinear realization via controllability and observability functions*, Proc. 1998 American Contr. Conf., Philadelphia, PA, pp. 3349-3353.

[247] Schulze-Riegert, R.W., Axmann, J.K., Haase, O., Rian, D.T. and You, Y.-L., *Optimization methods for history matching of complex reservoirs*, SPE 66393 (2001).

[248] Sheehan, B.N. and Saad, Y., *Higher order orthogonal iteration of tensors (hooi) and its relation to pca and glram.*, SDM, SIAM, 2007.

[249] Sinha, S., Kumar, R., Vega, L. and Jalali, Y., *Flow equilibration towards horizontal wells using downhole valves*, paper SPE 68635, presented at the SPE Asia Pacific Oil and Gas Conference and Exhibition, Jakarta, Indonesia, 17-19 April (2001).

[250] Sirovich, L., *Turbulence and the dynamics of coherent structures, parts 1, 2 and 3*, Quarterly Applied Mathematics, 45:561-590 (1987).

[251] Smaoui, N. and Gharbi, R.B., *Using Karhunen-Loève decomposition and artificial neural network to model miscible fluid displacement in porous media*, Applied Mathematical Modelling **24** (2000), 657–675(19).

[252] Smith, R.A., *Matrix equation $XA+BX = C$*, SIAM Journal on Applied Mathematics, 16(1):198–201, 1968.

[253] Snaith, N., Chia, R., Narayasamy, D. and Schrader, K., *Experience with operation of smart wells to maximise oil recovery from complex reservoirs*, SPE International Improved Oil Recovery Conference in Asia Pacific, 20-21 October 2003, Kuala Lumpur, Malaysia, 2003. SPE 84855.

[254] Sontag, E.D., *Mathematical control theory*, Springer-Verlag, New-York, NY, 2nd Edition, 1998.

[255] Spillete, A.G., Hilestad, J.G. and Stone, H.L., *A high-stability sequential solution approach to reservoir simulation*, paper SPE 4542 presented in the 48th Annual Fall Meeting, Las Vegas, Sept.-Oct., 1973.

[256] Stigter, J.D. and Peeters, R.L.M., *On a geometric approach to the structural identifiability problem and its application in a water quality case study*, In Proc. European Control Conference, pages 3450-3456. Koss,Greece, 2007.

[257] Stykel, T., *A modified matrix sign function method for projected Lyapunov equations*, Systems and Control Letters, vol.56, 695-701, 2007.

[258] Sudaryanto, B. and Yortsos, Y.C., *Optimization of fluid front dynamics in porous media using rate control. I. equal mobility fluids*, Physics of Fluids 12 (2000) No.7, pp. 1656-1670.

[259] Tavassoli, Z., Carter, J.N. and King, P.R., *Errors in history matching*, SPE Journal, 9(3):352-361, 2004.

[260] Ten, S.X.D. and He, L., *Advanced model order reduction techniques in vlsi design*, Cambridge University Press, New York, NY, USA, 2007.

[261] Thomas, G.W. and Thurman, D.H., *The mathematical basis of the adaptive implicit method*, paper SPE 10495 presented at the 6th SPE Symposium on Reservoir Simulation of the Society of Pefroleum Engineers of AlME held in New Orleans, LA, January 31 - February 3, 1982.

[262] _____, *Reservoir simulation using an adaptive implicit method*, paper SPE 10120 presented at the 56th Annual Fall Technical Conference and Exhibition of the SPE of AIME, held in Antonio, Texas, October 5-7, 1981.

[263] Tiwary, S.K. and Rutenbar, R.A., *Faster, parametric trajectory-based macromodels via localized linear reductions*, ICCAD (Soha Hassoun, ed.), ACM, 2006, pp. 876–883.

[264] Tromeur-Dervout, D. and Vassilevsky, Y., *Choice of initial guess in iterative solution of series of systems arising in fluid flow simulations*, J. Comput. Phys. **219** (2006), no. 1, 210–227.

[265] _____, *POD acceleration of inexact Newton method in massively parallel reservoir simulation*, Tech. Report CDCSP06-02, CDCSP, 2006.

[266] Truhar, N. and Veseli¸, K., *Bounds on the trace of a solution to the Lyapunov equation with a general stable matrix*, Systems and Control Letters, vol.56, 493-503, 2007.

[267] Tubino, F. and Solari, G., *Double proper orthogonal decomposition for representing and simulating turbulence fields*, Journal of Engineering Mechanics **131** (2005), no. 12, 1302–1312.

[268] Vakili-Ghahani, S.A. and Jansen, J.D., *Control-relevant upscaling*, paper SPE 113647 presented at the 2008 SPE Europec/EAGE Annual Conference and Exhibition held in Rome, Italy, 912 June 2008.

[269] Vakili-Ghahani, S.A., Markovinović, R. and Jansen, J.D., *Upscaling of large reservoir systems by using a control-relevant approach*, ECMOR XI, 8 - 11 September 2008 in Bergen, Norway.

[270] van der Veen, A-J., Deprettere, E. and Swindlehurst, A., *Subspace-based signal analysis using singular value decomposition*, Proceedings of the IEEE, 81(9):1277– 1308, Sept. 1993.

[271] van Doren, J.F.M., *Reduced-order optimal control of water flooding using proper orthogonal decomposition*, MSc thesis, TU Delft, 2004.

[272] van Doren, J.F.M., Markovinović, R. and Jansen, J.D., *Reduced-order optimal control of water flooding using proper orthogonal decomposition*, Computational Geosciences **10** (March 2006), 137–158(22).

[273] van Doren, J.F.M., Van den Hof, P.M.J., Jansen, J.D. and Bosgra, O.H., *Determining identifiable parameterizations for large-scale physical models in reservoir engineering*, Proc. International Federation for Automatic Control (IFAC) World Congress, Seoul, Korea, Portugal, 6-11 July 2008.

[274] ———, *Structural identifiability of grid block and geological parameters in reservoir simulation models*, XI European Conference Mathematical Oil Recovery (ECMOR), 8-11 September 2008, Bergen, Norway.

[275] Van Essen, G.M., Zandvliet, M.J., Bosgra, O.H., Van den Hof, P.M.J. and Jansen, J.D., *Robust waterflooding optimization of multiple geological scenarios*, paper SPE 102913 presented at 2006 SPE Annual Technical Conference and Exhibition, San Antonio, USA, 24-27 September.

[276] Van Overschee, P. and De Moor, B., *Subspace identification for linear systems: theory, implementation, applications*, 1996.

[277] ———, *Subspace algorithms for the stochastic identification problem*, Automatica 29, 649-660, 1993.

[278] ———, *A unifying theorem for three subspace system identification algorithms*, In Proc. 10th IFAC Symp. on Syst. Id., Copenhagen, Denmark, 1994.

[279] Varga, A., *Balancing-free square-root algorithm for computing singular perturbation approximations*, Proc. CDC'91, Brighton, UK, (1) 1991, pp. 1062–1065.

[280] ———, *Enhanced modal approach for model reduction*, Mathematical Modelling of Systems, (1) 1995, pp. 91 – 105.

[281] _____, *On modal techniques for model reduction*, Technical Report TR R136-93, Institute of Robotics and System Dynamics, DLR Oberpfaffenhofen, Wessling, Germany, 1993.

[282] Varga, A. and Parrilo, P., *Fast algorithms for solving h-infinity-norm minimization problems*, 40th IEEE Conference on Decision and Control, 11 2001, pp. 1 – 6.

[283] Vasilyev, D. and White, J., *A more reliable reduction algorithm for behavioral model extraction*, ICCAD, 2005, pp. 813–820.

[284] Vasilyev, D., Rewieński, M. and White, J., *Perturbation analysis of tbr model reduction in application to trajectory-piecewise linear algorithms for MEMS structures*, NSTI-Nanotech 2004, Vol. 2, 2004.

[285] Verhaegen, M., *Analysis of the asymptotic properties of the moesp type of subspace algorithms*, Automatica, 30, 61-74, 1994.

[286] Vermeulen, P.T.M., Heemink, A.W. and te Stroet, C.B.M., *Low-dimensional modelling of numerical groundwater flow*, Hydrol. Process. **18** (2004), 1487–1504.

[287] Villemagne, C.D. and Skelton, R. E., *Model reduction using a projection formulation*, Internat. J. Control, Vol. 46, pp. 2141-2169, 1987.

[288] Wachspress, E.L., *Extended application of alternating direction implicit iteration model problem theory*, **11** (1963), no. 4, 994–1016.

[289] _____, *Iterative solution of the Lyapunov matrix equation*, Appl. Math. Lett. **1** (1988), no. 1, 87–90.

[290] Wang, C., Li, G. and Reynolds, A. C., *Production optimization in closed-loop reservoir management*, SPE 109805. In: 2007 SPE Annual Technical Conference and Exhibition.

[291] Wen X.-H. and Gomez-Hernandez J.J., *Upscaling hydraulic conductivities in heterogeneous media: An overview*, Journal of Hydrology **183** (1996), 9–30.

[292] Willcox, K., *Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition*, Computers and Fluids **35** (2006), no. 2, 208–226.

[293] Willcox, K. and Peraire, J., *Balanced model reduction via the proper orthogonal decomposition*, AIAA Journal **40** (2002).

[294] Xu, D., Yan, S., Zhang, L., Zhang, H-J., Liu, Z. and Shum, H-Y., *Concurrent subspaces analysis*, CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2 (Washington, DC, USA), IEEE Computer Society, 2005, pp. 203–208.

[295] Yamashita, Y., *Image compression by weighted Karhunen-Loève transform*, ICPR '96: Proceedings of the 13th International Conference on Pattern Recognition (Washington, DC, USA), IEEE Computer Society, 1996, p. 636.

[296] Yang, J. and Zhang, D, *Stochastic analysis of saturated-unsaturated flow in heterogeneous media by combining Karhunen-Loève expansion and perturbation method*, J. Hydrology **294** (2004), 18–38.

[297] Yang,J., Zhang, D., Frangi, A.F. and Yang, J-Y., *Two-dimensional pca: A new approach to appearance-based face representation and recognition*, IEEE Trans. Pattern Anal. Mach. Intell. **26** (2004), no. 1, 131–137.

[298] Ye, J., *Generalized low rank approximations of matrices*, Mach. Learn. **61** (2005), no. 1-3, 167–191.

[299] Ye, L., *Generalized low rank approximations of matrices*, In International Conference on Machine Learning, ICML'04., 2004.

[300] Yeten, B., Durlofsky, L.J. and Aziz, K., *Optimization of nonconventional well type, location and trajectory*, SPE journal, pages 200-210, 2003. SPE 86880.

[301] Yu, S., Davies, D. R., and Sherrard, D. W., *The modelling of advanced "intelligent" well – an application*, SPE Annual Technical Conference and Exhibition, Dallas, Texas, 1-4 October (2000).

[302] Zandvliet, M., *Model-based lifecycle optimization of well locations and production settings in petroleum reservoirs*, PhD Dissertation, TU Delft, 2008.

[303] Zandvliet, M.J., van Doren, J.F.M., Bosgra, O.H., Jansen, J.D. and Van den Hof, P.M.J., *Controllability, observability and identifiability in single-phase porous media flow*, Accepted for publication in Computational Geosciences.

[304] Zhang, D. and Lu, Z., *An efficient, high-order perturbation approach for flow in random porous media via Karhunen-Loève and polynomial expansions*, J. Comput. Phys. **194** (2004), no. 2, 773–794.

[305] Zhang, F. and Reynolds, A.C., *Optimization algorithms for automatic history matching of production data*, 8th European Conference on the Mathematics of Oil Recovery - ECMOR VIII, Freiberg, Germany, Sept 3-6 2002.

[306] Zhang, F., Skjervheim, J.A., Reynolds, A.C. and Oliver, D.S., *Automatic history matching in a Bayesian framework: Example applications*, SPE Reservoir Evaluation and Engineering (2005) 8, 214.

[307] Zhou, K., Doyle, J.C. and Glover, K., *Robust and optimal control*, Prentice-Hall, Upper Saddle River, NJ, 1996.

[308] Zhou, Y., *Theoretical and practical aspects of linear and nonlinear model order reduction techniques*, PhD Thesis, MIT, 2008.

[309] _____, *Numerical methods for large scale matrix equations with applications in LTI system model reduction*, PhD Thesis, Rice University, 2002.

Glossary

$H(s)$	LTI SISO transfer function (s=Laplace variable)
J	Objective function
L_c	Controllability energy function
P_c	(here water-oil) capillary pressure $[Pa]$
P_d	entry capillary pressure $[Pa]$
$S_{N,\alpha}$	normalized phase saturation $[-]$
$S_{\alpha r}$	residual saturation of phase α $[-]$
S_α	saturation of phase α $[-]$
Ψ_α	phase potential $[Pa]$
α_{icv}	interval control valve parameter $[-]$
$\mathbf{\Phi}$	POD projection matrix
$\mathcal{C}_n$	Controllability matrix
$\mathcal{H}$	Hankel matrix
$\mathcal{H}_c$	continuous-time Hankel operator
$\mathcal{H}_d$	discrete-time Hankel operator
$\mathcal{K}$	denotes a Krylov subspace
$\mathcal{O}_n$	Observability matrix
$\boldsymbol{\lambda}$	Vector of Lagrange multipliers
$\boldsymbol{\phi}$	POD basis function
λ_α	mobility of phase α $[\frac{1}{Pa \cdot s}]$
λ_i	A pole of an LTI system
μ_α	viscosity of phase α $[Pa \cdot s]$
ω	well-index, but also frequency
ρ_α	density of phase α $[\frac{kg}{m^3}]$
σ_i	A Hankel singular value of an LTI system
$\mathbf{K}$	permeability tensor $[m^2]$
$\mathbf{P}$	Permutation Matrix
$\mathbf{R}$	State snapshots correlation matrix
$\mathbf{W}_c$	Controllability Gramian
$\mathbf{X}$	State snapshots matrix
$\mathbf{u}_\alpha$	phase α velocity vector $[\frac{m}{s}]$
φ	rock porosity $[-]$
$_\alpha$	(as subscript) fluid phase indicator ($\alpha = w, o$)
c_t	total compressibility $[\frac{1}{Pa}]$
f_α	fractional flow of phase α $[-]$

g	gravitational acceleration constant, 9.81 $[\frac{m}{s^2}]$
$k_{r\alpha}$	relative permeability of phase α $[-]$
$k^0_{r\alpha}$	phase α relative permeability end point $[-]$
n_α	empirical coefficient $[-]$ in the Brooks-Corey relation for the relative permeability of phase α
p_α	pressure of phase α $[Pa]$
p_{wf}	(bottom-hole) well flowing pressure $[Pa]$
q_α	well flow-rate of phase α $[\frac{kg}{m^3 s}]$
A(x),E(x)	state-dependent matrices
A,B,E	constant matrices describing LTI dynamics
$\mathbf{A}_r$	reduced-order system matrix
C,D	constant matrices describing LTI output
H(s)	LTI MIMO transfer function matrix
I	Identity matrix
$\mathbf{L}_{qu}$	input location matrix
V(x),W(x),T(x),F(x)	state-dependent two-phase flow matrices
V,W	matrices spanning projection subspaces
f(x,u),g(x,u)	state and input dependent vector fields
x,u,y	(dynamical) system state-,input/control- and output vector, resp.
AIM	Adaptive Implicit Method
BTR	Balanced Truncation
CVOD	Centroidal Voronoi Orthogonal Decomposition
CVT	Centroidal Voronoi Tessellation
EnKF	Ensemble Kalman Filter
EOF	Empirical Orthogonal Functions
EVD	Eigenvalue Decomposition
GLRAM	Generalized Low-Rank Approximation of Matrices
HM	History Matching
HMR	Hybrid Model Reduction
HSVs	Hankel Singular Values
ICCG	Incomplete Cholesky Conjugate Gradient
IMPES	IMplicit-Pressure Explicit-Saturation
KL	Karhunen-Loeve
KS	Krylov Subspace
LR-ADI	Low-Rank Alternating Direction Iteration

LTI	Linear Time-Invariant
LTV	Linear Time-Varying
MIMO	Multiple-Input Multiple-Output
MOR	Model Order Reduction
MTR	Modal Truncation
NPV	Net Present Value [$]
ODE	Ordinary Differential Equation
PCA	Principal Component Analysis
PDE	Partial Differential Equation
PID	Principal Interval Decomposition
POD	Proper Orthogonal Decomposition
SISO	Single-Input Single-Output
SPA	Singular Perturbation Approximation
SVD	Singular Value Decomposition
TFMM	Transfer Function Moment Matching
TPWL	Trajectory-based PieceWise Linear

List of Publications

Journal papers

Heijn, T., Markovinović, R., and Jansen, J.D., *Generation of low-order reservoir models using system-theoretical concepts*, SPE Journal, June 2004, 202-218.

Markovinovic, R. and Jansen, J.D., *Accelerating iterative solution methods using reduced-order models as solution predictors*, Int. J. for Numerical Methods in Engineering (IJNME) **68** (2006), 525–541.

van Doren, J.F.M., Markovinović, R. and Jansen, J.D., *Reduced-order optimal control of water flooding using proper orthogonal decomposition*, Computational Geosciences **10** (March 2006), 137–158(22).

Technical reports

Ibragimov, I., Markovinović, R., Ermolaev, A.J. and Naevdal, G., *Production Optimization of Thin-Oil Rims: Evaluation of a Stochastic Steepest-Ascent Approach*, International Research Institute of Stavanger (IRIS), Technical report (restricted). (Additional information: in a slightly shorter form submitted to the 13th IFAC Symposium on Information Control Problems in Manufacturing (INCOM'09), June 3 - 5, 2009, Moscow, Russia.)

Markovinović, R., *Towards Knowledge-Based Processing System (PRISM)*. A Marie Curie Research Training Network funded by European Commission (2004-2008). Contract Number: MRTN-CT-2004-512233. Final Report (Task 1.2:'Modeling of Integrated Oil and Gas Processes').

Conference proceedings

Markovinović, R., Geurtsen, E.L., and Jansen, J.D., *Subspace identification of low-order reservoir models*, XIV International Conference on Computational Methods in Water Resources, June 23-28, Delft, The Netherlands, p.281-288.

Markovinović, R., Geurtsen, E.L., Heijn, T., and Jansen, J.D., *Generation of low-order reservoir models using POD, empirical gramians and subspace identification*, 8th European Conference on the Mathematics of Oil Recovery - ECMOR VIII, Freiberg, Germany, Sept 3-6 2002.

Markovinović, R., Jansen, J.D. and Rommelse, J.R., *Reduced representations in reservoir simulation - extending POD to include more general optimality conditions*, In: Proc. 9th European conference on the mathematics of oil recovery (ECMOR IX), EAGE, Houten, The Netherlands, 2004, pp. B10–1–B10–6.

Vakili-Ghahani, S.A., Markovinović, R. and Jansen, J.D., *Upscaling of large reservoir systems by using a control-relevant approach*, ECMOR XI, 8 - 11 September 2008 in Bergen, Norway.

van Doren, J.F.M., Markovinović, R. and Jansen, J.D., *Use of POD in control of flow through porous media*, European Conference of Computational Fluid Dynamics (ECCOMAS CFD 2006), September 5-8, 2006, Egmond aan Zee, The Netherlands.

Summary

This thesis is concerned with low-order modeling of heterogeneous reservoir systems for the purpose of efficient simulation and optimization of flooding processes with multiple injection and production (smart) wells. Other important applications of low-order representations could e.g. be in the regularization of inversion procedures for identification of reservoir properties and/or for optimal placement of wells. Typically, and in order to incorporate 'as much physics as possible', one is initially equipped with a large-scale physics-based (or 'white-box') model consisting of $O(10^3 - 10^6)$ equations and parameters representing a (coupled) system of discretized PDEs (Chapter 2) defined on a geometric grid. Low-order approximations of such models are traditionally generated by "grid-coarsening" methods, which range from conventional upscaling to various multiscale techniques.

The methodology undertaken in this research is fundamentally different, in that none of the model-order reduction (MOR) approaches and techniques presented employs any coarse-grid approximation of the fine-grid problem. Instead, the reduced-order models are here based on 'system-theoretic' and dynamically-intrinsic properties of the fine-scale system. In single-phase flow problems that can be modeled as linear time-invariant state-space systems these properties are, e.g., the system's transfer function in the Laplace domain, the eigenstructure of the system matrix, or controllability and observability of the (particular state-space realization of the) system. For multi-phase flow problems, resulting in nonlinear state-space models, due to both the size and the overall complexity of these systems (moving fluid interfaces, coupled flow and transport), intrinsic information needs to be sought in data obtained by simulating the fine-scale model. The contribution of this thesis can be divided into three main themes, summarized below.

Standard 'projection-based' MOR

For slightly compressible single-phase fluid flow, the performance is assessed of *Modal Truncation* (MTR), *Singular Perturbation Approximation* (SPA) (or 'residualization'), *Transfer Function Moment Matching* (TFMM), and *Balanced Truncation* (BTR). For two-phase (waterflood) flow, the so-called *Proper Orthogonal Decomposition* (POD) technique is employed, a data-driven technique that guarantees optimal approximation of the collected data in the mean square error sense. All the methods, except the residualization, are projection-based, meaning that a macro-

model of the original large-scale dynamical system is generated by projecting it onto some low-dimensional subspace. The numerical tests show that these techniques can produce accurate reduced-order models of very low order. In waterflood applications, the tests indicate that the POD approach is quite robust against non-training input signals if the snapshots are generated by exciting the system through manipulation of the injection and/or production profiles consistent with the expected range of operating conditions.

Acceleration of solving the fine-scale problem

In iterative numerical methods for solving nonlinear evolution problems, the initial solution guess is usually taken to be equal to the last time-step solution. The POD-based algorithm developed in this study determines the initial guess by using a regularly updated reduced-order model. The solution of the reduced-order model can be interpreted as a 'shadow' running in parallel with the solution of high-order model. When applied to two-phase (incompressible) flow in an IMPES (IMplicit Pressure Explicit Saturation) formulation, the method seems to result in an improved computational efficiency except in pathological cases with steady-state inputs and near-time invariant (state-dependent) parameters. The method seems to be particularly attractive for problems with time-varying parameters or time-varying source terms.

Acceleration of adjoin-based waterflooding optimization

Modern approaches in reservoir optimization often require performing many reservoir flow simulations. The nested-loops iterative strategy developed in this study (jointly with J.F.M. van Doren) aims at speeding-up adjoint-based waterflooding optimization by employing a truncated POD-basis in the inner loop to calculate optimized injection and production rates. After convergence in this loop, the high-order model is simulated in the outer loop with the improved rates obtained in the inner loop. An adapted POD basis is used in the next inner loop to calculate new optimized injection and production rates, etc. In the numerical example, the NPV obtained by the full-order optimal control algorithm was approached closely by the NPV obtained by the reduced-order algorithm within a shorter simulation time.

Samenvatting

Dit proefschrift houdt zich bezig met het bepalen van lage-orde modellen van heterogene ondergrondse reservoirsystemen met als doel een efficiënte simulatie en optimalisatie van 'waterflooding' processen met meerdere injectie- en productie ('inteligente') putten. Andere denkbare toepassingen van lage-orde modellen zijn bijvoorbeeld de regularisatie van inversieprocedures voor identificatie van reservoirkenmerken (parameters and toestandsvariabelen) en/of voor optimale positionering van putten. Men heeft initiëel typisch beschikking over een grootschalig 'physics-based' (of 'white-box') model, welk meestal een systeem van ruimtelijk gediscretiseerde (gekoppelde) partiële differentiële vergelijkingen (PDEs) is, bestaand uit $\mathcal{O}(10^3 - 10^6)$ vergelijkingen en parameters gedefi-niëerd op een fijn geometrisch grid. Lage-orde benaderingen van zulke modellen worden traditioneel gegenereerd met "grid-coarsening' technieken, die lopen van de conventionele opschaling tot meer recente multiscale technieken.

Dit proefschrift benadert het problem volkomen anders, in de zin dat geen van de gepresenteerde methoden gebruik maakt van enige grof-grid representatie van het fijne-grid probleem. In plaats daarvan worden de lage-orde modellen hier bepaald op basis van 'systeem-theoretische' en dynamisch-intrinsieke eigenschappen van het fijne-grid probleem. Voor éénfase stroming gemodelleerd als een lineair tijdsinvariant toestandssysteem zijn deze eigenschappen o.a. de systemoverdrachtsfunctie in het Laplace domain, de eigenstruktuur van de systeemmatrix, of de regelbaarheid en observeerbaarheid van (de gegeven toestandsrealisatie van) het systeem. Voor meerfasestromingsproblemen maken de nietlineariteit, de omvang, en de algehele komplexiteit van deze systemen (bewegende vloeistofgrensvlakken, gekoppelde stroming en transport) het noodzakelijk intrinsieke informatie te zoeken in fijn-grid simulatiedata. De belangrijkste bijdragen en bevindingen van dit proefschrift kunnen in drie hoodthema's ingedeeld worden, opgesomd hieronder.

Standaard 'projectie-gebaseerde' model-orde reductie (MOR)

Voor zwak samendrukbare éénfase stroming worden de kenmerken en de prestaties van de volgende methoden onder de loep genomen: *Modal Truncation* (MTR), *Singular Perturbation Approximation* (SPA) (of 'residualisatie'), *Transfer Function Moment Matching* (TFMM), en *Balanced Truncation* (BTR). Voor tweefasen (water/oil)

stroming kan de zogeheten *Proper Orthogonal Decomposition* (POD) worden gebruikt, een op data gebaseerde techniek die de kleinste gemiddelde kwadratische fout in de approximatie van de data garandeert. Al deze methoden, op de residualizatie na, zijn projectie-gebaseerd, hetgeen wil zeggen dat er een macromodel van het originele grootschalige dynamische systeem mee wordt gegenereerd door het laatste op laagdimensionele deelruimtes te projecteren. De numerieke voorbeelden tonen aan dat met deze methoden nauwkeurige lage-orde modellen gegenereerd kunnen worden. In waterflooding applicaties blijkt de POD vrij robuust met betrekking tot voor het genereren van de data ('snapshots') niet-gebruikte ingangssignaalvormen (injectie- en/of productieprofielen) indien de snapshots gegenereerd worden met ingangsprofielen die consistent zijn met de verwachte span van exploatatiecondities.

Versnelling van het oplossen van het fijne-schaal probleem

In iteratieve numerieke methoden voor het oplossen van niet-lineaire tijdsafhankelijke dynamische problemen wordt voor de initiele schatting voor de oplossing meestal de voorgaande oplossing genomen. Het op POD-gebaseerde algorithme beschreven in dit proefschrift bepaalt de initiële schatting met behulp van een regelmatig bijgewerkt lage-orde model. De oplossing van het gereduceerde-orde model kan worden opgevat als een 'schaduw' die parallel loopt met de oplossing van het hoge-orde model. Toegepast op tweefasen (onsamendrukbare) vloeistofstroming in een IMPES (IMplicit Pressure Explicit Saturation) formulering schijnt de methode in een verbeterde rekenefficiëntie te resulteren met als uitzondering pathologische gevallen met steady-state inputsignalen en nagenoeg tijdsinvariante (toestandsafhankelijke) parameters. Meer winst lijkt haalbaar voor problemen met tijdsvarirende parameters of tijdsvariërende brontermen.

Versnelling van adjoint-gebaseerde waterflood optimization

Moderne methoden voor reservoir optimalisatie vereisen vaak een aanzienlijk aantal simulaties. De geneste lus iteratieve strategie uit dit proefschrift (ontwikkeld in samenwerking met J.F.M. van Doren) versnelt adjoint-gebaseerde waterflooding optimalisatie met gebruikmaking van een afgekapte POD basis in de binnenlus om verbeterde injectie- en productiedebieten te berekenen. Na de convergentie in deze lus wordt met deze verbeterde debieten het hoge-orde model gesimuleerd in de buitenlus. Voor het berekenen van nieuwe verbeterde injectie en productiedebieten wordt in de volgende binnenlus een aangepaste POD-basis gebruikt, etcetera. In numerieke voorbeelden ligt de op deze manier geoptimaliseerde NPV dicht bij de NPV verkregen door het volle-orde 'optimale controle' algoritme, terwijl de totale simulatietijd in de geneste lus methode verkort wordt.

About the author

Renato Markovinović was born in Živinice, Bosnia and Herzegovina (BiH, at that time SFR Yugoslavia) on the 8th of February 1970. He graduated from the Secondary Electrotechnical School in 1988 (Živinice). Prior to continuing his education at the Faculty of Electrical Engineering "ETF" in Banjaluka (BiH), he served one year in the Yugoslavian Army (JNA). His education at ETF was abruptly broken by the outbreak of the war in BiH in 1992. After few years of 'gaining real-life experience' in the war, as a refugee in Holland he made a 'fresh start' in his education at the Faculty of Electrical Engineering at TU Delft (The Netherlands) in Sep 1996, where he obtained his MSc degree in Feb 2001 (Cum Laude). After working for few months as control engineer, in August 2001 he started a Ph.D. at the Faculty of Civil Engineering and Geotechnology in Delft under the supervision of Prof.Dr.Ir. Jan Dirk Jansen. The most interesting results from his Ph.D. research are described in this thesis. In the period January 2006 - June 2007, he was employed at Schlumberger AbTC (UK) as a Marie Curie experienced researcher under an EU Research Training Network (RTN) Program. In October 2008 he started in Bergen (Norway) as a PostDoc at International Research Center of Stavanger (IRIS). He's a lucky husband and a proud father of three beautiful children.