Application of the Reduced Basis Method in Geophysical Simulations:

Concepts, Implementation, Advantages, and Limitations

Von der Fakultät für Georessourcen und Materialtechnik der
Rheinisch -Westfälischen Technischen Hochschule Aachen

zur Erlangung des akademischen Grades einer

Doktorin der Naturwissenschaften

genehmigte Dissertation

vorgelegt von

M. Sc. Denise Degen

aus Köln

Berichter: Prof. Florian Wellmann, Ph. D.
Prof. Karen Veroy, Ph. D.
Prof. Klaus Regenauer-Lieb, Ph. D.

Tag der mündlichen Prüfung: 10.07.2020

Diese Dissertation ist auf den Internetseiten der Universitätsbibliothek online verfügbar

RWTH Aachen University

Doctoral Thesis

Application of the Reduced Basis Method in Geophysical Simulations: Concepts, Implementation, Advantages, and Limitations

---

Author:
Denise Degen, M.Sc.
(from Cologne)

Supervisor:
Prof. Florian Wellmann, Ph.D.,
Prof. Karen Veroy-Grepl, Ph.D.

A thesis submitted in fulfillment of the requirements for the degree of Dr.rer.nat

in the

Faculty of Georesources and Materials Engineering

500000
Fakultät für Georessourcen und Materialtechnik
S. Frenzel-Gumlich
Intzestr. 1
52056 Aachen

Fax: 0241-80-92370
E-Mail: promotionen@fb5.rwth-aachen.de

Eidesstattliche Erklärung

1.)

Hiermit versichere ich eidesstattlich, dass ich die Dissertation selbstständig verfasst und alle in Anspruch genommenen Hilfen in der Dissertation angegeben habe. ☒

Unterschrift

Ort, Datum

2.) Sofern die Dissertation in einer Einrichtung außerhalb der RWTH entstanden ist:

Hiermit erkläre ich, dass mit Veröffentlichung der Dissertation keine bestehenden Betriebsgeheimnisse verletzt werden. ☐

Unterschrift

Ort, Datum

RWTH AACHEN UNIVERSITY

Kurzzusammenfassung

Fakultät für Georessourcen und Materialwissenschaften
Computational Geoscience and Reservoir Engineering

Dr.rer.nat

Anwendung der Reduzierten Basismethode in Geophysikalischen Simulationen: Konzepte, Implementierung, Vorteile, und Einschränkungen

von Denise Degen

Erneuerbare und nachhaltige Energieformen gewinnen immer mehr an Bedeutung und für deren Erhalt werden numerische Simulationen benötigt. Aufgrund der hohen Unsicherheiten von geophysikalischen Simulationen stellt dies eine große Herausforderung dar. Deshalb liegt der Fokus der Arbeit auf der Verbesserung numerischer Simulationen. Dadurch wird die Anwendung inverser Prozesse, wie Markov Chain Monte Carlo, für komplexe sedimentäre Becken ermöglicht, die andernfalls zu kostspielig sind.

Durch die Dissertation wird verdeutlicht, dass die reduzierte Basismethode ein wertvolles Werkzeug besonders für geowissenschaftliche Parameterstudien ist. Einfache Testbeispiele, wie ein Dreischicht und ein Störungsmodell, werden verwendet, um die Möglichkeiten der Methoden besser zu illustrieren. Anschließend werden die gleichen Methoden für existierende Sedimentbecken, zum Beispiel den Oberrheingraben und das Brandenburg Modell, genutzt. Die Simulationszeit beider Modelle ist – durch die Verwendung der reduzierten Basismethode – vier bis sechs Größenordnungen kleiner. Wohingegen die Approximierungen genauer als die Messungen sind. Dies führt dazu, dass sowohl die reduzierten als auch die vollen Modelle gleichwertig für Modellvorhersagen und Parameterstudien sind.

Die gewonnene Simulationszeit kann für die Durchführung von sowohl deterministischen als auch stochastischen inversen Prozessen, wie globalen Sensitivitätsstudien, Modellkalibrierungen und Markov Chain Monte Carlo, genutzt werden. Die Arbeit weist nach, dass die Rechenzeiten für diese Analysen von zehn bis hunderten Knotenjahre auf wenige Knotenstunden reduziert werden.

Die reduzierte Basismethode hat den Vorteil, dass sie im Gegensatz zu vielen anderen Ersatzmodellen nicht nur die gesuchten Größen sondern die gesamte Variable zurückgibt. Dadurch wird die Generierung von 2D und 3D Unsicherheitskarten, welche uns interessante Erkenntnisse über den Einfluss der Grenzbedingungen auf die gesamte Unsicherheitsverteilung liefern, ermöglicht. Im Gegensatz zu vielen anderen Machine-Learning Ansätzen ist die reduzierte Basismethode physikbasiert. Das erlaubt nicht nur die Einbeziehung unseres physikalischen Wissens sondern auch die Kompensierung für fehlende Daten.

Für die hier präsentierten Probleme können wir die reduzierte Basismethode mit Fehlerschranken einsetzen. Diese Fehlerschranken erlauben eine effiziente Basiskonstruktion und liefern eine Garantie über die Approximierungsqualität.

Die hier vorliegende Arbeit konzentriert sich auf geothermische konduktive Wärmeleitfähigkeitsprozesse. Es werden zusätzlich die Vorteile der reduzierten Basismethode für Grundwasserprobleme und elektrische Widerstandstomographie vorgestellt und erläutert. Im Falle der transienten Grundwasseranwendungen zeigt sich, dass der

Rechenzeitgewinn einige Größenordnungen höher ist als der stationärer Anwendungen.

Neben den Parameterstudien zeigt die Arbeit, unter welchen Bedingungen die reduzierte Basismethode für Datenassimilierung in Grundwasserstudien geeignet ist.

RWTH AACHEN UNIVERSITY

Abstract

Faculty of Georesources and Materials Engineering
Computational Geoscience and Reservoir Engineering

Dr.rer.nat

Application of the Reduced Basis Method in Geophysical Simulations: Concepts, Implementation, Advantages, and Limitations

by Denise Degen

Renewable and sustainable energy sources are becoming more and more important. To retrieve these earth's resources, we require reliable simulations. This is, however, a major challenge because of huge uncertainties related to geophysical simulations. Therefore, in this thesis, we focus on improving numerical simulations, to enable the usage of computationally demanding inverse processes, such as Markov Chain Monte Carlo, for complex sedimentary basins that are prohibitive with state-of-the-art finite element solvers.

We demonstrate that the reduced basis method is a powerful tool, especially for parameter estimation studies, in the field of Geosciences. We use simple benchmark examples, of a three-layer and a fault model, to better illustrate the employed methods and then use the same methods for the real-case basin-scale models of the Upper Rhine Graben and Brandenburg. For both models, we obtain speed-ups of four to six orders of magnitude, while maintaining an approximation accuracy higher than the measurement accuracy. Hence, we generate approximations that can be equally well used for model predictions and parameter studies as the original full model.

We illustrate how the obtained speed-up can be used to perform both deterministic and probabilistic inverse processes as, for instance, global sensitivity studies, model calibrations, and Markov Chain Monte Carlo studies. For these analyses, we are able to reduce the compute time from ten to hundreds of core years to a couple of core hours.

The reduced basis method has the advantage that it provides the full state variable, and not only the quantities of interest, such as many other surrogate models. This allows the generation of 2D and 3D uncertainty quantification maps, providing us with interesting insights into the influence of the kind of boundary conditions on the overall spatial distribution of the uncertainties. Opposed to many machine learning approaches, we use a physics-based approach. This allows to incorporate our physical knowledge and to compensate for problems as data sparsity.

For the here presented problems, we are able to use the certified reduced basis method. Hence, we obtain error bounds. These error bounds give us a guarantee of our approximation quality and allow for an effective basis construction.

We focus in this thesis on geothermal conductive heat transfer problems. However, we also present and discuss the benefits of the reduced basis method for groundwater flow, and electrical resistivity tomography problems. For the transient simulations of the groundwater flow problems, we are able to show that the gained speed-ups are several orders of magnitude higher than those for the steady-state problems.

In addition to the parameter estimation studies, we furthermore demonstrate under which conditions the reduced basis method can be used for data assimilation in groundwater studies.

Acknowledgements

First of all, I would like to express my deepest gratitude to my thesis advisors, Prof. Florian Wellmann, and Prof. Karen Veroy-Grepl, for their support, and encouragement during my doctoral studies. I am particularly thankful for the uncountable and fruitful discussions. I would also like to thank them for the opportunity to participate in many scientific conferences and workshops both in the field of geosciences and model order reduction.

I would further like to thank Prof. Magdalena Scheck-Wenderoth, from the German Research Centre for Geosciences for her fruitful collaboration. She greatly supported me by providing the real-case basin-scale models of the Upper Rhine Graben and Brandenburg, which enabled us to show the benefits of the reduced basis method for real-case applications. Regarding the Upper Rhine Graben, special thanks are directed to Jessica Freymark, for sharing her intensive knowledge of the Upper Rhine Graben and for the generation of the geological model itself. I would also specially thank Dr. Vera Noack, Federal Institute for Geosciences and Natural Resources, for the generation of the various Brandenburg models.

Furthermore, I would like to express my thanks to Prof. Martin Grepl, RWTH Aachen University, for his support and helpful discussion on the reduced basis method for parabolic problems, especially in the context of data assimilation.

Additionally, I would like to thank Dr. Mauro Cacace and Dr. Antoine Jacquey (German Research Centre for Geosciences) for their helpful discussion about the MOOSE Framework and its aspects related to high-performance computing.

I am most grateful for the support from my current and former colleagues at both the AICES and the CGRE institute. In particular, I like to thank Zoi Tokoutsi and Nicole Nellesen for their helpful and motivating discussion of the reduced basis method. I like to thank Nikhil Vaidya and Rahul Dhopeshwar for their discussion of various aspects of the implementation of the reduced basis method and the MOOSE Framework. Additionally, I would like to express my gratitude to Rhea von Bülow for here support regarding the electrical resistivity tomography. I would also like to thank Dr. Simon Virgo and Miguel de la Varga for their insightful discussions about geological modeling, in particular, the modeling software GemPy.

I also thankfully acknowledge the finical support by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) under the project GSC 111 and IRTG-2379. Furthermore, I gratefully acknowledge the computing time granted through the JARA-HPC on the supercomputer JURECA at Forschungszentrum Jülich.

Contents

Kurzzusammenfassung **v**

Abstract **vii**

Acknowledgements **ix**

1 Introduction **1**
1.1 Motivation . . . 1
1.2 Problems of Geoscientific Simulations . . . 2
1.3 Model Order Reduction – The Reduced Basis Method . . . 3
1.4 Scientific Aims . . . 4
1.5 Structure of the Thesis . . . 7
1.6 Publications . . . 8

2 Materials and Methods **11**
2.1 Certified Reduced Basis Methods – A General Introduction . . . 11
2.1.1 Elliptic - Steady State . . . 12
2.1.2 Parabolic - Transient . . . 16
2.1.3 RB for Applications in Geosciences . . . 18
Geothermal Conduction . . . 18
Darcy Flow Problem . . . 21
2.2 Software . . . 22
2.2.1 MOOSE Framework . . . 22
2.2.2 DwarfElephant . . . 23
2.2.3 Software Coupling . . . 24
Loose Coupling . . . 24
Direct Coupling . . . 24

3 Certified Reduced Basis Method for Geothermal Applications **27**
3.1 Geological Models . . . 27
3.1.1 Base and Fault Model . . . 27
3.1.2 Perth Basin . . . 32
3.2 Results . . . 34
3.2.1 Base and Fault Model . . . 34
3.2.2 Perth Basin . . . 37
3.3 Discussion . . . 40
3.3.1 Base and Fault Model . . . 40
3.3.2 Perth Basin . . . 44

4 Global Sensitivity Analysis for Model Calibrations **47**
4.1 Introduction . . . 47
4.2 Materials and Methods . . . 49
4.2.1 Sensitivity Analysis . . . 49

Local Sensitivity Analysis . . . 49
Global Sensitivity Analysis . . . 50
4.2.2 Forward Simulation . . . 52
4.2.3 Model Calibration . . . 53
4.3 Case Study of the Upper Rhine Graben . . . 53
4.3.1 Upper Rhine Graben . . . 53
4.3.2 Data and Weighting . . . 54
4.3.3 Model Calibration without Sensitivity Analysis . . . 55
4.3.4 Model Calibration with Local Sensitivity Analysis . . . 58
Sensitivity Analysis . . . 58
Model Calibration . . . 59
4.3.5 Model Calibration with Global Sensitivity Analysis . . . 62
Sensitivity Analysis . . . 63
Model Calibration . . . 63
4.3.6 Computational Cost . . . 71
4.4 Discussion . . . 71
4.4.1 Sensitivity Analysis . . . 71
4.4.2 Model Calibration . . . 72
4.4.3 Computational Cost . . . 76
4.4.4 Reliability . . . 76

5 Influence of the Model Dimension 77
5.1 Introduction . . . 77
5.2 Model Calibration . . . 78
5.2.1 Local Gradient-Based Methods . . . 78
5.2.2 Local Derivative-Free Methods . . . 79
5.2.3 Global Derivative-Free Methods . . . 81
5.3 Brandenburg Models . . . 83
5.4 Results . . . 87
5.4.1 Influence of the Model Dimension on the Model Predictions . . . 87
Sensitivity Analysis . . . 87
Model Calibration . . . 88
Approximation Quality and Computational Cost . . . 97
5.4.2 Automated Model Calibration Comparison . . . 97
5.5 Discussion . . . 102
5.5.1 Influence of the Model Dimension on the Model Predictions . . . 102
Sensitivity Analysis . . . 102
Model Calibration . . . 104
Computational Cost . . . 105
5.5.2 Automated Model Calibration Comparison . . . 105

6 Uncertainty Quantification For Basin-Scale Geothermal Models 107
6.1 Introduction . . . 107
6.2 Uncertainty Quantification . . . 108
6.3 Geological Models . . . 108
6.4 Results . . . 109
6.4.1 Brandenburg Model . . . 109
6.4.2 Thermal Conductivities . . . 109
Uncertainty Quantification Maps . . . 113
Computational Cost . . . 114
6.4.3 Upper Rhine Graben . . . 118

Thermal Conductivity . . . 118
Uncertainty Quantification Maps . . . 121
Computational Cost . . . 123
6.5 Discussion . . . 132
6.5.1 Brandenburg Model . . . 132
Thermal Conductivities . . . 132
Uncertainty Quantification Maps . . . 132
Reduced Order Model . . . 134
6.5.2 Upper Rhine Graben . . . 134
Thermal Conductivity . . . 134
Uncertainty Quantification Maps . . . 135
6.5.3 Computational Cost and Approximation Accuracy . . . 138

7 Certified Reduced Basis Method for Groundwater Applications 139
7.1 Darcy's Law . . . 139
7.2 Grid Search . . . 140
7.3 Fault Model . . . 140
7.4 Results . . . 142
7.4.1 Darcy Flow Problem . . . 143
7.4.2 Parameter-Dependent Initial Conditions . . . 149
7.4.3 Softwate Coupling . . . 156
7.5 Discussion . . . 157
7.5.1 Darcy Flow Problem . . . 157
7.5.2 Parameter-Dependent Initial Conditions . . . 158
7.5.3 Software Coupling . . . 159

8 Global Sensitivity Analysis for the Investigation of Fault Zone Behavior 161
8.1 Malm Fault Zone . . . 161
8.2 Results . . . 162
8.3 Discussion . . . 164

9 Reduced Basis Method for Sequential Data Assimilation 167
9.1 Introduction . . . 167
9.2 Data Assimilation – Ensemble Kalman Filter . . . 168
9.3 Box Model . . . 169
9.4 Results . . . 169
9.4.1 Forward Model . . . 172
9.4.2 Ensemble Kalman Filter – Finite Element Version . . . 174
Influence of the Observations Points . . . 174
Influence of the Boundary Conditions . . . 179
9.4.3 Ensemble Kalman Filter – Reduced Basis Version . . . 182
Modification of the Ensemble Kalman Filter . . . 182
Reduced Basis Method . . . 183
9.5 Discussion . . . 190
9.5.1 Forward Model . . . 190
9.5.2 Ensemble Kalman Filter – Finite Element Version . . . 190
9.5.3 Ensemble Kalman Filter – Reduced Basis Version . . . 191

10 Extension to Other Geophysical Applications **193**
10.1 Hydrothermal . . . 193
10.1.1 Forward Problem . . . 193
10.1.2 Entropy Production . . . 194
10.1.3 Results . . . 195
10.1.4 Discussion . . . 201
10.2 Electrical Resistivity Tomography . . . 201
10.2.1 Governing Equations . . . 202
10.2.2 Trench Model . . . 203
10.2.3 Results . . . 205
10.2.4 Discussion . . . 207

11 Outlook **209**
11.1 Hydrothermal Applications . . . 210
11.2 Data Assimilation . . . 212
11.3 Electrical Resistivity Tomography and Maxwell . . . 213

12 Overall Discussion **215**

A DwarfElephant **xxi**
A.1 Introduction . . . xxi
A.2 The Reduced Basis Method . . . xxi
A.3 Problem Description . . . xxii
A.4 Prerequisites and First Steps . . . xxv
A.5 Implementation . . . xxv
A.5.1 Mesh, Variables, and Global Params . . . xxv
A.5.2 Kernels . . . xxv
A.5.3 BCs . . . xxvi
A.5.4 Materials . . . xxvi
A.5.5 Problem, Executioner and UserObjects . . . xxvi
A.5.6 PostProcessors . . . xxvii
A.5.7 Outputs . . . xxvii
A.6 A Geothermal Conduction Problem . . . xxvii
A.6.1 Mesh . . . xxvii
A.6.2 Variables . . . xxvii
A.6.3 GlobalParams . . . xxviii
A.6.4 Kernels . . . xxviii
A.6.5 BCs . . . xxviii
A.6.6 Problem and Executioner . . . xxviii
A.6.7 UserObjects . . . xxix
A.6.8 Outputs . . . xxx
A.6.9 Run . . . xxx

B Loose Coupling to Dakota - A Tutorial **xxxi**
B.1 Prepare the Dakota inputfile . . . xxxi
B.1.1 Example File . . . xxxi
B.2 Prepare the Driver Script . . . xxxii
B.2.1 Pre-Processig . . . xxxii
B.2.2 Simulation . . . xxxii
B.2.3 Post-Processing . . . xxxii
B.2.4 Exampe File . . . xxxii

B.3 Prepare the MOOSE template xxxiii
B.3.1 Example File xxxiii
B.4 Execute the Analysis xxxiv

C Direct Coupling - A Tutorial **xxxv**
C.1 Python Libraries xxxv
C.1.1 Example File xxxv
C.2 Dakota Coupling xxxvi
C.2.1 Example File xxxvi

D Electrode Positions **xxxix**

List of Abbreviations

BC	**B**oundary **C**ondition
BRB	**BR**anden**B**urg
ENKF	**EN**semble **K**alman **F**ilter
ERT	**E**lectrical **R**esistivity **T**omography
FE	**F**inite **E**lement
HPC	**H**igh-**P**erformance **C**omputing
HSV	**H**ankel **S**ingular **V**alues
MCMC	**M**arkov **C**hain **M**onte **C**arlo
MOOSE	**M**ultiphysics **O**bject **O**rientated **S**oftware **E**nvironment
PDE	**P**artial **D**ifferential **E**quation
POD	**P**roper **O**rthogonal **D**ecomposition
RB	**R**educed **B**asis
RWTH	**R**heinisch-**W**estfälische **T**echnische **H**ochschule
URG	**U**pper **R**hine **G**raben
TRF	**T**rust **R**egion **R**eflective

Nomenclature

Symbol	Name	Unit
a	distance between the potential electrodes	m
$a(\cdot)$	bilinear form	-
A	cross-sectional area	m^2
B	Boussinesq number	-
c	specific heat capacity	$J\ K^{-1}$
E	electrical field	$V\ m^{-1}$
$\hat{e}(\cdot)$	Riesz representant	-
$f(\cdot)$	linear form, cost function	-
g	gravity	$m\ s^{-2}$
$g(t)$	control	-
GW	ground water level	Pa
h	hydraulic head	m
H	measurement matrix	-
I	current	A
$\hat{i}, \hat{j}, \hat{k}$	unit vector	-
j	current density	$A\ m^{-2}$
k	permeability	m^2
K_e	Kalman gain	-
l	length	m
L	distance between the current electrodes	m
$l(\cdot)$	linear functional	-
$m(\cdot)$	bilinear form	-
n	search direction	-
$N_{\dot{S}}$	entropy generation number	-
p	pressure	Pa
p_0	atmospheric pressure	Pa
P	probability	-
P_e	ensemble error matrix	-
pr	production rate	$m^3\ s^{-1}$
q	Darcy velocity/flux	$m\ s^{-1}$
Q	sink/source term	$kg\ s^{-1}$
r	distance	m
$r(\cdot)$	residual	-
R	resistance	Ω
Ra	Rayleigh number	-
R_e	ensemble covariance matrix	-

Symbol	Name	Unit
s	step size, scaling parameter	-
$s(\cdot)$	output of interest	-
S	radiogenic heat production, entropy production	W m^{-3}, W m^{-1} K^{-1}
S_s	specific storage	Pa^{-1}
t	time	s
T	temperature	K
T_0	reference temperature	K
u	unknown variable	-
U	potential	V
$u(\cdot)$	solution	-
v	test function	-
V	potential difference	V
w	trial function	-
x, y, z	spatial coordinates	m
y	data	-
α	thermal expansion coefficient	K^{-1}
$\alpha(\mu)$	coercivity constant	-
$\gamma(\mu)$	continuity constant	-
Δ	error bound, difference	-
ϵ	measurement errors	-
η	effectivity	-
$\theta(\mu)$	parameter-dependent part	-
κ	thermal diffusivity	m^2 s^{-1}
λ	thermal conductivity	W m^{-1} K^{-1}
μ	parameter	-
$\overline{\mu}$	fixed parameter	-
μ_f	dynamic fluid viscosity	Pa s m^{-2}
ρ	density, electrical resistivity	kg m^{-3}, Ω m
ρ_0	reference density	kg m^{-3}
σ	electrical conductivity	Ω^{-1} m^{-1}
$\sigma(\mu)$	coercivity constants	-
ϕ	porosity	-
Φ	viscous dissipation function	-
Ψ	ensemble	-

Symbol	Name	Unit
$\mathcal{D}$	parameter domain	-
X	function space	-
Ω	spatial domain	-
Γ	boundary	-
$\overline{\{\cdot\}}$	mean	-
*	dimensionless	-
^{a}	analysis step	-
$_a$	bilinear form, apparent	-
$_{A,B}$	current electrode position	-
$_{\text{bottom}}$	bottom of the model domain	-
$_{C,D}$	potential electrode position	-
$_{\text{eff}}$	effective	-
$_{\text{en}}$	energy	-
^{f}	forecast step	-
$_f$	linear form, fluid	-
^{k}	time step	-
$_l$	linear functional	-
$_{\text{LB}}$	lower bound	-
$_m$	bilinear form, medium	-
$_{\text{max}}$	maximum	-
$_N$	dimension of the reduced space	-
$_P$	parameter dimension	-
$_{\text{ref}}$	reference	-
$_{\text{rel}}$	output of interest	-
^{s}	output	-
$_s$	solid, source position	-
^{T}	transposed	-
$_{\text{therm}}$	irreversible heat transfer	-
$_{\text{top}}$	top of the model domain	-
$_{\text{total}}$	total	-
$_{\text{train}}$	training	-
$_{\text{visc}}$	fluid flow friction	-

Chapter 1

Introduction

1.1 Motivation

Clean water and clean energy are two of the seventeen sustainable development goals of the UN (United Nations, 2012). To achieve these sustainable development goals, we require predictions of the processes taking place in the Earth's subsurface. However, we do not have direct access to the entire subsurface of the earth. To obtain predictions, nonetheless, we need to perform numerical simulations based on physical models. But, why do we need numerical simulations? Why do we not use analytic simulations, or laboratory measurements instead? Analytic simulations are incredibly efficient in the areas where they are applicable. Unfortunately, these areas are limited since simulations considering, for instance, complex geometries have no analytic solutions. Therefore, we need to solve these problems numerically. Also, laboratory measurements have a few disadvantages. They face the challenge to represent the correct in-situ conditions and provide only pointwise information, especially for applications with deep regions of interest.

To obtain model predictions, the field of Geosciences aims at gaining an accurate understanding of the spatial distribution of the Earth's subsurface properties, and the involved physical processes. These processes are, for instance, fluid and heat transport, chemical species transport, and mechanical processes. To achieve this, we require inverse processes since in most cases, we want to recover the rock properties from, for instance, temperature or pressure measurements.

In order to account for various sources of uncertainty and to analyze, for example, the sensitivity of the model parameters, we need methods such as the Sobol sensitivity analysis, or the Markov Chain Monte Carlo method. The need for these analyses is identified in Eker et al. (2018), and Saltelli (2019). Unfortunately, they are often poorly executed (Saltelli, 2019). One of the reasons is that these methods have in common that they require many forward simulations. This makes them prohibitive for basin-scale analyses using standard numerical methods, such as the finite element (FE) method. For an example, we refer to Fig. 1.1.

If we would perform an uncertainty quantification for a basin-scale geological model with nearly four million degrees of freedom, assuming the example provided in Fig. 1.1, we would require 165 core-years, which is infeasible. So, we need to construct a surrogate model. One naive surrogate model would involve a simplification of the model geometry and the parameterization. This leads to an enormous reduction in compute time, but unfortunately, we are no longer able to represent our original problem. For this reason, we use the reduced basis (RB) method as a surrogate model. The RB method constructs a basis, which is a linear combination of finite element solutions. This reduced basis can be used to efficiently execute inverse processes while retaining a high accuracy.

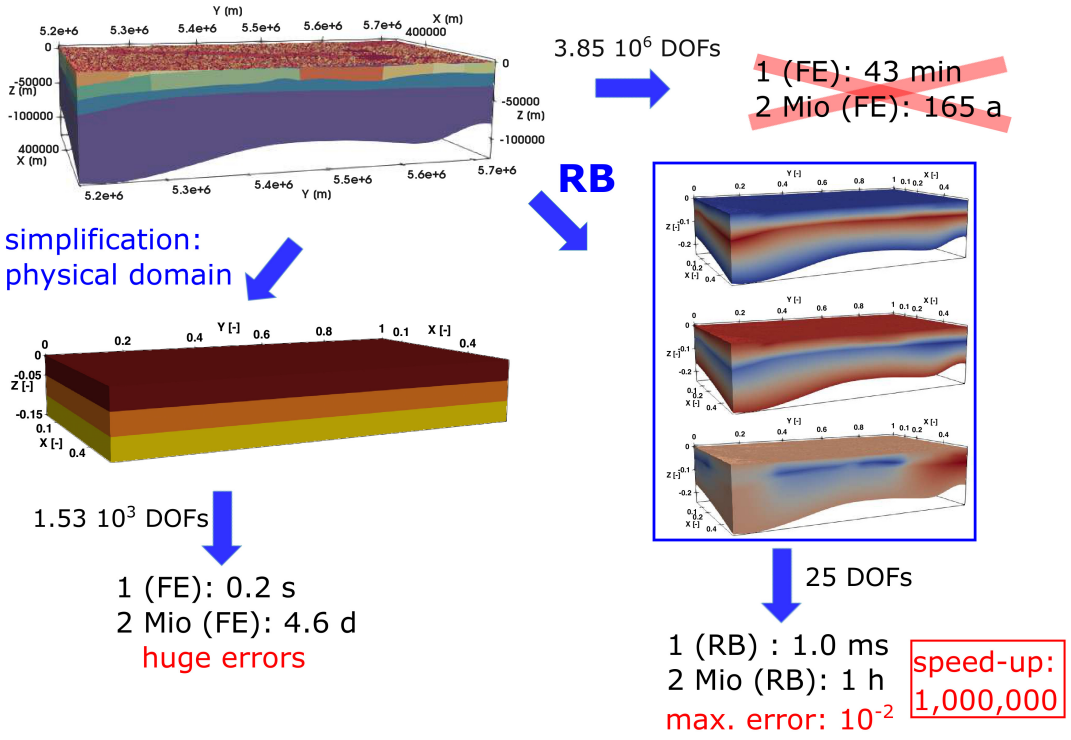


FIGURE 1.1: Illustrating the benefits of the RB method for geophysical inverse problems using the case study of the Upper Rhine Graben.

1.2 Problems of Geoscientific Simulations

Meaningful predictions present us with two main challenges. First, we have to analyze complex coupled processes over a large spatial and temporal domain. Additionally, the highly heterogeneous nature of the subsurface typically results in a high dimensional parameter space. This leads to the so-called "curse of dimensionality". Second, retrieving information from the subsurface is a non-trivial process due to, for instance, limited access to the areas of interest. Consequently, the measurements are associated with high uncertainties (Floris et al., 2001; Wellmann and Caumon, 2018). Accounting for these uncertainties requires numerous simulations, which are often prohibitively expensive when using standard approaches, such as the finite element method.

To overcome the "curse of dimensionality", many geoscientific applications focus solely on the area of interest. Therefore, they use a spatial and temporal discretization that is coarse enough to permit computation in a reasonable time. However, such approaches neglect possibly significant influences from regions outside the area of interest. Furthermore, attempts to compensate for such effects through special boundary conditions often result in insufficient descriptions of the actual problem. Similar problems occur for coarse spatial and temporal discretization because one needs to have methods that correctly upscale the properties from smaller to larger scales. Most of the methods used in Geosciences for reducing the dimensionality reduce the physical spatial and temporal domain by taking less data. In this thesis, we propose an alternative approach in which we systematically reduce the dimensionality of the mathematical space.

1.3 Model Order Reduction – The Reduced Basis Method

Model order reduction (MOR), has a long history in many different application fields. It aims at reducing the complexity of numerical models while preserving the input-output relationship of these models. Hence, it produces surrogate models that have a significantly lower dimension than the original problem but still capture the relevant physical processes. This leads to a much more efficient evaluation of the model, making MOR models suitable surrogate models for outer loop processes, such as probabilistic model calibrations.

There are many different model order reduction techniques available. We focus in this thesis on a projection-based MOR method (Benner et al., 2015), namely the reduced basis (RB) method. Other projection-based model order reduction techniques are the Proper Orthogonal Decomposition (POD) (Benner et al., 2015; Hesthaven et al., 2016; Volkwein, 2013) and balanced truncation (Antoulas, 2005).

POD is probably the most widely used MOR method and, depending on the application field, is also known as Karhunen-Loève expansion/decomposition, principal component analysis, and singular value decomposition. In POD, we first calculate the snapshots and construct the reduced basis by performing a singular value decomposition. There, we keep only the singular vectors that correspond to the largest singular values. This yields an orthonormal POD basis, which is turned into the reduced model via a Galerkin projection (Benner et al., 2015; Hesthaven et al., 2016; Volkwein, 2013).

Balanced truncation is the most widely used MOR technique for dynamical systems. It calculates the Hankel Singular Values (HSV) of the controllability and observability Gramians of the system. The controllability defines which initial states are able to reach the final state, and the observability describes how difficult it is to distinguish between the initial and zero states, given the output in the case of zero input (Antoulas, 2005; Benner et al., 2015). States that are associated with small HSV values are both hard to reach and compute; therefore, the reduced model is obtained by truncating these states. For more information regarding balanced truncation, we refer to Antoulas (2005), and Benner et al. (2015). There are also other projection-based MOR methods, for an overview of these methods we refer to Benner et al. (2015).

With the use of model order reduction, we can potentially gain a significant speed-up by addressing the “curse of dimensionality”, while retaining a predefined level of accuracy. The RB method computes low-dimensional approximations to solutions of parametrized partial differential equations. It is especially beneficial in the many-query or real-time context. Application fields are, for instance, uncertainty quantification, inversion, and sensitivity analysis where many forward simulations are required to obtain meaningful results. The RB method, however, is not suitable for single high precision simulations because of the offline construction time of the reduced space. Consequently, the method pays off only after a certain amount of simulations (Hesthaven et al., 2016; Prud’homme et al., 2002; Quarteroni et al., 2015), as illustrated in Section 3.3.1.

The RB method was thoroughly studied and presented by, for instance, (Grepl and Patera, 2005; Hesthaven et al., 2016; Prud’homme et al., 2002; Quarteroni et al., 2015) and used in, for instance, medical applications (Kärcher et al., 2017). However, it is relatively unknown in the field of Geosciences. So far, the applications of reduced order modeling in Geosciences have been mainly limited to the use of Proper Orthogonal Decomposition, e.g. for thermal (Rousset et al., 2014) and groundwater studies (Ghasemi and Gildin, 2016; Rizzo et al., 2017; Gosses et al., 2018). Compared to the RB method, POD has the disadvantage of being less efficient for a large number of parameters (Jung, 2012; Volkwein, 2011).

1.4 Scientific Aims

To help to provide both clean water and energy, we require an accurate, simulation-based assessment of temperature and pressure states in the subsurface. However, relevant subsurface properties (e.g., thermal conductivity, permeability, porosity) are always subject to uncertainties. In that regard, we investigate in this thesis:

1. **If the reduced basis method is a suitable approach to obtain accurate simulation results while considering the full scale and complexity of geophysical models.**

2. **How the consideration of global sensitivity studies can be used to improve automated model calibration, by, for instance, incorporating parameter correlations.**

   Sensitivity analysis aims at identifying relevant model parameters and reveal which parameters can be inferred from the data (Saltelli et al., 2004). We distinguish between local and global sensitivity analysis (Sobol, 2001). Wainwright et al. (2014) present a comparison of local and global sensitivity analysis for a hydrological model. In van Griensven et al. (2006), Tang et al. (2007), Cloke et al. (2008), Zhan et al. (2013), Baroni and Tarantola (2014), and Song et al. (2015) global sensitivity analysis are used for hydrological studies. Cannavó (2012) provides an example of a global sensitivity analysis for volcanic source modeling and Fernández et al. (2017) for geothermal heat exchangers. In contrast, to local sensitivity analyses, global sensitivity analyses are computationally demanding. This makes them prohibitive for finite element solutions at basin-scale. In this thesis, we present the comparison of local and global sensitivity analyses using real-case basin-scale models.

   Model calibration is a necessary method because the numerical simulations that we use in this thesis are based on geological models that are suitable representations of "reality" for specific purposes. Both the rock properties and the model geometry are associated with errors. These errors are arising, for instance, from, measurement errors and uncertainties of the geometry. Another issue is that we always use simplifications and/or generalizations of reality, which introduces another source of errors (Wellmann and Caumon, 2018). The construction of reliable models is also a challenge for geothermal applications because most geothermal measurements are shallow, and only very few deep measurements are available.

   So far, many calibrations for sedimentary basins are performed via a "trial-and-error" model calibration (Freymark et al., 2017; Noack et al., 2012). These model calibrations neither consider the sensitivities of the various parameters nor their correlations. Automated model calibrations considering sensitivities, but not global sensitivity studies, have been presented in Wellmann and Reid (2014) for the Perth Basin (Australia). Fuchs and Balling (2016) present model calibrations without sensitivities. Furthermore, local sensitivity studies are presented in Ebigbo et al. (2016).

3. **Evaluating the feasibility of uncertainty quantification for the entire state (i.e., temperature and pressure distribution) of large-scale complex geophysical systems.**

Uncertainty quantification methods, as Markov Chain Monte Carlo (MCMC), are probabilistic model calibration methods. So, similar to the previously introduced model calibrations, they aim at characterizing model errors. In contrast to deterministic model calibration methods, they are computationally demanding. So why do we need probabilistic methods when deterministic methods are much faster to solve? The issue in Geosciences is that our problems to do not possess a unique solution. This is caused by, for instance, high uncertainties in the parameter ranges (Freymark et al., 2017; Lehmann et al., 1998; Vogt et al., 2010; Wagner and Clauser, 2005). With uncertainty quantification methods, we want to both characterize and reduce the uncertainties in the parameter ranges through our observation data. Linde et al. (2017) give an overview of the state of uncertainty quantification for hydrogeological models. Furthermore, Wellmann and Caumon (2018) present the state of uncertainty quantification for structural modeling. However, uncertainty quantification is uncommon for basin-scale process simulations due to the enormous computational costs.

Furthermore, we would like to analyze:

1. **If model calibrations are necessary and how they improve our geological models.**
2. **If it is possible to generate predictive 2D and 3D uncertainty quantification maps of basin-scale models.**
3. **How the boundary conditions influence our model predictions.**
4. **Whether the RB method is suitable for sequential data assimilation problems.**

   The Ensemble Kalman Filter, being a **sequential data assimilation** technique, was first introduced by Evensen (1994), and later clarified by Burgers et al. (1998). It is a modification of the Kalman Filter that aims at overcoming the issue of evaluating the full covariance matrix (Evensen, 1994, 2003).

   The Ensemble Kalman Filter and its various modifications, such as the dual approach (Hendricks Franssen and Kinzelbach, 2008), are widely used in groundwater modeling for the estimation of the initial conditions (i.e., Hendricks Franssen and Kinzelbach, 2008; Li et al., 2012; Panzeri et al., 2014; Rodell et al., 2007; Zaitchik et al., 2008; Zhou et al., 2011) and also in combination with global sensitivity analysis (Vrugt et al., 2005). An overview of the various modifications of the Ensemble Kalman Filter is presented in Keller et al. (2018). However, the Ensemble Kalman Filter has been applied to groundwater basin-scale models only using satellite data by Zaitchik et al. (2008). Applications of the Ensemble Smoother for geothermal models are presented in Békési et al. (2018) and Limberger et al. (2018). Also, the question of the computational cost, for large spatial and temporal geophysical models and large ensemble sizes has not been addressed.

   Pagani et al. (2017) investigate the RB method for the Ensemble Kalman Filter. However, with the focus on the mathematical aspects and not related to geophysical applications. Furthermore, for the in Pagani et al. (2017) presented case studies, data assimilation is used for the parameter estimation and not the state estimation. The state estimation is the typical geoscientific application for data assimilation. We present, in this thesis, how we can apply the RB method to groundwater problems using the Ensemble Kalman Filter for estimating the

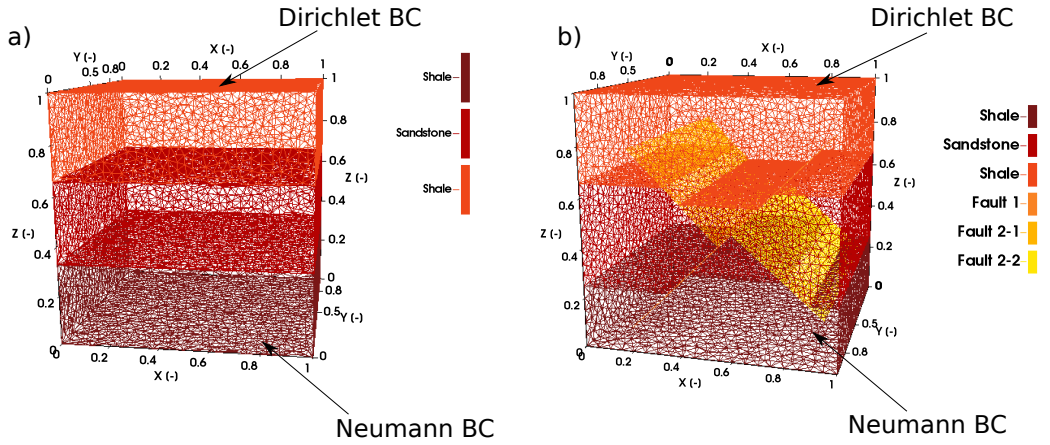


FIGURE 1.2: Mesh representation of a) the Base model consisting of three parallel layers (11, 269 degrees of freedom) and b) the Fault model consisting of three layers and two faults (23,860 degrees of freedom). Both models have a Dirichlet boundary condition at the top and a Neumann boundary condition at the bottom of the respective model. Note that the mesh is dimensionless for performance reasons.

state. We illustrate the benefits and limitations of the method for this application area.

To achieve the presented scientific aims for complex sedimentary basins, we developed an application, namely the DwarfElephant package, using the Multiphysics Object-Orientated Simulation Environment (MOOSE) (Alger et al., 2019; Permann et al., 2020), implementing the RB method. MOOSE, primarily developed by the Idaho National Laboratory, offers a built-in parallelization. A built-in parallelization is especially advantageous since constructing parallelized problems for high-performance-computing (HPC) infrastructures is often anything but trivial. This is crucial because of the computationally demanding nature of large-scale sedimentary basins (Fig. 1.1). The framework builds upon the libMesh (Kirk et al., 2006) and PETSc (Balay et al., 2017) libraries. The RB implementation mainly uses the rbOOmit package (Knezevic and Peterson, 2011) provided by libMesh. For further details regarding the software packages, refer to Section 2.2.2.

For benchmarking the code package and illustrating the methodology of the reduced basis method, we are using a simple three-layer and a fault model (Fig. 1.2) with a conductive heat transfer and Darcy flow application.

Afterwards, we extend the investigation to real-case basin-scale applications. We first focus on the conductive heat transfer problem, a typical application for geophysical simulations in geothermal studies. Hence, we consider a steady-state heat flow with the radiogenic heat production as a source term. As the first case study, we use a sedimentary basin located in the south-west of Germany, the Upper Rhine Graben (Freymark et al., 2017). The second case study is again a sedimentary basin located in the north-east of Germany, the Brandenburg model (Noack et al., 2012, 2013). Both models (Fig. 1.3) have been constructed in previous studies (Freymark et al., 2017; Noack et al., 2012, 2013).

The Darcy flow problem, as a typical application in groundwater studies, is investigated for a case study in southern Germany (close to Munich). We refer to the model as the Malm Fault Zone model (Konrad et al., 2019) and present it in Fig. 1.3.

The geothermal problems are considered for the time-independent case and the Darcy flow problem for the time-dependent case. Additionally, we present how the

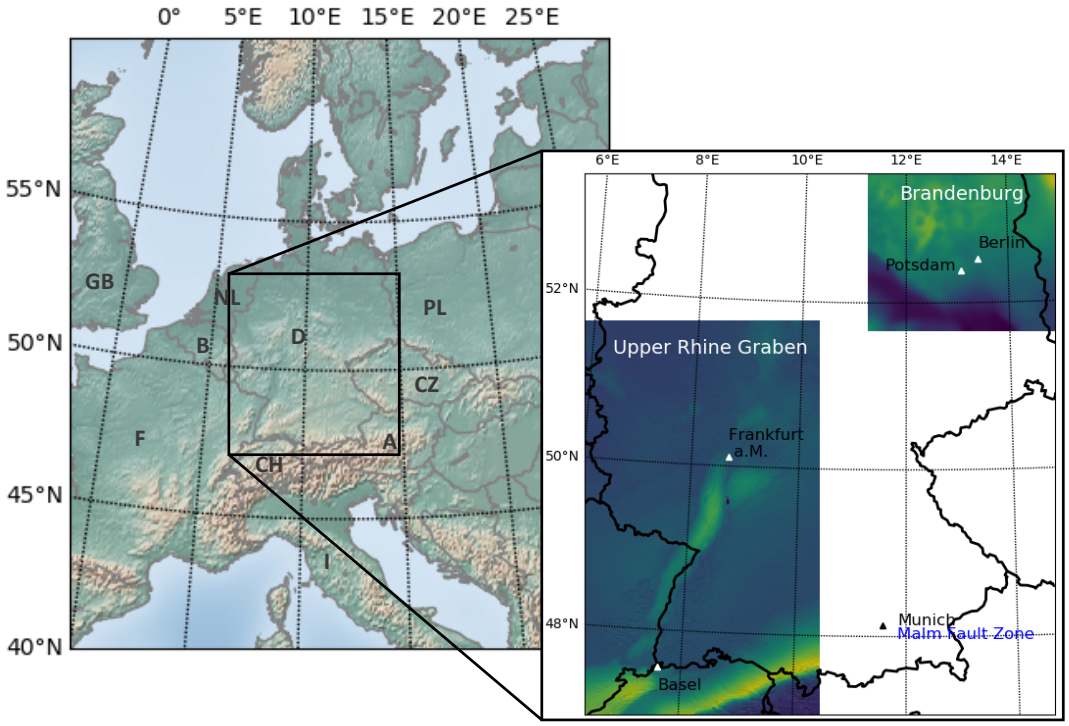


FIGURE 1.3: Location of the case studies of the Upper Rhine Graben, Brandenburg, and the Malm Fault Zone. The letters indicate the various countries. For the Upper Rhine Graben and Brandenburg model, we show the posterior standard deviation maps at a depth of 5 km. For more details regarding these maps including the colorbar we refer to Sections 6.4.3 and 6.4.1.

application of the RB method can be extended to other, possibly coupled, geophysical applications.

1.5 Structure of the Thesis

The thesis is structured as followed: First, we briefly introduce, in Chapter 2, the concepts of the reduced basis method for both the time-dependent and -independent case. In Chapter 2, we also present the developed software package DwarfElephant and the various coupling options to other external software packages. The Materials and Methods Chapter is followed by a presentation of the RB method for synthetic geothermal forward problems in Chapter 3.

Afterwards, we present the benefits of the method for conductive heat transfer real-case basin-scale applications. Therefore, in Chapter 4, we show how basin-scale model calibrations benefit from global sensitivity analyses. The concepts are illustrated using the case study of the Upper Rhine Graben. In Chapter 5, we investigate the influence of the model dimension on geothermal model predictions and provide a comparison of various model calibration techniques, using the Brandenburg model. Chapter 6 follows with a discussion of the necessity of uncertainty quantification for geothermal applications, demonstrated through the case studies of the Upper Rhine Graben and the Brandenburg model.

Afterwards, we switch the focus from geothermal to groundwater flow applications. Therefore, we first introduce, in Chapter 7, the concepts using a simple benchmark problem. In Chapter 8, this is extended to the real-case application of the Malm

Fault Zone. Transient groundwater flow problems face the challenge to derive the initial conditions from measurement data. Hence, we present in Chapter 9 how we apply the RB method to data assimilation problems.

In Chapter 10, we demonstrate the benefits of the RB method for other geophysical applications. These applications are hydrothermal and electrical resistivity tomography applications. This is followed by a discussion of the future work in Section 11 and finalized by an overall discussion in Section 12. In the Appendix, we provide tutorials on how to use the DwarfElephant package, and on how to couple it to external libraries and software packages.

1.6 Publications

Parts of this thesis are published or in preparation in the following papers:

The theoretical background of the RB method (Section 2.1.1) illustrated for a geothermal conductive heat flow problem (Section 2.1.3), together with the results and discussion for simple benchmark problems, as well as the Perth Basin (Chapter 3) is presented in:

- Degen, D., Veroy, K., Wellmann, F., 2020. Certified Reduced Basis Method in Geosciences Addressing the challenge of high dimensional problems. Computational Geosciences, 24, 241–259,

which is available as a preprint in:

- Degen, D., Veroy, K., Wellmann, F., 2019. Certified Reduced Basis Method in Geosciences Addressing the challenge of high dimensional problems. `doi:10.31223/osf.io/neh9j`.

We present the global sensitivity study in combination with a model calibration for the Upper Rhine Graben (Chapter 4) in:

- Degen, D., Veroy, K., Freymark, J., Scheck-Wenderoth, M., Wellmann, F., 2020. Global Sensitivity Analysis to Optimize Basin-Scale Conductive Model Calibration – Insights of the Upper Rhine Graben. Geothermics (under review),

which is also available as preprint in:

- Degen, D., Veroy, K., Freymark, J., Scheck-Wenderoth, M., Wellmann, F., 2020. Global Sensitivity Analysis to Optimize Basin-Scale Conductive Model Calibration – Insights on the Upper Rhine Graben. `doi.org/10.31223/osf.io/neh9j`.

Moreover, we elaborate the influence of the model dimension for the Brandenburg model (Section 5.3, 5.4.1, and 5.5.1) in:

- Degen, D., Veroy, K., Scheck-Wenderoth, M., Wellmann, F., 2020. How Well Do We Know Our Boundary Conditions?. JGR: Solid Earth (in preparation).

Furthermore, we discuss the uncertainty quantification for the Brandenburg model (Section 5.3, 6.4.1, and 6.5.1) in:

- Degen, D., Veroy, K., Wellmann, F., 2020. Uncertainty Quantification for Basin-Scale Conductive Models (in preparation),

which is also available as preprint in:

- Degen, D., Veroy, K., Wellmann, F., 2020. Uncertainty Quantification for Basin-Scale Conductive Models. Earth and Space Science Open Archive ESSOAr. doi.org/10.1002/essoar.10504276.1.

In Chapter 8, we present how the reduced basis method can be applied for the case study of the Malm Fault Zone model. This work will be presented in:

- Konrad, F., Savvatis, A., Degen, D., Einsiedl, F., Wellmann, F., Zoßeder, K., 2020. Productivity enhancement of geothermal wells through fault zones: efficient numerical evaluation of a parameter space for the Upper Jurassic aquifer of the North Alpine Foreland Basin. Geothermics (in preparation).

Here, our contribution is the construction of the reduced basis model and the support for the global sensitivity study, in the form of explaining the methodology and providing the necessary software coupling.

As presented in Section 10.2, we construct a reduced basis for the electrical resistivity applications in order to allow global sensitivity studies for this field this will be presented in:

- Von Bülow, R., Degen, D., Klitsch, N., Veroy, K., Wellmann, F., 2020. Global Sensitivity Analysis to Investigate the Application Areas of ERT Configurations (in preparation).

Our contribution to this is the construction of the reduced model and the performance of the global sensitivity studies. The other authors of the paper perform the physical interpretation. Furthermore, the authors will adapt the global sensitivity study for the specific needs of the ERT community (precisely the definition of the quantity of interest).

Chapter 2

Materials and Methods

2.1 Certified Reduced Basis Methods – A General Introduction

In this thesis, we consider certified reduced basis methods for elliptic and parabolic parameterized partial differential equations (PDEs). We first introduce the theoretical concepts of the method and then illustrate it with typical geoscientific problems. We take into account several parameter variations (Fig. 2.1):

1. Variations of the various rock properties such as thermal conductivity and permeability (denoted in green in Fig. 2.1).

2. Variations of the boundary conditions (denoted in orange in Fig. 2.1).

Another possibility is to vary, for instance, the depth of the interface between two geological units (denoted with blue bars in Fig. 2.1). These are geometrical variations, which we do not take into account in this thesis. For further information regarding geometrical parameters, we refer to Hesthaven et al. (2016), Rozza et al. (2007), and Haasdonk (2017).

In all cases, we consider only PDEs that are linear in the state variable. However, the RB method is also applicable to non-linear problems. We refer to Barrault et al. (2004), and Haasdonk (2017) for further information about the non-linear case.

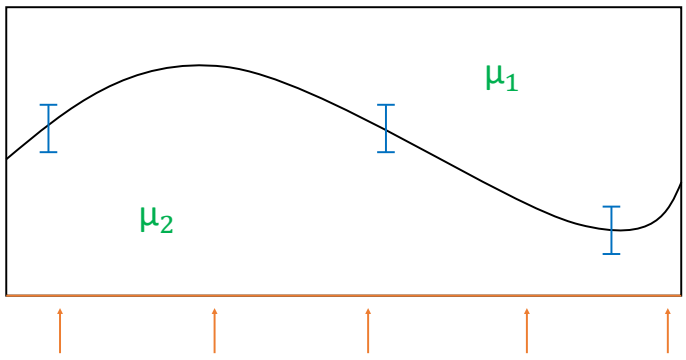


FIGURE 2.1: Schematic representation of possible parameter variations: rock properties (green), boundary conditions (orange), positions of layer interfaces (blue).

2.1.1 Elliptic - Steady State

As previously mentioned, the RB method is, so far, fairly unknown in Geosciences. Although, other model order reduction techniques such as the Proper Orthogonal Decomposition are used (Rousset et al., 2014; Ghasemi and Gildin, 2016; Rizzo et al., 2017; Gosses et al., 2018). We present the theoretical background in greater detail to offer a brief overview of the method itself, besides providing a geoscientific example to illustrate the methodology in a geoscientific context.

In general, the RB method can be used with various discretization schemes. However, since we are using an FE solver in this thesis, we explain the background of the RB approach by assuming an FE problem.

We introduce now the problem itself. We consider $\mu \in \mathcal{D}$ - where $\mathcal{D}$ is a preset, compact parameter domain in $\mathbb{R}^P$ (with P being the number of parameters) - and wish to find $u(\mu) \in X$ that satisfies (Haasdonk, 2017; Hesthaven et al., 2016; Prud'homme et al., 2002; Quarteroni et al., 2015):

$$a(u(\mu), v; \mu) = f(v; \mu), \qquad \forall v \in X. \tag{2.1}$$

The space X ($H_0^1(\Omega) \subset X \subset H^1(\Omega)$) for a spatial domain $\Omega \subset \mathbb{R}^3$ is a Hilbert space with inner product $\langle \cdot, \cdot \rangle_X$ and induced norm $\| \cdot \|_X$. Here, $a : X \times X \times \mathcal{D} \to \mathbb{R}$ is a bounded bilinear form that depends on the parameter μ. Both the variable u, for which we are solving, and the test function v are in the function space X. The functional $f : X \times \mathcal{D} \to \mathbb{R}$ is a bounded linear form on X. Given $u(\mu)$, we evaluate the output of interest $s(\mu)$ (Haasdonk, 2017; Hesthaven et al., 2016; Prud'homme et al., 2002; Quarteroni et al., 2015):

$$s(\mu) = l(u(\mu); \mu). \tag{2.2}$$

Here, $l : X \times \mathcal{D} \to \mathbb{R}$ is a linear form on X. It is a linear functional that connects the input μ to the output through our field variable u. Hence, in the context of Geosciences, it can be seen as a specific type of post-processor.

We define a parameter-dependent energy inner product by $\left(\left(\left(\cdot, \cdot\right)\right)\right)_\mu$ (Eq. 2.3) and let $\left|\left|\left|\cdot\right|\right|\right|_\mu$ denoted the induced norm (Eq. 2.4) (Haasdonk, 2017; Hesthaven et al., 2016; Prud'homme et al., 2002; Quarteroni et al., 2015):

$$\left(\left(\left(w, v\right)\right)\right)_\mu = a(w, v; \mu), \qquad \forall w, v \in X \tag{2.3}$$

$$\left|\left|\left|w\right|\right|\right|_\mu = \sqrt{a(w, w; \mu)}, \qquad \forall w \in X \tag{2.4}$$

In addition, we define the parameter-independent X-inner product (Eq. 2.5) and induced norm (Eq. 2.6) for a fixed μ (indicated by $\overline{\mu} \in \mathcal{D}$) as (Haasdonk, 2017; Hesthaven et al., 2016; Prud'homme et al., 2002; Quarteroni et al., 2015):

$$(w, v)_X = \left(\left(\left(w, v\right)\right)\right)_{\overline{\mu}}, \qquad \forall w, v \in X \tag{2.5}$$

$$\|w\|_X = \sqrt{(w, w)_X}, \qquad \forall w \in X \tag{2.6}$$

Furthermore, we assume that all linear and bilinear forms are affine parameter-dependent. Affine means that they can be expressed as a sum of terms, where each term may consist of a parameter-dependent (coefficient function) and a parameter-independent part (linear functional/bilinear form) (Haasdonk, 2017; Hesthaven et al.,

2016; Prud'homme et al., 2002; Quarteroni et al., 2015):

$$l(v;\mu) = \sum_{q=1}^{Q_l} \theta_l^q(\mu)\ l^q(v), \qquad \forall v \in X, \forall \mu \in \mathcal{D}, \tag{2.7}$$

$$f(v;\mu) = \sum_{q=1}^{Q_f} \theta_f^q(\mu)\ f^q(v), \qquad \forall v \in X, \forall \mu \in \mathcal{D}, \tag{2.8}$$

$$a(w,v;\mu) = \sum_{q=1}^{Q_a} \theta_a^q(\mu)\ a^q(w,v), \qquad \forall v,w \in X, \forall \mu \in \mathcal{D}. \tag{2.9}$$

Here, $\theta_l^q, \theta_f^q, \theta_a^q : \mathcal{D} \to \mathbb{R}$ are the parameter-dependent terms of the PDE and $l^q : X \to \mathbb{R}$, $f^q : X \to \mathbb{R}$, and $a^q : X \times X \to \mathbb{R}$ denote the parameter-independent parts.

We take into account that f, l, and a are continuous on X. We further assume that a is symmetric and parametrically coercive (Eq. 2.11) (Haasdonk, 2017; Hesthaven et al., 2016; Prud'homme et al., 2002; Quarteroni et al., 2015).

$$\exists\, \bar{\gamma} > 0 :\ \gamma(\mu) = \sup_{w \in X} \sup_{v \in X} \frac{a(w,v;\mu)}{\|w\|_X \|v\|_X} < \bar{\gamma} < \infty, \qquad \forall \mu \in \mathcal{D} \tag{2.10}$$

$$\exists\, \underline{\alpha} > 0 :\ \alpha(\mu) = \inf_{w \in X} \frac{a(w,w;\mu)}{\|w\|_X^2} > \underline{\alpha} > 0, \qquad \forall \mu \in \mathcal{D} \tag{2.11}$$

The principal idea of the RB approach is to take advantage of the decomposition (Eq. 2.7 – 2.9) by implementing an offline-online procedure.

During the offline stage, we perform all the expensive pre-computations. The cost of these pre-computations is dependent on the high-dimensional FE solution because several solves of the full FE problem are required to build a low-order approximation space, as explained in the following. Hence, if available, it is advisable to perform the offline stage for large-scale models, where the solution of a single FE problem is already extremely time-consuming, even on HPC infrastructures.

To construct the low-order approximations, we choose a parameter set $S_N = \{\mu^n, n = 1, ..., N\}$ with snapshots $u^n = u(\mu^n)$ (Fig. 2.2) and construct the RB approximation space $X_N = \text{span}\{u(\mu^n), 1 \leq n \leq N\}$. Consequently, we define RB solutions as the Galerkin projection onto this space, i.e., for μ find $u_N(\mu) \in X_N$ such that Eq. 2.12 holds (Haasdonk, 2017; Hesthaven et al., 2016; Prud'homme et al., 2002; Quarteroni et al., 2015):

$$a(u_N(\mu), v_N; \mu) = f(v_N; \mu) \qquad \forall v_N \in X_N. \tag{2.12}$$

Afterwards, we evaluate the approximate output of interest (Eq. 2.13):

$$s_N(\mu) = l(u_N(\mu); \mu). \tag{2.13}$$

The interesting question is how to retrieve and combine the snapshots in a meaningful way such that the obtained RB space is, for our applications, a suitable representation of the full FE space. In order to explain the algorithm for the basis construction, we introduce first the error bounds since they are an essential component of the algorithm. Error bounds are, as the name implies, bound the error (i.e., difference between full and reduced solution). We require them because the computation of the truth error requires the solve of the FE system. However, during the online phase, we do not want to calculate the full system. Hence, we use inexpensive error bounds. To evaluate the quality of the error bounds we introduce the effectivity. The effectivity describes how

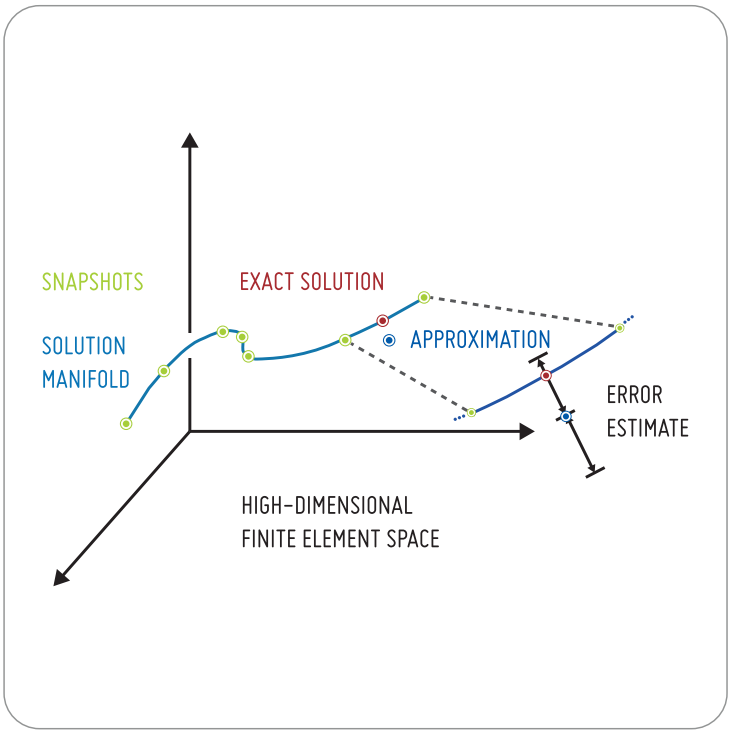


FIGURE 2.2: Schematic representation of the RB method. The snapshots are denoted in green, the exact solution in red, the approximation and the solution manifold in blue and the space and error bound in black. Note, that the error bound is an approximation to the truth error. The error bound is for our examples larger than the truth error.

much the error bound overestimates the truth error.

We first have to define a coercivity constant lower bound of our problem. To achieve this, we employ the min-θ method (Rozza et al., 2007) and define:

$$\alpha_{LB}(\mu) \coloneqq \theta_a^{\min,\overline{\mu}}(\mu) \coloneqq \min_{q\in\{1,\dots,Q_a\}} \frac{\theta_a^q(\mu)}{\theta_a^q(\overline{\mu})}\,\alpha(\bar{\mu}). \tag{2.14}$$

Here, LB stands for lower bound. The min-theta method means that our coercivity constant lower bound is dependent on the minimum parameter value of the affine decomposition.

We now define the residual as (Haasdonk, 2017; Hesthaven et al., 2016; Prud'homme et al., 2002; Quarteroni et al., 2015):

$$r(v;\mu) \coloneqq f(v;\mu) - a(u_N(\mu), v;\mu), \qquad \forall v \in X, \forall \mu \in \mathcal{D}. \tag{2.15}$$

We then find the Riesz representant $\hat{e}$ such that (Haasdonk, 2017; Hesthaven et al., 2016; Prud'homme et al., 2002; Quarteroni et al., 2015):

$$(\hat{e}(\mu), v)_X = r(v;\mu), \qquad \forall v \in X. \tag{2.16}$$

In the following, we present briefly the definitions of various error bounds and their effectivities. For further information regarding the benefits and disadvantages of each of those bounds we refer to (Haasdonk, 2017; Hesthaven et al., 2016; Prud'homme et al., 2002; Quarteroni et al., 2015).

An energy error bound (Eq. 2.17) and effectivity (Eq. 2.18) are (Prud'homme

et al., 2002):

$$\Delta_N^{\text{en}}(\mu) \equiv \frac{\|\hat{e}(\mu)\|_X}{\sqrt{\alpha_{LB}(\mu)}}, \qquad (2.17) \qquad \eta_N^{\text{en}}(\mu) \equiv \frac{\Delta_N^{en}(\mu)}{|||\hat{e}(\mu)|||_\mu}. \qquad (2.18)$$

Analogously, the error bound (Eq. 2.19) and effectivity (Eq. 2.20) for the output are defined, respectively, as (Prud'homme et al., 2002):

$$\Delta_N^{\text{s}}(\mu) \equiv \frac{\|\hat{e}(\mu)\|_X^2}{\alpha_{LB}(\mu)}, \qquad (2.19) \qquad \eta_N^{\text{s}}(\mu) \equiv \frac{\Delta_N^{s}(\mu)}{s(\mu) - s_N(\mu)}, \qquad (2.20)$$

and those for the relative output error bound (Eq. 2.21) are (Prud'homme et al., 2002):

$$\Delta_N^{\text{s,rel}}(\mu) \equiv \frac{\|\hat{e}(\mu)\|_X^2}{\alpha_{LB}(\mu)s_N(\mu)}, \qquad (2.21) \qquad \eta_N^{\text{s,rel}}(\mu) \equiv \frac{\Delta_N^{\text{s,rel}}(\mu)}{\frac{s(\mu)-s_N(\mu)}{s(\mu)}}, \qquad (2.22)$$

In this thesis, we use the output error bound within the Greedy algorithm, explained below.

The RB method uses a Greedy algorithm to construct the reduced basis. Performing the Greedy algorithm involves the following steps (Bertsimas and Tsitsiklis, 1997; Veroy et al., 2003):

Given, $\Xi_{\text{train}} \subset \mathcal{D},\ \mu_1 \in \Xi_{\text{train}},\ N = 0,\ \epsilon_{\text{tol,min}} > 0,\ X_N = 0,\ \Delta_N^{\text{max}} = \infty,\ \mu^* = \mu_1$
while $\Delta_N^{\text{max}} \geq \epsilon_{\text{tol,min}}$
$N = N + 1;$
$\mu_N^* = \arg \max_{\mu \in \Xi_{\text{train}}} \Delta_N(\mu);$
$S_N = S_{N-1} \cup \mu_N^*;$
$X_N = X_{N-1} + \text{span}\{u^{\mathcal{N}}(\mu_N^*)\};$
end

In essence, while the error is above a certain tolerance, the Greedy algorithm adds a new basis function by determining the member of the training set Ξ_{train} at which the error bound is largest. This new parameter μ_N^* is then added to the sample set S_N, and X_N is expanded in direction of the orthonormalized basis function $u(\mu_N^*)$. We repeat the procedure until we reach the chosen error tolerance. For details regarding the convergence rates of greedy algorithms in reduced basis methods, we refer to Binev et al. (2011). In general, the error tolerance can be user-defined; in a geoscientific context, the tolerance can thus be adjusted according to, for instance, the measurement error. Section 3.3.1 illustrates the benefits of this aspect.

In the offline stage, performed once, we solve for the RB functions. Hence, the computational cost depends on the finite element space dimension $\mathcal{N}$. In the online stage, performed many times (for each new μ) the computational cost depends on the dimension N. Considering $N \ll \mathcal{N}$, this means that the online stage has a much lower computational cost because of the largely reduced dimensionality of the problem (Haasdonk, 2017; Hesthaven et al., 2016; Prud'homme et al., 2002; Quarteroni et al., 2015; Veroy and Patera, 2005).

2.1.2 Parabolic - Transient

The RB method for parabolic parametrized partial differential equations with an affine parameter dependency has been studied by, e.g., Grepl and Patera (2005), Haasdonk (2017), Rovas (2003), and Rovas et al. (2006).

For the time-dependent problem, we have a simultaneous dependence of the model on parameters and time. The RB method treats time as an additional parameter. The outputs of interest may be, for the time-dependent case, a function of both the parameters and time, which is, for instance, important for control problems.

In this case, for $\mu \in \mathcal{D}$ - where $\mathcal{D}$ is again the compact parameter domain in $\mathbb{R}^P$ (with P being the number of parameters) - we wish to find $u(t;\mu) \in L^2(0,t_f;X(\Omega))$ that satisfies (Grepl and Patera, 2005; Haasdonk, 2017; Hesthaven et al., 2016; Rovas, 2003; Rovas et al., 2006):

$$m(\frac{\partial u}{\partial t}(t;\mu), v;\mu) + a(u(\mu), v;\mu) = f(v;\mu)g(t), \qquad \forall v \in X, \forall t. \tag{2.23}$$

Here, $m : X \times X \times \mathcal{D} \to \mathbb{R}$ is a bilinear form, which is in our case affine in μ, symmetric, continuous, and coercive with respect to $L^2(\Omega)$ uniformly for all μ over $\mathcal{D}$. The control is denoted with g. The remaining parameters and variables are the same as in Eq. 2.1.

Given $u(t;\mu)$, we evaluate, analogously to the elliptic case, the output of interest $s(t;\mu)$ (Grepl and Patera, 2005; Haasdonk, 2017; Hesthaven et al., 2016; Rovas, 2003; Rovas et al., 2006):

$$s(t;\mu) = l(u(t;\mu);\mu). \tag{2.24}$$

For the introduction of the method, we assume zero initial conditions. The affine decomposition is the same as for the elliptic case with the additional decomposition of the bilinear form m (Grepl and Patera, 2005; Haasdonk, 2017; Hesthaven et al., 2016; Rovas, 2003; Rovas et al., 2006):

$$m(w, v;\mu) = \sum_{q=1}^{Q_m} \theta_m^q(\mu)\, m^q(w, v), \qquad \forall v, w \in X, \forall \mu \in \mathcal{D}. \tag{2.25}$$

Here $\theta_m^q : \mathcal{D} \to \mathbb{R}$ is the parameter-dependent part and $m^q : X \times X \to \mathbb{R}$ is the parameter-independent part of the bilinear form m.

For the parameter-independent X-inner product, and induced norm, we use the same definitions as in the elliptic case (see Eq. 2.5, and 2.6). The parameter-independent L^2-inner product and induced norm are defined as (Grepl and Patera, 2005; Haasdonk, 2017; Hesthaven et al., 2016; Rovas, 2003; Rovas et al., 2006):

$$(w, v)_{L^2} \neq (((w, v)))_{\overline{\mu}}, \qquad \forall w, v \in X, \tag{2.26}$$

$$\|w\|_{L^2} = \sqrt{(w, w)_{L^2}}, \qquad \forall w \in X. \tag{2.27}$$

Define, $1 \leq k \leq K$ with K being the total number of time steps and a backward Euler approach for the time discretiziation, we define the parameter-dependent spatial-temporal energy inner product, and induced norm as (Grepl and Patera, 2005; Haasdonk, 2017; Hesthaven et al., 2016; Rovas, 2003; Rovas et al., 2006):

$$\left(\left(\left(w^k, v^k\right)\right)\right)_\mu = m(w^k, v^k; \mu) + \sum_{k'=1}^{k} \Delta t \; a(w^{k'}, v^{k'}; \mu), \qquad \forall w, v \in X, \tag{2.28}$$

$$\left|\left|\left| w^k \right|\right|\right|_\mu = \sqrt{m(w^k, w^k; \mu) + \sum_{k'=1}^{k} \Delta t \; a(w^{k'}, v^{k'}; \mu)}, \qquad \forall w \in X \tag{2.29}$$

Analogous to the elliptic case, we take into account coercive and continuous bilinear forms a and m. For this reason, we can define the coercivity constants as (Grepl and Patera, 2005; Haasdonk, 2017; Hesthaven et al., 2016; Rovas, 2003; Rovas et al., 2006):

$$\exists \, \underline{\alpha} > 0 : \alpha(\mu) = \inf_{w \in X} \frac{a(w, w; \mu)}{\|w\|_X^2} \geq \underline{\alpha} > 0, \qquad \forall \mu \in \mathcal{D}, \tag{2.30}$$

$$\exists \, \underline{\sigma} > 0 : \sigma(\mu) = \inf_{w \in X} \frac{m(w, w; \mu)}{\|w\|^2} \geq \underline{\sigma} > 0, \qquad \forall \mu \in \mathcal{D}, \tag{2.31}$$

and the continuity constants as (Grepl and Patera, 2005; Haasdonk, 2017; Hesthaven et al., 2016; Rovas, 2003; Rovas et al., 2006):

$$\exists \, \bar{\gamma_a} > 0 : \gamma_a(\mu) = \sup_{w \in X} \sup_{v \in X} \frac{a(w, v; \mu)}{\|w\|_X \|v\|_X} \leq \bar{\gamma_a} < \infty, \qquad \forall \mu \in \mathcal{D}, \tag{2.32}$$

$$\exists \, \bar{\gamma_m} > 0 : \gamma_m(\mu) = \sup_{w \in X} \sup_{v \in X} \frac{m(w, v; \mu)}{\|w\| \|v\|} \leq \bar{\gamma_m} < \infty, \qquad \forall \mu \in \mathcal{D}. \tag{2.33}$$

To determine lower bounds of the coercivity constants, we use, as in the elliptic case, the min-θ approach. Defining the residual as (Grepl and Patera, 2005; Haasdonk, 2017; Hesthaven et al., 2016; Rovas, 2003; Rovas et al., 2006):

$$r^k(v; \mu) = f(v; \mu) \; g(t^k) \; - \; m\left(\frac{u_N(t^k; \mu) - u_N(t^{k-1}; \mu)}{\Delta t}, v; \mu\right) \; - \; a(u_N(t^k; \mu), v; \mu), \qquad \forall v \in X, \; \forall k \in \mathbb{K}, \tag{2.34}$$

and the Riesz representant as the unique solution to (Grepl and Patera, 2005; Haasdonk, 2017; Hesthaven et al., 2016; Rovas, 2003; Rovas et al., 2006):

$$(\hat{e}^k, v)_X = r^k(v; \mu), \qquad \forall v \in X. \tag{2.35}$$

A particular output error bound which we use in this thesis can then be defined as:

$$\Delta_N^{sk}(t^k; \mu) = \sigma_{LB}^{-1}(\mu) \; (\sup_{v \in X} \frac{l(v; \mu)}{\|v\|}) \; \Delta_N^k(t^k; \mu), \qquad 1 \leq k \leq K. \tag{2.36}$$

Again, we provide only the brief definition of the error bound. The aim of this section is to give an insight of the RB method for the field of Geosciences. For further information regarding this and alternative error bounds, we refer to (Grepl and Patera, 2005; Haasdonk, 2017; Hesthaven et al., 2016; Rovas, 2003; Rovas et al., 2006).

For the basis construction of the parabolic problem, the Greedy algorithm is extended to a POD(t)-Greedy(μ) sampling routine. Thereby, a small Proper Orthogonal Decomposition (POD) in time is combined with an exhaustive Greedy search in the parameter space (Grepl and Patera, 2005; Haasdonk, 2017; Hesthaven et al., 2016; Rovas, 2003; Rovas et al., 2006). The algorithm is as follows:

Given, $\Xi_{\text{train}} \in \mathcal{D},\ \mu_1 \in \Xi_{\text{train}},\ N = 0,\ \epsilon_{\text{tol,min}} > 0,\ X_N = 0,\ \mu^* = \mu_1,\ \Delta_N^{k,\max} = \infty$
while $\Delta_N^{k,\max} \geq \epsilon_{\text{tol,min}}$
$e^k_{N,\text{proj}}(\mu^*) = u^k(\mu^*) - \text{proj}_{X,X_N} u^k(\mu^*), \qquad 1 \leq k \leq K$
$S_{N+1} = S_N \cup \mu^*;$
$X_{N+1} = X_N + POD_X(\{e^k_{N,\text{proj}}(\mu^*), 1 \leq k \leq K\}, 1);$
$N = N + 1;$
$\mu_N^* = \arg \max_{\mu \in \Xi_{\text{train}}} \frac{\Delta_N^k(\mu)}{|||u_N^k(\mu)|||};$
$\Delta_N^{k,\max} = \frac{\Delta_N^k(\mu)}{|||u_N^k(\mu)|||};$
end

2.1.3 RB for Applications in Geosciences

In the following, we use the geothermal conductive heat flow problem to explain step by step how one decomposes the PDE for time-independent problems. Afterwards, the Darcy flow problem illustrates the decomposition of time-dependent geoscientific problems. We illustrate the required decomposition in great detail since the RB method is relatively unknown in the field of Geosciences. We aim to illustrate with this step by step explanation to clarify the concepts of the method with physical examples well known among the community.

Geothermal Conduction

To illustrate the concepts of the decomposition, we consider the geothermal conduction problem governed by the following PDE:

$$-\lambda \nabla^2 T = 0, \qquad \text{in } \Omega. \tag{2.37}$$

For simplicity, we take into account a steady-state problem without any source or sink terms, where λ denotes the thermal conductivity, T the temperature, and Ω our domain. As the geological model, we use a simple three-layer model (Fig. 2.3) with equal sediment thicknesses. In this model, a sandstone layer interrupts two shale layers. The thermal conductivities in each layer are isotropic and homogeneous. Furthermore, we apply a Dirichlet boundary condition, with a value of zero (normed to the surface temperature), at the top of the domain and a Neumann boundary condition with the boundary value *e* at the bottom (corresponding to a normal geothermal gradient). For this model, the exact thermal conductivity values are unknown. That is the reason why we want to be able to vary all three thermal conductivities later on.

We need to reformulate our geothermal conduction problem in the following manner:

1. We need to decompose the linear form f into its parameter-dependent and -independent part. The linear form contains all contributions arising from surface

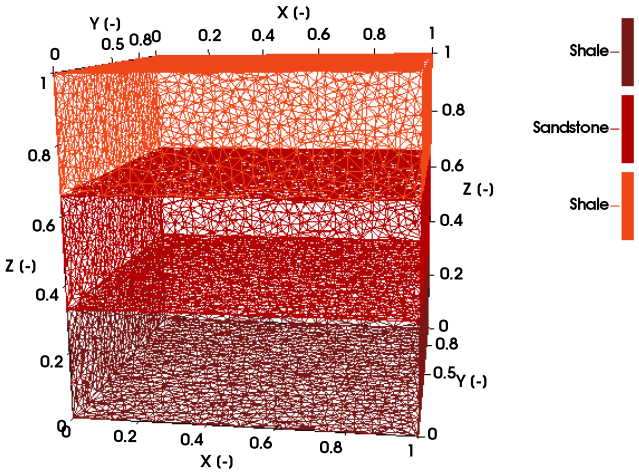


FIGURE 2.3: Mesh representation of the Base model consisting of three parallel layers. The model has a Dirichlet boundary condition at the top and a Neumann boundary condition at the bottom of the respective model. Note, that the mesh is dimensionless for performance reasons.

integrals within our PDE. In our example, these are the boundary conditions. To decompose the linear form, we need to follow these steps:

(a) Determine the number of non-homogeneous (meaning non-zero) boundary conditions we have. In our example, there is only one non-zero Neumann boundary condition (at the bottom of the model domain).

(b) Do we want to vary the value of that boundary condition? In our case, we are interested in variations of the thermal conductivities, and we do not want to vary the flux at the boundary. We thus obtain the following expression for the linear form:

$$f(v) = \int_\Gamma v\, e\, d\Gamma, \qquad \forall v \in X. \tag{2.38}$$

Here, Γ is the boundary in $\mathbb{R}^3$ and e is the boundary heat flux.

Comparing this to the general expression given by Eq. 2.8, we see that Q_f is equal to one because we have only one non-zero Neumann boundary condition. Furthermore, $\theta_f^q(\mu)$ is also equal to one since we do not want to vary the flux at the boundary.

In case the variation of the flux is desired, we need to express the linear form as:

$$f(v;s) = s\int_\Gamma v\, e\, d\Gamma, \qquad \forall v \in X,\ \forall s \in \mathcal{D}, \tag{2.39}$$

where s describes the variation of the flux.

2. We consider the compliant case, meaning that the linear output functional is equal to 2.38.

3. Furthermore, we need to decompose the bilinear form a. The bilinear form contains, in our case, all volume integrals of the PDE. Note that in case of Robin

boundary conditions, we need to account also for their contributions to the bilinear form a. For decomposing the bilinear form, we execute the following steps:

(a) We need to determine which parameters we want to vary. This is, for the given example, the thermal conductivity.

(b) Do we have a linear relationship between the parameter and the remaining part of the equation? For the geothermal conduction problem, this is the case. Therefore, we can use the RB method in the form presented here.

(c) How many different thermal conductivities do we have, and do we want to vary all of them? In the given example, we have one thermal conductivity for each layer, and we want to vary all of them. Hence, we end up with three thermal conductivities that have to be varied.

(d) How are θ and μ related? As a reminder, θ is the parameter-dependent part of the parameterized PDE that depends on the parameter μ. In our case, θ and μ are identical. However, this is not always the case.

(e) Finally, we are ready to formulate the bilinear form as:

$$a(w, v; \lambda) = \sum_{q=1}^{3} \lambda^q \int_{\Omega_q} \nabla w \nabla v \, d\Omega_q, \quad \forall v, w \in X, \forall \lambda \in \mathcal{D}. \tag{2.40}$$

Here, λ^q denotes the thermal conductivities for the respective layers, and Ω_q is the integration volume. Comparing Eq. 2.40 to the general expression (Eq. 2.9), we observe that in the geothermal example Q_a is equal to three. That is the reason because we have in total three thermal conductivities that we need to vary. Furthermore, we define $\theta_a^q(\mu) = \mu^q = \lambda^q$.

4. Our upper boundary condition is a Dirichlet boundary condition, which in contrast to Neumann and Robin boundary conditions does not appear in the integral form. We need to impose it on the system, as we would need to do it for the finite element method.

To sum up, for a geothermal conduction example consisting of three layers (e.g., two shale layers interrupted by a sandstone layer), where we want to vary the thermal conductivity in each of these layers, we end up with Q_a equal to three. Defining a homogeneous Dirichlet boundary condition and a Neumann boundary condition for the top and bottom of the model, respectively, we have Q_f equal to one. The diffusive part of the equation is always the same; only the thermal conductivity varies for different rock types. That does not mean that both parts can be evaluated independently of each other. Following (Eq. 2.40), the thermal diffusion term ($\nabla^2 T$) is the parameter-independent part of the PDE, whereas the thermal conductivity (λ) is the parameter-dependent part.

The decomposition, shown in this Chapter schematically, still poses difficulties when implemented in a numerical solver. For this reason, a full package, named DwarfElephant, was built on top of the finite element simulation package MOOSE, in the context of this thesis. For further details regarding this package, we refer to Section 2.2.

After decomposing the PDE into its parameter-dependent and -independent part, we need to define the coercivity constant lower bound for the calculation of the error bound. For this, we use the min-θ approach described in Rozza et al. (2007). Hence, in

our geothermal example, this lower bound is equal to the lowest thermal conductivity since we set the reference thermal conductivity $\bar{\lambda}$ to 1.

With this decomposition, we can now use the RB method as provided in the Dwarf-Elephant package. This section intends to demonstrate which preparational work is required for the RB method. For the practical usage of the RB method within the DwarfElephant package, we refer to Appendix A.

Darcy Flow Problem

We now explain how to decompose a specific time-dependent problem. To illustrate this, we use the hydraulic head formulation of the Darcy law:

$$S_s \frac{\partial h}{\partial t} - \nabla q = 0, \qquad \text{with} \quad q = -\frac{k}{\mu}\nabla h, \qquad \text{with} \quad h = z + \frac{p}{\rho g}. \tag{2.41}$$

Here, h is the hydraulic head, k the permeability, μ the dynamic viscosity, z the depth, p the pressure, ρ the fluid density, g the gravity acceleration, and S_s the specific storage. For simplicity, we do not take into account any source or sink terms. As the geological model, we consider the model from the previous section. However, we apply now Dirichlet boundary conditions, with the value 0 and h_{right}, to the left and right side of the model domain. For this example, we take the permeabilities of the various layers as unknown. The remaining parameters are assumed to be known. Hence, we have in total three permeabilities that we want to vary later on.

For the decomposition of the time-dependent problem, we need to follow these steps:

1. We need to decompose the linear form f. Therefore, we follow the steps 1a and b from the previous section. We have now one non-zero Dirichlet boundary condition. We need to use a lifting function to compensate for this. The lifting function $g(x)$ (see Eq. 2.42) transfers the boundary to volume contributions. Hence, after applying the lifting function, we obtain a zero Dirichlet boundary condition. Again, we do not want to vary the value of the boundary condition. This results in the following expression for the linear form:

$$\begin{aligned} f(v;k) =& \frac{1}{\mu}\sum_{q=1}^{3} k_q \int_{\Gamma_q} \nabla v \; g(x) \; d\Gamma_q, \qquad \forall v \in X, \; \forall k \in \mathcal{D}, \\ \text{with g(x)} =& h_{\text{right}} \; x, \end{aligned} \tag{2.42}$$

   where Γ_q form the right boundary in $\mathbb{R}^3$, $g(x)$ is the lifting function, and h_{right} is the hydraulic head at the right model domain boundary.

2. Next, we need to decompose the bilinear form a, this is analogous to the previous section. Hence, we follow the steps 3a to e.

$$a(w, v; k) = \frac{1}{\mu}\sum_{q=1}^{3} k_q \int_{\Omega_q} \nabla w \; \nabla v \; d\Omega_q, \qquad \forall v, w \in X, \; \forall k \in \mathcal{D}. \tag{2.43}$$

3. The difference between a time-independent and -dependent problem is that, in addition to the bilinear form a, we also have the bilinear form m. For the decomposition of the bilinear form m, we have to follow these steps:

   (a) Do we have a scaling parameter for the time derivative? In our example, this is the specific storage because we have a transient Darcy flow problem.

(b) Do we want to vary this scaling parameter? If yes, in how many layers do we want to be able to vary the scaling parameter? We assumed that the specific storage is known in our example, and we assume that it is the same for all layers. Hence, we do not want to vary it yielding the following formulation of the bilinear form m:

$$m(w, v; S_s) = \int_\Omega S_s \frac{\partial h(t;k)}{\partial t} \nabla w \ \nabla v \ d\Omega, \quad \forall v, w \in X, \ \forall S_s \in \mathcal{D}, \ \forall k \in \mathcal{D}, \ \forall t. \tag{2.44}$$

Note that in case we consider a different specific storage for each layer, and we want to vary those, the decomposition changes to:

$$m(w, v; S_s) = \sum_{q=1}^{3} S_{s,q} \int_{\Omega_q} \frac{\partial h(t;k)}{\partial t} \nabla w \ \nabla v \ d\Omega_q, \quad \forall v, w \in X, \ \forall S_s \in \mathcal{D}, \ \forall k \in \mathcal{D}, \ \forall t. \tag{2.45}$$

4. Finally, analogous to the elliptic case and in the same way as for the standard finite element problem, we need to force the Dirichlet boundary conditions onto the system by a modification of the stiffness matrix.

Also, for the time-dependent problem, we need to determine the coercivity constant lower bound. We again employ the min-θ approach yielding the lowest permeability as the coercivity constant lower bound since we set the reference permeability $\bar{k}$ to 1.

Analogous, to the elliptic case this section introduces the necessary preparational work for employing the RB method for parabolic geophysical problems. The usage of the RB method within the DwarfElephant package is analogous to the elliptic case presented in Appendix A.

Also, note that this and the previous section do not intend to provide a tutorial of how to decompose in general time-dependent and -independent problems. Instead, we want to provide typical examples of these types of problems to offer a guideline for decomposing geoscientific problems. Be aware, that the decomposition is highly problem specific, as indicated here, and has to be adapted for each new problem.

2.2 Software

We make use of various open-source software packages in this thesis. For the forward simulations, we developed the DwarfElephant package (Degen et al., 2020c), an application to the MOOSE Framework (Permann et al., 2020). In the following sections, we present both software packages. For the deterministic model calibrations, we use the Dakota framework (Adams et al., 2017), and the Python library SciPy (Jones et al., 2014). The sensitivity analyses are executed using the Python library SALib (Herman and Usher, 2017) and the Markov Chain Monte Carlo simulations using the Python library PyMC (Patil et al., 2010).

2.2.1 MOOSE Framework

The MOOSE Framework is in contrast to some other FE solvers, fully coupled and has fully implicit solvers. FE solvers are highly advantageous for geological studies since they allow discrete fracture analyses, as we demonstrate later on. Additionally,

MOOSE is built-up on top of the PETSc library, allowing an easy interfacing to advanced preconditioner settings.

In the case of the RB method, it was furthermore beneficial that MOOSE is also based on the libMesh library since libMesh already has a package implementing the RB method. So far, we focused our work, within the software environment, on integrating the existing libMesh package into the MOOSE Framework. Additionally, we extend the functionality of the libMesh package for the various applications, presented in this thesis. This work has been realized within our MOOSE application named DwarfElephant.

Using the MOOSE environment in addition to the libMesh library provides the user with several advantages such as a highly simplified model setup, an easier interchange of different physics, and simplified mesh modification. MOOSE uses a dimension independent physics; together with the object-oriented structure of the software, this allows uncomplicated reuse and extensibility of pre-existing code structures. Another advantage of MOOSE is the built-in parallelization, meaning the same code can be used on desktop computers and HPC infrastructures (Permann et al., 2020).

2.2.2 DwarfElephant

The DwarfElephant package is based on the MOOSE framework and aims at offering user-friendly access to the reduced basis method. It is an open-source, high-performance library written in the programming language C++. We based it on the libMesh implementation of the reduced basis method. Using the libMesh implementation directly, and not via the MOOSE framework, has several disadvantages for geoscientific problems. Within the geoscientific community, we have to deal with rapidly changing geological models. Using the RB package within libMesh means that the entire working routine must be constructed again for each new model, which is an unnecessarily time-consuming task. The DwarfElephant package instead takes full advantage of the MOOSE capabilities. This allows an easy and fast interchange of different models due to several automated processes.

It is furthermore possible to automatically apply different physics to different subdomains of the model because MOOSE operates on the individual subdomains separately. We take advantage of that by automatically separating the stiffness matrix with respect to the subdomains but also allowing every other decomposition. This automatic separation provides an easier and faster way to simulate new models with the RB method. The load vector is not automatically separated, since this is often not required, but as for the stiffness matrix, any decomposition is possible.

We developed the software on purpose in a physics-independent framework to allow usability of the software also outside of the geoscientific community. The DwarfElephant package is furthermore easily compatible with other MOOSE applications, such as GOLEM (Cacace and Jacquey, 2017) and REDBACK (Poulet et al., 2014). In Appendix A we provide a tutorial of how to use the RB method for a simple three-layer geothermal problem. Ganther (2018) benchmarks the geothermal and Darcy flow problem using the FE, and RB method.

We evaluate the performance of the RB method for geoscientific applications. Several software packages are available for the RB method:

- rbMIT (Huynh et al.),
- RBmatlab (University of Stuttgart et al.),
- rbOOmit (Knezevic and Peterson, 2011),
- pyMOR (Milk et al., 2016),

- Feel++ (Prud'homme et al., 2012),
- FreeFem++ (Hecht et al., 2007),
- RBniCS (Ballarin et al., 2017), (Hesthaven et al., 2016).

However, some of these packages, e.g., the MATLAB packages rbMIT and RBmatlab, are based on proprietary software. This makes it challenging to provide results that are openly verifiable and reproducible. Furthermore, they often lack high-performance capabilities, thus making them unsuitable for large-scale models. Other packages, such as pyMOR and libMesh, are constructed as libraries. Thus, setting up a specific problem requires much work. Python wrappers are only available for pyMOR, RBnicS, and FreeFem++. Some of these libraries are used by higher-level software environments, for example, RBniCS uses FEniCS (`http://fenicsproject.org/`), and MOOSE is based on libMesh, which contains the rbOOmit package. So, with the DwarfElephant package, we provide an easy, and open-source access to the RB method. We are taking full advantage of the user-friendly MOOSE environment, allowing an easy set-up of new models and a quite straightforward approach to extend the package further.

Additionally, several previous geophysical studies use MOOSE applications (Cacace and Jacquey, 2017; Jacquey et al.; Poulet and Veveakis, 2016; Poulet et al., 2014). Our package thus offers a direct use of the RB method. It furthermore allows a merging with already existing MOOSE applications, something that is not possible with any of the other presented packages.

2.2.3 Software Coupling

In general, there are two ways of coupling two software libraries, direct, and loose coupling. The loose coupling has the advantage of being easier to implement, but less efficient. In contrast, the direct coupling is more intensive from the implementation side but also more efficient. In the following, we describe which coupling types we provide within the DwarfElephant package for which software libraries and methods.

Loose Coupling

The loose coupling between Dakota and MOOSE is usable for any MOOSE application. We developed it to allow the execution of, for instance, optimization algorithms provided by Dakota with MOOSE applications as forward solvers. A loose coupling has the advantage of being fast and easy to implement, but unfortunately not very efficient due to the constant writing and reading of files. This I/O overhead can be avoided by using a direct coupling.

Our coupling is usable for both the FE and RB simulations, although we strongly recommend, for efficiency reasons, to use the direct coupling in case of the RB method. For more information about the coupling and how to use it, we refer to the tutorial in Appendix B.

Direct Coupling

For the RB method, we offer a direct coupling to Dakota via their Python interface. Furthermore, we can use the same coupling to interface with many other Python libraries as, SALib and PyMC, or other software packages that provide Python interfaces. For the coupling, we implemented the online stage of the RB method in Python. We

also have a Theano (Bergstra et al., 2010) version of the online stage, which has been developed by Liao et al. (2019), allowing an interfacing with the PyMC3 library (Salvatier et al., 2016). For more information about the coupling and how to use it, we refer to the tutorial in Appendix C.

After presenting the concept of the RB method with all its computational aspects, we present in the next chapter the benefits of the method for geothermal conductive heat transfer problems.

Chapter 3

Certified Reduced Basis Method for Geothermal Conductive Heat Transfer Applications

This chapter aims to provide an introduction to the concepts of the RB method for steady-state geophysical applications. As an example, we are using a geothermal conductive heat transfer problem. To better illustrate the concepts of the RB method for the geothermal applications, we use the simple benchmark problems of both the Base and the Fault model, presented in Section 3.1.1. As an extension to real-case applications, we use a simplified version of the Perth Basin (Australia) to demonstrate that the method is applicable for real-case applications. All variables are displayed in their nondimensional form since this is a conceptual chapter, and we want to demonstrate how the method and the software operates. This chapter is part of the publication Degen et al. (2020c).

3.1 Geological Models

For our geothermal conductive heat transfer problem, we consider three geological models. The Base and Fault model (Section 3.1.1) are benchmarking problems. The Perth Basin (Australia, Section 3.1.2) is a simplified real-case model illustrating the advantages of the RB method for real-case geothermal applications.

3.1.1 Base and Fault Model

For the benchmarking of the geothermal and Darcy flow problem (see Chapter 7), we use two different synthetic models. For performance issues, we consider dimensionless meshes with an extension in x-, y-, and z-direction from 0 to 1. The reference length is 1,000 m.

The first model consists of three parallel layers (further referred to as the "Base model"), and the second has the same three layers and additionally two faults (the "Fault model"). The younger fault offsets the layers by 0.06 (corresponding to 60 m) and the older fault by 0.05 (50 m). The layer offset created by the older fault is 0.02 (20 m). Figure 3.1 displays a schematic representation of the two different models. In both cases, the top and bottom layers are shale layers, whereas the middle layer consists of a more conductive sandstone. For all rock types, typical thermal conductivities (see Table 3.1) are assumed (Midttømme et al., 1998). The faults, only present in the second model, have the same thermal conductivities. These are the highest thermal conductivities within the entire model. The faults have a dip of 45°. Note that the thermal conductivity is also a nondimensional parameter.

For the meshing, we used the MeshIT software developed by the Geoforschungszentrum Potsdam (Cacace and Blöcher, 2015) and Gmsh (Geuzaine and Remacle, 2009). Both models have tetrahedron meshes. Furthermore, the faults are considered as discrete 2D elements in the 3D mesh and consisting of triangular elements. Table 3.2 lists the different meshes and their properties. For the Base model, the degrees of freedom are varying from 1,426 to 1,512,161. All meshes have the same geometries and only differ in their global refinement level. Note that for each global refinement factor a new mesh has been created. We did not refine the pre-existing mesh in order to prevent errors by refining possible ill-created mesh elements. The meshes of the Fault model, varying from 4,600 degrees of freedom to 1,969,077, are generated in the same way as for the Base model. However, we use the meshing software Gmsh instead of MeshIT to demonstrate the usability of different meshing software within the package. The faults have a global refinement that is twice as fine as the layers.

All simulation-relevant parameters for the FE and RB method are listed in Tab. 3.3. Note that for the RB method the same parameters have been chosen as for the FE method. Hence, they are not listed again in the RB section of Tab. 3.3. In this section, we mention only parameters additionally required for the RB method. The larger meshes of the Fault model required a higher solver tolerance in order to avoid rounding errors during the greedy algorithm. We thus made another set of simulations with smaller solver tolerances of 10^{-9} for the RB and FE model. Note that we are using

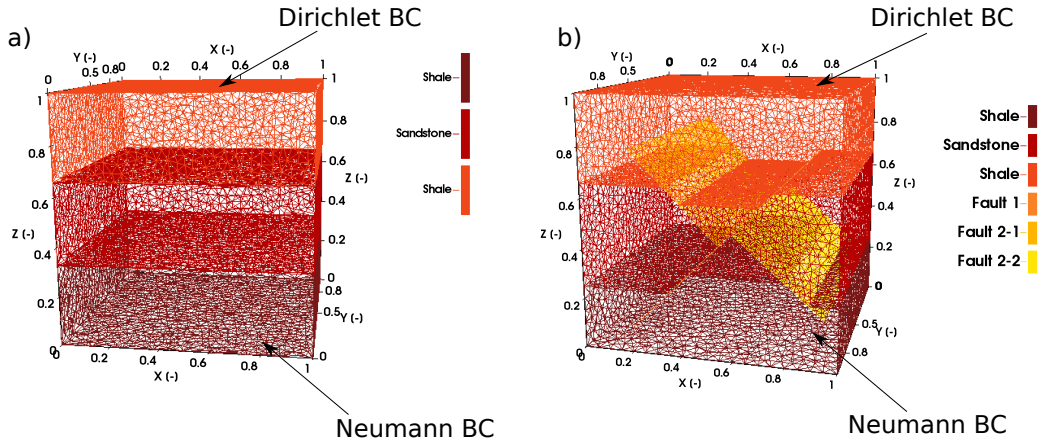


FIGURE 3.1: Mesh representation of a) the Base model consisting of three parallel layers (11, 269 degrees of freedom) and b) the Fault model consisting of three layers and two faults (23,860 degrees of freedom). Both models have a Dirichlet boundary condition at the top and a Neumann boundary condition at the bottom of the respective model. Note that the mesh is dimensionless for performance reasons.

the same physical model and affine decomposition as presented in Section 2.1.3. For both models, we used a reference thermal conductivity of 1.05 W m^{-1} K^{-1} (i.e., the smallest thermal conductivity), and the reference temperature of 10 °C.

Both models have a Dirichlet boundary condition of 10 °C at the top and a Neumann boundary condition of 0.039 W m^{-2} at the bottom. The Neumann condition represents a normal geothermal gradient with a temperature increase of about 30 °C per kilometer. At the top, we define the average annual temperature as a Dirichlet boundary condition. Correspondingly, the dimensionless top boundary condition is 0. The bottom boundary condition is 3.71 (choosing a standard basin heat flux).

Note for the RB method, we have considered three parameters in case of the Base model, and five parameters in case of the Fault model. That means that although

we made our simulations with equal thermal conductivities for layer 1 and 3 and equal thermal conductivities for both faults, we have the flexibility to set them to non-equal values. For using the RB method, we decompose our problem into a parameter-dependent and -independent part.

TABLE 3.1: Thermal and model properties for both synthetic models (Midttømme et al., 1998). Take into account that both the thermal conductivity and the length are dimensionless (with normalization factor $\lambda_{\text{ref}} = 1.05$ W m^{-1} K^{-1} and $l_{\text{ref}} = 1000$ m).

unit	thermal conductivity [W m^{-1} K^{-1}] / thermal conductivity [-]	extension in		
		x-direction [-]	y-direction [-]	z-direction [-]
Base model				
top layer (shale)	1.05 / 1.00	0.00 - 1.00	0.00 - 1.00	0.66 - 1.00
middle layer (sandstone)	2.50 / 2.38	0.00 - 1.00	0.00 - 1.00	0.33 - 0.66
bottom layer (shale)	1.05 / 1.0	0.00 - 1.00	0.00 - 1.00	0.00 - 0.33
Fault model				
top layer - section 1 (shale)	1.05 / 1.00	0.00 - 0.34	0.00 - 1.00	0.66 - 1.00
top layer - section 2 (shale)	1.05 / 1.00	0.40 - 0.70	0.00 - 1.00	0.60 - 1.00
top layer - section 3 (shale)	1.05 / 1.00	0.76 - 1.00	0.00 - 1.00	0.66 - 1.00
middle layer - section 1 (sandstone)	2.50 / 2.38	0.00 - 0.34	0.00 - 1.00	0.25 - 0.66
middle layer - section 2 (sandstone)	2.50 / 2.38	0.43 - 0.70	0.00 - 1.00	0.33 - 0.60
middle layer - section 3 (sandstone)	2.50 / 2.38	0.89 - 1.00	0.00 - 1.00	0.25 - 0.66
bottom layer - section 1 (shale)	1.05 / 1.00	0.00 - 0.35	0.00 - 1.00	0.00 - 0.25

bottom layer - section 2 (shale)	1.05 / 1.00	0.43 - 0.80	0.00 - 1.00	0.00 - 0.33
bottom layer - section 3 (shale)	1.05 / 1.00	0.89 - 1.00	0.00 - 1.00	0.00 - 0.25
Fault 2	3.00 / 2.86	0.20 - 0.90	0.00 - 1.00	0.10 - 0.80
Fault 1 - section 1	3.00 / 2.86	0.20 - 0.55	0.00 - 1.00	0.45 - 0.80
Fault 1 - section 2	3.00 / 2.86	0.60 - 0.95	0.00 - 1.00	0.20 - 0.50

TABLE 3.2: Mesh properties for both synthetic models. Listed are the refinement, the number of nodes, and the number of elements for the various meshes. We denote the parameters that are not applicable with n/a.

model	refinement		# elements	# nodes
	layer	fault		
Base model	0.12	n/a	6,351	1,426
	0.1	n/a	10,975	2,378
	0.05	n/a	56,222	11,269
	0.04	n/a	94,749	18,692
	0.03	n/a	179,976	34,989
	0.02	n/a	437,183	83,924
	0.015	n/a	808,468	154,181
	0.01	n/a	1,908,931	361,967
	0.009	n/a	2,374,382	449,662
	0.008	n/a	3,035,457	574,241
	0.007	n/a	4,004,907	756,814
	0.006	n/a	5,510,748	1,040,333
	0.005	n/a	8,021,256	1,512,161
Fault model	0.1	0.05	26,742	4,600
	0.05	0.025	1142,576	23,860

	0.025	0.0125	896,885	149,490
	0.0125	0.00625	6,235,267	1,040,325
	0.01	0.005	11,796,240	1,969,077

TABLE 3.3: Simulation parameters for the finite element and the reduced basis method of both synthetic models. We denote the parameters that are not applicable with n/a.

Simulation parameter	Value	
	Base model	Fault model
FE method		
Solver type	Newton	
Linear solver tolerance	10^{-8}	$10^{-8}/10^{-9}$
Nonlinear solver tolerance	10^{-8}	$10^{-8}/10^{-9}$
Preconditioner type	hypre	
Hypre type	boomerang	
ksp_gmres_restart	101	
RB method		
Relative training tolerance	10^{-5}	
Parameter range (Layer 1)	0.5 to 5.0	
Parameter range (Layer 2)	0.5 to 7.0	
Parameter range (Layer 3)	0.5 to 5.0	
Parameter range (Layer 4)	0.5 to 5.0	
Parameter range (Layer 5)	0.5 to 5.0	
Parameter range (Layer 6)	0.5 to 5.0	
Parameter range (Faults)	n/a	0.5 to 4.0
Number of parameters	3	5
Number of training samples	5	100

Maximum number of basis functions	5	100

3.1.2 Perth Basin

For a first test of the RB method for real-case large-scale geological models, we consider the Perth Basin (Australia) modified after Wellmann and Reid (2014) and displayed in Fig. 3.2. We use a reference length of 70,000 m, a reference thermal conductivity of 4.3 W m^{-1} K^{-1} (i.e., the largest thermal conductivity), and a reference temperature of 22 °C. It has a spatial extension of 0.90 (63 km) in x-direction, 1.00 (70 km) in y-direction and 0.27 (19 km) in z-direction. It consists of six layers and is displayed in Figure 3.2. We vary the thermal conductivities in all six layers. At the top of the model, a Dirichlet boundary condition with a value of 0 (22 °C) is applied, corresponding to the annual average temperature of Australia. For the bottom boundary condition, we use a Neumann boundary condition with a value of 19.24 (0.026 W m^{-2}), which is the average basin heat flux of the Perth Basin. Table 3.4 contains all thermal properties, and Table 3.5 all simulation properties. The model itself was generated using GemPy (de la Varga et al., 2019). We have meshed it using the "GeneratedMesh" functionality of MOOSE, resulting in a hexahedral mesh with 1,771,561 degrees of freedom and 1,728,000 elements.

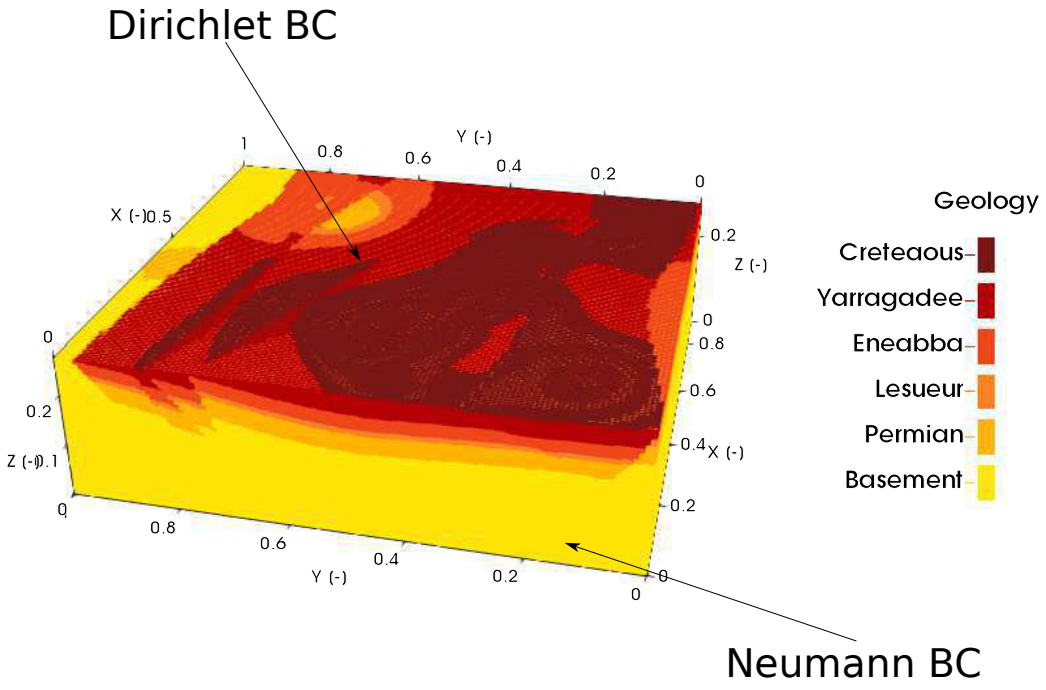


FIGURE 3.2: Mesh representation of the Perth Basin model with 1,771,561 degrees of freedom. The model has a Dirichlet boundary condition at the top and a Neumann boundary condition at the bottom of the model. Note that the mesh is dimensionless for performance reasons.

TABLE 3.4: Thermal properties for the Perth Basin (Wellmann and Reid, 2014). Take into account that both the thermal conductivity and the length are dimensionless (normalization factor λ_{ref} = 4.3 W m^{-1} K^{-1} and l_{ref} = 70,000 m).

unit	thermal conductivity [W m^{-1} K^{-1}] / thermal conductivity [-]
Cretaceous	3.9 / 0.91
Yarragadee	4.3 / 1.00
Eneabba	3.85 / 0.90
Lesueur	3.8 / 0.88
Permian	3.1 / 0.72
Basement	2.7 /0.63

TABLE 3.5: Listed are the simulation parameters for the Perth Basin model for both the FE and the RB method.

Simulation parameter	Value
FE method	
Solver type	Newton
Linear solver tolerance	10^{-8}
Nonlinear solver tolerance	10^{-8}
Preconditioner type	hypre
Hypre type	boomerang
ksp_gmres_restart	101
RB method	
Rel. training tolerance	10^{-3}
Parameter range (Layer 1)	0.453 to 1.360
Parameter range (Layer 2)	0.500 to 1.500
Parameter range (Layer 3)	0.448 to 1.343
Parameter range (Layer 4)	0.441 to 1.325

Parameter range (Layer 5)	0.360 to 1.081
Parameter range (Layer 6)	0.313 to 0.941
Number of parameters	6
Number of training samples	100
Maximum number of basis functions	50

3.2 Results

We first present the results of applying the RB method to a conductive heat transfer problem for the Base and Fault model. After presenting the benchmark problems, we demonstrate how one applies the RB method to real-case models by using the case study of the Perth Basin (Wellmann and Reid, 2014).

We performed all simulations on the RWTH compute cluster on an Intel Westmere X5675 (3.07 GHz, 6 cores per chip, 12 cores per node and 24 GB memory per node) using one CPU core. Only the performance tests (see Section 3.3.1) have been simulated using Intel Xeon E5-2680 v3 Haswell nodes on the JURECA compute cluster (2.5 GHz, 24 cores per node and 128 GB memory per node) (Jülich Supercomputing Centre, 2016).

In order to achieve convergence for the larger models, a preconditioner is required. We chose the HYPRE preconditioner of the type BoomerAMG. This PETSc preconditioner is a matrix element-based preconditioner that can be found in the LLNL package HYPRE (Balay et al., 2017). Additionally, one should note that MOOSE is a state-of-the-art forward solver. Thus, we have a highly efficient FE solve, which affects the speed-up (shown in Section 3.2.1) since the preconditioning does not influence the online times because of the dense nature.

3.2.1 Base and Fault Model

The main reason for using a surrogate model such as the RB method is certainly the speed-up. Therefore, Fig. 3.3 compares the runtime of the FE and RB simulations and additionally plots the speed-ups obtained by the RB approach. First of all, we can see that we get a significant speed-up with a maximum value of 1,009.5 for the Base model with 1,512,161 degrees of freedom. Moreover, maybe even more important is to note that this speed-up is higher for the larger models. Hence, we observe a speed-up variation from 33.4 to 1,009.5 for an increase in the degrees of freedom from 1,426 to 1,512,161.

For the Fault model, we observe speed-ups between 32.0 and 1,138.5 for solver tolerances of 10^{-9}. We get smaller speed-ups for the Fault than for the Base model for similar model sizes. The reasons for this are illustrated in Section 3.3.1.

After showing how much faster simulations are with the RB method, we have to validate that the results for the method are qualitatively and quantitatively comparable to the FE solutions. Therefore, we simulated both models, with the properties given in the previous section, taking the standard finite element and the reduced basis method into account.

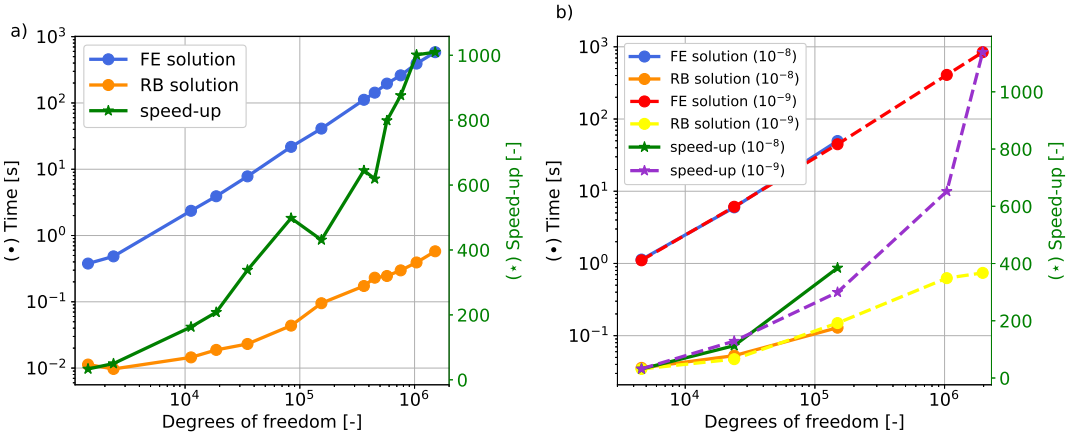


FIGURE 3.3: Runtime comparison between the full finite element solution –denoted in blue– and the RB solution –denoted in orange– for the a) Base model and b) Fault model with permeable faults with a solver tolerance of 10^{-8}. Additionally, in b) the red dashed line denotes the FE solution and the yellow dashed line the RB solution for a tolerance of 10^{-9}. Furthermore, the blue line illustrates the speed-up with a solver tolerance of 10^{-8} and the violet dashed line the speed-up with a solver tolerance of 10^{-9}.

Figure 3.4 shows the simulation results for the Base and the Fault model under consideration of the RB method. Note that Figure 3.4 shows only the results of the RB method. The results of the FE method are not plotted since they are not distinguishable in the temperature plot due to the very small errors of the RB method (see Figure 3.5).

In Fig. 3.4a, we observe the expected temperature diffusion for a three-layer scenario with a more conductive middle layer. So, the top and bottom layers have the highest temperature increase of 1.26 (12.6 °C) and 1.22 (12.2 °C), respectively. This is due to the lower thermal conductivity. In contrast, the middle layer has a lower temperature increase of 0.52 (5.2 °C) because of the higher thermal conductivity. For the Base model, we encounter parallel temperature isolines following the geological layers and varying from a minimum temperature of 0.0 (corresponding to 10.0 °C) to a maximum temperature of 3.0 (corresponding to 40.0 °C).

The image slightly changes for the Fault model (Fig. 3.4b). Here, we consider permeable faults (Fig. 3.4b). In case of the permeable faults, the highly conductive faults yield a lower maximum temperature of 2.22 (32.2 °C) and curved temperature isolines following both fault structures. We thus observe an x-like structure for the isolines around the faults. These results are in accordance with our expectations since the heat transport follows the high conductive faults, as seen in Fig. 3.4b.

After validating that we qualitatively reproduce the FE solutions with the RB method, we want to illustrate the quantitative differences between both methods. As with any approximation, we face the question of the price we have to pay for our speed-up. Usually, we pay by losing accuracy; the critical aspect is naturally how high this accuracy loss is. For this purpose, we performed the same simulation with the same conditions and within the same software with *i)* the finite element method and *ii)* the reduced basis method. Afterwards, we generated a difference plot between both simulations (Fig. 3.5). For both synthetic models, we reach our given relative error tolerance of 10^{-5}. The largest errors are located at the layer interfaces, where an abrupt change of the parameter values occurs.

Additionally, we present a plot of the reduced basis convergence rate in Fig. 3.6. In this figure, the maximum error bound is plotted as a function of the number of basis

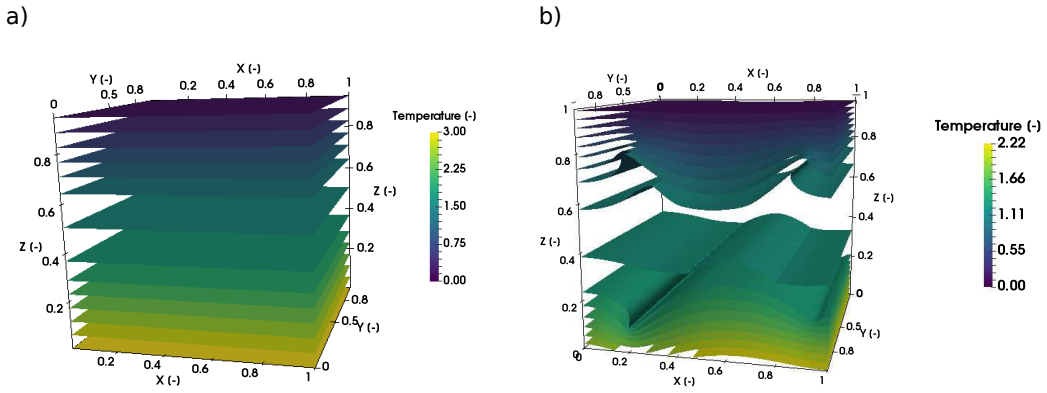


FIGURE 3.4: Temperature distribution for a) the Base model, and b) Fault model with permeable faults. Note that both the mesh dimensions, and the temperature have been normalized for numerical reasons, and are hence dimensionless. Both models have a reference temperature of 10 °C, a reference length of 1,000 m and solver tolerances of a) 10^{-8} and b) 10^{-9}.

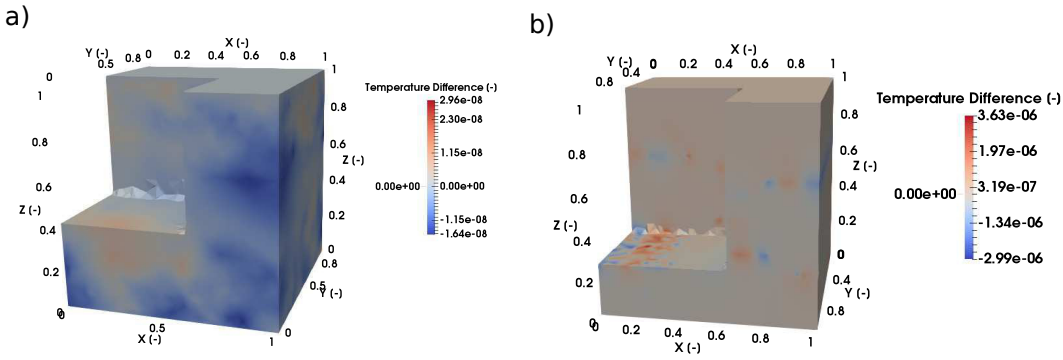


FIGURE 3.5: Difference plot between the full finite element and the reduced basis solution for the a) Base model and b) Fault model with permeable faults. The relative error tolerance was set for all cases to 10^{-5}, and the solver tolerances to a) 10^{-8} and b) 10^{-9}.

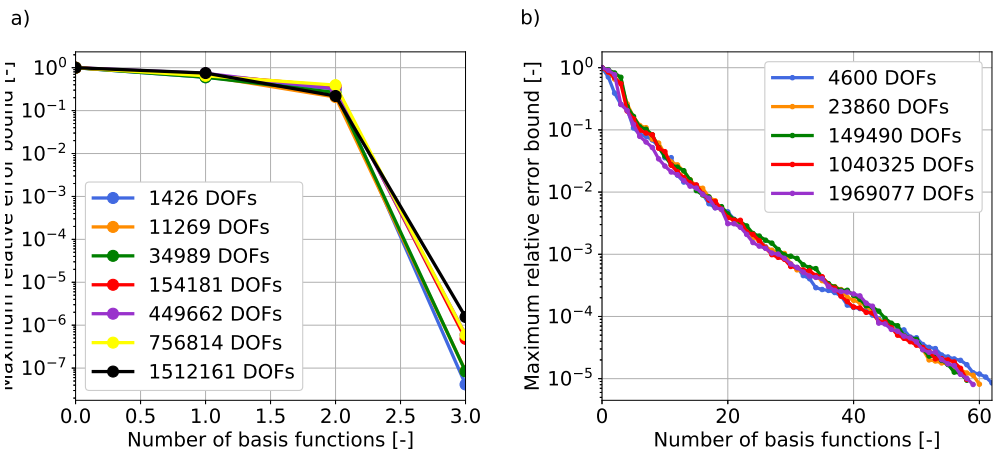


FIGURE 3.6: Convergence rate of the construction of the RB space for the a) Base model and b) Fault model for varying degrees of freedom and a relative error tolerance of 10^{-5} and solver tolerances of a) 10^{-8} and b) 10^{-9}.

functions. We considered the Base and Fault model for varying numbers of degrees of freedom again. In the case of the Base model, the observations are two-fold: First,

we generally have lower error bounds for smaller model sizes; and second, adding the first and second basis functions has a much smaller impact on the error bound than adding the third basis function.

In case of the Fault model, we cannot observe generally higher error bounds for the larger models. Instead, no specific pattern is observable. For the Fault model, we obtain a varying number of basis functions in order to reach our predefined tolerance. The random selection of training parameters and not the model size causes the variation. As for the Base model, we see the same behavior in the convergence rate independent of the degrees of freedom.

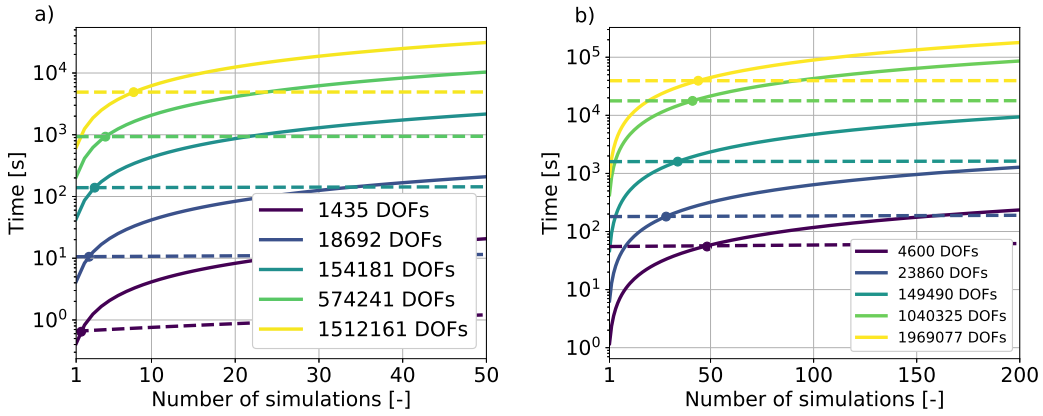


FIGURE 3.7: Comparison of the runtimes for the FE approach (solid lines) and the combined offline and online times of the RB approach (dashed lines) for varying degrees of freedom for a) the Base model and b) the Fault model over 50 and 200 simulations, respectively.

In Fig. 3.7 we show the runtimes (including the offline phase) for both the FE and RB simulations considering several simulations. For these results, we only made one simulation and extrapolated the runtime for the other simulations. Thus, we do not account for performance fluctuations of the cluster.

For the Base model, already two to seven simulations are enough for the RB method (with the cost of offline and online stages combined) to be more efficient than the FE method. For the Fault model, more basis functions are necessary; they vary between 58 and 62 (Fig. 3.8). Hence, the offline phase of the Fault model is more time-consuming than the one of the Base model since every new basis function requires an FE solve.

Figure 3.9 shows that for the Base model, 1,426 degrees of freedom are already enough to reach convergence, whereas, for the Fault model, we reach convergence after 1,040,325 degrees of freedom. For the latter model, we need a much finer mesh because the simulation results are highly affected by the two faults. The degrees of freedom that are necessary to reach convergence could be reduced by, for instance, adaptive mesh refinement methods.

3.2.2 Perth Basin

After the demonstration of the benefits of the RB method for geothermal benchmark problems, we shift the focus to real-case applications. Therefore, we first investigate the speed-up of the simplified Perth Basin model. The FE solution of the Perth Basin model requires 160.24 s, whereas the RB solution takes only 0.21 s. Hence, the speed-up of the Perth Basin model is 752.3 for a solver tolerance of 10^{-8}.

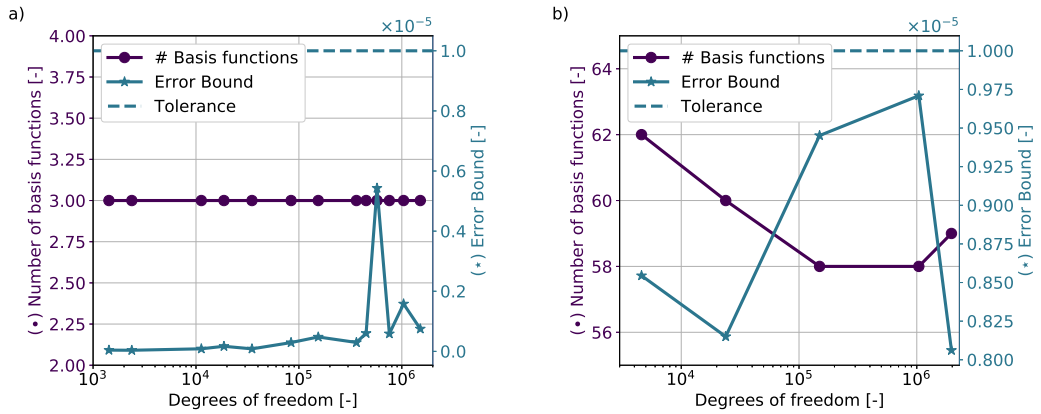


FIGURE 3.8: The evolution of the maximum relative error bound at the final basis function, and the number of basis functions required for reaching a relative tolerance of 10^{-5} for the a) Base model and b) Fault model.

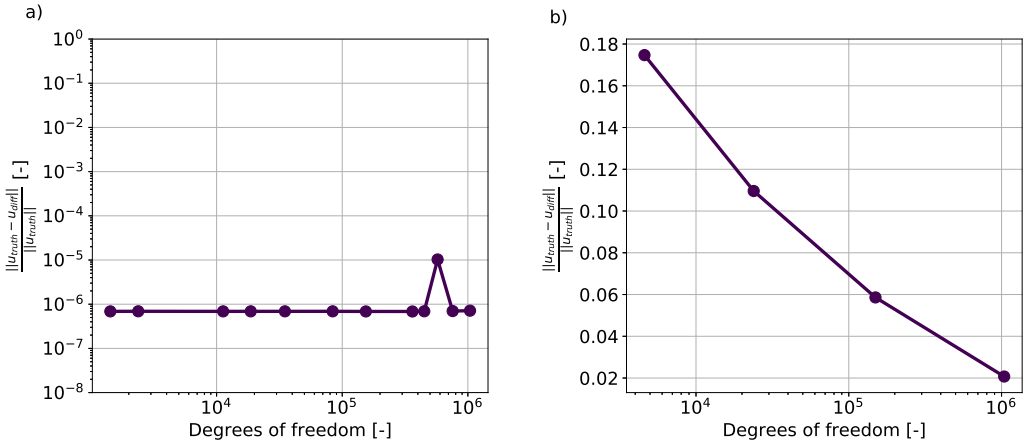


FIGURE 3.9: H_0^1 norm for the a) Base model and b) Fault model for varying degrees of freedom and a relative error tolerance of 10^{-5}.

Analog to the benchmark problems, we evaluate the RB simulation both qualitatively and quantitatively. For the qualitative control, we focus on the temperature distribution of the Perth Basin. This distribution (Fig. 3.10) resembles the one from the Base model, with the differences that the isolines are slightly curved towards the center of the basin. The temperature gradient has its minimal nondimensional value of 0.0 (corresponding to 22.0 °C) at the top of the model and its maximum value of 7.7 (corresponding to 191.4 °C in the dimensional model) at the bottom of the model. Hence, the temperature distribution is following our expectations. Note that we are again only plotting the results of the reduced basis solution since due to the small errors (Fig. 3.11), no differences are visible in this representation.

Now, we evaluate the quantitative performance of the RB method. Therefore, we focus on the error tolerance. We have a different error tolerance for the Perth Basin than for the Base and Fault model in order to demonstrate how this parameter is adjustable to measurement data. We choose the tolerance such that the obtained simulations are more accurate than the measurements to ensure that the reduction itself introduces no additional errors. For the Perth Basin, we reach the tolerance

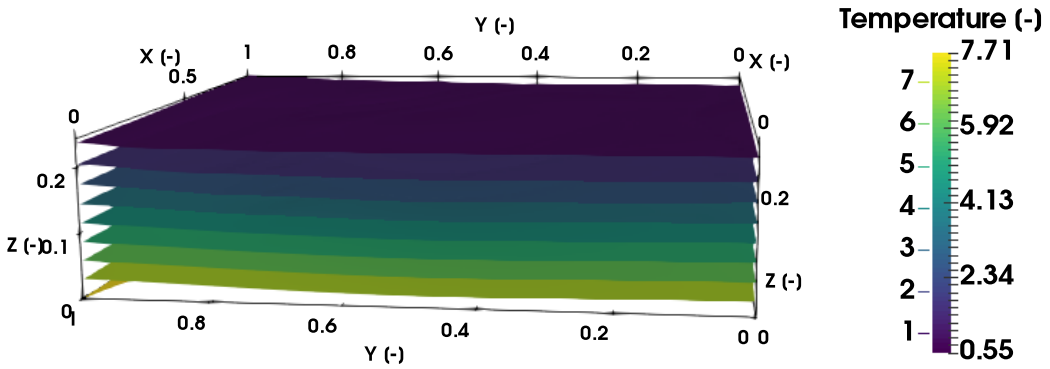


FIGURE 3.10: Temperature distribution for the Perth Basin model. Note that both the mesh dimensions and the temperature have been normalized for numerical reasons and are hence dimensionless. The model has a reference temperature of 22 °C, a reference length of 70,000 m and solver tolerances of 10^{-8}.

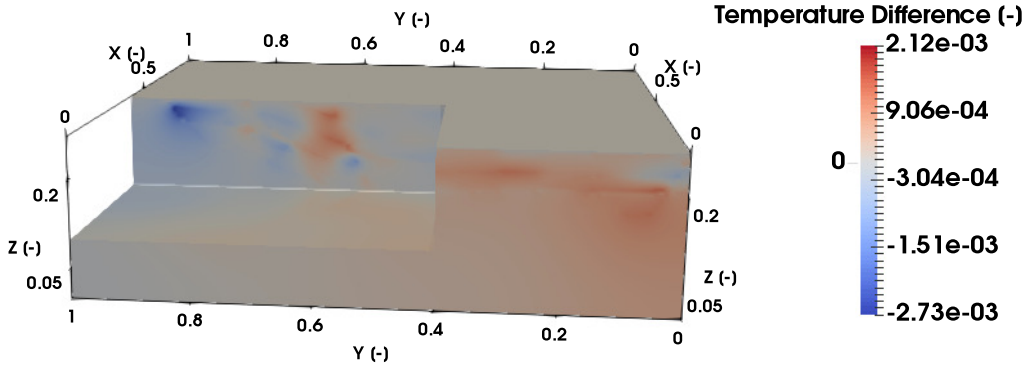


FIGURE 3.11: Difference plot between the full finite element and the reduced basis solution for the Perth Basin model. The relative error tolerance was set to 10^{-3} and the solver tolerances to 10^{-8}. The L2-norm of the FE solution is $5.559 \cdot 10^{4}$.

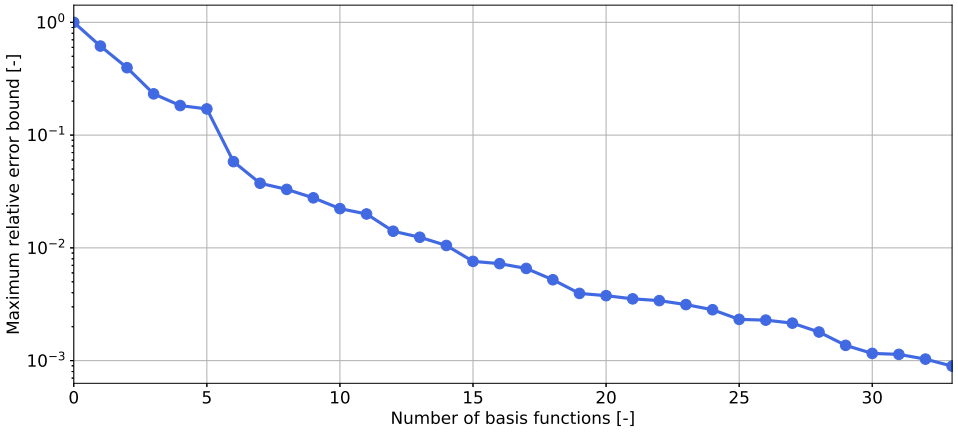


FIGURE 3.12: Convergence rate of the greedy construction of the RB space for the Perth Basin model for a relative error tolerance of 10^{-3} and solver tolerances of 10^{-8} over a set of 100 training samples.

of 10^{-3}. Analog to the Base and Fault model, we observe the largest errors at the layer interfaces, caused by the abrupt change of the parameter value. The maximum relative error (Fig. 3.12) decreases similarly to the Fault model.

Finally, we present the pay-off study for the Perth Basin. This analysis shows how many simulations we require for being more efficient with the RB than the FE method. Considering that the offline stage of the Perth Basin takes 12,719.78 s, we need 80 simulations for the pay-off.

3.3 Discussion

As for the Results Section, we first discuss the results of applying the RB method to geothermal problems for the Base and Fault model and afterwards for the Perth Basin model.

3.3.1 Base and Fault Model

We observe in Fig. 3.3 an increase in the speed-up for both the Base and the Fault model with increasing degrees of freedom. An explanation of this is found by having a closer look at the dimensionality changes of the FE and RB space (Fig. 3.8). The dimension increase of the RB space is much smaller than that of the FE space. It is because although the dimension of the FE space has increased, the complexity of the parameter dependence for the two problems remains more or less the same. In Fig. 3.8, we observe the different dimensionality. There, we see that for the RB method, three basis functions suffice to represent the parameter-dependence of the problem. The RB method thus has great potential for large-scale models which have an underlying low-dimensional parameter dependence. We demonstrate this for the Perth Basin model in the next section. This might be especially beneficial for geoscientific applications, which are usually high-dimensional such as seismic applications (Herrmann et al., 2012), stochastic problems (Curtis and Lomax, 2001) and data assimilation problems (van Leeuwen, 2010).

In Fig. 3.3 we can also see that we do not always get a higher speed-up for more degrees of freedom. The problem is that the online time requires, depending on the model size, between 9.7 ms and 578.2 ms. Fluctuations in the cluster performance due to different workloads have a significant influence because of these small runtimes. Thus, we always observe a speed-up, but the increase in speed-up might be smaller than expected.

In order to explain the decreased speed-ups for the Fault model, we have to look at the runtimes for the FE simulations. If we take, for instance, the Base model with 2,378 degrees of freedom and the Fault model with 4,600 degrees of freedom, we observe an increase in the runtime from 0.485 s to 1.106 s (an increase of 128 %). For the online times, we observe an increase from 0.097 s to 0.0346 s (257 %), which results in the smaller speed-up. The reason why the online time has increased by this amount is due to the increase in the number of basis functions from 3 to 62 (Fig. 3.8).

Furthermore, note that we defined three parameters for the Base model and five parameters for the Fault model, although we consider equal thermal conductivities in layer 1 and 3 and both faults. Thus, two parameters in the case of the Base model and three parameters for the Fault model would be sufficient to describe our problem. Using three instead of five parameters would lead to a reduced number of basis functions, which would yield a faster offline and online phase. As previously mentioned, we have chosen a higher number of parameters for obtaining a more flexible model.

In the case of the Base model, we reach an error tolerance that is far below our threshold due to the significant error reduction introduced by the third basis function (Fig. 3.6). The simplicity of the model, three layers with varying thermal conductivities, causes this. For this problem, we require three basis functions to capture the variability. More complex models would naturally require more basis functions. The reduction in the error is even larger than the defined tolerance for the third basis function since we can now fully describe the model. In contrast, for the first and second basis functions, crucial information is missing to describe the model entirely. The Fault model does not show this behavior since it is more complex, and the reduced model does not yet fully capture the entire complexity.

The accuracy loss is, therefore, far below the typical measurement error of a geothermal temperature measurement. We do not introduce additional error sources through the RB method. In this chapter, we have chosen a lower tolerance value than necessary in order to demonstrate the precision of the RB method. If desired, even more precise results can be obtained by further lowering the tolerance. However, we should take into account that a higher precision results in a higher number of basis functions which in turn decreases the speed-up and increases the time required for the basis construction. For standard temperature simulations, a relative tolerance of 10^{-3} is sufficient. With this tolerance, we guarantee that the introduced error through the approximation is below our measurement error.

We observed generally increasing error bounds with increasing model sizes for the Base and Fault model. This indicates that if we would continue increasing our model size, we would, at some point, require additional basis functions for a good representation of the FE space. Furthermore, take into account that the general trends in the convergence rate do not change for different model sizes.

A major shortcoming of the RB method is the expensive and time-consuming offline phase, where the construction of the reduced basis takes place. However, although it is time-consuming, it only has to be performed once. For that reason, the method is more efficient than the FE method when we require numerous simulations, as shown in Fig. 3.7. Again, we did not account for performance fluctuations of the cluster. Nonetheless, we get a realistic representation of the simulations required for the RB method to be beneficial over the FE method. This is the reason because the fluctuations greatly influence only the online time. As mentioned previously, the online times are extremely small, and therefore the fluctuation effects will be barely visible for these kinds of analyses. To further demonstrate this, we show during the performance test, the same analysis performed with varying degrees of freedom simulated at different times and hence with different workloads of the cluster.

In Fig. 3.7 the intersection between the dashed lines (RB) and the solid lines (FE) denotes the number of simulations required for the RB method to be more efficient than the FE method. The position of the intersection is for both models not dependent on the degrees of freedom. For the Base model, it is caused by the mentioned fluctuations of the compute cluster. This is also one of the reasons for the variability of the intersection position for the Fault model. The other reason is that we chose our training parameters randomly. Hence, with a specifically tailored training set, the number of required basis functions for each realization would be lower. Due to the randomness, the difference to this optimal set varies for each of these realizations, and therefore the position of the intersection changes.

Regarding the Base model, three basis functions have been enough for describing the model. Due to the low number of basis functions, the construction of the basis is relatively fast, and therefore the RB method pays off rather quickly. Furthermore, the complexity of the offline phase depends on the dimension of the FE problem.

The increase in the online time with increasing degrees of freedom is negligible in comparison to the FE runtimes. Accordingly, no relationship between the model size and the threshold at which the RB method is more efficient than the FE method is observable. Note that, as an example, for a model with 1,512,161 degrees of freedom, 50 FE simulations would require approximately 8 hours, whereas the same number of simulations using the RB method (including the offline time) are approximately six times faster. In case the offline stage has already been performed, the simulation would require approximately 0.1 % of the FE compute time.

The offline phase of the Fault model took more computation time than the one for the Base model because more basis functions are required to represent the model. Consequently, it takes more simulation runs for the RB method to be more efficient than the FE method. In our example, we required between 29 and 48 simulations. Taking, for instance, 200 runs of the model with 1,969,077 degrees of freedom into account, we would need nearly 50 hours with the FE approach and are 4.5 times faster with the RB method (including the offline time). The online stage for 200 simulations needs less than 0.1 % of the FE compute time. Furthermore, the number of basis functions is highly model-specific. Consider the Fault model and apply the boundary conditions to the left and right side of the model. Then, we only need around 20 basis functions, which decreases the offline phase and hence results in an earlier pay-off.

This demonstrates that the RB method is not only useful for large models. It is also helpful for small scale models if a very high number of simulations is necessary. One should only keep in mind that the smaller the model, the more simulation runs are generally required before it pays off to use the RB instead of the FE method. This pay-off behavior is also dependent on the complexity of the model. The Base model showed a similar number of simulations required for the RB method to be more efficient than the FE method, independent of the model size. The reason for this is that we needed only three basis functions. Hence, the construction of the basis was relatively cheap, and we obtained similar computational savings for all model sizes.

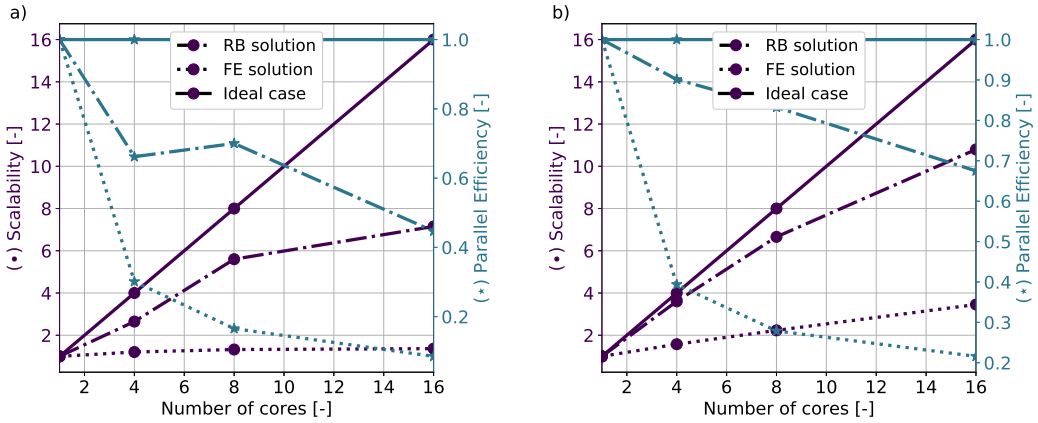


FIGURE 3.13: Performance analysis for the a) Base model (1,512,161 degrees of freedom) and b) Fault model (1,969,077 degrees of freedom). Shown are the strong scalability (purple lines) and the parallel efficiency (blue lines). The performance test has been done on Intel Xeon E5-2680 v3 Haswell cores (2.5 GHz).

Fig. 3.13 shows another aspect. Here we compare the performance for the RB and the standard FE method. In the case of the RB method, we obtain a strong scalability

that varies, in case of the Base model, from 1.00 to 7.15 for 1 and 16 cores, respectively. The FE method has, for the same number of cores, a strong scalability changing between 1.00 and 1.37. For comparison, the ideal values for the strong scalability vary between 1.00 and 16.00. Consequently, the RB method shows much stronger scalability than the FE method. Similarly, for the parallel efficiency, we get values between 1.00 and 0.45 for the RB method and from 1.00 to 0.09 for the FE method. This means that the RB method shows at all times better performance than the FE method. Also, note that the difference in performance is larger for a smaller number of cores.

For geothermal conduction problems, we do not require the full capabilities of a standard HPC cluster due to the simplicity of the problem. Hence, we still have computing resources left that can be used to compensate for the additional time of the basis construction by using more cores. This also means the problem is not ideal for scaling tests because few cores are sufficient to compute the problem. What we want to emphasize is that, for many-query problems, the RB method is more efficient than the FE method. A trend that is also expected, for other problems that are more complex in terms of model geometry and the underlying physics. Using the provided MOOSE implementation, no additional work has to be invested in parallelizing the problem since MOOSE offers a built-in parallelization that hides all MPI processes from the user.

Fig. 3.13b shows a first indication that this is true. Here, we did the same performance test for the Fault model, and we observe a slightly better performance of the RB method than for the Base model for up to 16 cores. The strong scalability increases from 7.15 to 10.80, and the parallel efficiency from 0.45 to 0.67, although the model has only 23 % more degrees of freedom.

Furthermore, the size of the training sample has, depending on the error bound used, an influence on the offline times. We used the dual residual norm to calculate the Riesz representor. The error bound applied here is thus relatively insensitive to the size of the training set. So, a general way to reduce the offline time is to choose a training set that is as small as possible.

However, one needs to be careful that the training set is not too small. This would result in a significant accuracy loss because the domain is under-sampled. We now consider the case that we define a too small set of training samples. The basis function is calculated using the parameter values within the training set by determining the maximum error overall training samples. In our examples, the package would provide us with an error bound. However, this is not representative for the "truth" problem since our training samples do not represent the full complexity of the "truth" model. Furthermore, one needs to ensure that we have always more training samples than basis functions. Keep in mind that these numbers are problem specific. In our example, it is, therefore, more important to have a training set that is large enough in order to avoid a recalculation than choosing a smaller training set.

For the Base model, a small training set is enough due to the low parameter complexity of the model. We performed simulations with a training size of 5, 100, and 1000 parameters using the Base model with 756,814 degrees of freedom. For this model, we got runtimes of 1,284.77 s (5 training parameters), 1,294.65 s (100 training parameters), and 1,285.9 s (1,000 training parameters). We would expect a slight increase in the computation times for more training parameters. However, the fluctuations due to different workloads of the cluster are in the same range as the increase in computation time. Hence, they mask these effects. The Fault model needs more training samples for our simulations because of the higher parameter complexity of the model.

We explained in Chapter 2 that the number of basis functions for the RB space is, among other things, determined by the user-defined error tolerance. This is advantageous for geoscientific applications since the measurement error we obtain widely varies for different methods and settings. By defining the error tolerance, we adjust to this variety. Having a slower but very accurate approximation of the FE space while having high measurement errors, and vice versa is not beneficial at all. This means that we can take the measurement error minus a safety factor for the error tolerance. So, we get an optimal speed-up (as few basis functions as possible) without losing accuracy.

As previously mentioned, we consider nondimensional parameters only. This yields a better solver performance since the residuals are without dimensions. In the case of this geothermal example where the scales of the parameters are similar, this does not play a major role. However, if we additionally take, for instance, the pressure into account, it would have a significant influence since the scale differences between pressure and temperature are large. We use the nondimensional description, especially because of the RB method. The error bound we used in our Greedy algorithm is sensitive to the mesh dimensions. Consequently, a consideration of the length in its dimensional form results in a high error bound, which in turn leads to a premature termination of the Greedy algorithm. Thus, resulting in a lower approximation accuracy. This is especially true for geoscientific applications since our models generally have an extension of several hundreds of meters to several ten thousands of meters.

Another reason to use only nondimensional variables are the Dirichlet boundary conditions. As in many other mathematical approaches, inhomogeneous (i.e., nonzero), Dirichlet boundary conditions are problematic. In contrast to, for instance, Neumann and Robin (natural) boundary conditions, the Dirichlet boundary conditions are so-called forced or essential boundary conditions. That means they do not appear in the integral (weak) form of the PDE but are later on enforced on the system of equations. For the implementation of the RB method in libMesh, inhomogeneous Dirichlet boundary conditions yield inaccurate approximations due to the treatment of the affine decomposition. If we consider the nondimensional problem instead, such that the boundary is equal to zero (as done in this thesis), we can resolve the problem.

Keep in mind that the issue with the Dirichlet boundary conditions is not a specific problem of the RB method but appears for several approaches. Hence, it is a good idea to take advantage of the nondimensional form to achieve i) a better solver performance and reduce numerical problems and ii) to gain a better performance of the method itself.

3.3.2 Perth Basin

As previously mentioned, the Perth Basin model acts in this thesis as a transition model between the synthetic models and the real-case applications. Hence, we will only briefly evaluate the performance of the RB method for this specific model. A more detailed discussion of the RB method for real-case large-scale models, in the context of inverse problems and the benefits for Geosciences, will follow for the models of the Upper Rhine Graben and Brandenburg (see Chapter 4 - 6).

Before discussing the performance a brief note concerning the error tolerance: We have chosen in the real-case example a more realistic tolerance to demonstrate how the tolerance is adjustable to the measurement error.

Regarding the performance, the Perth Basin shows, as the Fault model, a decreased speed-up in comparison to the Base model, which is induced by the higher complexity

due to the variation of six instead of three parameters. We require at least 80 simulations to be more efficient with the RB method than the FE method. If we retake the example of 200 simulations, then we require for the FE method approximately 9 hours and the RB method requires slightly more than 0.1 % of the FE compute time.

Nonetheless, in all cases, the number of runs is far below the thousands to millions of runs usually required for stochastically inversions, and uncertainty quantification. Consequently, the gained time for these types of applications is expected to be very high.

Chapter 4

Global Sensitivity Analysis to Optimize Basin-Scale Conductive Model Calibrations – Insights of the Upper Rhine Graben

In this chapter, we illustrate the advantages of global sensitivity analyses for the optimization of model calibration. Therefore, we use the example of the Upper Rhine Graben (Freymark et al., 2017). The content of this chapter is presented in Degen et al. (2020a)

4.1 Introduction

Geothermal applications, like many other geoscientific applications, face the major challenge to construct representative models despite major sources of uncertainty. Naturally, the determined thermal parameters and the model geometry is not entirely correct. In order to compensate for these model errors, arising from, for instance, measurement errors, generalizations, and geometrical uncertainties (Wellmann and Caumon, 2018), we need model calibrations to obtain reliable and robust models.

Reliable and robust model calibrations are a challenge for deep geothermal systems because most temperature measurements are at a shallow depth. Only few deeper measurements are available for the model validation. Sensitivity analyses help in identifying relevant model parameters and reveal which parameters can be inferred from the data (Saltelli et al., 2004). We demonstrate in this thesis how the model calibration process significantly improves by incorporating global sensitivity analyses into the process. We further show why global sensitivity analyses are needed and why local sensitivity analyses are not sufficient for deep geothermal studies.

Global sensitivity analyses in Geosciences have been performed for hydrological studies (van Griensven et al., 2006; Tang et al., 2007; Cloke et al., 2008; Zhan et al., 2013; Baroni and Tarantola, 2014; Song et al., 2015), for volcanic source modeling (Cannavó, 2012), and for geothermal heat exchangers (Fernández et al., 2017). Furthermore, a comparison of local and global sensitivity analysis has been performed, for a hydrological model, in Wainwright et al. (2014). There is no comparison of local and global sensitivity analyses for basin-scale heat flow models available, as we present it in this thesis.

Naturally, the question arises why do we not use a local sensitivity analysis that requires only very few forward solves? Why do we require instead the computationally demanding global sensitivity analysis? We need global analyses because the local

ones fail in the context of deep geothermal applications. The reason is that local sensitivity analyses explore only in the vicinity of the initially guessed parameters. In our geothermal applications, the parameter ranges (i.e., thermal conductivities, and radiogenic heat productions) are usually associated with high uncertainties (Freymark et al., 2017; Lehmann et al., 1998; Vogt et al., 2010; Wagner and Clauser, 2005), which is the reason why we apply model calibrations. These high uncertainties quickly lead to wrong initial guesses of the various parameters bearing the high risk of exploring the wrong part of the parameter space with a local sensitivity analysis. For geothermal inverse problems, we will never be able to get a unique solution. Therefore, it is not only essential to determine the parameter set but also to reveal their correlations. Something that the local sensitivity analysis is not capable of.

As a case study, we use the Upper Rhine Graben. The Upper Rhine Graben is due to its increased geothermal gradient of high interest for deep geothermal explorations. We see this in, for instance, Soultz-sous-Forêts, Landau, and Bruchsal (Agemar et al., 2013, 2014; Illies, 1972; Pauwels et al., 1993; Tiefe Geothermie, 2007; Vidal et al., 2015). So far, only "trial-and-error" model calibrations, without taking into account the sensitivities of the various parameters, are available for the Upper Rhine Graben (Freymark et al., 2017). Automated model calibrations considering sensitivities, but not global sensitivity studies, have been presented by Wellmann and Reid (2014) for the Perth Basin (Australia), and without sensitivities in Fuchs and Balling (2016). Furthermore, local sensitivity studies are presented in Ebigbo et al. (2016).

That obtaining reliable temperature distributions is a major challenge is seen in the ongoing discussion (Agemar et al., 2013, 2014; Freymark et al., 2017, 2019; GeORG-Projektteam, 2013; Stober and Bucher, 2015) of determining the factors that control the temperature distribution within the Upper Rhine Graben. We are dependent on numerical models that are suitable representations of "reality" for a specific purpose. However, generating these models is a non-trivial process. We have to include all relevant physical processes, and the correct geometrical parameterization, factors that are associated with high uncertainties. Therefore, we require automated sensitivity-driven model calibrations to compensate for all possible error sources.

In this thesis, we present how one can use the model calibration as a reliable tool for identifying model errors. This is only possible by incorporating the global sensitivity results and using automated calibrations to avoid subjective calibrations and to take into account all possible correlations between the parameters. As we show, the local sensitivity analysis fails to reveal the relevant parameters. Therefore, it yields a non-robust model calibration, and can, hence, not be considered for the model error identification.

Global sensitivity studies are computationally very demanding due to the large number of forward simulations. Therefore, they have been prohibitive for basin-scale models, using the finite element method. To overcome this problem, we propose the usage of a physics-based machine learning approach, namely the reduced basis (RB) method (Hesthaven et al., 2016; Prud'homme et al., 2002). It reduces the computational costs of the Upper Rhine Graben conductive heat flow simulations by several orders of magnitude. At the same time, it still provides highly accurate estimates of temperature predictions.

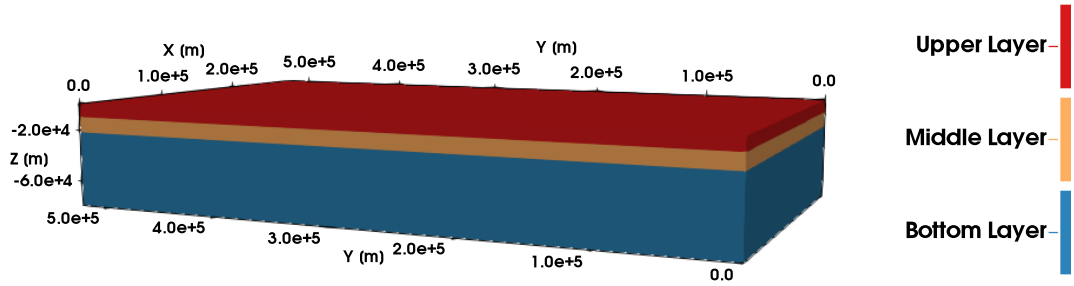


FIGURE 4.1: Simplified basin-scale model for illustrating the concepts of the methods presented in this chapter.

4.2 Materials and Methods

In the following, we briefly demonstrate the concepts of both local and global sensitivity analysis through a simplified geological basin-scale model. Afterwards, presentations of the physical model, the RB method, and the concepts of model calibration follow.

4.2.1 Sensitivity Analysis

Consider a function $u = f(x)$, where $x = (x_1, ..., x_N)$ are the model parameters and let $u^* = f(x^*)$ be the desired solution, then the sensitivity analysis aims at determining the influence of the model parameters x on the model $f(x)$ (Sobol, 2001; Wainwright et al., 2014). The evaluation of these sensitivities differs for the local and global sensitivity analysis, as discussed in the next sections. The sensitivity analysis is a natural preparation step for a model calibration (Ray et al., 2015). As we demonstrate in this thesis, the efficient reduction of the parameter state leads to more robust model calibration results. The sensitivity analysis can be subdivided into the local and global study (Sobol, 2001).

In the following, we will briefly describe the main difference between both analyses. We illustrate the concepts by using a simplified geological basin-scale model (Fig. 4.1). We use the same physical model as later on for the case study. The top boundary condition is set to 10 °C (corresponding to the annual average surface temperature in Germany), and the bottom boundary condition is set to the 1300 °C isotherm (Bayer et al., 1997). The model extends 292 km in the x-direction, 525 km in the y-direction and -80 km in the z-direction, resembling the dimension of the Upper Rhine Graben model (Freymark et al., 2017). The parameter values are taken from Freymark et al. (2017), and the top layer corresponds to the Cenozoic Volcanics, the middle layer to the Saxothuringian, and the bottom layer to the Lithospheric Mantle. We investigate the sensitivities of all thermal conductivities and the radiogenic heat production of the Saxothuringian. Note that we perform the simulation with the nondimensional equation. Hence, we have nondimensional temperatures between 0 and -1. Therefore, we do not overestimate the sensitivity of the deeper temperature measurements. That would be the case for the dimensional case because deeper measurements have higher temperatures and therefore, a possible higher difference to the calibration data.

Local Sensitivity Analysis

If we consider again the function $u = f(x)$, with $x = (x_1, ..., x_N)$ being the model parameters, and the desired solution $u^* = f(x^*)$, then the local sensitivity analysis

evaluates the sensitivity by taking $\left(\frac{\partial u}{\partial x_k}\right)_{x=x^*}$. Hence, the local sensitivity analysis investigates the influence of the various model parameters with respect to a pre-defined reference parameter set. For this reason, they focus on the sensitivity in the vicinity of the input parameters (Sobol, 2001; Wainwright et al., 2014). Hence, in case of the local sensitivity analysis, we allow a defined variation of, e.g., 1 %. The local sensitivity analysis does not consider parameter correlations. It assumes that the influence observed for each realization is solely arising from the changed model parameter itself. This assumption is not valid for general geoscientific applications.

In case of a geothermal application, we would, for instance, determine the influence of the different thermal conductivities with respect to a reference set, which would usually be our initial guess. That already shows the major issue with local sensitivity analyses: we have to define a reference. With this reference parameter set, we calculate the reference temperatures at our observation points, using our physical model. These reference temperatures are however not identical to our observed temperatures, due to model and measurement errors. To conclude, we have to use our "best" knowledge as the reference. If we consider the uncertainties related to geothermal applications this "best" knowledge likely does not represent the "true" parameter values. Hence, we face the high risk of failing with the local sensitivity analysis.

For our simple basin-scale model, we obtain the sensitivities presented in Fig. 4.2. As the reference thermal conducitivities, we take the calibrated thermal conductivity values from Freymark et al. (2017). We investigate the influence of the model parameters on the absolute misfit between the simulated and the reference temperature data. Using the observed temperatures instead of the reference temperatures leads to equal sensitivities for all parameters because these temperatures lay outside the 1 % variation range. For the comparison with the reference temperatures, we observe that the model is sensitive to all investigated parameters. For further analyses, that means that we have to consider all model parameters.

Global Sensitivity Analysis

For the global analysis, we use the Sobol sensitivity analysis with the Saltelli sampler, which is a variance-based sensitivity analysis (Sobol, 2001; Saltelli, 2002; Saltelli et al., 2010; Wainwright et al., 2014). It does not set $x = x^*$, but evaluates instead the entire model $f(x)$. Assuming a square-integrable function with orthogonal members, we can decompose it into the following form (Sobol, 2001):

$$\int f^2(x)\mathrm{d}x - f_0^2 = \sum_{s=1}^{n} \sum_{i_1<\cdots<i_s}^{n} \int f_{i_1\cdots i_s}^2 \mathrm{d}x_{i_1} \cdots \mathrm{d}x_{i_s}, \tag{4.1}$$

where f_0 is a constant, and f_i a function. Then the variances D are defined as (Sobol, 2001):

$$D = \int f^2 \mathrm{d}x - f_0^2, \qquad D_{i_1\cdots i_s} = \int f_{i_1\cdots i_s}^2 \mathrm{d}x_{i_1} \cdots \mathrm{d}x_{i_s}. \tag{4.2}$$

The global sensitivity indices S can be derived from the variances by taking the ratio (Sobol, 2001):

$$S_{i_1\cdots i_s} = \frac{D_{i_1\cdots i_s}}{D}. \tag{4.3}$$

We use the global sensitivity analyses to investigate the influence of the model parameters on the total absolute misfit between the simulated and the observed temperature data. In contrast to the local sensitivity analysis, the global one does not require a

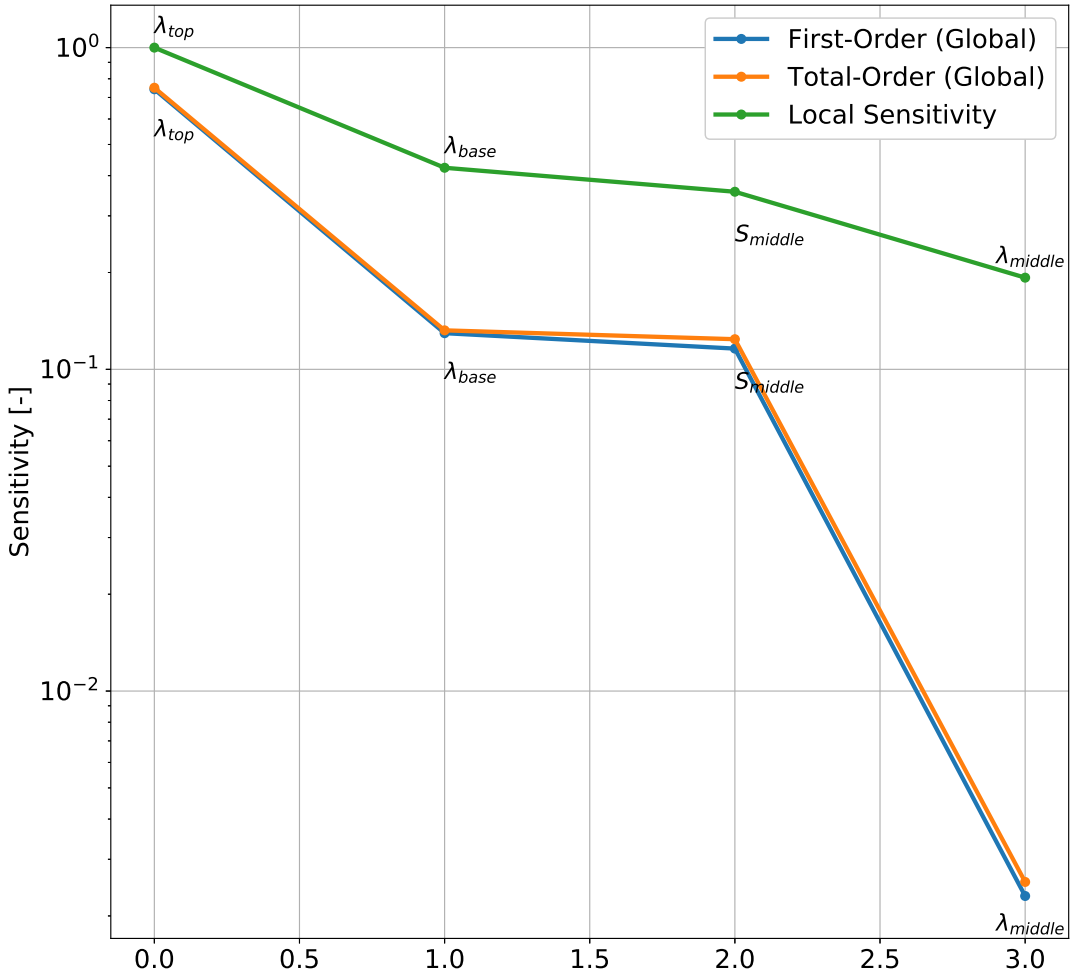


FIGURE 4.2: We present a comparison of the local and global sensitivity analysis for the benchmark problem. The local sensitivity indices are denoted in green, the first-order contributions (global sensitivity analysis) in blue, and the total-order contributions (global sensitivity analysis) in orange.

reference and investigates the influences of all parameters within the pre-defined parameter ranges (Sobol, 2001). Hence, we are no longer restricted to small parameter variations. Furthermore, we no longer need to know the parameter distribution in advance.

In contrast to the local sensitivity analysis, the global one considers correlations. They can be investigated by looking at the Sobol sensitivity indices. The first-order index represents the influence on the model from the model parameter itself. The second-order index describes the influence of the correlation between two parameters. Further, higher-order indices are available to investigate the influence of the correlation between more than two parameters (Sobol, 2001). For further details regarding the definition of the sensitivity indices, we refer to Sobol (2001), and for further information regarding the sampling procedure, we refer to Saltelli (2002), and Saltelli et al. (2010).

Coming back to our geothermal examples, this means that we no longer need to input our possibly wrong "best" knowledge of the thermal properties. We performed the global sensitivity analysis for the simple basin-scale model (Fig. 4.2), where we use 10,000 realizations per parameter to reduce the statistical error to eliminate negative sensitivities. The analysis is executed using the Python library SALib (Herman and Usher, 2017). The results significantly differ from the ones of the local sensitivity analysis. We still observe the dominant influence form the thermal conductivity of the top layer; however, we see that the model is insensitive to the thermal conductivity of the middle layer. So, the local sensitivity analysis overestimates the influence of this model parameter. Furthermore, the local analysis overestimates the influence of both the thermal conductivity of the bottom layer and the radiogenic heat production of the middle layer. So, in contrast to the local sensitivity analysis, we can reduce the dimension of the parameter space from four to three.

Additionally, to the total sensitivities, we get the information about the parameter correlations. For the presented model, we observe nearly exclusive first-order contributions. Hence, the parameter correlations in the presented model are small. The local sensitivity analysis already fails for this model, although it is ideally suited for it since it has negligible parameter correlations. However, the global sensitivity analysis has the issue that it requires many forward simulations. In Section 4.3, we describe how we overcome this disadvantage.

4.2.2 Forward Simulation

For the forward simulations of the temperatures, we consider a steady-state geothermal heat conduction problem with a radiogenic heat production after Bayer et al. (1997).

$$-\lambda \nabla^2 T + S = 0, \tag{4.4}$$

where λ is the thermal conductivity, T the temperature and S the radiogenic heat production. For efficiency reasons, we are taking into account nondimensional parameters and variables only, which leads to Eq. 4.5:

$$-\frac{\lambda}{\lambda_{\text{ref}}\, S_{\text{ref}}}\, \frac{\nabla^2}{l_{\text{ref}}^2}\, \left(\frac{T - T_{\text{ref}}}{T_{\text{ref}}}\right) + \frac{S\, l_{\text{ref}}^2}{S_{\text{ref}}\, T_{\text{ref}}\, \lambda_{\text{ref}}} = 0 \tag{4.5}$$

Here, λ_{ref} is the reference thermal conductivity, T_{ref} the reference temperature, S_{ref} the reference radiogenic heat production, and l_{ref} the reference length.

4.2.3 Model Calibration

We consider the trust region reflective (TRF) method from SciPy (Branch et al., 1999; Jones et al., 2014) as our calibration method. Take the following problem into account: $\min_{x \in \mathbb{R}^n} \{f(x) : \quad l \leq x \leq u\}$, where we want to minimize the function $f(x)$ with x being our model parameter. Furthermore, the model parameters are bounded below by the lower bound l, and above by the upper bound u. The method defines a trust region N around the current "best" solution to improve the convergence rate. In standard trust region methods, the quadratic model $q(s)$ is assumed to approximate the original objective function inside the trust region. We derive the quadratic model by taking the first two steps of the Taylor expansion of f at x. The trial step s is computed by minimizing over the trust region: $\min_s \{q(s), s \in N\}$. The current point is updated to $x+s$ if $f(x+s) < f(x)$ and the trust region is updated. If we do not fulfill this condition, the current point stays the same. Then, we decrease the trust region and repeat the computation of the trial step. The trust region optimization problem is then expressed as: $\min \left\{ \frac{1}{2} s^T H s + s^T g \text{ such that } \|Ds\| \leq \Delta \right\}$. Here, g is the gradient, H the hessian matrix, D a diagonal scaling matrix, and Δ a positive scalar. We repeat these steps until we reach convergence. Minimizing a quadratic function – instead of the original, possibly non-linear, objective function – results in the reduction in computation time. More details about the method can be found in Branch et al. (1999).

Furthermore, we take the results of the sensitivity analysis into account, meaning that we calibrate only those parameters to which the model is sensitive.

4.3 Case Study of the Upper Rhine Graben

In the next section, we use the Upper Rhine Graben, presented in Freymark et al. (2017), as a case study to illustrate the benefits of a combined global sensitivity study and a model calibration. This is used to optimize the temperature predictions of real-case basin-scale conductive heat flow models.

4.3.1 Upper Rhine Graben

This thesis uses the model of the Upper Rhine Graben presented in Freymark et al. (2017), and in this section, we give an introduction into its numerical aspects.

The model extends 292 km in the x-direction, 525 km in the y-direction, and in the vertical direction down to the lithosphere-asthenosphere boundary (LAB). It consists of 24 geological layers and is discretized using deformed eight-noded prisms. The horizontal resolution is one km, and the vertical resolution corresponds to the layer thickness yielding 3,852,950 degrees of freedom. At the top and the bottom of the model, we apply Dirichlet boundary conditions, and we set all other boundaries to no-flow boundaries. For the upper boundary condition, we use the annual average surface temperatures, as presented in Freymark et al. (2017). In contrast, the lower boundary condition is set to 1300 °C isotherm at the LAB (Bayer et al., 1997).

We constructed two reduced models. The first one considers only thermal conductivities. It consists of 12 different parameters since we combined the layers with equal thermal conductivities into one. The second one contains the thermal conductivity of the Cenozoic Rift Sediments and the radiogenic heat productions from the Upper Crust, resulting in eight parameters. In both models, we define the allowed variations of the parameters in accordance with the ranges given in Freymark et al. (2017). If

no range is defined in Freymark et al. (2017), we allow a variation of $\pm$ 50 % from the initial value.

For the nondimensional representation of the Upper Rhine Graben model, we set the reference thermal conductivity λ_{ref} to the maximum thermal conductivity of 6.0 W m^{-1} K^{-1}. Analogously, we choose the maximum temperature of 1300 °C as the reference temperature T_{ref} and the maximum radiogenic heat production of 3.0 μW m^3 as the reference radiogenic heat production S_{ref}. The reference length l_{ref} corresponds to the maximum y-extent of the mesh (525,000 m).

We are using the RB solution as a surrogate model for the FE method. In order to apply the method, we need to decompose the PDE into its parameter-dependent and -independent part. This leads to the following decomposition of the bilinear form a for both reduced models:

$$a(w, v; \lambda) = -\sum_{q=0}^{n} \lambda_q \int_{\Omega} \nabla w \, \nabla v \, d\Omega, \qquad \forall v, w \in X, \; \forall \lambda \in \mathcal{D}. \tag{4.6}$$

Here, w is the trial function, v the test function, the index "q" denotes the number of the training parameters (see also Tab. 4.1 – 4.3), X the function space ($H_0^1(\Omega) \subset X \subset H_1(\Omega)$), Ω the spatial domain in $\mathbb{R}^3$, and $\mathcal{D}$ the parameter domain in $\mathbb{R}^{12}$. For the reduced model with varying thermal conductivities n is equal to 11 and for the one with varying thermal conductivities and radiogenic heat production n is equal to one.

The decomposed linear form f has the following form for the first reduced model:

$$\begin{aligned} f(v; \lambda) = &-\sum_{q=0}^{11} \lambda_q \int_{\Gamma} \nabla v \; g(x, y, z) \; d\Gamma + \\ &\int_{\Gamma} \nabla v \; S \; d\Gamma, \qquad \forall v \in X, \; \forall \lambda \in \mathcal{D}, \\ \text{with } \mathrm{g(x, y, z)} = &\, T_{top} \frac{h(x, y, z) - z_{bottom}(x, y)}{d(x, y)}, \end{aligned} \tag{4.7}$$

where Γ is the boundary in $\mathbb{R}^3$, $g(x, y, z)$ the lifting function, T_{top} the temperature at the top of the model, $h(x, y, z)$ the location in the model, $z_{bottom}(x, y)$ the depth of the bottom surface, and $d(x, y)$ the distance between the bottom and top surface.

The decomposition of the linear form of the second reduced model is similar:

$$\begin{aligned} f(v; \lambda) = &-\int_{\Gamma} \lambda_{(1)} \nabla v \; g(x, y, z) \; d\Gamma - \; \lambda_1 \int_{\Gamma} \nabla v \; g(x, y, z) + \\ &d\Gamma \int_{\Gamma} \nabla v \; S_{(14-20)} \; d\Gamma + \sum_{q=14}^{20} S_q \int_{\Gamma} \nabla v \; d\Gamma, \quad \forall v \in X, \; \forall \lambda \in \mathcal{D}, \\ \text{with } \mathrm{g(x, y, z)} = &\, T_{top} \frac{h(x, y, z) - z_{bottom}(x, y)}{d(x, y)}. \end{aligned} \tag{4.8}$$

Here, $\lambda_{(1)}$ are all thermal conductivities except the thermal conductivity of the Cenozoic Rift Sediment, λ_1 the thermal conductivity of the Cenozoic Rift Sediments, S_{14-20} the radiogenic heat productions of the upper crust layers, and $S_{(14-20)}$ all remaining radiogenic heat productions.

4.3.2 Data and Weighting

Data are very unequally distributed throughout the spatial domain. In order to compensate for this, we consider weights. For the calculation of the weights, we consider

two ways: First, we calculate them automatically via a distance matrix provided by SciPy, and second via user-input. The user-defined weighting is obtained by splitting the data into three regions (Fig. 4.3). We see that in region 3, northern part of the Upper Rhine Graben, we have only very few data points. To compensate for the data sparsity in this region, we apply a weight of 10 to the misfit. Many observation data are available in the Hesse area, while significantly fewer data are available for the Upper Rhine Graben. In order to compensate for this unequal data distribution, we apply a weight of 0.1 to region 1 and of 1.0 to region 2.

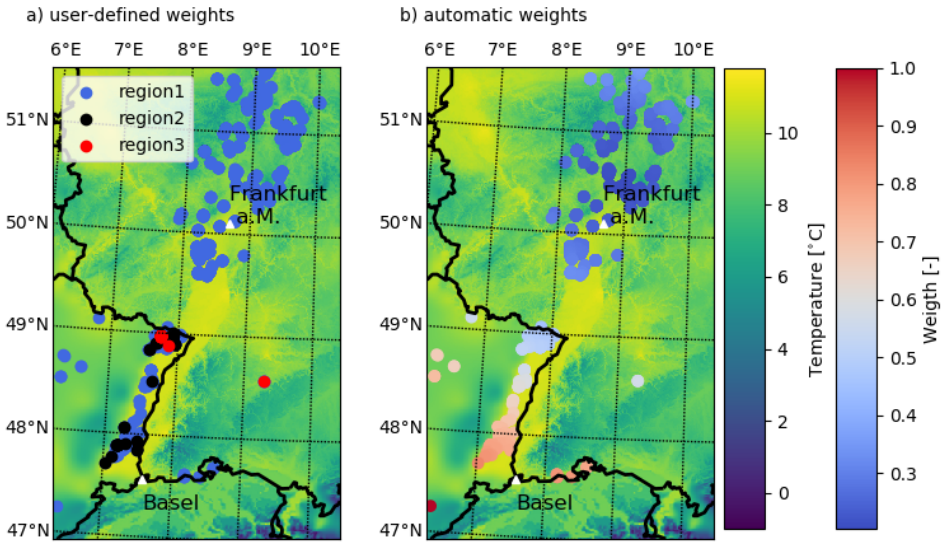


FIGURE 4.3: Comparison of the distribution of the weights for the automated model calibration. On the left side is the distribution derived from expert knowledge, and on the right side the weighting scheme using the distance matrix.

4.3.3 Model Calibration without Sensitivity Analysis

For all calibrations, we only vary the thermal conductivities since the influence of the radiogenic heat production is negligible in comparison to the thermal conductivity. We demonstrate this in the following sections. As the loss function, we use the smooth approximation of the L1 absolute value loss with a value of 0.1 for the soft margin between inlier and outlier residuals. We set the tolerances for the termination to 10^{-5} for changes of the cost function, 10^{-8} for the norm of the gradient, and 10^{-5} for changes of the independent parameters. We perform the calibration three times, once without weighing the misfit of observed and simulated data and once with applying the different weighting methods to the misfit.

If we do not incorporate any sensitivity result into the model calibration, we have to calibrate our model with all thermal conductivities consequently. Tab. 4.1 summarizes the parameter values for all considered scenarios. The result from the model calibration without weights and the model calibration with automatic weights are very similar. Significant differences are observed for the model calibration with user-defined weights. The major differences occur for the Cenozoic Volcanics and Rift Sediments and the Lithospheric Mantle. The Cenozoic Volcanics and Rift Sediments show a decrease in thermal conductivity to the initial values for the model calibration with user-defined weights, and the other two show an increase. The decrease for the Cenozoic Rift Sediments in case of the model calibration with user-defined

weights is with 0.4 W m^{-1} K^{-1} enormous, and also the increase in thermal conductivity of 0.9 W m^{-1} K^{-1} for the Cenozoic Volcanics for the model calibration without and with automatic weights. Also, for the Cenozoic Folded Molasse/Cenozoic Foreland Molasse/Buntsandstein/Jura Mountains/Odenwald, the Dogger/Lias/Keuper/Saxothuringian/Moldanubian, the Muschelkalk, the Zechstein/Upper Crust - Alps, and the Mid-German Crystalline High, significant differences in the calibrated thermal conductivities are observed. The model calibrations without weights and with automatic weights follow the same trend, whereas the model calibration with user-defined weights mostly follows the reversed trend. The other layers show similar calibrated model parameters for all model calibration versions.

Furthermore, for all three versions of the model calibration, many of the calibrated model parameters reach the upper or lower bound of their pre-defined variation range, as indicated in Tab. 4.1.

TABLE 4.1: The thermal properties before and after the calibration for the Upper Rhine Graben without considering the results of a sensitivity study. The initial thermal properties are taken from Freymark et al. (2017). Note that UC stands for Upper Crust, ub for upper bound, and lb for lower bound.

ID	Layer	Training Parameter	Initial Bulk Thermal Conductivity [W m^{-1} K^{-1}]	Calibrated Thermal Conductivity [W m^{-1} K^{-1}]			Radiogenic Heat Production [$\mu W m^{-3}$]
				user-defined weights	automatic weights	no weights	
CV	Cenozoic Volcanics	0	1.8	1.6	2.7 (ub)	2.7 (ub)	0.2
CRS	Cenozoic Rift Sediments	1	1.2	0.8	1.4	1.4	1.0
TS	Tertiary salt	2	6.0	5.5 (lb)	5.5 (lb)	5.5 (lb)	0.0
A	Alps	3	2.2	1.3	1.4	1.5	0.3
CM	Cenozoic Folded Molasse	4	3.0	4.5 (ub)	3.4	3.5	1.0

CFM	Cenozoic Foreland Molasse	4	3.0	4.5 (ub)	3.4	3.5	1.0
C	Cretaceous	3	2.2	1.3	1.4	1.5	0.5
M	Malm	5	2.7	2.8 (ub)	2.8 (ub)	2.8 (ub)	1.4
DLK	Dogger/Lias-/Keuper	6	2.5	2.4 (lb)	3.2	3.5 (ub)	1.6
MK	Muschelkalk	7	2.0	3.0 (ub)	2.1	1.5	1.2
B	Buntsandstein	4	3.0	4.5 (ub)	3.4	3.5	1.0
Z	Zechstein	8	2.3	2.5	3.4 (ub)	3.4 (ub)	0.8
PC	Permo-Carboniferous	3	2.2	1.3	1.4	1.5	1.0
JM	Jura Mountains	4	3.0	4.5 (ub)	3.4	3.5	0.5
S	UC: Saxothuringian	6	2.5	2.4 (lb)	3.2	3.5 (ub)	2.5
O	UC: Odenwald	4	3.0	4.5 (ub)	3.4	3.5	1.8
MCH	UC: Mid-German Crystalline High	9	2.4	2.7 (ub)	2.3 (lb)	2.3 (lb)	1.8
NPZ	UC: Nothern Phyllite Zone	5	2.7	2.8 (ub)	2.8 (ub)	2.8 (ub)	3.0
R	UC: Rhenohercynian	5	2.7	2.8 (ub)	2.8 (ub)	2.8 (ub)	1.0
MN	UC: Moldanubian	6	2.5	2.4 (lb)	3.2	3.5 (ub)	2.6
UC-A	UC: Alps	8	2.3	2.5	3.4 (ub)	3.4 (ub)	2.4

LC	Lower Crust	10	2.1	2.1 (lb)	2.3	2.1 (lb)	0.5
LM1	Lithospheric Mantle 1	11	3.95	4.1	5.9 (ub)	5.9 (ub)	0.03
LM2	Lithospheric Mantle 2	11	3.95	4.1	5.9 (ub)	5.9 (ub)	0.03

4.3.4 Model Calibration with Local Sensitivity Analysis

In this section, we investigate the model calibration of the Upper Rhine Graben under consideration of a local sensitivity analysis again with all three model weighting versions. First, we present the sensitivity results, and then we show how to incorporate these results into the model calibration procedure.

Sensitivity Analysis

The local sensitivity analysis without weights – for the reduced model with varying thermal conductivities (Fig. 4.4, 4.5) – has high sensitivities for the thermal conductivities of the Cenozoic Rift Sediments and the Lithospheric Mantle. Also, the Cenozoic Folded Molasse/Cenozoic Foreland Molasse/Buntsandstein/Jura Mountains/Odenwald layers have a significant influence on the model response. The influences of the Malm/Northern Phyllite Zone/Rhenohercynian, the Mid-German Crystalline High, and the Lower Crust have a comparable influence on the model response. This influence is the third highest from all model parameters. The last parameter that influences the model is the thermal conductivity of the Cenozoic Volcanics. The remaining layers have a sensitivity below 0.1 and are for the further analyses ignored.

The sensitivities for the case with automatic weights are similar except that now also the Cenozoic Volcanics are insensitive to the temperature distribution. For the analyses with user-defined weights, we observe slightly smaller sensitivities for all parameters except the sensitivity of the thermal conductivity of the Cenozoic Rift Sediments. Now also the Malm/Northern Phyllite Zone/Rhenohercynian layers do not influence the temperature distribution.

For the local sensitivity analysis, using the reduced model with varying radiogenic heat production rates (Fig. 4.6), we see again that the highest sensitivity is arising from the thermal conductivity of the Cenozoic Rift Sediments. The sensitivities form the radiogenic heat productions of the Upper Crust are significantly lower. These results are independent of the weighting scheme.

For the analysis without weights, we obtain similar sensitivities for the radiogenic heat productions of the Saxothuringian, the Mid-German Crystalline High, the Northern Phyllite Zone, and the Moldanubian. The one of the Rhenohercynian is significantly lower, and the those of the Odenwald and Alps are negligible.

In case of the local sensitivity analysis with weights, we generally observe lower the sensitivities of the radiogenic heat production of the Upper Crust. For these analyses also the sensitivities of the Mid-German Crystalline High, the Northern Phyllite Zone, and the Rhenohercynian, the Odenwald, and the Alps become negligible.

Because of the small influence of the radiogenic heat production, we focus in the model calibration solely on the thermal conductivities.

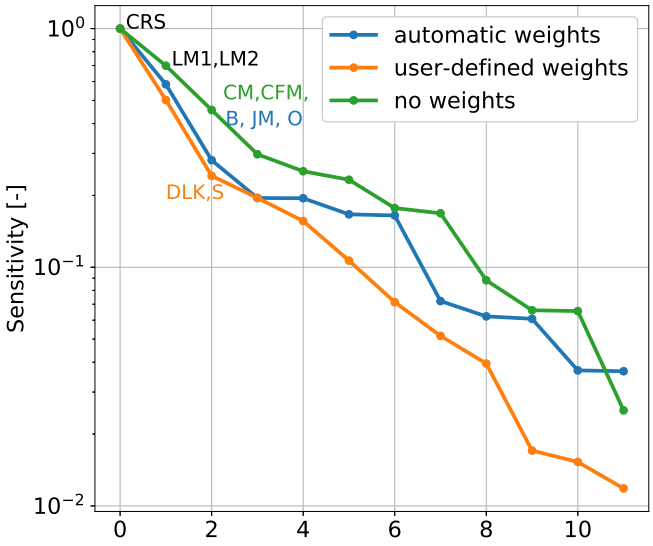


FIGURE 4.4: Local sensitivity analysis, without weights (green), with user-defined weights (orange), and with automated weights (blue) for the reduced order model varying only the thermal conductivities.

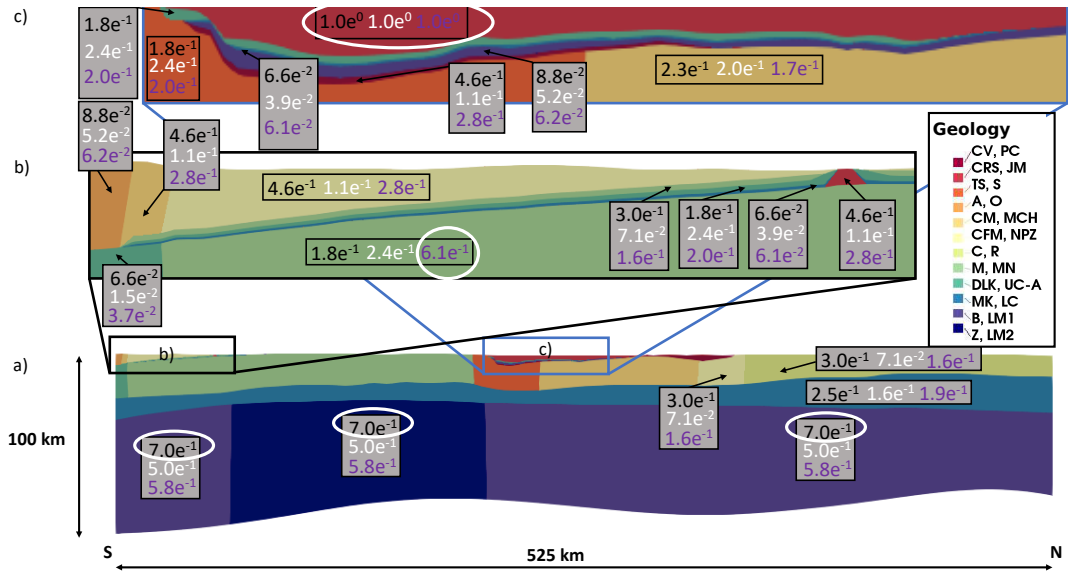


FIGURE 4.5: We show a cross-section of the Upper Rhine Graben with the local sensitivities of the different geological layers. The local sensitivities without considering weights are denoted in black, the ones with user-defined weights in white, and the ones with automatic weights in purple.

Model Calibration

The local sensitivity analyses helped to reduce the parameter space, such that for the model calibration with user-defined weights, we have to consider six parameters. The model calibration with automatic weights requires seven parameters and the model calibration without weights eight parameters instead of twelve. For all weighting versions, we can ignore the Tertiary Salt, the Alps, the Muschelkalk, and the Zechstein/Upper Crust - Alps layers since they are insensitive to the model response. The

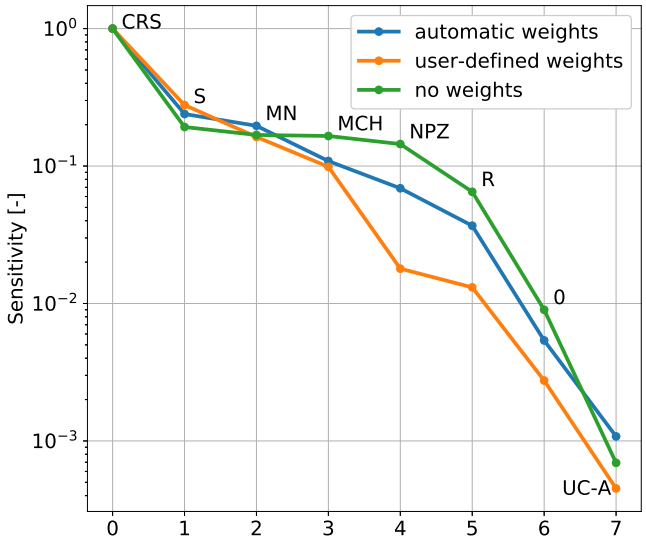


FIGURE 4.6: Local sensitivity analysis, without weights (green), with user-defined weights (orange), and with automated weights (blue) for the reduced order model varying the thermal conductivity of the Cenozoic Rift Sediments and the radiogenic heat productions of the Upper Crust.

Cenozoic Volcanics do not need to be considered for the model calibrations with user-defined, and automatic weights. Also, the Malm/Northern Phyllite Zone/Rhenohercynian are insensitive to the model response for the case of user-defined weights.

The model calibration results for all calibrate thermal conductivities are very similar to those obtained by the model calibration without considering a sensitivity study (Tab. 4.2). Only for the Lower Crust and the Lithospheric Mantle differences are observed. We still see that many of the model parameters reach their upper or lower bound.

TABLE 4.2: The thermal properties before and after the calibration for the Upper Rhine Graben with considering the results of the local sensitivity study. The initial thermal properties are taken from Freymark et al. (2017). We denote all parameters that are not involved in the model calibration, due to too low sensitivities, with n/a. Note that UC stands for Upper Crust, ub for upper bound, and lb for lower bound.

ID	Layer	Training Parameter	Initial Bulk Thermal Conductivity [W m^{-1} K^{-1}]	Calibrated Thermal Conductivity [W m^{-1} K^{-1}]			Radiogenic Heat Production [$\mu W m^{-3}$]
				user-defined weights	automatic weights	no weights	
CV	Cenozoic Volcanics	0	1.8	n/a	n/a	2.7 (ub)	0.2
CRS	Cenozoic Rift Sediments	1	1.2	0.8	1.4	1.4	1.0
TS	Tertiary salt	2	6.0	n/a	n/a	n/a	0.0
A	Alps	3	2.2	n/a	n/a	n/a	0.3
CM	Cenozoic Folded Molasse	4	3.0	4.5 (ub)	3.3	3.3	1.0
CFM	Cenozoic Foreland Molasse	4	3.0	4.5 (ub)	3.3	3.3	1.0
C	Cretaceous	3	2.2	n/a	n/a	n/a	0.5
M	Malm	5	2.7	n/a	2.8 (ub)	2.8 (ub)	1.4
DLK	Dogger/Lias-/Keuper	6	2.5	2.4 (lb)	3.2	3.5 (ub)	1.6
MK	Muschelkalk	7	2.0	n/a	n/a	n/a	1.2

B	Buntsandstein	4	3.0	4.5 (ub)	3.3	3.3	1.0
Z	Zechstein	8	2.3	n/a	n/a	n/a	0.8
PC	Permo-Carboniferous	3	2.2	n/a	n/a	n/a	1.0
JM	Jura Mountains	4	3.0	4.5 (ub)	3.3	3.3	0.5
S	UC: Saxothuringian	6	2.5	2.4 (lb)	3.2	3.5 (ub)	2.5
O	UC: Odenwald	4	3.0	4.5 (ub)	3.3	3.3	1.8
MCH	UC: Mid-German Crystalline High	9	2.4	2.7 (ub)	2.3 (lb)	2.3 (lb)	1.8
NPZ	UC: Nothern Phyllite Zone	5	2.7	n/a	2.8 (ub)	2.8 (ub)	3.0
R	UC: Rhenohercynian	5	2.7	n/a	2.8 (ub)	2.8 (ub)	1.0
MN	UC: Moldanubian	6	2.5	2.4 (lb)	3.2	3.5 (ub)	2.6
UC-A	UC: Alps	8	2.3	n/a	n/a	n/a	2.4
LC	Lower Crust	10	2.1	2.1 (lb)	2.6	2.1 (lb)	0.5
LM1	Lithospheric Mantle 1	11	3.95	4.3	5.9 (ub)	5.9 (ub)	0.03
LM2	Lithospheric Mantle 2	11	3.95	4.3	5.9 (ub)	5.9 (ub)	0.03

4.3.5 Model Calibration with Global Sensitivity Analysis

In the following, we present the results from the global sensitivity study and the model calibration using these sensitivity results. Again, we perform all analyses for the scenarios: i) no weights, ii) automatic weights, iii) user-defined weights.

Sensitivity Analysis

Now, we consider the reduced model with varying thermal conductivities and the scenario without weights. The Sobol sensitivity analysis (Fig. 4.7, 4.8) shows that the Cenozoic Rift Sediments dominantly influence the model. The major part of the sensitivity is first-order, and only a minor part is originating from higher-order contributions. The second highest sensitivity is coming from the Cenozoic Volcanics and also has dominantly first-order contributions. Note that the third highest sensitivity is arising from the thermal conductivity of the Lithospheric Mantle, which has mainly higher-order contributions. Also, the Cenozoic Folded Molasse/Cenozoic Foreland Molasse/Buntsandstein/Jura Mountains/Odenwald layers are minorly influencing the model with predominantly first-order contributions. The influences from all remaining layers are negligible because they are below our threshold value of $1 \cdot 10^{-2}$.

The Sobol sensitivity analysis for the same reduced model with user-defined weights results in a slightly different pattern. For this scenario, the model is still predominately influenced by the Cenozoic Rift Sediments, but it has nearly no higher-order contributions. Furthermore, first-order contributions from the Lithospheric Mantle influence the model, again the higher-order contributions are nearly non-existent. The influences from the Cenozoic Volcanics and the Cenozoic Folded Molasse/Cenozoic Foreland Molasse/Buntsandstein/Jura Mountains/Odenwald layers are no longer observable. The Sobol sensitivity analysis with automatic weight yields nearly the same results as the analysis without weights. The analyses took in total 260,000 function evaluations each yielding an execution time of under 500 s.

Furthermore, we performed a Sobol sensitivity analysis (with and without weights) with eight parameters (Fig. 4.9): the thermal conductivity of the Cenozoic Rift Sediments and the radiogenic heat production rates of the Upper Crust. From all radiogenic heat productions, only the one from the Saxothuringian influences the model for both the analysis with and without weights. However, the influence is minimal in comparison to the thermal conductivity of the Cenozoic Rift Sediments. Higher-order contributions dominate the radiogenic heat production of the Saxothuringian, for the analysis without weights. Similar to the first reduced model, we observe a drastic increase in the first-order contributions for the analysis with user-defined weights. Again, for the analysis with automatic weights, we get the same results as for the analysis without weights. For each of these analyses, we need 180,000 function evaluations, and less than 90 s compute time.

Model Calibration

For the calibration, we considered only the thermal conductivities since the Sobol sensitivity analysis showed that the influence of the radiogenic heat production is small in comparison to the thermal conductivity. Furthermore, we calibrate, in the no weight and automatic weight case, with only four instead of 12 parameters. The global sensitivity analysis revealed that the model is insensitive to the other eight parameters. For the case with user-defined weights, we consider only two parameters. The calibrations require at most seven function evaluations, resulting in a computation time of under 100 ms.

Region 2 is located in the southern and central part of the Upper Rhine Graben, whereas region 3 is mostly located in the central part of the Upper Rhine Graben (Fig. 4.10). We observe the highest misfit for the temperatures of the initial parameter distribution and the parameter distribution after the calibration without weights in the southern and central part of the Upper Rhine Graben (Fig. 4.10). Focusing on the

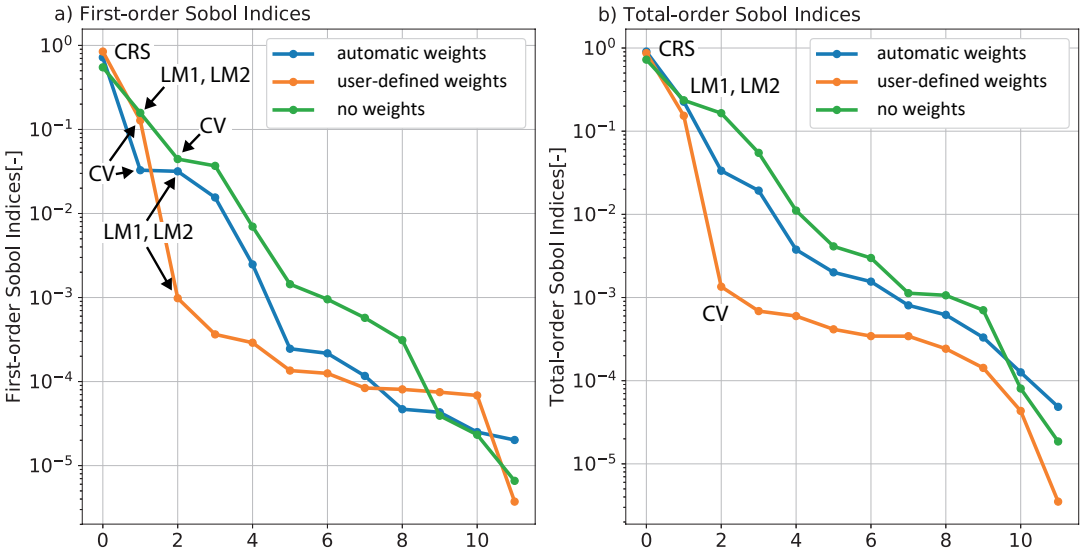


FIGURE 4.7: Global sensitivity analysis, without weights (green), with user-defined weights (orange), and with automated weights (blue) for the reduced order model varying only the thermal conductivities.

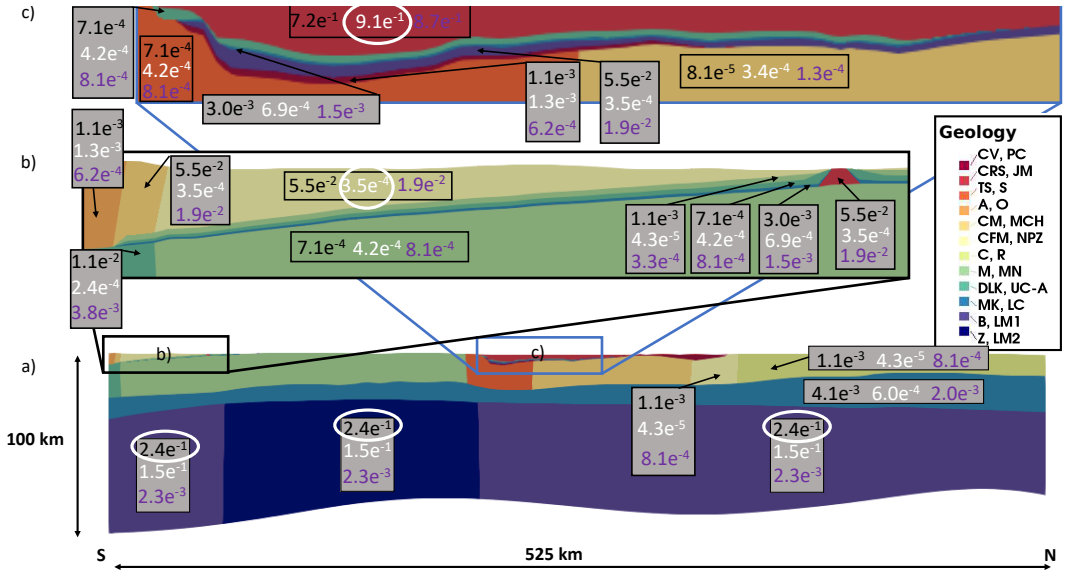


FIGURE 4.8: We show a cross-section of the Upper Rhine Graben with the global sensitivities of the different geological layers. The global sensitivities without considering weights are denoted in black, the ones with user-defined weights in white, and the ones with automatic weights in purple.

area of the highest misfit, we observe in the central part of this area underestimated temperature values yielding a temperature misfit of -10 °C to -40 °C. In the southern and northern part, we observe overestimated and underestimated temperature values ranging from -20 °C to 70 °C.

For the parameter distribution after the calibration with user-defined weights, we see a similar pattern. Note that the underestimation of the temperature in the central part of the misfit area increased. This results in a temperature misfit of around -70 °C. In contrast, the overestimation in the southern and northern part decreased, yielding temperature differences of around 10 °C to 50 °C. This is the main reason why the

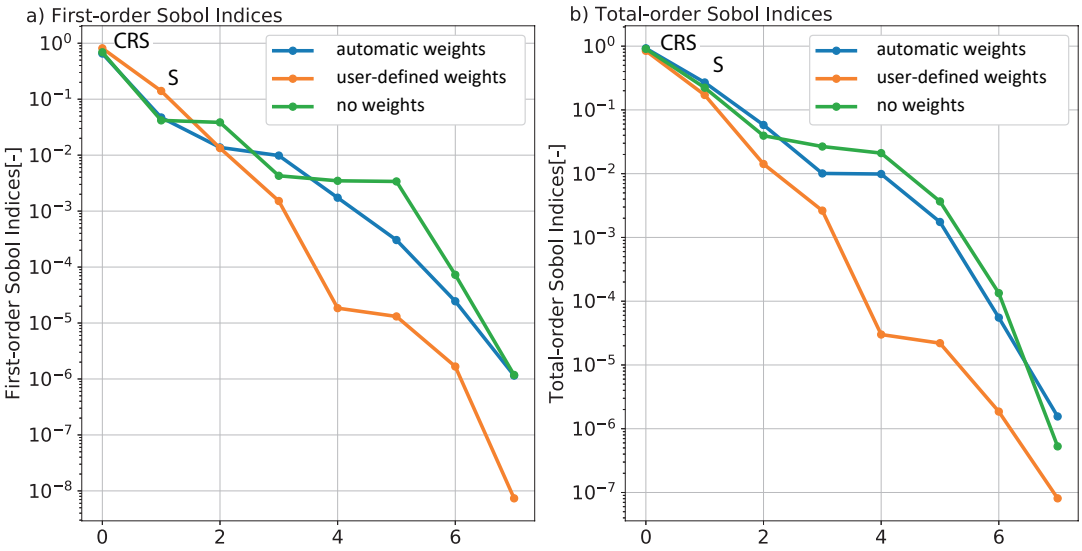


FIGURE 4.9: Global sensitivity analysis, without weights (green), with user-defined weights (orange), and with automated weights (blue) for the reduced order model varying the thermal conductivity of the Cenozoic Rift Sediments and the radiogenic heat productions of the Upper Crust.

average misfit in region 2 decreased from 29 °C to 17 °C. The misfit distribution for the parameters after the calibration with automatic weights is similar to the one without weights.

The initial temperature distribution (Fig. 4.11) in the overall model of the Upper Rhine Graben varies from -1 °C to 1300 °C. At the upper part of the model, we observe much steeper thermal gradients than in the lower part of the model. The misfit between observed and simulated temperatures has its highest values in the area of the Upper Rhine Graben and there in the regions 2 and 3.

We now focus on the calibrated thermal conductivities (Tab. 4.3). Regarding the calibration without weights (Fig. 4.12), we observe, for the Cenozoic Volcanics and Rift Sediments and the Lithospheric Mantle, an increase in thermal conductivity in contrast to the initial thermal conductivities. This yields thermal conductivities of 2.4 W m^{-1} K^{-1}, 1.3 W m^{-1} K^{-1}, and 4.9 W m^{-1} K^{-1}, respectively. The Cenozoic Folded Molasse/Cenozoic Foreland Molasse/Buntsandstein/Jura Mountains/Odenwald layers have with a thermal conductivity of 3.0 W m^{-1} K^{-1} the same value before and after the calibration. The calibration with automatic weights yields similar thermal conductivities as the one without weights.

For the calibration with user-defined weights (Fig. 4.13), we observe for the Rift Sediments a decrease in thermal conductivity after the calibration, resulting in an extremely low thermal conductivity of 0.8 W m^{-1} K^{-1}. The Lithospheric Mantle has a slightly decreased thermal conductivity of 3.6 W m^{-1} K^{-1} in contrast to the initial thermal conductivity of 3.95 W m^{-1} K^{-1}.

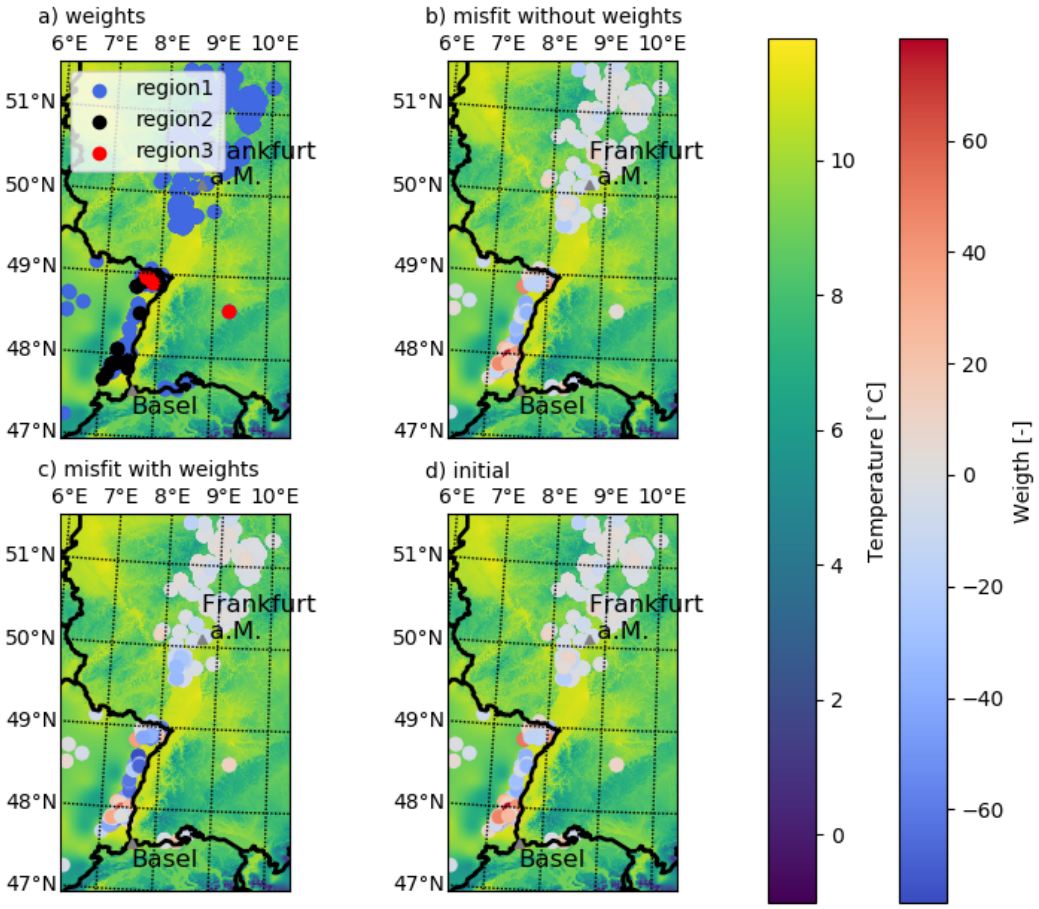


FIGURE 4.10: Spatial distribution of a) the user-defined weights b) the misfit in temperature for the calibration without weights c) the misfit in temperature after the calibration with user-defined weights, and d) the misfit in temperature for the initial parameter distribution. In the background of all images are the annual regional average surface temperature values plotted to outline the region of the Upper Rhine Graben.

TABLE 4.3: The thermal properties before and after the calibration for the Upper Rhine Graben considering the results of the global sensitivity study. The initial thermal properties are taken from Freymark et al. (2017). We denote all parameters that are not involved in the model calibration, due to too low sensitivities, with n/a. Note that UC stands for Upper Crust.

ID	Layer	Training Parameter	Initial Bulk Thermal Conductivity [W m^{-1} K^{-1}]	Calibrated Thermal Conductivity [W m^{-1} K^{-1}]			Radiogenic Heat Production [$\mu W m^{-3}$]
				user-defined weights	automatic weights	no weights	
CV	Cenozoic Volcanics	0	1.8	n/a	2.6	2.4	0.2
CRS	Cenozoic Rift Sediments	1	1.2	0.8	1.3	1.3	1.0
TS	Tertiary salt	2	6.0	n/a	n/a	n/a	0.0
A	Alps	3	2.2	n/a	n/a	n/a	0.3
CM	Cenozoic Folded Molasse	4	3.0	n/a	2.9	3.0	1.0
CFM	Cenozoic Foreland Molasse	4	3.0	n/a	2.9	3.0	1.0
C	Cretaceous	3	2.2	n/a	n/a	n/a	0.5
M	Malm	5	2.7	n/a	n/a	n/a	1.4
DLK	Dogger/Lias-/Keuper	6	2.5	n/a	n/a	n/a	1.6
MK	Muschelkalk	7	2.0	n/a	n/a	n/a	1.2
B	Buntsandstein	4	3.0	n/a	2.9	3.0	1.0
Z	Zechstein	8	2.3	n/a	n/a	n/a	0.8

PC	Permo-Carboniferous	3	2.2	n/a	n/a	n/a	1.0
JM	Jura Mountains	4	3.0	n/a	2.9	3.0	0.5
S	UC: Saxothuringian	6	2.5	n/a	n/a	n/a	2.5
O	UC: Odenwald	4	3.0	n/a	2.9	3.0	1.8
MCH	UC: Mid-German Crystalline High	9	2.4	n/a	n/a	n/a	1.8
NPZ	UC: Nothern Phyllite Zone	5	2.7	n/a	n/a	n/a	3.0
R	UC: Rhenohercynian	5	2.7	n/a	n/a	n/a	1.0
MN	UC: Moldanubian	6	2.5	n/a	n/a	n/a	2.6
UC-A	UC: Alps	8	2.3	n/a	n/a	n/a	2.4
LC	Lower Crust	10	2.1	n/a	n/a	n/a	0.5
LM1	Lithospheric Mantle 1	11	3.95	3.6	5.2	4.9	0.03
LM2	Lithospheric Mantle 2	11	3.95	3.6	5.2	4.9	0.03

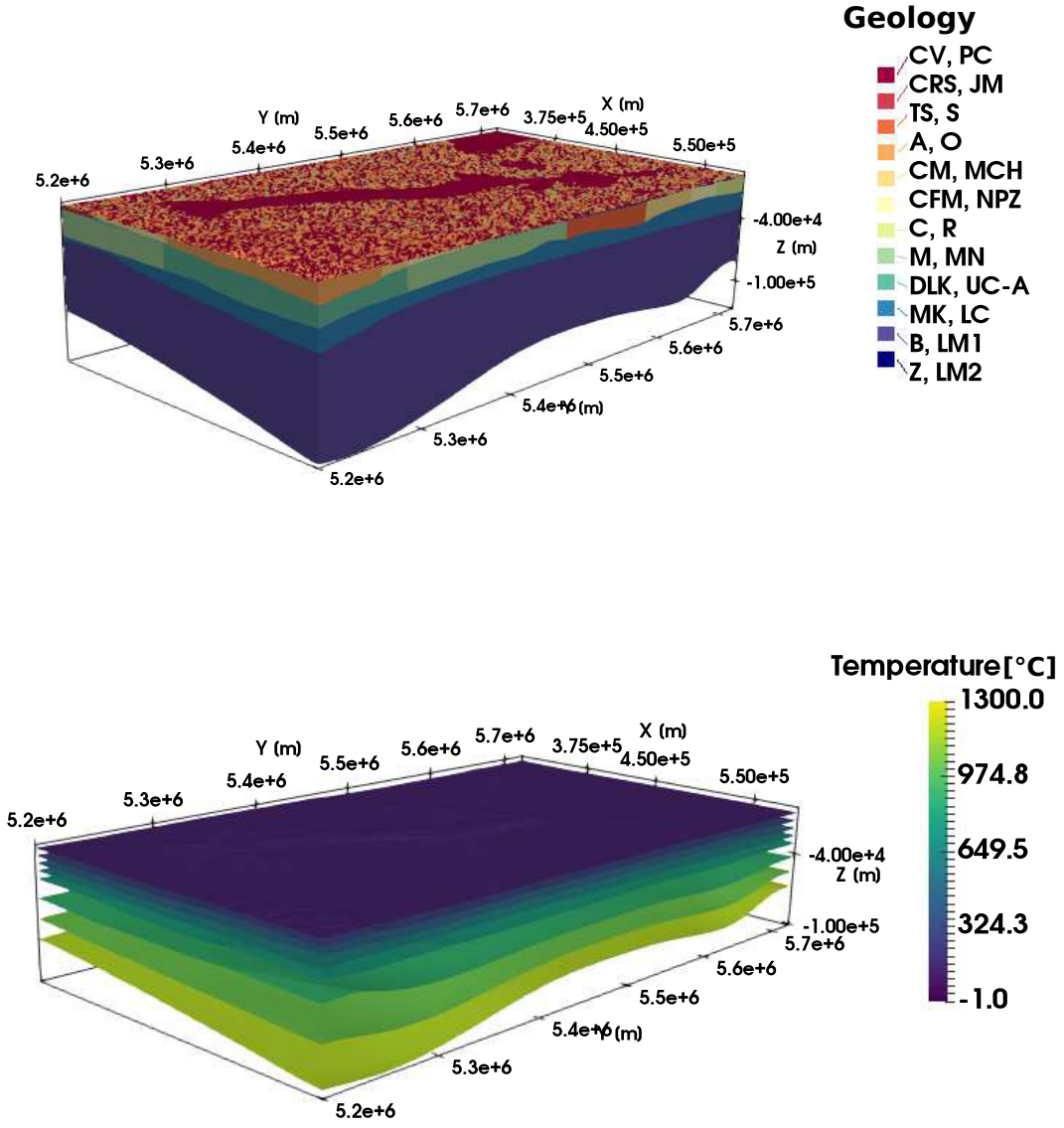


FIGURE 4.11: Image of the geology, and the initial temperature distribution of the Upper Rhine Graben model (for the layer IDs refer to Tab. 4.3).

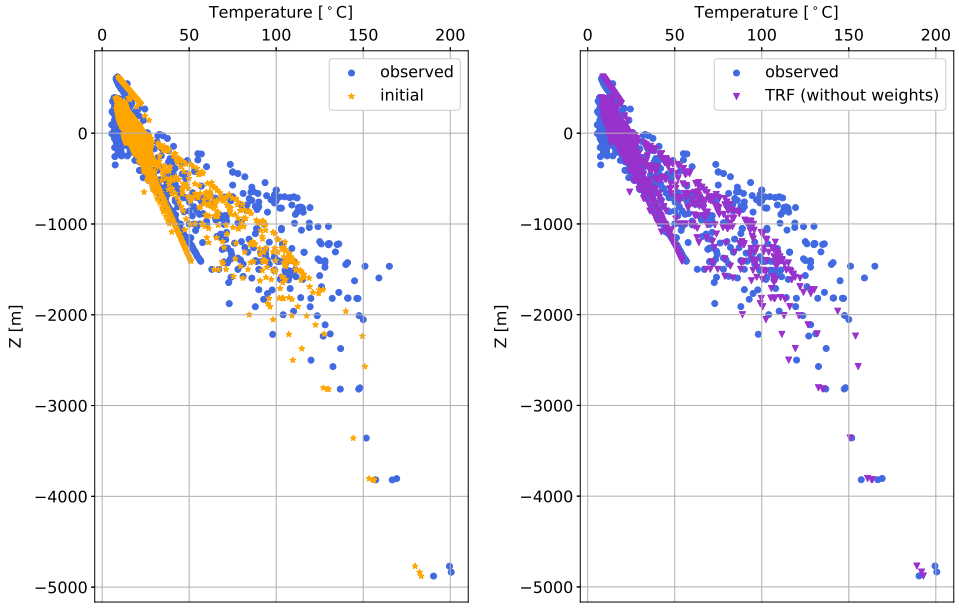


FIGURE 4.12: On the left side, the temperature distribution of the observed data and the initial parameter distribution are shown. The right image shows the observed data and the temperature distribution of the parameters after the trust region reflection (TRF) calibration without weights.

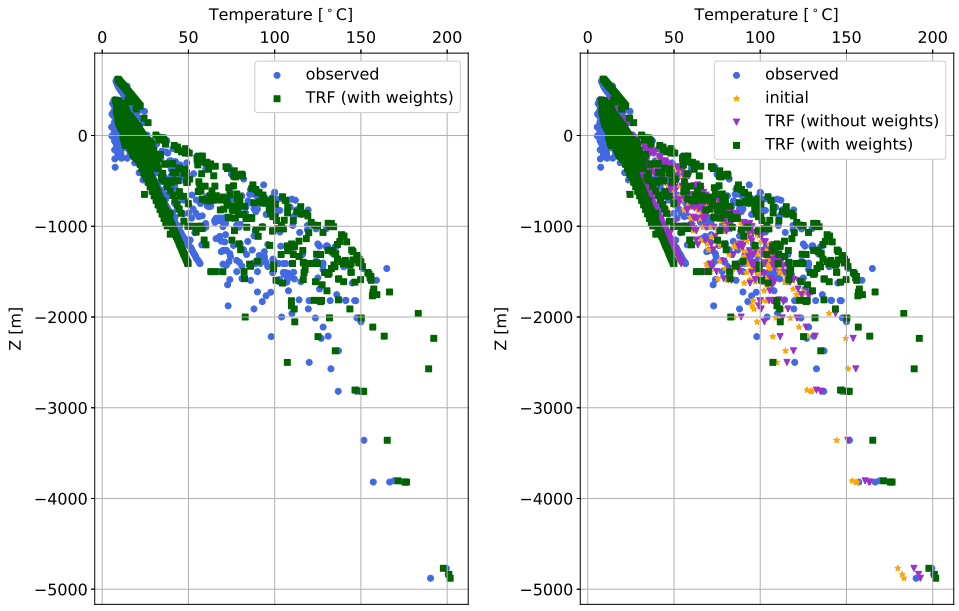


FIGURE 4.13: On the left side, the observed data and the temperature distribution after the trust region reflective (TRF) calibration with user-defined weights are shown. The right image shows the temperature distribution of the observed data, the initial parameter distribution, and the parameter distribution after the calibration with and without weights. Only for the temperature distribution after the calibration with user-defined weights, we observe a good agreement of the temperature values for all parts of the model.

4.3.6 Computational Cost

The reduction requires 128 basis functions to describe the model for a relative error tolerance of $5{\cdot}10^{-4}$ for the Upper Rhine Graben model, where we vary only the thermal conductivities (Fig. 4.14). We can reduce the compute time of a single forward simulation from 44 min to 3 ms, yielding a speed-up of $9.2{\cdot}10^{5}$. For the model with a varying thermal conductivity of the Cenozoic Rift Sediments and the varying radiogenic heat productions of the Upper Crust, we require 25 basis functions for reaching an error tolerance of $1{\cdot}10^{-5}$. Here, we reduce the compute time to 1 ms, resulting in a speed-up of $2.7{\cdot}10^{6}$.

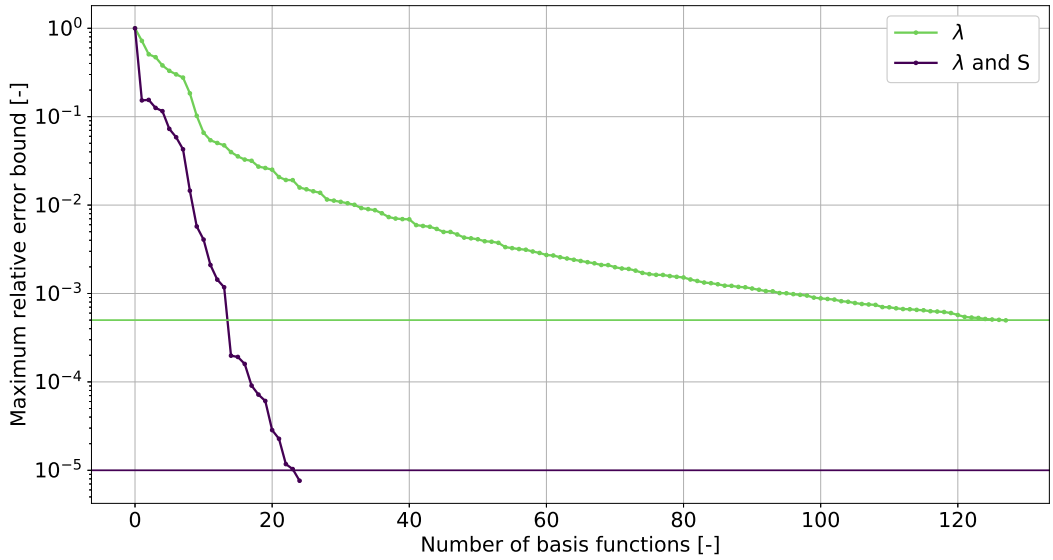


FIGURE 4.14: Convergence of the relative maximum error bound for both reduced order models. The error bound is relative to the approximation of the "truth" (FE solution). For that reason, it is a measure of the approximation quality.

4.4 Discussion

In this section, we discuss the advantages of a global sensitivity analysis over a local one. Additionally, we present the benefits of an automated model calibration in comparison to a "trial-and-error" model calibration. A discussion of the computational cost of using the RB instead of the classical FE method follows.

4.4.1 Sensitivity Analysis

The global sensitivity analysis, without weights, indicated that first-order contributions from the thermal conductivity of the Cenozoic Rift Sediments dominate the model. Hence, the most substantial influence from that layer is coming from the layer itself and not from correlations with other layers. So, one might conclude that for this specific model, a local sensitivity analysis is sufficient. This is, however, not the case because:

1. Even if we observe nearly no correlation, the local sensitivity analysis overestimates the influence of the model parameters on the model response. As we have seen it already for the simplified basin-scale model.

2. The thermal conductivity of the Lithospheric Mantle, on the other hand, has nearly only influences arising from correlations to other parameters, which is only visible through performing a global sensitivity analysis.

The thermal conductivities of the Cenozoic Volcanics and the Lithospheric Mantle have a significantly smaller influence on the model. They both have similar total sensitivities. Moreover, the thermal conductivity of the Cenozoic Volcanics has mostly influences arising from itself and nearly no influences originating from the correlation with other parameters. In case of the Lithospheric Mantle, the large layer thickness causes the strong influence.

We expect that the model is mostly influenced by the upper layers, because of the measurements located in the upper layers (deepest measurement at about 7 km depth). The upper boundary condition causes the smaller influence from the Cenozoic Volcanics. At the top of the model, we applied a Dirichlet boundary condition, which fixes the temperatures to the regional average annual temperature values. Hence, the possible model variations in the uppermost part of the model are limited, and therefore, also the sensitivities are smaller.

Furthermore, the model is to a smaller degree influenced by the thermal conductivity of the Cenozoic Folded Molasse/Cenozoic Foreland Molasse/Buntsandstein/Jura Mountains/Odenwald layers itself. For the global sensitivity analysis, with user-defined weights, we observed that the sensitivity towards the thermal conductivities of the Cenozoic Volcanics and Cenozoic Folded Molasse/Cenozoic Foreland Molasse/Buntsandstein/Jura Mountains/Odenwald layers are vanishing. Furthermore, the first-order contributions are drastically increasing. Further emphasizing that the model is mainly influenced by the thermal conductivities of the layers themselves. So, not through the correlation with other thermal conductivities.

If we compare the results with the local sensitivity studies, we observe that the local study is overestimating the influence of the remaining layers. Furthermore, we cannot distinguish between first- and higher-order contributions. Therefore, it is not possible to investigate the correlation between the parameters.

The influences of the radiogenic heat productions are negligible in comparison to the thermal conductivities. This is the reason why we did not account for these parameters in further analyses.

4.4.2 Model Calibration

If we look at the model calibration using a global sensitivity study, we see that independent of the weighting, no calibrated parameter reaches its bound. However, the cost of the sensitivity-driven model calibration is exceptionally high because of the global sensitivity analysis. So, why can we not use a local sensitivity study? The reasons become apparent if we look at the results of the model calibration using a local sensitivity analysis. The calibrated parameter values are nearly identical to the ones without a sensitivity study. We are able to reduce the number of parameters from twelve to six, seven, or eight parameters depending on the weighting. Nonetheless, many parameters are still reaching their pre-defined bounds. This either shows that the bounds were not placed far enough from the initial parameter distribution or that the parameters are insensitive to the temperature distribution. Considering, the already extensive variation range (up to $\pm$ 50 %), it is more likely that the parameters

that reach their bounds are insensitive to the model response. If we consider the case of insensitive parameters. Then, the model calibration fails to determine the parameter values correctly since all possible parameter values result in the same temperature distribution. Having insensitive parameters after performing a local sensitivity study, means consequently that the local sensitivity study fails to reduce the parameter space successfully assuming well defined bounds.

We further investigate the problem with the bounds by decreasing the allowed variation range from $\pm$ 50 % to $\pm$ 10 % in step sizes of 10 %. For this analysis, only the model calibration with a global sensitivity analysis behaves in accordance with our expectations. Both the model calibration without a sensitivity study and with a local sensitivity study behave non-physically. As an example, if we define a variation range of $\pm$ 10 % the thermal conductivity of the Cenozoic Folded Molasse/Cenozoic Foreland Molasse/Buntsandstein/Jura Mountains/Odenwald layers decrease to their lower bound of 2.7 W m^{-1} K^{-1}. Note that the initial value is 3.0 W m^{-1} K^{-1}. However, if we now increase the variation range to $\pm$ 20 % then the thermal conductivities increase to their upper bound of 3.6 W m^{-1} K^{-1}, which is physically not plausible.

With a global sensitivity driven model calibration, we are not only able to identify the most influencing parameters but also to investigate to which degree the individual parameters are correlated with each other. Furthermore, the automated calibration is a systematic way to explore the entire parameter space. A "trial-and-error" model calibration allows us to compute a handful of realizations, but it is far away from giving a guarantee to find the optimal parameter set. Additionally, it does not consider any correlations with other parameters and results in more qualitative than quantitative results. Automated model calibrations are reproducible, something that is not true for the "trial-and-error" model calibration. Within this chapter, we perform model calibrations with and without weights.

Using the model calibration with user-defined weights and a global sensitivity analysis leaves us with two parameters that can be calibrated: the Cenozoic Rift Sediments, and the Lithospheric Mantle. This is in accordance with our geological understanding of the model. In case of the user-defined weights, we concentrate the calibration nearly exclusively on the area of the Upper Rhine Graben, in order to explain the mismatch of the temperature values in region 2 from the initial parameter distribution. If we focus our analysis on the Upper Rhine Graben itself, the calibration is majorly impacted by the dominant geological layer of the Upper Rhine Graben: the Cenozoic Rift Sediments. Only, the global sensitivity analysis represents our geological understanding of the model. Note, that if we focus the model calibration on the entire model, we can also determine the behavior of other layers such as the Cenozoic Volcanics.

In order to control the robustness of the different model calibration versions, we performed the model calibration without a sensitivity study, with a local sensitivity study, and with a global sensitivity study with different initial guesses. All model calibrations were tested using the user-defined weights. We tested as initial guess the calibrated values provided by Freymark et al. (2017), the lower and upper parameter bounds, and ten randomly chosen parameter sets. The least robust model calibration is the one without a sensitivity analysis. It has the highest differences, which are in the order of 10^{-2}. The model calibration with a local sensitivity analysis yields differences in the order of 10^{-5}, and the one with a global sensitivity 10^{-4}. Hence, a first conclusion might be that both the model calibrations with the local and the global sensitivity study result in robust model calibrations since they converge to the same results. However, if we look a bit closer at the model calibration with the local sensitivity study, this impression is lost. Those parameters that reach the bounds show in most realizations a difference of precisely zero with respect to the initial parameter set of

Freymark et al. (2017). That is extremely suspicious since at least differences caused by numerical errors should be visible. This leads to the conclusion that the model calibration with the local sensitivity study is robust for the model parameters that do not reach their bounds. It is stuck for those that reach one of their bounds. Overall, it is a non-robust model calibration. For both the model calibrations without a sensitivity study and with a local sensitivity study, many of the calibrated parameter values yield unphysical parameter values. This, in combination with the non-robustness of both model calibrations, leaves only the model calibration with a global sensitivity study as a robust and reliable model calibration.

Using a sensitivity-driven automated model calibration instead of a "trial-and-error" model calibration, as done in the previous studies, has several advantages:

1. Through the sensitivity analysis, it is possible to identify the parameters that do not influence the model. Hence, we reduce the number of parameters that we have to consider in further analyses, which reduces the compute time.

2. A global sensitivity study, as the Sobol analysis, has further the advantage that it provides both first-order and higher-order sensitivities. This further emphasizes the need for global sensitivity studies. Independent of how correlated the individual parameters are, the local sensitivity study fails to reduce the parameter space due to the overestimation of the sensitivities of the model parameters.

The difficulty arising from the observation data is the very uneven spatial distribution. We have 2,282 data points in region 1, 53 in region 2, and 12 data points in region 3. Hence, region 1 contains one order of magnitude more data than region 2 and two orders of magnitude more data than region 3. Performing a calibration without applying any weights to the misfit will, therefore, reduce the misfit in region 1. However, it nearly not considers the misfit in region 2 and 3 since the contribution of the misfit of these regions to the overall misfit is tiny. Fig. 4.11 clearly shows this by comparing the temperature values of the model calibration without weights and the initial temperature distribution. The difference between these two is insignificant especially under consideration of the measurement accuracy. If we are only interested in region 1 of the model, it is sufficient to follow the approach of a model calibration without any weights. If we are, however, interested in the entire model, this is not enough. In order to compensate for the unequal distribution of observations, we apply user-defined weights to the different regions as described in Section 4.3.2. This results in a slightly worse fit for region 1 compared to the initial temperature distribution. However, we decrease the misfit in region 2 from 29 °C to 17 °C. The misfit in region 3 is comparable for both realizations.

The spatial distribution of the misfit demonstrates that in the area outside the southern and central part of the Upper Rhine Graben, a conductive model is sufficient to describe the temperature distribution. In the mentioned area, we observe a significant misfit to the observed temperatures. The calibration with user-defined weights shows that this misfit is caused by an effect that leads to a decreased thermal conductivity in the Cenozoic Volcanics and Rift Sediments. By having a closer look at this area in the literature, we find that especially the area of the central part of the Upper Rhine Graben is characterized by highly permeable sediments with a dense fault network (Buchmann and Connolly, 2007; Bauer et al., 2015; Freymark et al., 2019; Vidal and Genter, 2018; Meixner et al., 2016). Combining these structural characteristics, and the discrepancy in the thermal conductivity of the upper two layers, we can conclude that the misfit might be related to not having considered fluid movement and related effects on heat transport. Fluids moving in the pore space, being influenced by the

two border faults, could explain the obtained extremely low "effective" thermal conductivity (Fig. 4.15). These would lead to an increased heat transfer inside the graben and due to the no-flow boundaries to a higher amount of heat in the layer and hence to a lower effective thermal conductivity. The flow pattern, as shown here, has also been modeled in Freymark et al. (2019). In contrast to previous studies, we are able to identify the two layers that are responsible for the misfit and systematically determine which of these layers is of higher importance. Furthermore, we can determine through the sensitivity analysis that the two layers themselves cause most of these misfits. The correlation with other parameters introduces only a significantly smaller part.

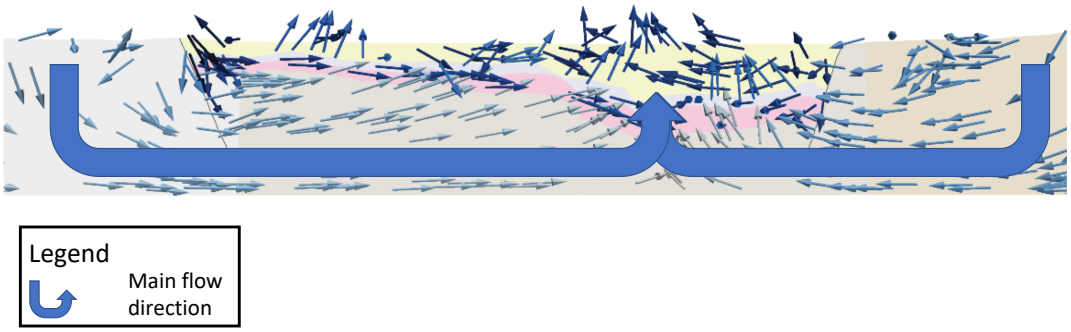


FIGURE 4.15: Representation of the fluid flow in the Upper Rhine Graben that explains the lower effective thermal conductivity for the Cenozoic Rift Sediments (modified after Freymark et al. (2019)).

The question arises why the calibration with automatic weights yields very similar results to the calibration without weights. We explain it by having a closer look at the spatial distribution of the misfit inside the southern and central part of the Upper Rhine Graben. In there we can identify three regions: i) in the central part a region of overall underestimated temperatures and ii) north and south to it regions of overall overestimated temperatures. The latter regions correspond to region 2 and are therefore the region in which the conductive model fails to represent the measured temperature distribution. If we use the automated weighting scheme that considers distances only, we apply a higher weight to both the over- and underestimated temperature regions. The model calibration method minimizes the total misfit. By applying a higher weight to both the over- and underestimated temperature regions in the Upper Rhine Graben, we increase for both parts the contribution to the total misfit. However, only the overestimated temperatures correspond to region 2. Hence, if we only apply a higher weight to the overestimated temperature region, as done in the weighting scheme based on expert knowledge, we increase the contribution to the total misfit of this region alone. Only then, we obtain a model that represents all regions.

Regarding the thermal conductivities, we observe a significant decrease in thermal conductivity in the upper two layers. We see no changes for the Cenozoic Folded Molasse/Cenozoic Foreland Molasse/Buntsandstein/Jura Mountains/Odenwald layers and only a slight decrease in the Lithospheric Mantle. This demonstrates that except for the Cenozoic Volcanics and Rift Sediment layer, a conductive model is sufficient to describe the temperature distribution of the Upper Rhine Graben. It is also in accordance with previous studies (for instance, Freymark et al. (2017)). However, it also demonstrates that the model is in its current form, insufficient to describe the temperature distribution arising from the upper two layers. That is not seen in the model calibration without weights because of the above-described reasons.

4.4.3 Computational Cost

The offline stage of the Upper Rhine Graben with varying thermal conductivities required 128 basis functions and took 35.7 h. We have performed the computation on an Intel Westmere X675 machine (3.07 GHz 6 cores per chip, 12 cores per node and 24 GB memory per node) using 48 cores. The other offline stage, with varying radiogenic heat productions, required 25 basis functions yielding a total compute time of 23.3 h on the same machine. If we compare that to the total number of function evaluations of around 440,000, it becomes evident that we are far more efficient with the reduced basis than with the finite element method. Furthermore, take into account that the reduction is a one-time cost. So, we can use the same reduced model for further investigations and do not require another time-consuming offline stage.

A disadvantage of the RB method is that it provides efficient error bounds, at the current stage, only for elliptic and parabolic partial differential equations (PDE). For a geothermal application, this has the consequence that the method is only applicable to conductive studies if we want to obtain a guarantee for the approximation error. The RB method is also applicable for hyperbolic problems but in this case we only obtain estimates of the approximation error. Also note, that we need to decompose the PDE into a parameter-dependent and -independent part. This is naturally given for linear problems, but not for non-linear ones. Nonetheless, the method is also applicable for non-linear PDEs by using the empirical interpolation method (Barrault et al., 2004), which approximates the nonlinear part of the PDE.

4.4.4 Reliability

We showed that only the model calibration in combination with the global sensitivity results in both a robust and reliable model calibration. Only this calibration can be used to identify model errors and possible to compensate for these errors at a later stage. We cannot only identify the areas with unaccounted physical properties but can furthermore quantify their influence on the thermal conductivity. This is the reason why we can construct a conductive model that compensates for the errors in the underlying physics. Resulting in a model that provides more reliable predictions. Although it does not account for all relevant physical processes.

The combination of a global sensitivity study and an automated model calibration is, unfortunately, computationally prohibitive with state-of-the-art finite element simulations. Therefore, we use the RB method as a surrogate model for the inverse processes. This reduces the compute time from 36 core-a to 10 core-min.

We have seen that the upper boundary condition influences the sensitivity results. In the next chapter, we want to investigate the influences of the boundary conditions on the model response in greater detail. We especially want to analyze the consequences of constructing geological models with a relatively low vertical extent, as common practice in the field.

Chapter 5

Influence of the Model Dimension and Automated Model Calibrations – Insights from the Brandenburg Model

Basin-scale models tend to be computationally extremely demanding. One possible approach to reduce the computational cost of these models is to reduce the spatial dimension of the models. In this chapter, we discuss the consequences of this spatial reduction and present with the RB method an alternative to it. For the investigation, we use three different versions of the Brandenburg model. Section 5.3 presents all models and their respective decompositions. These analyses are in preparation for publication in Degen et al. (2020b).

Furthermore, we use the Brandenburg model to perform a comparison of various model calibration techniques. The physical model is for all analyses the same, a geothermal conductive heat transfer with the radiogenic heat production as the source term. For further details regarding the physical model, we refer to Chapter 4.

5.1 Introduction

As mentioned, one problem of geophysical and geoscientific applications is that they suffer from their computationally intensive nature. To overcome this issue a common way is to generate models that have a large horizontal extension but a very low vertical extent (Pribnow and Schellschmidt, 2000; Pribnow and Clauser, 2000; Mottaghy et al., 2011; Vogt et al., 2013; Kastner et al., 2015). The boundary conditions for these models are usually retrieved from larger models (Noack et al., 2013). This has the advantage of generating computationally much less demanding models. However, as we demonstrate in this thesis, the models are dominated by the boundary conditions. To address the computational challenges, we are replacing, in this thesis, the finite element solution of our forward solves with the reduced basis solution. We perform global sensitivity analyses on several model realizations of the Brandenburg model (Germany) to demonstrate the influence of the lower boundary condition on the simulation.

Additionally, we perform an automated model calibration to provide an objective and reproducible way to compensate for the errors of both the physical and geological model. To recap, sensitivity analysis for basin-scale models has been performed before in Noack et al. (2012) and also been combined with automated model calibrations (Wellmann and Reid, 2014), and without sensitivities in Fuchs and Balling (2016). Furthermore, local sensitivity studies are presented in Ebigbo et al. (2016).

However, none of these can address the computationally demanding nature of the problem. Therefore, they are limited in the number of parameters, sensitivity analyses, and model calibrations they can perform. By using a physics-based machine learning approach instead of the finite element method, we can reduce the compute time of the forward solve by several orders of magnitude. It allows, in turn, to perform global sensitivity analysis and full flexibility in the model calibration.

As a reminder, global sensitivity analyses have been performed for hydrological problems (van Griensven et al., 2006; Tang et al., 2007; Cloke et al., 2008; Zhan et al., 2013; Baroni and Tarantola, 2014; Song et al., 2015), for volcanic source modeling (Cannavó, 2012), and for geothermal heat exchangers (Fernández et al., 2017). In Chapter 4, we have investigated the influence of both local and global sensitivity studies for the Upper Rhine Graben. We want to use the combination of the global sensitivity study and model calibration, as presented in Chapter 4, to investigate the influence of the placement of the boundaries on the model predictions.

5.2 Model Calibration

Geoscientific applications have many sources of uncertainty. Constructing reliable models, despite these sources of uncertainty presents us with a major challenge and yields into Georg Box famous quote that "All models are wrong, but some are useful" (Box, 1979). Model calibration is a tool to make the models "useful" by compensating for model errors through the adaptation of the model parameters. For instance, inaccurate geometrical parameterization, or incomplete knowledge of the governing physical laws are causing these model errors (Wellmann and Caumon, 2018).

Model calibration techniques can be subdivided into three main categories: local gradient-based, local derivative-free, and global derivative-free methods. In the following, we briefly present each of these categories and point out the main advantages and disadvantages of them. For the illustration of the search behavior of the different calibration methods, we use the Rosenbrock function (Rosenbrock, 1960):

$$(1 - x_1)^2 + 100(x_2 - x_1^2)^2, \tag{5.1}$$

where x_1 and x_2 are the model parameters.

5.2.1 Local Gradient-Based Methods

Local gradient-based methods are best suited for an efficient exploration of the parameter space in the vicinity of the initial parameters. They have the fastest convergence rate of the different optimization methods and consequently need the least number of forward solves. However, if the initial guess is far away from the true parameters, then the local gradient-based methods are failing since they usually get trapped in a local minimum. Furthermore, the accuracy of the gradient calculation is crucial. A too coarse approximation of the gradient also leads to the failure of this method type. In this thesis, we are considering numerical gradients (Adams et al., 2017; Goldberg, 1989; Geem, 2012). We chose a very high accuracy to avoid a failure of the model calibration. This leads to higher costs in the gradient calculation, which is not noticeable in our applications due to the very fast forward simulations by using the RB method. We take into account two local gradient-based methods: the conjugate gradient method, and the trust region reflection method.

For the conjugate gradient method, we consider the implementation provided by Dakota (Adams et al., 2017), since it can handle, in contrast to the SciPy implementation (Jones et al., 2014) also bounded parameter ranges. The conjugate gradient method is a modification of the steepest descent method. First, we chose an initial parameter x_0 for which we evaluate the cost function $f(x_0)$ and its gradient $\nabla f(x_0)$. Furthermore, we chose an error tolerance ϵ. Afterwards, we calculate the search direction by using (Chang, 2014; Fletcher and Reeves, 1964):

$$n_0 = n\left(x_0\right) = -\frac{\nabla f\left(x_0\right)}{\|\nabla f\left(x_0\right)\|}. \tag{5.2}$$

In contrast to the steepest-descent method that uses Eq. 5.2 for all following search directions, the conjugate gradient method searches along the conjugate gradient direction defined by (Chang, 2014; Fletcher and Reeves, 1964):

$$n_k = -\frac{\nabla f\left(x_k\right)}{\|\nabla f\left(x_k\right)\|} + \beta_k n_{k-1}. \tag{5.3}$$

Different definitions of β_k are available. Dakota uses the definition of Fletcher and Reeves (1964):

$$\beta_k = \frac{\|\nabla f\left(x_k\right)\|}{\|\nabla f\left(x_{k-1}\right)\|}. \tag{5.4}$$

Due to the slight modification in the calculation of the search direction, the convergence rate significantly improves in comparison to the steepest-descent method (Chang, 2014; Fletcher and Reeves, 1964).

Fig. 5.1 shows the walkthrough behavior of the conjugate gradient method for the parameter space of the Rosenbrock function. We used the same settings as Dakota in their example file for the conjugate gradient method (Adams et al., 2017). Hence, the maximum number of iterations is 100. The lower bound of both variables is -2.0 and the upper bound 2.0. We start with the initial point 1.2 for x_1 and 1.0 for x_2. The convergence tolerance is 10^{-4}. We calculate the gradients numerically using forward differences with a gradient step size of 10^{-5}. We observe that the conjugate gradient method can recover the local minimum in about 80 iterations.

The second local gradient-based method we consider in this chapter is the trust region reflective method, which is discussed in Section 4.2.3.

5.2.2 Local Derivative-Free Methods

We use local derivative-free methods in cases where the gradient calculation becomes too expensive. Furthermore, they have the advantage of being more robust than local gradient-based methods so that they can overstep local minima. The disadvantage is that they are computationally more demanding than, for instance, the conjugate gradient method since they require many more iterations. Furthermore, these methods are still prone to end up in a local minimum instead of the global one. For this thesis, we use the pattern search method (Adams et al., 2017; Goldberg, 1989; Geem, 2012), provided by the Dakota framework, to illustrate the advantages and disadvantages of local derivative-free optimization methods.

The naming of the pattern search method is founded in its searching behavior because the method explores the parameter space through a stencil. The size of the stencil and the contraction of it allows overstepping local minima. Otherwise, the method will, as the local gradient-based method, end up in a local minimum. This is

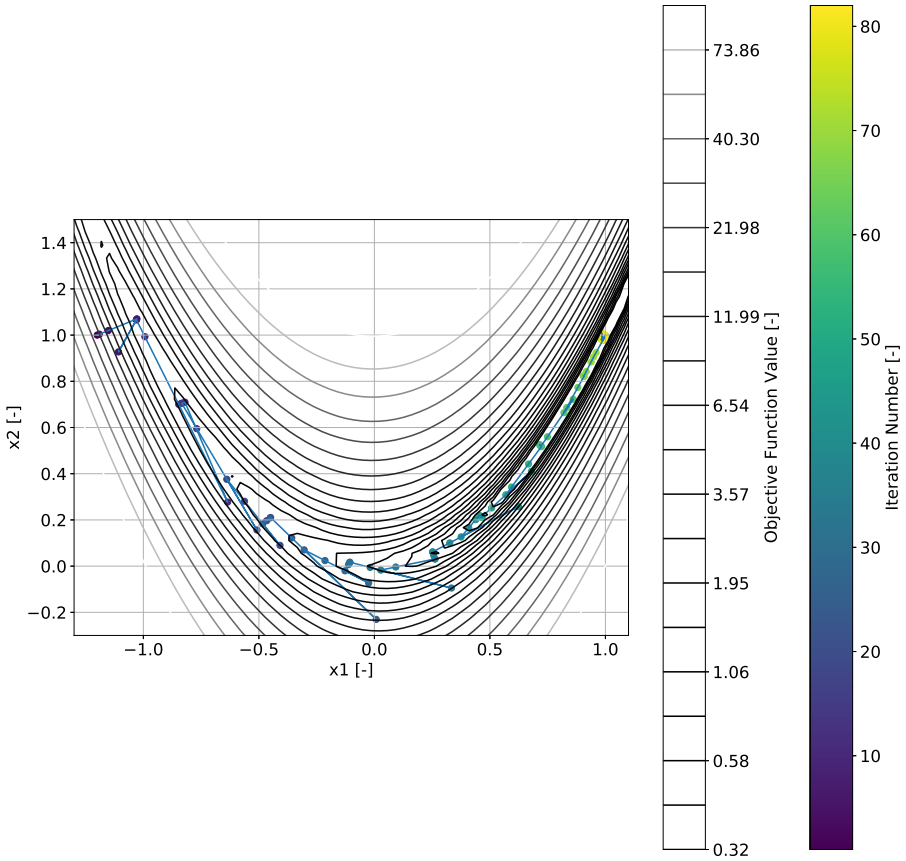


FIGURE 5.1: Exploration of the parameter space for the Rosenbrock function using the conjugate gradient method.

the reason why these methods are best suited for an efficient search of a local minimum in the vicinity of the initial point. The algorithm of the pattern search method resembles the one of the trust region approach. At the current points x_k, the convergence is controlled; if the convergence is not yet reached the cost function $f(x_k)$ is evaluated at the current points. Afterwards, a step size s_k is determined. If we fulfill the condition $f(x_k) > f(x_k + s_k)$, then we set the next points x_{k+1} to $x_k + s_k$. Otherwise, we keep the current points. Finally, we update the search pattern (Adams et al., 2017; Goldberg, 1989; Geem, 2012).

Fig. 5.2 shows the walkthrough behavior of the pattern search algorithm for the parameter space of the Rosenbrock function. We used the same settings as Dakota in their example file for the pattern search algorithm (Adams et al., 2017). Meaning that we have an initial stencil size of 0.5, a contraction factor of 0.75, and a tolerance of 10^{-4} for both the stopping criterion based on the objective function value and the one based on the step length. Furthermore, the maximum number of iterations is 1,000, and the maximum number of function evaluations 2,000. The lower bound of both variables is -2.0 and the upper bound 2.0.

We start with a large initial stencil and allow it to shrink from iteration to iteration. We observe that the pattern search can find the minimum of the Rosenbrock function, but it requires many more forward simulations than the conjugate gradient method. The parameter dimension strongly influences this behavior. Note that the Rosenbrock function has two parameters only. So, for real-case applications with a significantly higher parameter space, this leads to an even bigger difference in the performance of both methods. Regarding the search behavior, we can clearly identify the used stencil and how its size decreases towards the minimum of the Rosenbrock function.

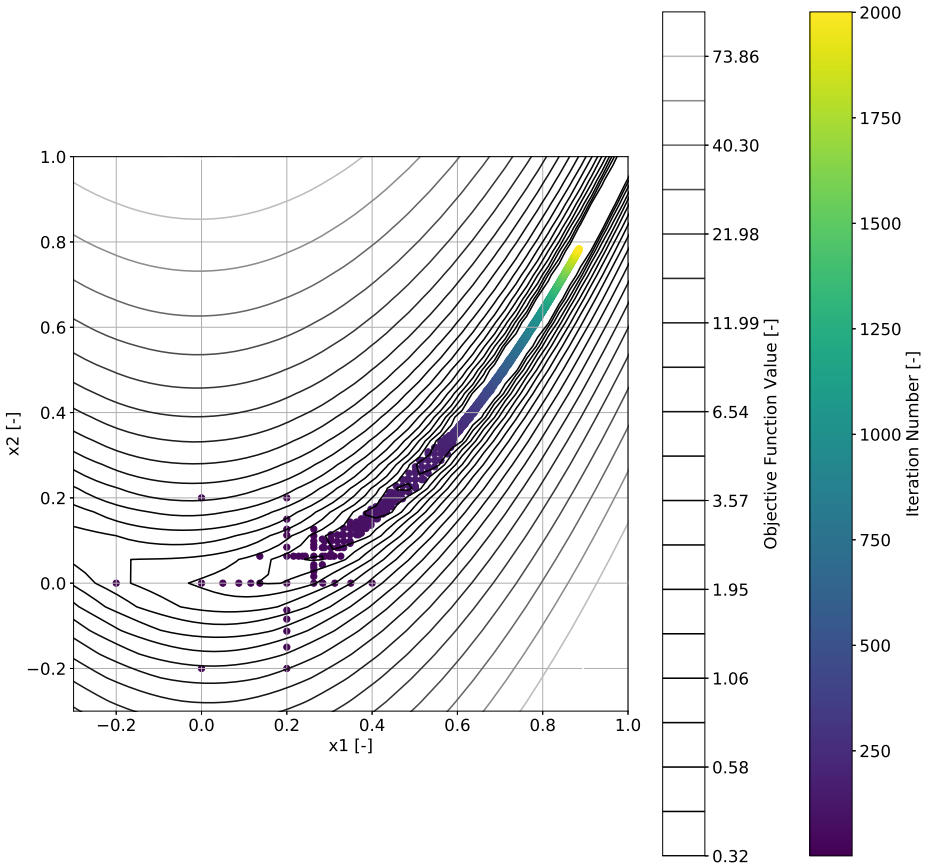


FIGURE 5.2: Exploration of the parameter space for the Rosenbrock function using a pattern search algorithm.

5.2.3 Global Derivative-Free Methods

Global derivative-free methods have similar advantages and disadvantages as the local derivative-free methods. The further advantage is that they can overstep local minima. Hence, they find the global minimum. For this thesis, we chose the evolutionary algorithm as the method to demonstrate the benefits and shortcomings of this method class for real-case applications. The evolutionary algorithm follows Darwin's law of

the survival of the “fittest”. It consists of four steps: the initialization, the selection, the reproduction, and the termination (see Fig. 5.3). During the initialization, one selects the parameters randomly. The selection follows, where we evaluate the objective functions for all chosen parameters. These parameters form a sequence, similar to the DNA, and only the parameters that have the lowest objective function value “survive" and evolve further. The evolution of the parameters is realized in the reproduction step. One distinguishes between two mechanisms for the reproduction:

1. crossover, meaning we combine several parameters to a new parameter, and
2. mutation, here we generate new parameters by applying small changes to individual parameters.

We repeat these steps until we meet a pre-defined termination criterion (Adams et al., 2017; Goldberg, 1989; Geem, 2012; Larrañaga and Lozano, 2001).

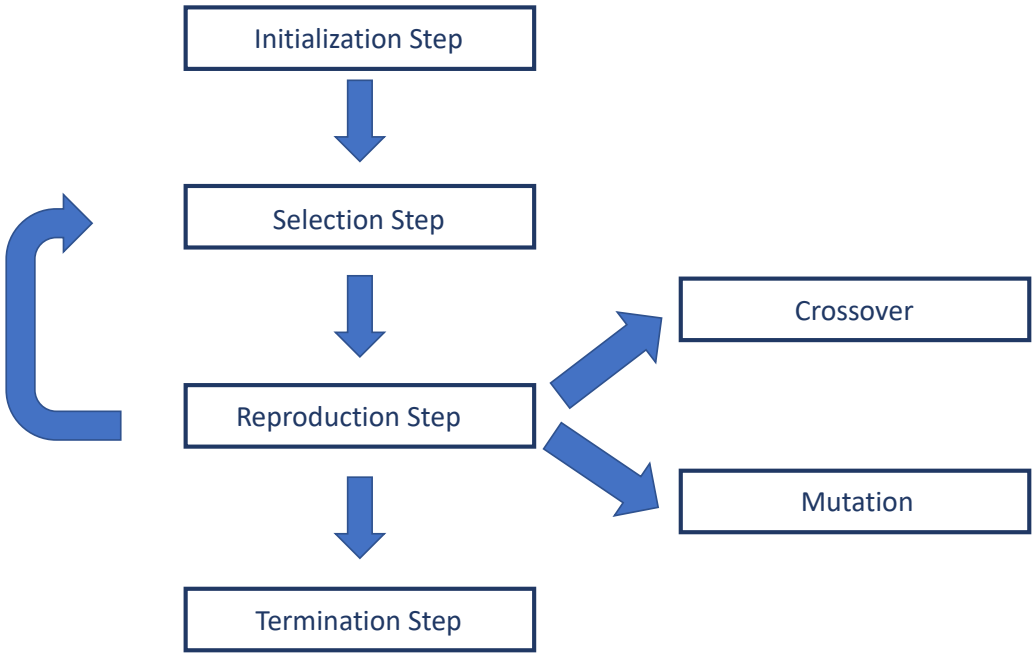


FIGURE 5.3: Graphical representation of the evolutionary algorithm.

Fig. 5.4 shows the walkthrough behavior of the evolutionary algorithm for the parameter space of the Rosenbrock function. We used the same settings as Dakota in their example file for the evolutionary algorithm (Adams et al., 2017). Hence, the population size is 50, the maximum number of iterations 100, the maximum number of function evaluations 2,000, and the mutation rate 1.0, where the mutation offset follows a normal distribution. The lower bound of both variables is -2.0 and the upper bound 2.0. The fitness type is a merit function, which considers both the aim of minimizing the objective function and satisfying the constraints. We use a two-point crossover type with a crossover rate of 0.0. Furthermore, we generate a new population by using replacement with a replacement size of 10.

Again, the method is capable of finding the minimum of the Rosenbrock function. However, at a computational cost that is higher than the one of the pattern search

method. Keep in mind that the Rosenbrock function possesses one minimum; hence, it is not surprising that the conjugate gradient method performs best for this example. Regarding, the search behavior, we nicely see the random component of the method and the clustering of the parameter distribution at and around the minimum.

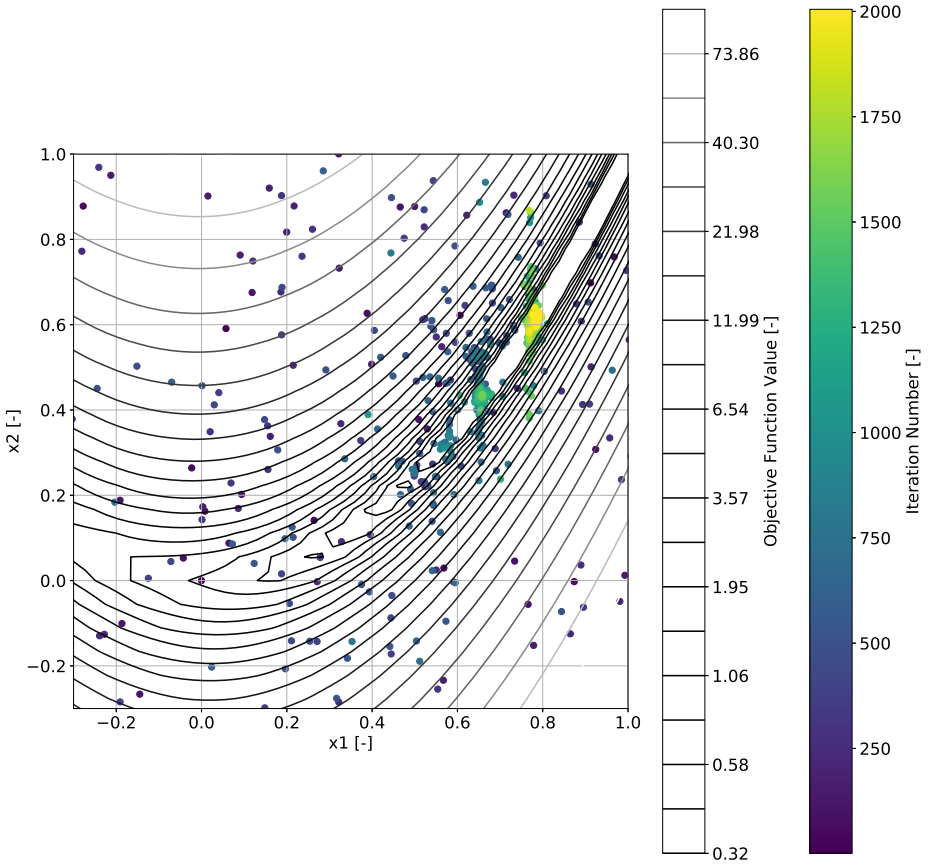


FIGURE 5.4: Exploration of the parameter space for the Rosenbrock function using an evolutionary algorithm.

5.3 Brandenburg Models

In this thesis, we are using three different versions of the Brandenburg model. The first version, from now on denoted as Brandenburg LAB model, has already been presented in Noack et al. (2012) and can be seen in Figure 5.5. It has an extension of 250 km in the x- and of 210 km in the y-direction and extends down to the lithosphere-asthenosphere boundary (LAB). The model consists of 15 lithological units and the mesh consists of deformed eight-noded prisms. The grid resolution is one km in the horizontal directions, whereas the vertical length of the layers corresponds to the vertical element length, resulting in a mesh with 840,000 degrees of freedom.

The second model, referred to in the following as the Brandenburg 6 km model (Figure 5.6), has the same horizontal extent but extends to a depth of 6 km instead of down to the LAB. It is presented in Noack et al. (2013) and consists of 12 lithological units. The model is discretized into a tetrahedron mesh. In comparison, to the

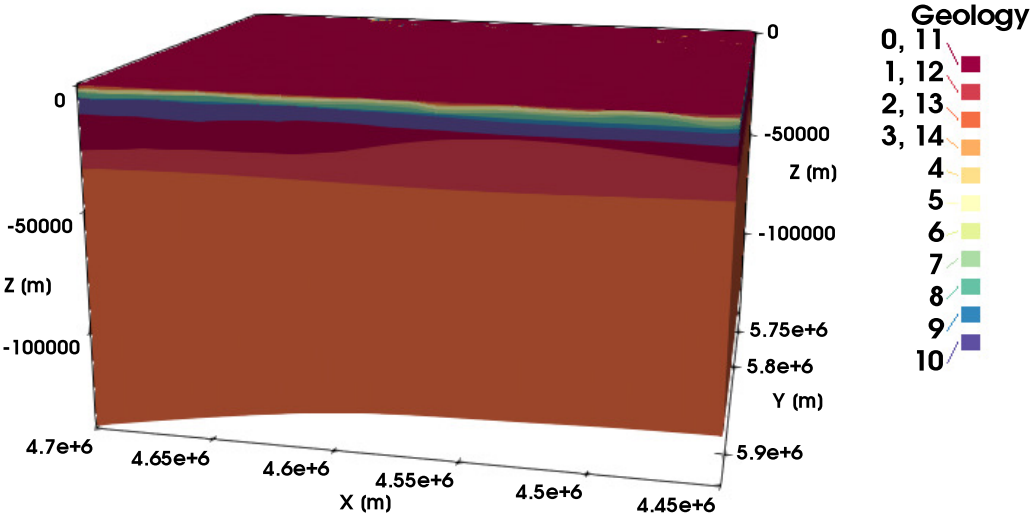


FIGURE 5.5: Geology of the Brandenburg LAB model – 0: Quaternary, 1: Tertiary, 2: Upper Cretaceous, 3: Lower Cretaceous, 4: Jurassic, 5: Keuper, 6: Muschelkalk, 7: Buntsandstein, 8: Zechstein, 9: Sedimentary Rotliegend, 10: Permo-Carboniferous Volcanics, 11: Pre-permian, 12: Upper Crust, 13: Lower Crust, 14: Lithospheric Mantle.

Brandenburg LAB model, it is refined in both geological and grid resolution terms. We have a horizontal element resolution of 0.22 km^2 and a vertical resolution that is interpolated from the z-evaluations of the geological layers with a minimum thickness of 0.1 m, resulting in a mesh of 1,546,675 degrees of freedom.

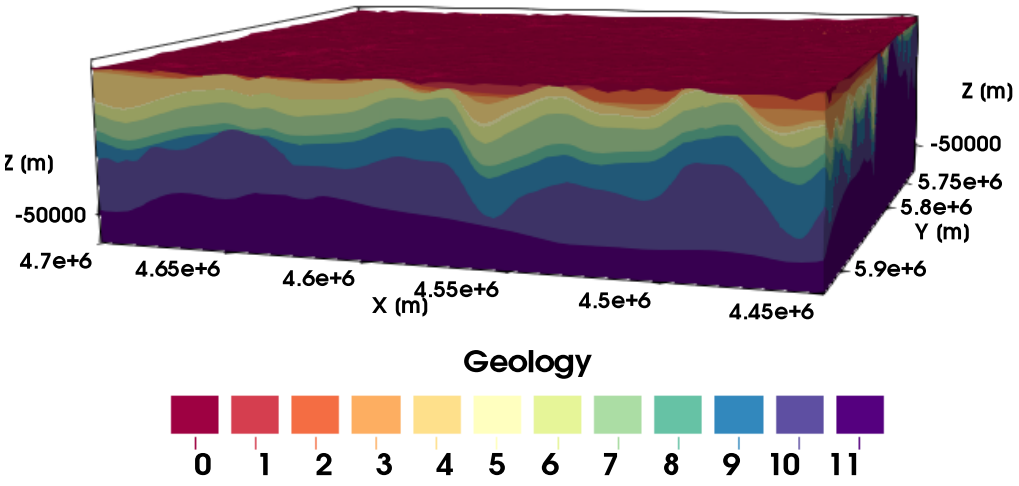


FIGURE 5.6: Geology of the Brandenburg 6 km model – 0: Quaternary, 1: Tertiary-post-Rupelian, 2: Tertiary Rupelian-clay, 3: Tertiary-pre-Rupelian-clay, 4: Upper Cretaceous, 5: Lower Cretaceous, 6: Jurassic, 7: Keuper, 8: Muschelkalk, 9: Buntsandstein, 10: Zechstein, 11: Basement.

Combining the Brandenburg 6 km model, the Brandenburg LAB model, and removing the minimal vertical thickness of 0.1 m results in the third version of the Brandenburg model, denoted as Brandenburg combined model (Figure 5.7). Consequently, this model consists of 17 geological layers, where the upper 11 layers have the same resolution as in the Brandenburg 6 km model. The lower six layers have the same vertical resolution of the Brandenburg LAB model and the same horizontal resolution

as the Brandenburg 6 km model. This results in a tetrahedron mesh with 2,141,550 degrees of freedom.

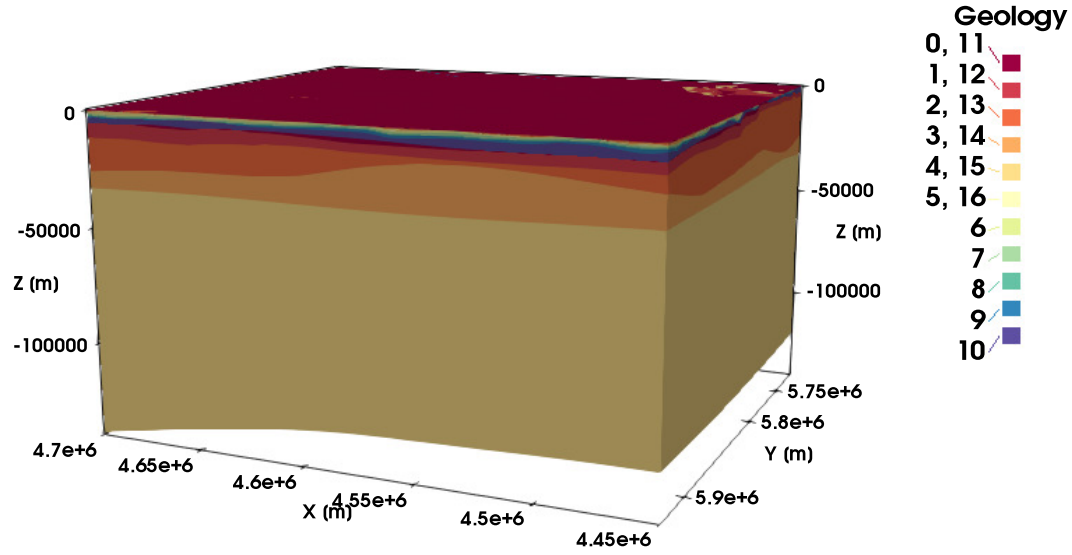


FIGURE 5.7: Geology of the Brandenburg combined model – 0: Quaternary, 1: Tertiary-post-Rupelian, 2: Tertiary Rupelian-clay, 3: Tertiary-pre-Rupelian-clay, 4: Upper Cretaceous, 5: Lower Cretaceous, 6: Jurassic, 7: Keuper, 8: Muschelkalk, 9: Buntsandstein, 10: Zechstein, 11: Sedimentary Rotliegend, 12: Permo-Carboniferous Volcanics, 13: Pre-permian, 14: Upper Crust, 15: Lower Crust, 16: Lithospheric Mantle.

For the validation of the models we are using temperature measurements presented in Noack et al. (2012, 2013) and corrected after Förster (2001). The temperature consists of 81 temperature measurements from 44 wells in the area of Brandenburg. It has been measured at various depth and stratigraphic levels.

The calibration of the Brandenburg 6 km model is challenging because of the lower boundary condition. The conventional way to define this boundary condition is to extract it from the Brandenburg LAB model and apply it to the Brandenburg 6 km model. However, which version of the Brandenburg LAB model do we choose? The calibrated or the uncalibrated and is the calibration also valid for the Brandenburg 6 km model? First, the calibration for the larger model is not valid for its smaller version, because of the refinement of the smaller model. Nonetheless, we remain with the question of how to define the lower boundary condition. Therefore, we compare the model calibration using the boundary condition from two uncalibrated Brandenburg LAB model versions and various hierarchical model calibrations. For the hierarchical models, we chose as the lower boundary condition either the boundary condition from the calibrated BC or a boundary condition obtained via Kriging.

We set the reference thermal conductivity λ_{ref} to the maximum thermal conductivity of the Brandenburg LAB model of 3.95 W m^{-1} K^{-1}. For the Brandenburg LAB and the Brandenburg combined model, the maximum temperature of 1300 °C is the reference temperature T_{ref}, whereas for the Brandenburg 6 km model a reference temperature of 8 °C is chosen. As previously mentioned, the RB method achieves a better performance in case of homogeneous Dirichlet boundary conditions (Degen et al., 2020c). The Brandenburg 6 km model has a constant Dirichlet boundary condition at the top. At the bottom, the model has a Dirichlet boundary condition with a different temperature value for each element. We set the top boundary condition to zero by using the value of the top boundary as our reference parameter. The bottom boundary condition is set to zero via a lifting function. In case of the Brandenburg LAB and combined

model we have for both Dirichlet boundary conditions constant values, hence we can use both of them as our reference parameter. We chose the value of the lower boundary condition to better reduce the magnitude of the temperatures, which yields a better performance. The maximum radiogenic heat production of the Brandenburg LAB model of 2.5 μW m^3 is the reference radiogenic heat production S_{ref}. The reference length l_{ref} corresponds to the maximum x-extent of all models (250,000 m).

For both the Brandenburg LAB and Brandenburg combined model, we apply a Dirichlet boundary condition of 8 °C, corresponding to the average annual temperature, at the top of the model. Moreover, we set a Dirichlet boundary condition of 1300 °C at the base of the LAB. Additionally, we allow a scaling of this boundary condition of $\pm$ 10 % to account for errors in the geometrical description of the LAB. The Brandenburg 6 km model has the same upper boundary condition, but at the bottom, we use various Dirichlet boundary conditions derived from the Brandenburg LAB model. All thermal properties are summarized in Table 5.1. The forward simulations are performed using the DwarfElephant package (Degen et al., 2020c) with a linear and nonlinear solver tolerance of 10^{-10}. Due to the nondimensional nature of the problem, no preconditioners are needed for the finite element solves.

For all Brandenburg models, the bilinear form a has the following decomposition:

$$a(w, v; \lambda) = -\sum_{q=0}^{n} \lambda_q \int_{\Omega} \nabla w \, \nabla v \, d\Omega, \qquad \forall v, w \in X, \; \forall \lambda \in \mathcal{D}, \tag{5.5}$$

where w is the trial function, v the test function, the index "q" denotes the number of the training parameter (for more information see Tab. 5.1), X the function space ($H_0^1(\Omega) \subset X \subset H_1(\Omega)$), Ω the spatial domain in $\mathbb{R}^3$, and D the parameter domain in $\mathbb{R}^n$. We denoted the number of trained thermal conductivities with n. Consequently, n equals thirteen, nine, and fourteen for the Brandenburg LAB, Brandenburg 6 km, and Brandenburg combined model, respectively.

For all Brandenburg models, except the Brandenburg 6 km model with a lower boundary condition derived via Kriging, the linear form f is decomposed in the following way:

$$\begin{aligned} f(v; \lambda, s) = &-\sum_{q=0}^{n} \lambda_q \, s \int_{\Gamma} \nabla v \; g(x, y, z) \; d\Gamma + \\ &s \int_{\Gamma} \nabla v \; S \; d\Gamma, \qquad \forall v \in X, \; \forall \lambda \in \mathcal{D}, \\ \text{with } \mathrm{g(x, y, z)} = T_{top} &\frac{h(x, y, z) - z_{bottom}(x, y)}{d(x, y)}. \end{aligned} \tag{5.6}$$

Here, Γ is the boundary in $\mathbb{R}^3$, s the scaling parameter for the lower boundary condition, $g(x, y, z)$ the lifting function, T_{top} the temperature at the top of the model, $h(x, y, z)$ the location in the model, $z_{bottom}(x, y)$ the depth of the bottom surface, and $d(x, y)$ the distance between the bottom and top surface.

For the Brandenburg 6 km with a Kriging lower boundary condition, the linear form slightly changes to the following:

$$
\begin{aligned}
f(v;\lambda,s) = &-\sum_{q=0}^{8}\sum_{i=0}^{3}\lambda_q\ s_i \int_\Gamma \nabla v\ g_i(x,y,z)\ d\Gamma + \\
&s_2 \int_\Gamma \nabla v\ S\ d\Gamma, \quad \forall v \in X,\ \forall \lambda \in \mathcal{D}, \\
\text{with } \mathrm{g_1(x,y,z)} = g_3(x,y,z) &= 1 - \frac{h(x,y,z) - z_{bottom}(x,y)}{d(x,y)}, \\
g_2(x,y,z) = &(\frac{3\ d(x,y)}{2a} - \frac{1}{2}(\frac{d(x,y)}{a})^3) \\
&(1 - \frac{h(x,y,z) - z_{bottom}(x,y)}{d(x,y)}).
\end{aligned}
\tag{5.7}
$$

Here g_1, g_2, and g_3 are again the lifting functions, with s_1 being the nugget, s_2 the partial sill, s_3 the scaling parameter for the mean temperature, and a the range.

5.4 Results

We first present the results from the investigation of the influence of the model dimension. Afterwards, we provide a comparison of various model calibration techniques to emphasize, which type of model calibrations are suitable for geothermal applications.

5.4.1 Influence of the Model Dimension on the Model Predictions

First, we present the results of the global sensitivity analysis. Afterwards, we show how we incorporate these results into the model calibration and how the model calibrations differ for the various versions of the Brandenburg model. As the calibration method, we are using the trust region reflective method, as discussed in Section 4.2.3. Finally, we present a brief overview of the computational cost associated with all analyses.

Sensitivity Analysis

The sensitivity analyses are performed with 13 (Brandenburg LAB model – Fig. 5.5), 11 (Brandenburg 6 km model – Fig. 5.6), 14 parameters (Brandenburg combined model – Fig. 5.7) and with 10,000 realizations for each parameter to reduce the statistical error. Note that for the Brandenburg 6 km model we show exemplarily the results using the Kriging lower boundary condition. The results of the sensitivity analyses using the other boundary conditions are analog to the one shown in this thesis. We only vary thermal conductivities in this section. We fix the radiogenic heat productions, to reduce the number of parameters within the reduction and all further analyses. We fix them and not the thermal conductivities because their influence on the overall temperature distribution is smaller. We allow a variation of $\pm$ 50 % from the initial thermal conductivities. Also, for the nugget and the partial sill, we allow a variation of $\pm$ 50 %. For the scaling parameter of the lower boundary of both the Brandenburg LAB model and Brandenburg combined model we allow a variation $\pm$ 10 % and for the scaling parameter of the mean temperature at the lower boundary condition of the Brandenburg 6 km model $\pm$ 20 %, in order to account for the uncertainties related to those boundary conditions.

Regarding the sensitivities, the Brandenburg LAB (Fig. 5.8) is mostly influenced by the Lower Cretaceous/Jurassic/Buntsandstein layer. The first-order sensitivity index is dominant over the higher-order indices. Furthermore, the model is sensitive to the Quaternary/Tertiary layer and the Lithospheric Mantle. For the Quaternary/Tertiary layer, we again have predominantly first-order influences, whereas the Lithospheric Mantle mostly impacts through higher-order contributions. Less pronounced are the influences from the Keuper, Zechstein layers and the scaling parameter of the lower boundary condition. The observed influences have similar first- and higher-order contributions. The other layers have only a minor influence on the model.

For the Brandenburg combined model (Fig. 5.9), we observe a similar pattern. The highest influences, dominated by first-order contributions, are arising from the Lower Cretaceous/Jurassic/Buntsandstein layer. The influence of both the Lithospheric Mantle and the scaling parameter of the lower boundary condition increased, but higher-order contributions still dominate both parameters. The Tertiary-pre-Rupelian-clay/Upper Cretaceous, the Keuper, and the Zechstein layers have a minor influence on the model and have similar first- and higher-order contributions.

The Brandenburg 6 km model is only influenced by the Basement layer and by the variability of the lower boundary condition (Fig. 5.10). The influence of the scaling parameter of the mean temperature is significantly higher than the one from the Basement layer. Higher-order contributions dominate both parameters. Note that the Basement layer has nearly no first-order contributions, whereas the scaling parameter has non-dominant first-order contributions.

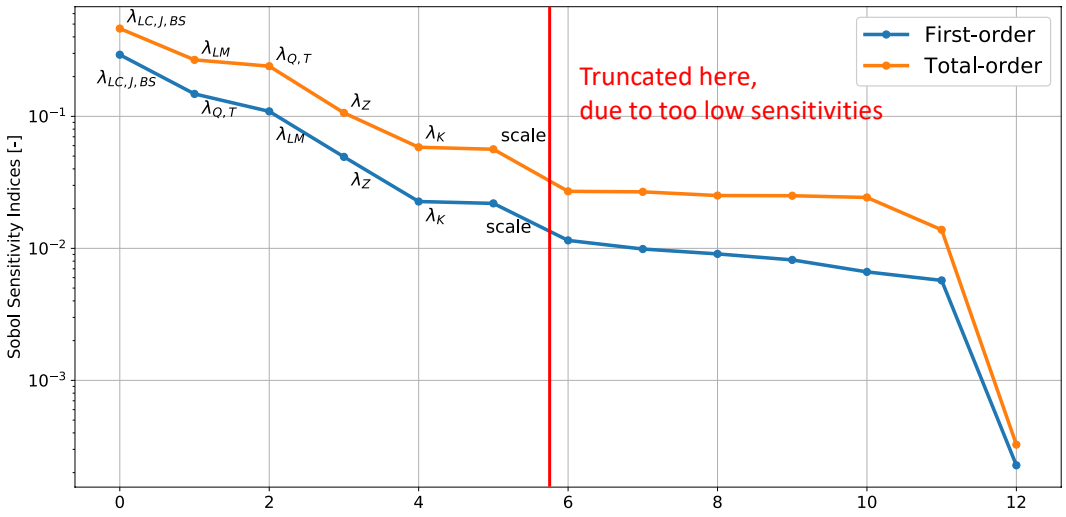


FIGURE 5.8: Global Sensitivity analysis for the Brandenburg LAB model. We show the first- (blue) and total-order contributions (orange).

Model Calibration

We take the results from the global sensitivity analysis as an input for the following model calibration. Model calibration is necessary to account for model errors of the Brandenburg model. For the calibration, we use the smooth approximation of the L1 absolute value loss with a value of 0.1 for the soft margin between inlier and outlier residuals as the loss function. The tolerances for the termination are 10^{-5} for changes of the cost function, 10^{-8} for the norm of the gradient, and 10^{-5} for changes of the

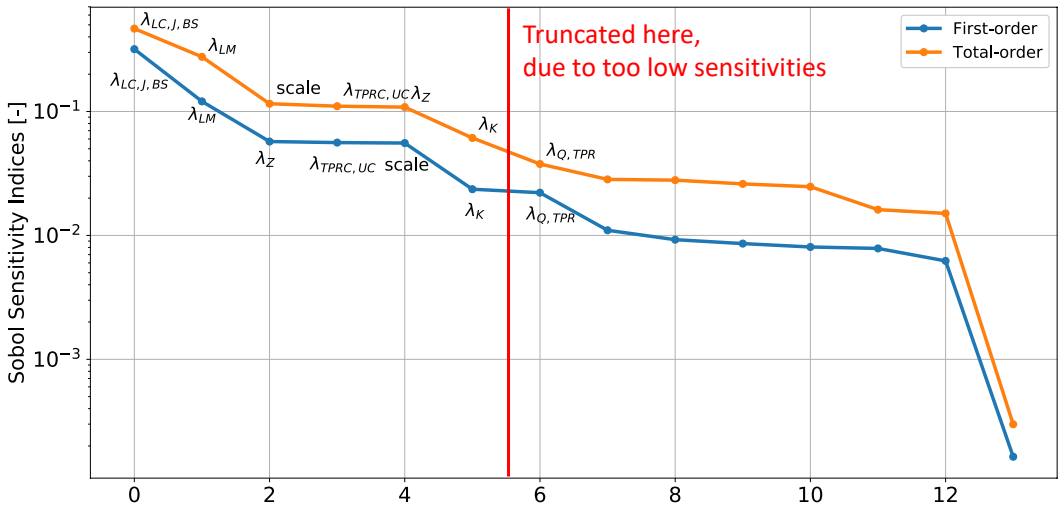


FIGURE 5.9: Global Sensitivity analysis for the Brandenburg combined model. We show the first- (blue) and total-order contributions (orange).

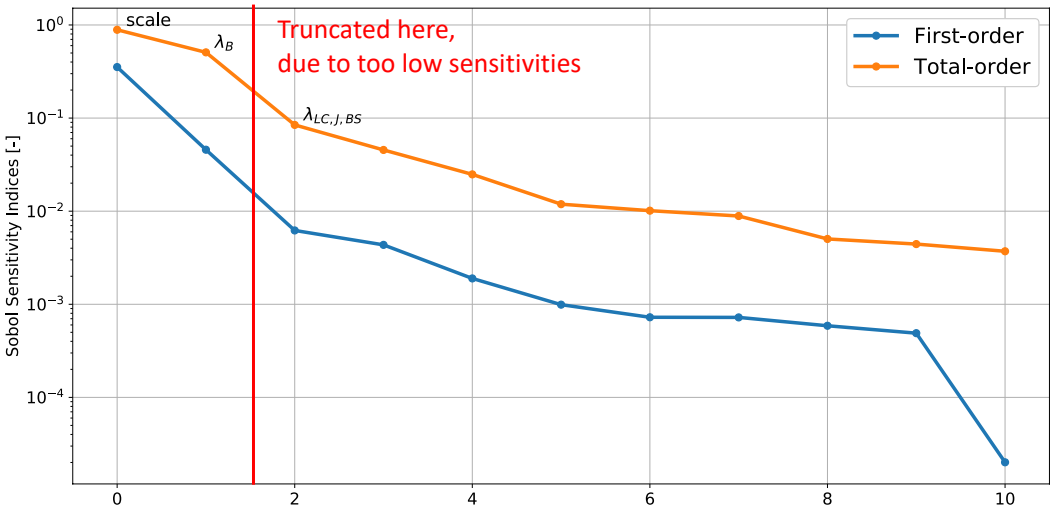


FIGURE 5.10: Global Sensitivity analysis for the Brandenburg 6 km model. We show the first- (blue) and total-order contributions (orange).

independent parameters. The calibrations use the results from the sensitivity analyses. Hence, we vary six parameters, two parameters, and six parameters within the calibration for the Brandenburg LAB (Fig. 5.11), Brandenburg 6 km (Fig. 5.12), and Brandenburg combined model (Fig. 5.13), respectively.

Using the initial thermal conductivities, we observe a deviation from the measured data up to a depth of 6 km for the Brandenburg LAB model. Below 6 km, we see a good agreement of simulated and measured data. After the calibration, we see a much better agreement between the simulated and measured value for the region above 6 km. Below 6 km, the agreement is slightly worse than before.

In Figure 5.12 we compare the model calibrations using various lower boundary conditions of the Brandenburg 6 km model. We show that the hierarchical model calibration with a Kriging lower boundary condition performs the best, although the other versions perform only slightly worse. Note that the hierarchical model calibration with

an extracted lower boundary condition from the Brandenburg LAB model performs better up to 3 km depth than the Kriging lower boundary condition. However, below 3 km depth, it performs significantly worse than the version with the Kriging boundary condition. In the case of the Brandenburg combined model, the calibration yields a very good agreement with the observed data throughout the entire model (Fig. 5.13).

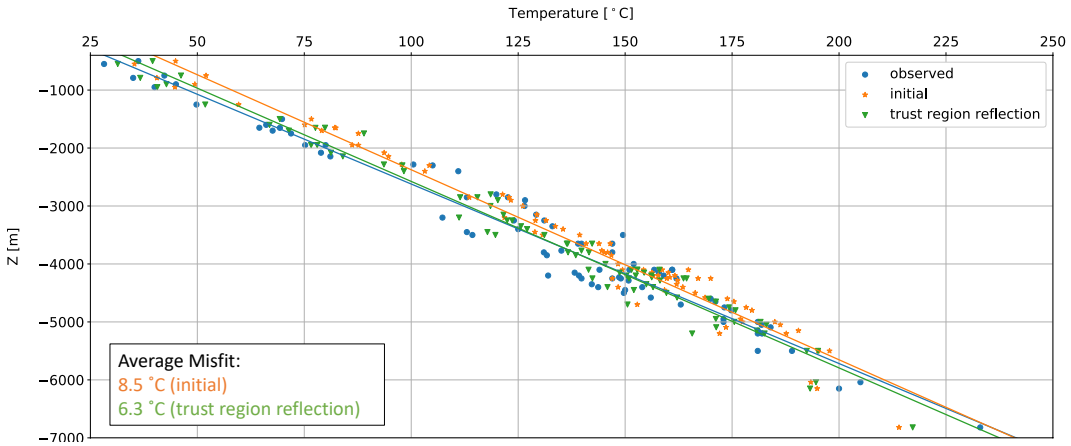


FIGURE 5.11: Observed temperature distribution (blue), temperature distribution for the initial parameter distribution (orange), and after the calibration (green) for the Brandenburg LAB model.

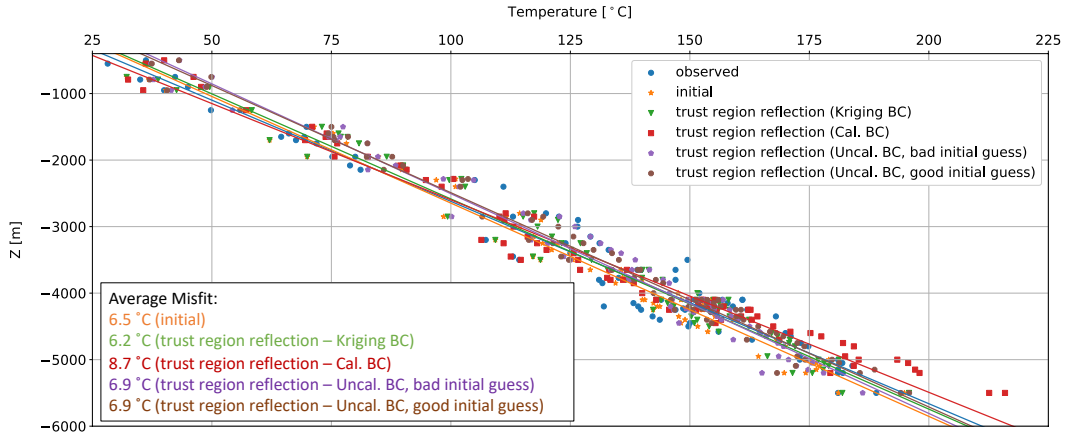


FIGURE 5.12: Observed temperature distribution (blue), temperature distribution for the initial parameter distribution (orange) and after the calibration: with a Kriging boundary condition (green), with the boundary condition from the calibrated model (red), with the boundary condition from the uncalibrated model and a bad initial guess (purple), and with a good initial guess (brown) for the Brandenburg 6 km model.

Table 5.1 presents the changes in the thermal conductivity between the initial and the best-calibrated values for all models. Regarding the Brandenburg LAB model, we observe an increase from 2.0 W m^{-1} K^{-1} to 2.63 W m^{-1} K^{-1} for the Lower Cretaceous-/Jurassic/Buntsandstein layers. For the Brandenburg combined model, we see a more pronounced increase, resulting in a value of 2.99 W m^{-1} K^{-1}. At the same time, we observe a significant decrease in thermal conductivity for the Keuper layer for the Brandenburg LAB and the Brandenburg combined model. Furthermore, a substantial increase in thermal conductivity is observed for the Lithospheric Mantle, resulting in conductivities of 5.93 W m^{-1} K^{-1} for the Brandenburg LAB and combined model. Additionally, the scaling parameter for the lower boundary condition shows an increase

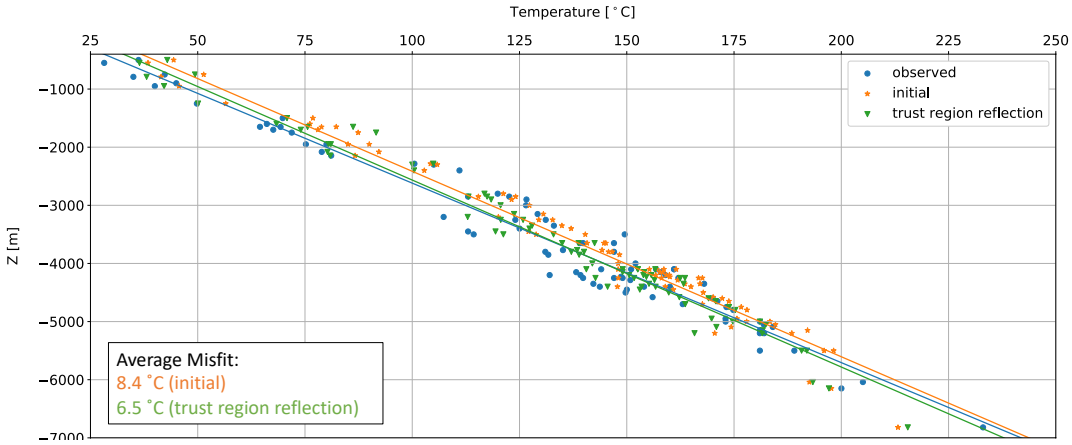


FIGURE 5.13: Observed temperature distribution (blue), temperature distribution for the initial parameter distribution (orange), and after the calibration (green) for the Brandenburg combined model.

to 1.10 for both the Brandenburg LAB model and Brandenburg combined model. Furthermore, we obtain increased thermal conductivities for the Zechstein layer of 4.03 W m^{-1} K^{-1} (Brandenburg LAB model), and 3.84 W m^{-1} K^{-1} (Brandenburg combined model). The calibration of the Brandenburg LAB model leads to an increased thermal conductivity of 1.89 W m^{-1} K^{-1} for the Quaternary/Tertiary layer, and the calibration of the Brandenburg combined model to an increased thermal conductivity of the Tertiary-pre-Rupelian-clay/Upper Cretaceous layer to 2.14 W m^{-1} K^{-1}. The parameter distribution for the Brandenburg 6 km model shows an increased thermal conductivity of 2.78 W m^{-1} K^{-1} for the Basement layer. The scaling factor for the mean temperature is 21.54 after the calibration resulting in a mean temperature of 180 °C.

The calibration leads in case of the Brandenburg LAB model to temperature changes of -162 °C to 3 °C with respect to the uncalibrated model (Fig. 5.14). In case of the Brandenburg 6 km model, the variations are smaller and range from -25 °C to 12 °C (Fig. 5.15). The Brandenburg combined model shows greater variations of -143 °C to 7 °C (Fig. 5.16) again. All RB based trust region reflection calibrations require less than 2 s (up to 15 iterations).

The temperature distribution (Fig. 5.17 - 5.19) is the same for all three models. We observe a steep temperature gradient for the uppermost part of the model and a less steep gradient for the lower part of the model. Note, that the Brandenburg 6 km model extends down to 6 km only, therefore only the steeper gradient is observable.

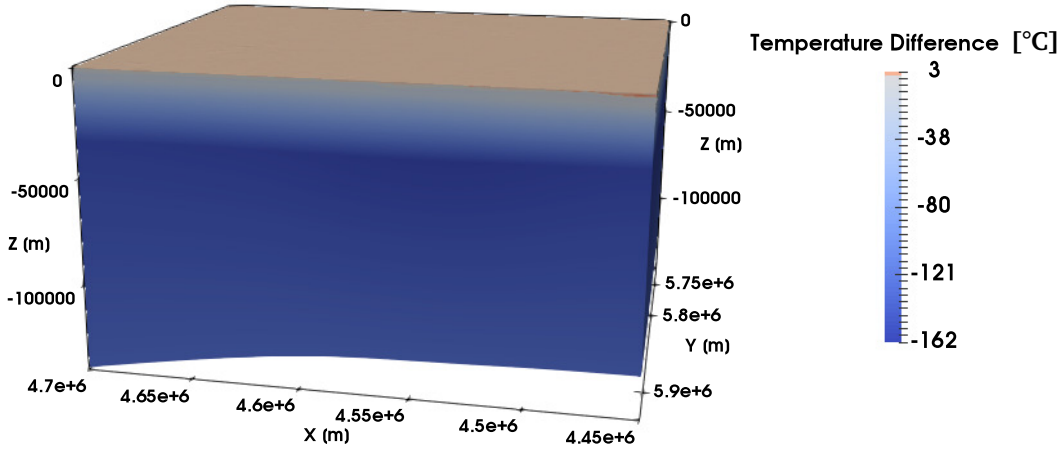


FIGURE 5.14: Resulting temperature difference between the initial and calibrated parameter distributions for the Brandenburg LAB model.

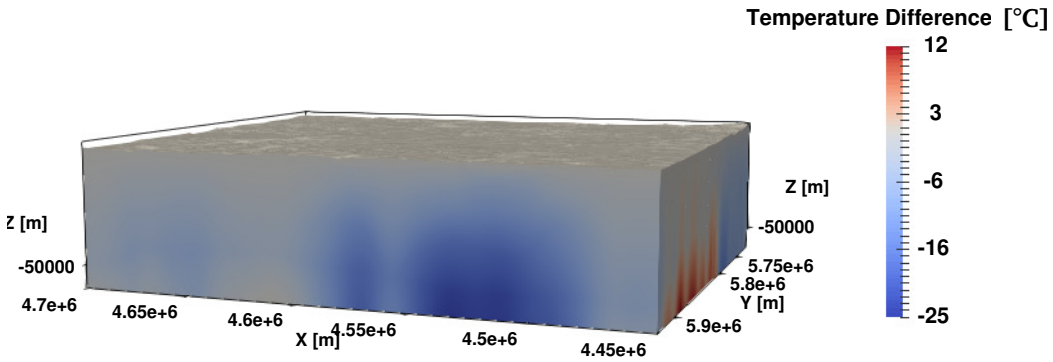


FIGURE 5.15: Resulting temperature difference between the initial and calibrated parameter distributions for the Brandenburg 6 km model.

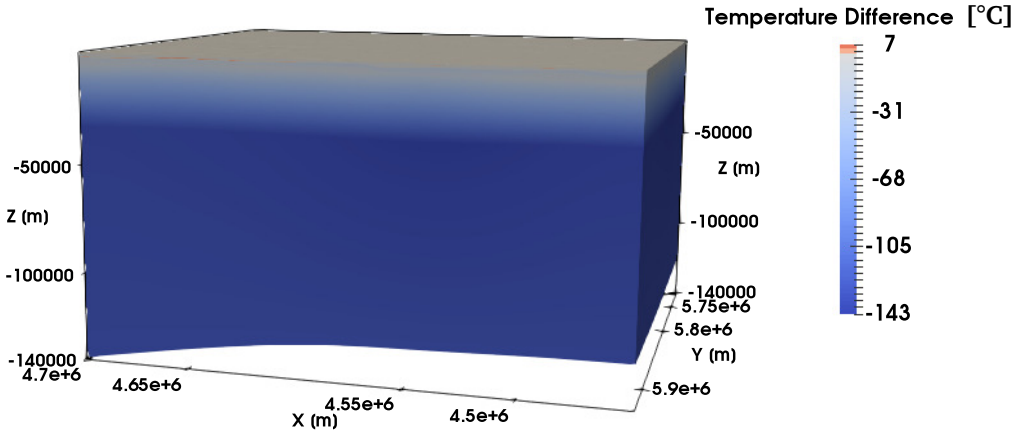


FIGURE 5.16: Resulting temperature difference between the initial and calibrated parameter distributions for the Brandenburg combined model.

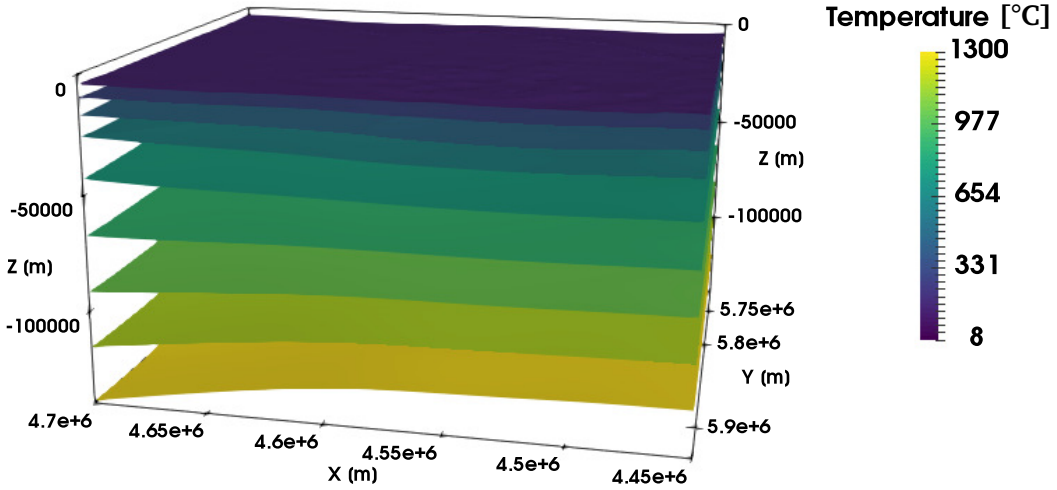


FIGURE 5.17: Initial temperature distribution of the Brandenburg LAB model.

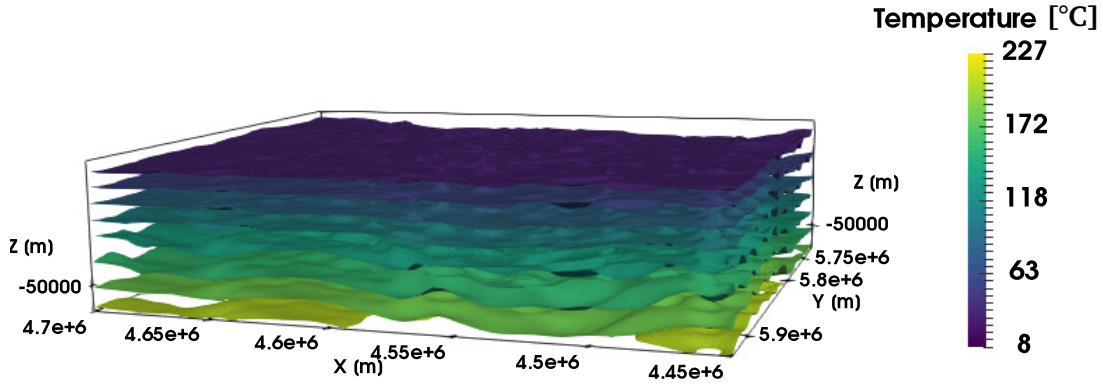


FIGURE 5.18: Initial temperature distribution of the Brandenburg 6 km model.

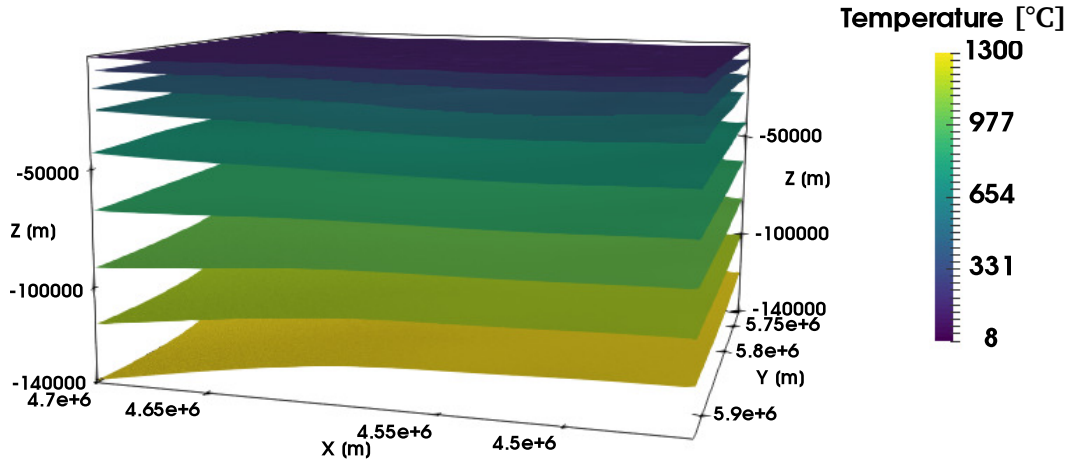


FIGURE 5.19: Initial temperature distribution of the Brandenburg combined model.

TABLE 5.1: Thermal Properties of all models for and after the automated model calibration. The initial thermal properties are from Noack et al. (2012, 2013). We denote all parameters that are not involved in the model calibration, due to too low sensitivities or that are not applicable for the specific model version, with n/a. Additionally, the training parameter ID is provided.

ID	Layer	Training Parameter			Radiogenic Heat Production $[\mu W m^{-3}]$	Initial Thermal Conductivity $[W m^{-1} K^{-1}]$	Calibrated Thermal Conductivity $[W m^{-1} K^{-1}]$		
		BRB LAB	BRB 6 km	BRB combined			BRB LAB	BRB 6 km	BRB combined
Q	Quaternary	0	0	0	0.7	1.50	1.89	n/a	n/a
T	Tertiary	0	n/a	n/a	0.7	1.50	1.89	n/a	n/a
TPR	Tertiary-post-Rupelian	n/a	0	0	0.7	1.50	n/a	n/a	n/a
TRC	Tertiary Rupelian-clay	n/a	1	1	0.45	1.00	n/a	n/a	n/a
TPRC	Tertiary-pre-Rupelian-clay	n/a	2	2	0.3	1.90	n/a	n/a	2.14
UC	Upper Cretaceous	1	2	2	0.3	1.90	n/a	n/a	2.14
LC	Lower Cretaceous	2	3	3	1.4	2.00	2.63	n/a	2.99
J	Jurassic	2	3	3	1.4	2.00	2.63	n/a	2.99
K	Keuper	3	4	4	1.4	2.30	1.77	n/a	1.52

M	Muschel-kalk	4	5	5	0.3	1.85	n/a	n/a	n/a
BS	Bunt-sand-stein	2	3	3	1.0	2.0	2.63	n/a	2.99
Z	Zechstein	5	6	6	0.09	3.50	4.03	n/a	3.84
B	Basement	n/a	7	n/a	1.5	2.50	n/a	2.78	n/a
SR	Sedi-mentary Rot-liegend	6	n/a	7	1.0	2.16	n/a	n/a	n/a
PCV	Permo-Carboni-ferous Vol-canics	7	n/a	8	2.0	2.50	n/a	n/a	n/a
PP	Pre-permian	8	n/a	9	1.5	2.65	n/a	n/a	n/a
UCR	Upper crust	9	n/a	10	2.5	3.10	n/a	n/a	n/a
LCR	Lower crust	10	n/a	11	0.8	2.70	n/a	n/a	n/a
LM	Litho-spheric Mantle	11	n/a	12	0.03	3.95	5.93	-	5.93

	Parameter	Training Parameter			Initial Value [-]	Calibrated Value [-]		
		BRB LAB	BRB 6 km	BRB com-bined		BRB LAB	BRB 6 km	BRB com-bined
	Nugget	n/a	8	n/a	0.40	n/a	n/a	n/a
	Partial Sill	n/a	9	n/a	1.58	n/a	n/a	n/a
	Scale	12	10	13	1.00	1.10	21.54	1.10

Approximation Quality and Computational Cost

For the Brandenburg LAB and the Brandenburg combined model, we defined a relative error tolerance of $5{\cdot}10^{-4}$ and for the Brandenburg 6 km model a tolerance of $1{\cdot}10^{-3}$ for the model reduction. We reach these error tolerances in all cases (Fig. 5.20). Our most accurate measurements are 10^{-1} accurate, and the deepest measurement is at a depth of around 7 km. We chose these error tolerances to ensure that we do no introduce additional errors through the approximation in the entire model.

For the Brandenburg LAB model, we require 248 basis functions to reach our predefined error tolerance. This leads to a reduction of compute time for single forward simulation from 73 s to 4 ms. Hence, we have a speed-up of $1.81{\cdot}10^{4}$. Similar, we require 143 basis functions to describe the Brandenburg 6 km model and 273 for the Brandenburg combined model (Fig. 5.20). Hence, we reduce the compute time for the Brandenburg 6 km model from 526 s to 2 ms and for the Brandenburg combined model from 501 s to 5 ms. Consequently, we obtain speed-ups of $2.77{\cdot}10^{5}$ and $1.02{\cdot}10^{5}$ for the Brandenburg 6 km and the Brandenburg combined model, respectively. The simulations for the speed-up have been performed on a MacBook Pro (Intel Core i7, 2.5 GHz, 16 GB memory) using a single core. Using a physics-based machine learning approach resulted in computation times of 15.5 min (Brandenburg LAB model), 5.4 min (Brandenburg 6 km model), and 47.8 min (Brandenburg combined model) for the Sobol sensitivity analysis with 280,000, 240,000, and 300,000 function evaluations, respectively.

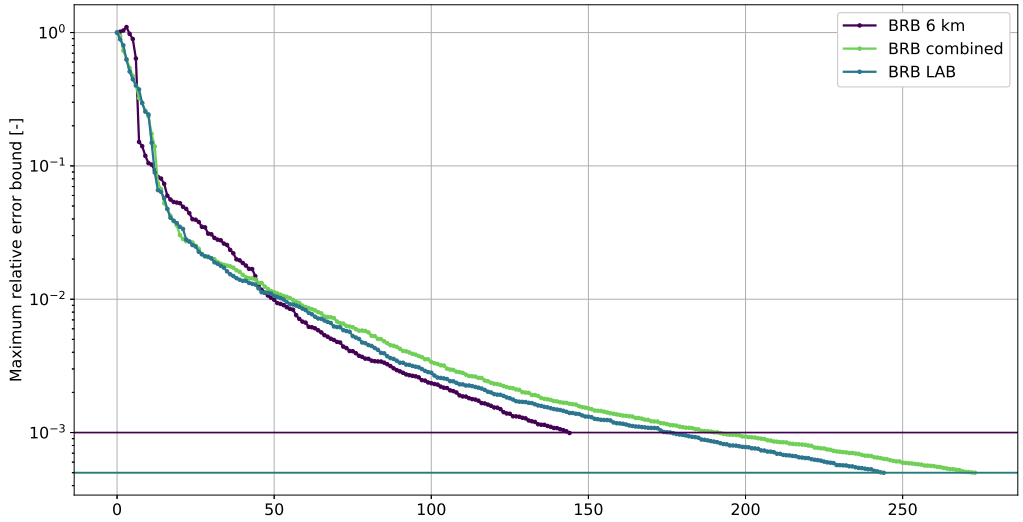


FIGURE 5.20: Convergence of the maximum relative error bound for the Brandenburg 6 km model (denoted in purple), the Brandenburg combined model (denoted in green), and the Brandenburg LAB model (denoted in blue). We are using an error tolerance of $1{\cdot}10^{-3}$ for the Brandenburg 6 km model, and an error tolerance of $5{\cdot}10^{-4}$ for the Brandenburg combined and LAB model.

5.4.2 Automated Model Calibration Comparison

Additionally, we want to compare the different calibration methods for the Brandenburg combined model. Therefore, we perform three model calibrations: conjugate gradient, pattern search, and evolutionary algorithm as representatives of the three main categories of local gradient-based, local derivative-free, and global methods. As

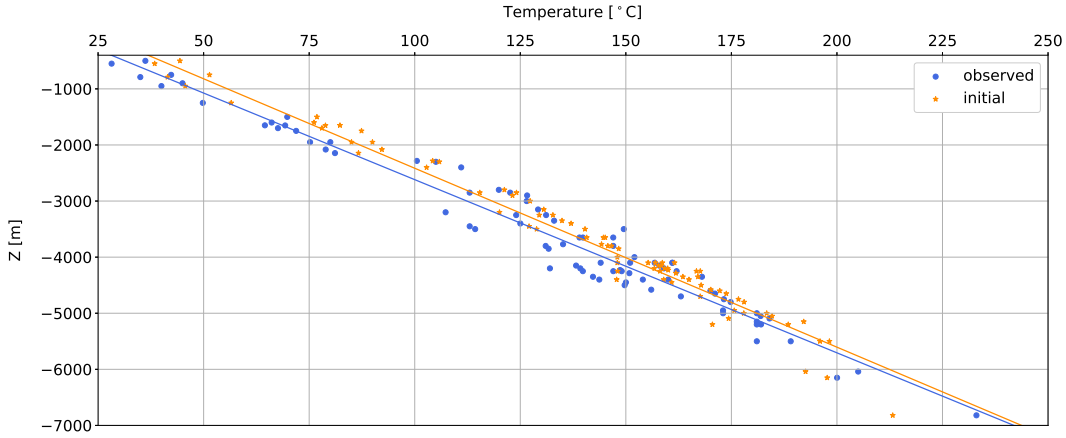


FIGURE 5.21: Displayed are the observed temperature data (blue) and the simulated temperatures using the initial parameter distribution (orange).

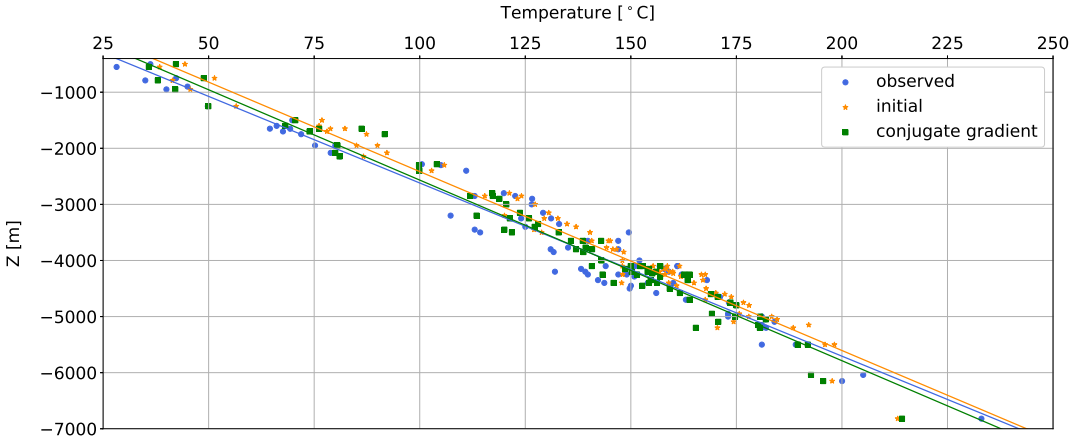


FIGURE 5.22: Displayed are the observed temperature data (blue) and the simulated temperatures using the initial parameter distribution (orange), and the parameter distribution obtained by the conjugate gradient method (green).

presented in the previous section, we again consider the results of the global sensitivity study for our model calibrations. Hence, we calibrate six instead of fourteen parameters.

In the previous section, we already presented the deviation of the simulated temperatures, obtained by the initial parameter distribution from the observed data (Fig. 5.21). This section aims to compare the various available calibrations methods.

We first discuss the differences between the methods concerning the resulting temperature distributions. The temperature distribution, resulting from the conjugate gradient method, is very similar to the one obtained by the trust region reflective method (Fig. 5.22). As for the trust region reflective method, the calibration leads to a broader spread of the temperature data and yields overall decreased temperature values in comparison to the initial temperature distribution. For the conjugate gradient method, we set the convergence tolerance to 10^{-5}, and the number of maximum iterations to 1,000. The gradients are calculated numerically with a central difference approach and a step size of 10^{-5}, in order to avoid errors caused by the numerical calculation of the gradients. Using these settings, we need 308 function evaluations to obtain the presented results. The operation requires about 3 s computation time.

The pattern search method results in a temperature distribution nearly identical to

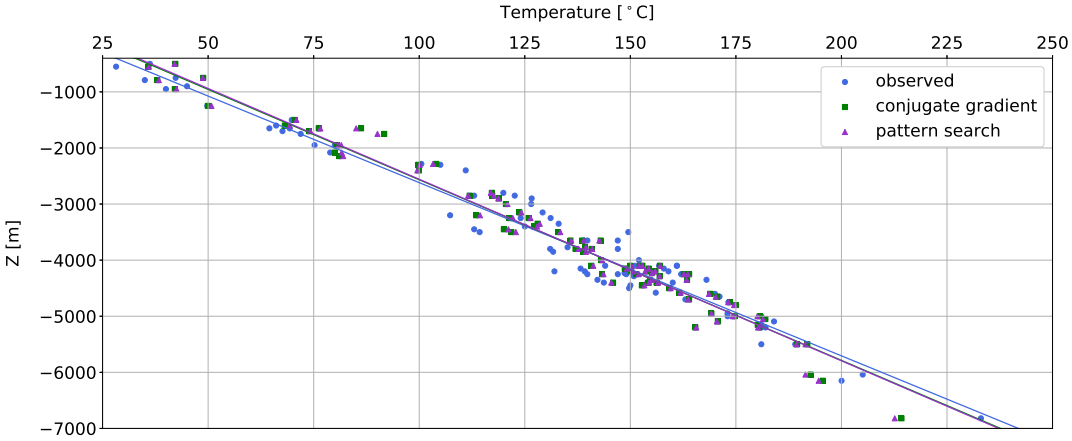


FIGURE 5.23: Displayed are the observed temperature data (blue) and the simulated temperatures using the parameter distribution obtained by the conjugate gradient method (green), and the parameter distribution obtained by the pattern search method (purple).

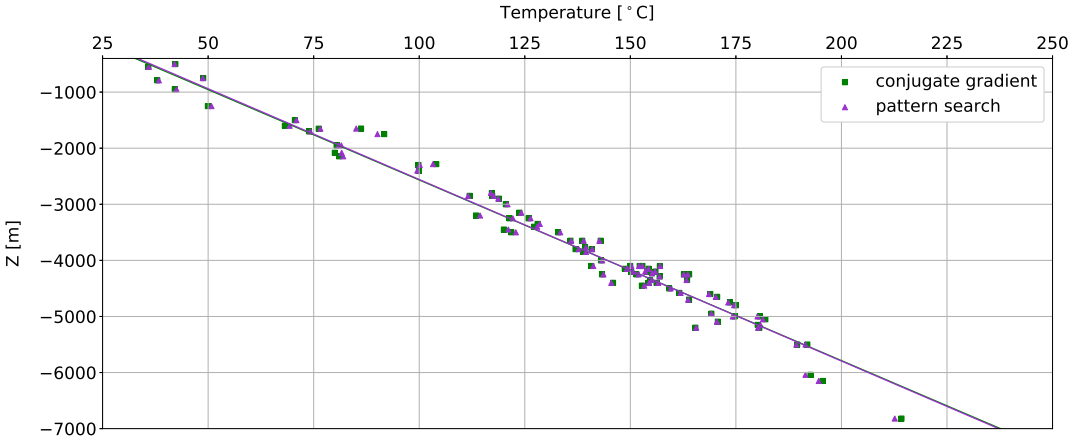


FIGURE 5.24: Displayed are the simulated temperatures using the parameter distribution obtained by the conjugate gradient method (green), and the parameter distribution obtained by the pattern search method (purple).

the one of the conjugate gradient method (Fig. 5.23 and 5.24). For the pattern search method, we set the maximum number of iterations to 5,000, the maximum number of function evaluations to 100,000, and the convergence tolerance to 10^{-5}. For the exploration of the parameter space, we use a stencil with an initial step size of 0.1 and a contraction factor of 0.75. In order to obtain the presented results, we require 51,044 function evaluations, which results in an execution time of 330 s.

Also, the evolutionary algorithm results in a nearly identical temperature distribution as both the temperature distributions of the conjugate gradient and pattern search method (Fig. 5.25 and 5.26). For the evolutionary algorithm, we set the maximum number of iterations to 10,000, the maximum number of function evaluations to 1,000,000, and the population size to 10,000. To gain reproducible results, we fix the random seed to 11,011,011. For the fitness function, we use a merit function because it balances both the aim of reducing the objective function and fulfilling the constraints (Adams et al., 2017). The mutation offset is a normal distribution with a mutation rate of 1.0. We use a two-point crossover-type – that combines the middle of one parent with the end of another parent– with a crossover rate of 0 (Adams et al., 2017). Furthermore, we create new populations by using the replacement. In order

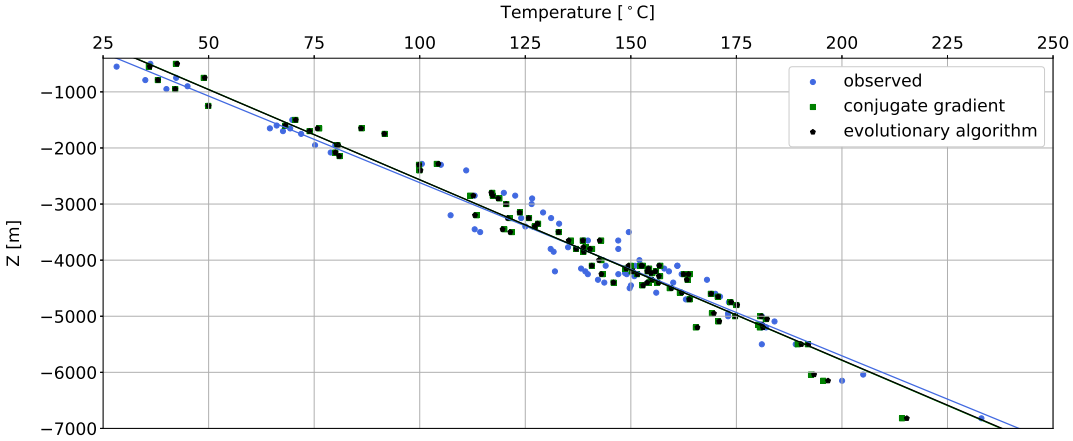


FIGURE 5.25: Displayed are the observed temperature data (blue) and the simulated temperatures using the parameter distribution obtained by the conjugate gradient method (green), and the parameter distribution obtained by the evolutionary algorithm (black).

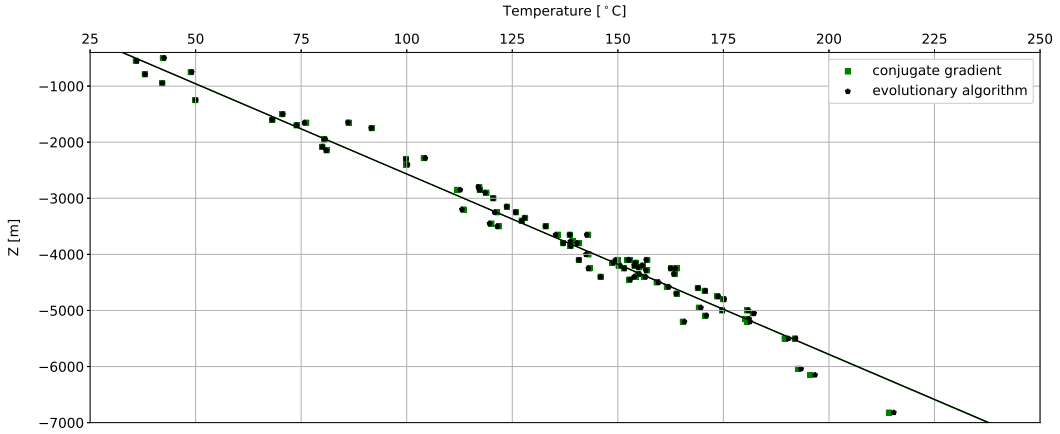


FIGURE 5.26: Displayed are the simulated temperatures using the parameter distribution obtained by the conjugate gradient method (green), and the parameter distribution obtained by the evolutionary algorithm(black).

to obtain the presented results, we required 1,009,000 function evaluations yielding an execution time of about 11,000 s (about 3 h).

Fig. 5.27 shows all temperature misfits with respect to the observed temperature data. Here, we can see that the conjugate gradient method and the evolutionary algorithm have the lowest mean misfit with a value of 6.5 °C. A slightly higher mean misfit is obtained for the pattern search method with 6.6 °C. The initial parameter distribution has with 8.4 °C a significantly higher mean misfit. The highest negative misfit is for all parameter distributions the same. For the highest positive misfit, we observe the same pattern as for the mean misfits.

Tab. 5.2 presents the parameter distributions obtained by the various calibration methods. As already seen in the previous section, we obtain extremely high thermal conductivities for the Lithospheric Mantle for all calibration methods. The one resulting from the evolutionary algorithm is the highest. We observe a significant increase in thermal conductivity for the Lower Cretaceous/Jurassic/Buntsandstein layers. In contrast, the Keuper layer shows a significant decrease in thermal conductivity.

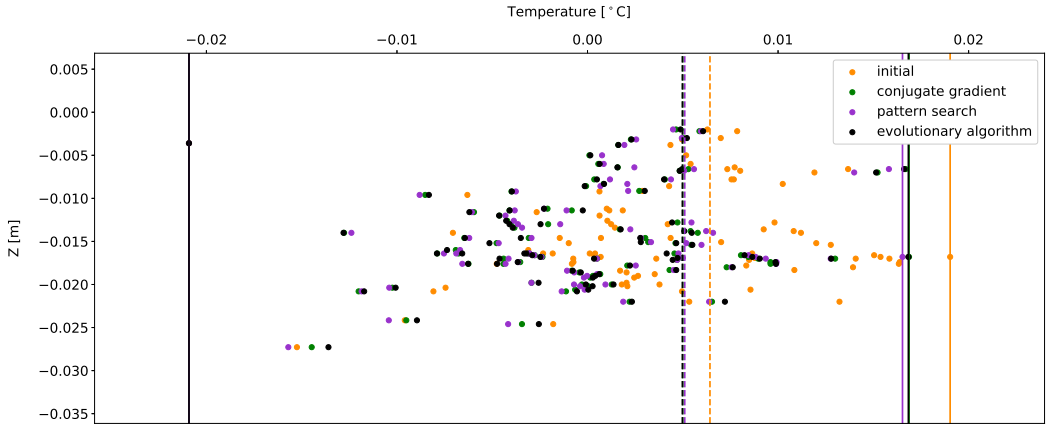


FIGURE 5.27: Displayed are the temperature misfits between the simulated temperatures obtained by the conjugate gradient method (green), pattern search method (purple), evolutionary algorithm (black), the initial parameter distribution (orange) and the observed temperatures.

TABLE 5.2: Thermal properties of the Brandenburg combined model before and after the model calibrations. The initial thermal properties are from Noack et al. (2012, 2013). We denote all parameters that are not involved in the model calibrations, due to too low sensitivities, with n/a. Additionally, the training parameter ID is provided.

Layer	Training Parameter	Initial Thermal Conductivity $[Wm^{-1}K^{-1}]$	Thermal Conductivity (Conjugate Gradient) $[Wm^{-1}K^{-1}]$	Thermal Conductivity (Pattern Search) $[Wm^{-1}K^{-1}]$	Thermal Conductivity (Evolutionary Algorithm) $[Wm^{-1}K^{-1}]$
Quaternary	0	1.50	n/a	n/a	n/a
Tertiary-post-Rupelian	0	1.50	n/a	n/a	n/a
Tertiary Rupelian-clay	1	1.00	n/a	n/a	n/a
Tertiary-pre-Rupelian-clay	2	1.90	2.07	1.99	2.15
Upper Cretaceous	2	1.90	2.07	1.99	2.15
Lower Cretaceous	3	2.00	2.95	2.75	2.99
Jurassic	3	2.00	2.95	2.75	2.99

Keuper	4	2.30	1.48	1.56	1.51
Muschelkalk	5	1.85	n/a	n/a	n/a
Buntsandstein	3	2.00	2.95	2.75	2.99
Zechstein	6	3.50	3.73	3.60	3.84
Sedimentary Rotliegend	7	2.16	n/a	n/a	n/a
Permo-Carboniferous Volcanics	8	2.50	n/a	n/a	n/a
Pre-permian	9	2.65	n/a	n/a	n/a
Upper crust	10	3.10	n/a	n/a	n/a
Lower crust	11	2.70	n/a	n/a	n/a
Lithospheric Mantle	12	3.95	5.56	5.21	5.93
Parameter	Training Parameter	Initial Value [-]	Value (Conjugate Gradient) [-]	Value (Pattern Search) [-]	Value (Evolutionary Algorithm) [-]
Scale	13	1.00	1.10	1.10	1.10

5.5 Discussion

Analog to the Results Section, we first discuss the results for the model dimension investigation. Afterwards, we discuss the different model calibration techniques and their importance for geothermal applications.

5.5.1 Influence of the Model Dimension on the Model Predictions

First, we elaborate on how the global sensitivity analysis can be used to determine the influence of the model dimension on the temperature response. Afterwards, we discuss which consequences this has for model calibrations. Finally, we evaluate the computational costs of the concepts presented here.

Sensitivity Analysis

We presented an automated sensitivity-driven model calibration at the basin-scale. Considering that the global sensitivity analysis requires 280,000, 240,000, and

300,000 for the Brandenburg LAB, Brandenburg 6 km, and Brandenburg combined model, respectively, it is clear that a model order reduction is needed to enable such an analysis. Using the reduced basis method showed extremely promising results because we obtain speed-ups of $1.81 \cdot 10^4$ to $1.65 \cdot 10^5$ without introducing approximation errors above the measurement error.

We see entirely different sensitivities for the Brandenburg 6 km model than for the Brandenburg LAB and combined models. For the Brandenburg 6 km model, we fixed the boundary condition at 6 km depth, resulting in an entirely boundary dominated model. This is observable due to the enormous sensitivity of the model to the: i) Basement layer, ii) scaling parameter of the respective boundary condition and iii) correlation between both parameters. Consequently, all information that we obtain from the Brandenburg 6 km model is coming from the boundary condition. Hence, we have a model that is uninformative with respect to the upper layers. However, these are the layers we are interested in. This demonstrates that generating diffusive models with an extremely low vertical to horizontal length ratio is to be avoided at any cost.

The results of the global sensitivity analysis of the Brandenburg LAB and combined model are matching our expectations. We observe a high sensitivity for the upper layers, which is caused by the shallow measurements (500 m to 6,820 m). First-order contributions of the Lower Cretaceous/Jurassic/Buntsandstein layers mostly impact the model. That means that the thermal conductivities of these layers are influencing the model themselves and not through a correlation with other layers. For the Brandenburg LAB model, we combined the thermal conductivity of the Quaternary and the Tertiary layer into one training parameter. For the Brandenburg combined model, we combined the thermal conductivities of the Quaternary and Tertiary-post-Rupelian, and the Tertiary-pre-Rupelian-clay and Upper Cretaceous. Combining the sensitivity analysis of both the Brandenburg LAB model and combined model, we can conclude that the Tertiary-pre-Rupelian-clay is the layer that the model is sensitive to. We can rule out the Quaternary, and the Tertiary-post-Rupelian layer because the Brandenburg combined model is insensitive to it. Furthermore, we can also eliminate the Upper Cretaceous because the Brandenburg LAB model is insensitive to it. Also, the influence of the thermal conductivity of the Tertiary-pre-Rupelian-clay is mainly originating from the parameter itself and not from interactions between various parameters.

Additionally, we obtain for both models a significant influence of the Lithospheric Mantle, and the scaling parameter for the lower boundary condition. Higher-order contributions dominate both of these parameters, and the second-order sensitivity indices show that both parameters are correlated with each other. Both the Keuper and the Zechstein layer have similar influences in both model versions and are less significant in comparison to the overall influences. The parameters themselves and the correlations among each other are causing similar orders of magnitude in the sensitivity.

To conclude, the only meaningful way to construct the model is by inserting the refined model into the original Brandenburg LAB model. This results in the Brandenburg combined model, which again shows the expected sensitivity distribution. One needs to keep in mind that this means an increase in degrees of freedom from 1,546,675 to 2,141,550. Nonetheless, both the finite element and the online execution time for both models are comparable since the complexity in these two models remains similar. This demonstrates that a reduction in the mathematical and not in the physical space is advantageous since it is much less restrictive.

Model Calibration

At first hierarchical model calibrations seem to be a way to transfer the knowledge from large-scale coarse models to smaller-scale fine discretized models. However, the sensitivities clearly show that the smaller model becomes uninformative towards the upper layers. That is especially dangerous because it is not noticeable looking only at the temperature distributions. Also, the parameter distributions of the different calibrations for the Brandenburg 6 km model does not fully reveal the problem. The hierarchical model calibration using the calibrated boundary condition from the Brandenburg LAB model yields an unrealistic low thermal conductivity of the Basement. However, the Kriging model calibration, the model calibration using the uncalibrated boundary condition (assuming a good and bad initial guess) performs well without having unrealistically thermal conductivities for the Basement. Only the sensitivity analysis revealed the uninformative nature of the model, further emphasizing the general necessity of global sensitivity studies.

For the remaining discussion of the calibration results, we talk about the results from the Brandenburg LAB and combined model because of the uninformative nature of the Brandenburg 6 km model. We observe similar trends for both the original (Brandenburg LAB) and the geologically refined (Brandenburg combined) model. Nevertheless, the parameter distribution of the Brandenburg combined model after the calibration is closer to the initial parameter distribution than the one from the Brandenburg LAB model. This demonstrates the need for model calibration. It is incredibly challenging and time-consuming to construct a model that accounts for all structural effects. Taking the lack of data into account, it becomes a somehow impossible task. Therefore, we follow a different approach in this thesis. We compensate for the model errors by replacing the physical by effective thermal conductivities. In that way, we obtain a representative model. Also, keep in mind that the major shortcoming of the Brandenburg 6 km model could only be revealed using a global sensitivity analysis. However, this requires so many forward simulations that it is not realizable, even with state-of-the-art finite element solvers, on a basin-scale without using surrogate models.

Considering the geological setting, the significant increase in thermal conductivity for the Lower Cretaceous/Jurassic/Buntsandstein layers (Brandenburg LAB model and Brandenburg combined model) is most likely caused by unresolved salt structures. Further investigations are required to analyze whether all layers (Lower Cretaceous/Jurassic/Buntsandstein) or only one layer contain unaccounted structures. The increase in thermal conductivity for the Zechstein layer is also most likely caused by unaccounted salt structures. Note that the increase in thermal conductivity for the Zechstein layer is significantly lower for the Brandenburg combined model than for the Brandenburg LAB model leading to the conclusion that the geological refinement captured successfully missing salt structures. The Keuper layer has after the calibration nearly the same value as the Muschelkalk layer leading to the assumption that we underestimated the sediment thickness of the Muschelkalk.

Combining the increase in thermal conductivity of the Lithospheric Mantle and the variation of the lower boundary conditions leads to the conclusion that a wrong geometrical parameterization of the LAB causes this increase. The correlation between both parameters confirms this. The fact that fixing the boundary condition to 1300 °C leads to an even higher increase in thermal conductivity further emphasizes this. For further studies, one could either allow a larger variation at the lower boundary condition or replace the constant value at the boundary with a function to better compensate for the geometrical effects.

From the sensitivity analysis, we know that from the upper layers, the influence is arising from the Tertiary-pre-Rupelian-clay. Again, the increase is less dominant for the Brandenburg combined model than for the Brandenburg LAB model, thus leading us to the conclusion that structural effects mainly cause this mismatch. The refined model version much better resolves this. Regarding the Brandenburg 6 km model, we already stated that it is completely boundary dominated. Therefore, we exclude the results from the discussion.

For the calibration, in general, we observe that, as expected, the local calibration method is sufficient due to the linearity of the problem.

Computational Cost

The offline stage of the Brandenburg combined model was computed on the RWTH compute cluster. We used two Intel Xeon Platinum 8160 CPUs (24 cores, 2.1 GHz, 192 GB of RAM), and the computations took 5.4 h. The other offline stages were computed on an Intel Westmere X675 machine (3.07 GHz 6 cores per chip, 12 cores per node and 24 GB memory per node). They required 2.9 h (Brandenburg LAB model) using 50 cores, and 57 min (Brandenburg 6 km model) using 48 cores. These time-consuming offline stages could be calculated faster using more cores. Nonetheless, the RB method is more efficient than the FE method because the offline stages contain up to 273 FE solves. This number is substantially lower than the number of function evaluations in this thesis. Note that we would have required 0.6 to 4.7 years for the Sobol sensitivity analysis using the FE problem one a single core. In contrast, we only required 5.4 min to 47.8 min using the RB method. Also, note that the forward solves are parallelizable, whereas most inversion routines are not. Additionally, the RB method allows on the fly adjustments of the parameters in the field, something that is not possible for the full model.

5.5.2 Automated Model Calibration Comparison

We have seen that the conjugate gradient method, being a local gradient-based method, performs best for the Brandenburg model. It has the smallest misfit for the observed data, and it requires the least number of function evaluations. The evolutionary algorithm also has the same low mean misfit, but it needs more than 3,000 times more function evaluations. This is not surprising since the PDE used is linear. Additionally, we use as our initial guess the parameter distributions obtained by a "trial-and-error" model calibration (Noack et al., 2012, 2013). Hence, we expect that we are already in the vicinity of the "true" parameter distribution. Consequently, this is a textbook example of a local gradient-based method.

Also, the pattern search method can significantly reduce the temperature misfit. However, it requires more than 184 times more function evaluations than the conjugate gradient method. This proves that the pattern search method is not ideal for the Brandenburg model since we obtain slightly worse results for a significant increase in computation time.

Comparing the parameter distributions of the conjugate gradient, the pattern search, the evolutionary, and the trust region reflective method, we observe the same trends for all methods. For this reason, we refer to the geological interpretation of the results to the previous section.

In this chapter, we focused on deterministic model calibrations. From our geological background, we are aware that more than one solution describes our observations.

Therefore, it is interesting to investigate the uncertainties related to both the thermal parameters and the temperatures.

Chapter 6

Uncertainty Quantification for Basin-Scale Conductive Heat Transfer Models

Within this chapter, we aim to demonstrate the importance of uncertainty quantification methods, such as Markov Chain Monte Carlo (MCMC), for basin-scale geothermal applications. Using standard techniques, such as the finite element method, this would be prohibitively expensive as these methods often require a large number of simulations. Therefore, we propose to use the RB method to construct suitable surrogate models. Two case studies are used to illustrate the benefits of the concepts proposed here: i) the Brandenburg model (Noack et al., 2012, 2013), and ii) the Upper Rhine Graben (Freymark et al., 2017). The case study of the Brandenburg model is in preparation for publication in Degen et al. (2020d).

6.1 Introduction

Geophysical and geoscientific applications have many sources of uncertainties, arising from, for instance, unaccounted and unknown physical processes, inaccurate geometrical information, and uncertain parameter distributions. Identifying and quantifying these uncertainties is a non-trivial process. Methods such as MCMC that easily require a million forward simulations, make this task not only non-trivial but computationally prohibitive for basin-scale geological heat flow models using state-of-the-art finite element solvers.

As a solution, the finite element method is replaced by a surrogate model such as, for instance, Kriging (Miao et al., 2019; Mo et al., 2019), or polynomial chaos expansions (Navarro et al., 2018). The issue with these surrogate models is that they are defined in the observation space only. Values outside this space need to be determined via inter- and extrapolation. For geothermal studies, however, we are interested in the entire temperature distribution at a particular target depth. Therefore, we use in this thesis a physics-based machine learning approach, the reduced basis method (RB) (Prud'homme et al., 2002; Veroy et al., 2003; Hesthaven et al., 2016), as the surrogate model. The RB method has, in contrast to other surrogate models, the advantage that it retrieves the temperature distribution in the entire model and is not restricted to the observation space. That enables an evaluation of the uncertainties in the entire model. Furthermore, the RB method provides, for the here presented geothermal application, an error bound allowing an objective assessment of the approximation quality.

The utility of model order reduction for Bayesian inversion has been investigated in previous studies. A data-driven POD approach is discussed in Cui et al. (2015). Himpe and Ohlberger (2015), and Lieberman et al. (2010) use a parameter-state

model reduction with a greedy algorithm, for addressing the computational challenges of uncertainty quantification. A POD approach for non-linear PDEs is presented in Galbally et al. (2010). Manzoni et al. (2016) addressed the computational issue with an RB model in combination with an error model. Furthermore, Chen and Schwab (2015, 2016) present a sparse-grid reduced basis version for Bayesian inversion for both linear and non-linear PDEs.

However, these papers focus on the methodology, and the presented case studies do not capture the typical complexity of geothermal basin-scale applications. We present a global sensitivity driven stochastical model calibration for complex basin-scale applications. We use these results to generate predictive uncertainty maps for the geothermal target depth.

6.2 Uncertainty Quantification

For the uncertainty quantification, we use the Monte Carlo Markov Chain (MCMC) implementation of PyMC with a Metropolis sampling Patil et al. (2010). Bayes Theorem is the basis of the MCMC method (Iglesias and Stuart, 2014):

$$P(u|y) \propto P(y|u)\ P(u). \tag{6.1}$$

The prior $P(u)$ describes our knowledge about the unknown variable without taking the data into account. The posterior $P(u|y)$, is the knowledge we have about our unknown variable u given the data y. Furthermore, $P(y|u)$ is the likelihood, which describes the likelihood of the parameters given the observation data. Often, we do not have a very accurate or detailed knowledge of our unknowns, which means that determining the priors is often challenging. If we would not use an uncertainty quantification method, such as MCMC, we would simply sample from our prior and determine the uncertainties from that. This would result in relatively large uncertainties. MCMC, on the other hand, incorporates the data to reduce the uncertainties through the given data. MCMC is a method to draw samples from a probability distribution. This is based on the generation of a Markov Chain. A Markov Chain develops only based on the knowledge of the present and previous event (Iglesias and Stuart, 2014).

6.3 Geological Models

In this chapter, we use both the case studies of the Brandenburg model and the Upper Rhine Graben. We illustrate both: i) how the RB method is beneficial in the context of uncertainty quantification and ii) why the consideration of uncertainties is crucial for geothermal applications. In case of the Brandenburg model, we use the Brandenburg combined model that we presented in Chapter 5. In the following, we denote it as Brandenburg model since we only use the Brandenburg combined model in this chapter. Note that we are using the Upper Rhine Graben with the same geological and physical model and with the same decomposition, as presented in Chapter 4. For both models, we consider a geothermal conductive heat transfer problem with a radiogenic heat production as the source term. This physical model is identical to the one of the previous chapters, and for further details regarding the physical model, we refer to Chapter 4 and 5.

6.4 Results

In this section, we first present the results of the uncertainty quantification, in terms of the parameter uncertainties and the 2D and 3D uncertainty quantification maps of the Brandenburg model, followed by the presentation of the Upper Rhine Graben.

6.4.1 Brandenburg Model

For the uncertainty quantification of the Brandenburg model, we perform a Markov Chain Monte Carlo (MCMC) analysis with a Metropolis sampling using the Python library PyMC (Patil et al., 2010). We reduce the 14 parameters that we are able to vary to six parameters. We do not consider the remaining eight parameters within the uncertainty quantification. A previously performed Sobol sensitivity analysis with the Saltelli sampler, performed with the Python library SALib (Herman and Usher, 2017), showed that the model is insensitive to those parameters. To reduce the statistical errors within the sensitivity analysis, we have 10,000 realizations per parameter resulting in 300,000 forward solves. For all thermal conductivities in the sensitivity analysis and the MCMC algorithm, we allow a variation of $\pm$ 50 %. The number of function evaluations for the MCMC run is set to 1,000,000 with a thinning of 1,000 and 10,000 burn-in-simulations. For the priors, we use normally distributed parameters. The mean of each parameter corresponds to the fitted thermal conductivity values of Noack et al. (2012, 2013). Both the standard deviation and proposal standard deviation are set to:

- one for the Tertiary-pre-Rupelian-clay/Upper Cretaceous and Lower Cretaceous/Jurassic layer
- two for the Keuper layer
- four for the Zechstein layer and the Lithospheric Mantle
- 0.002 for the scaling parameter of the lower boundary condition

and afterwards divided by their respective mean values. The standard deviations have been determined such that the values do not exceed a range of $\pm$ 50 % of their mean values. Analog to the deterministic model calibration, we use the temperature data presented in Noack et al. (2012, 2013). The bottom-hole temperatures of this database have been measured during the drilling process and were later on corrected after Förster (2001). This correction might not fully capture the perturbation of the temperature field. Therefore, we apply a standard deviation of 0.02 for the observation data.

6.4.2 Thermal Conductivities

First, we discuss the posterior distribution of the thermal conductivities obtained by the MCMC analysis (Tab. 6.1). Fig. 6.1 shows the Quantile-Quantile plots of the individual thermal conductivities. All parameters have a similar pattern. In the center part of the Quantile-Quantile plots, they follow a normal distribution, and towards the borders, they deviate from the normal distribution. The reason for this deviation is that we use truncated normal distributions as our priors instead of normal distributions. The usage of truncated distribution is unavoidable since we have to ensure that we stay within our pre-defined training range. Otherwise, we cannot use the RB model as our surrogate model. Nonetheless, normal distributions quite well describe our parameter.

Hence, we discuss in the following only the posterior mean and standard deviations of the thermal conductivities.

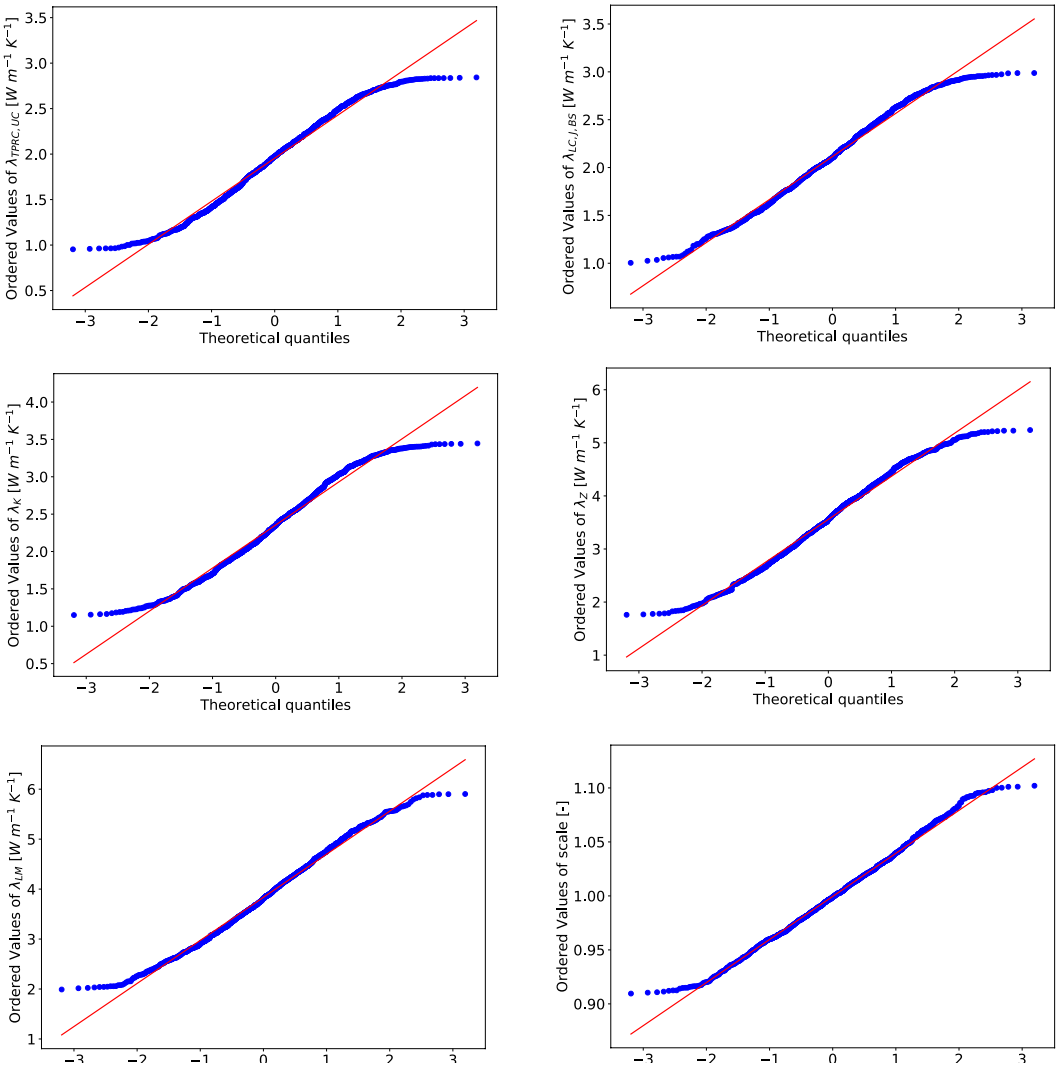


FIGURE 6.1: Quantile-Quantile plots for all thermal conductivities considered in the uncertainty quantification of the Brandenburg model.

In Fig. 6.2 we obtain for the Tertiary Rupelian-clay/Upper Cretaceous layer a slight increase in the posterior mean thermal conductivity of 0.05 W m^{-1} K^{-1} in contrast to the prior thermal conductivity. The parameter follows a normal distribution with a standard deviation of 0.47 W m^{-1} K^{-1}. We observe a posterior thermal conductivity of:

- 2.11 W m^{-1} K^{-1} $\pm$ 0.45 W m^{-1} K^{-1} for the Lower Cretaceous/Jurassic/Buntsandstein layer,
- 3.56 W m^{-1} K^{-1} $\pm$ 0.81 W m^{-1} K^{-1} for the Zechstein layer,
- and 2.35 W m^{-1} K^{-1} $\pm$ 0.58 W m^{-1} K^{-1} for the Keuper layer.

Hence, all three cases show an increase in the posterior thermal conductivity in comparison to the prior thermal conductivity, and they are also normally distributed. The Lithospheric Mantle shows a decrease in the posterior mean thermal conductivity of 0.11 W m^{-1} K^{-1} in comparison to the prior thermal conductivity and has a posterior standard deviation of 0.86 W m^{-1} K^{-1}. The scaling parameter has a posterior mean

value of 1.00, which is identical to the prior value, and a posterior standard deviation of 0.04. All parameters follow a normal distribution. The execution of the MCMC algorithm took in total about 4.5 hours.

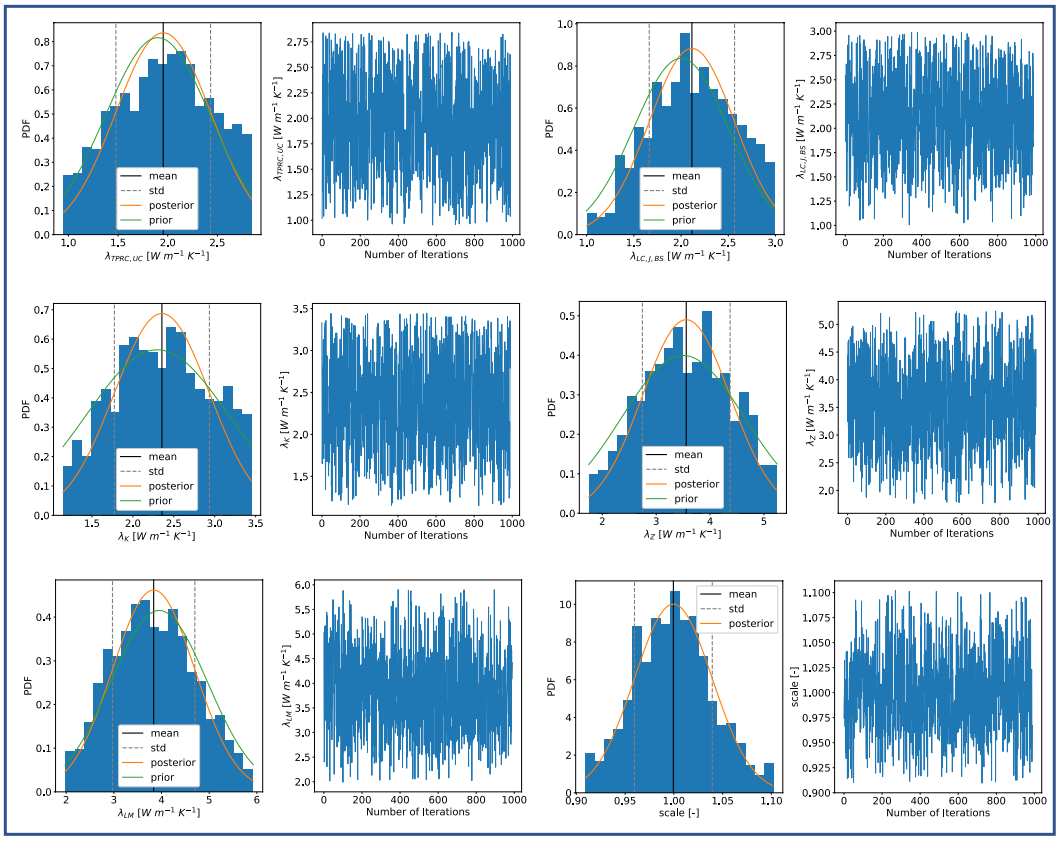


FIGURE 6.2: Prior and posterior parameter distributions of all six parameters and their respective traces.

TABLE 6.1: Thermal Properties of the Brandenburg model before and after the uncertainty quantification. The prior thermal properties are from Noack et al. (2012, 2013). We denote all parameters that are not involved in the uncertainty quantification, due to too low sensitivities, with n/a. Additionally, the affine decomposition for the model reduction is provided.

Layer	Training Parameter	Radiogenic Heat Production $[\mu W m^{-3}]$	Prior Thermal Conductivity $[W m^{-1} K^{-1}]$	Posterior Mean Thermal Conductivity $[W m^{-1} K^{-1}]$	Posterior Std Thermal Conductivity $[W m^{-1} K^{-1}]$
Quaternary	0	0.7	1.50	n/a	n/a
Tertiary-post-Rupelian	0	0.7	1.50	n/a	n/a

Tertiary Rupelian-clay	1	0.45	1.00	n/a	n/a
Tertiary-pre-Rupelian-clay	2	0.3	1.90	1.95	0.47
Upper Cretaceous	2	0.3	1.90	1.95	0.47
Lower Cretaceous	3	1.4	2.00	2.11	0.45
Jurassic	3	1.4	2.00	2.11	0.45
Keuper	4	1.4	2.30	2.35	0.58
Muschelkalk	5	0.3	1.85	n/a	n/a
Buntsandstein	3	1.0	2.0	2.11	0.45
Zechstein	6	0.09	3.50	3.56	0.81
Sedimentary Rotliegend	7	1.0	2.16	n/a	n/a
Permo-Carboniferous Volcanics	8	2.0	2.50	n/a	n/a
Pre-permian	9	1.5	2.65	n/a	n/a
Upper crust	10	2.5	3.10	n/a	n/a
Lower crust	11	0.8	2.70	n/a	n/a
Lithospheric Mantle	12	0.03	3.95	3.84	0.86
Parameter	Training Parameter	Prior Value [-]		Posterior Mean Value [-]	Posterior Std Value [-]
Scale	13	1.00		1.00	0.04

Uncertainty Quantification Maps

Now, we use the parameter distributions of the MCMC analysis to generate 2D and 3D uncertainty quantification maps. We make here also use of the RB method, which allows us to compute model realizations for samples from the posterior distribution to obtain temperature state values everywhere in space.

For the generation of uncertainty quantification maps, we first have to choose a suitable representation. Therefore, we pick nine points at a depth of 5 km (Fig. 6.3). For these nine points, we plot the distribution of the temperatures for all parameter values obtained by the previous MCMC analysis (Fig. 6.4). The temperatures at all nine points follow a normal distribution. Therefore, we plot in the following the posterior mean temperatures and their standard deviations to achieve a suitable representation of the temperature uncertainties.

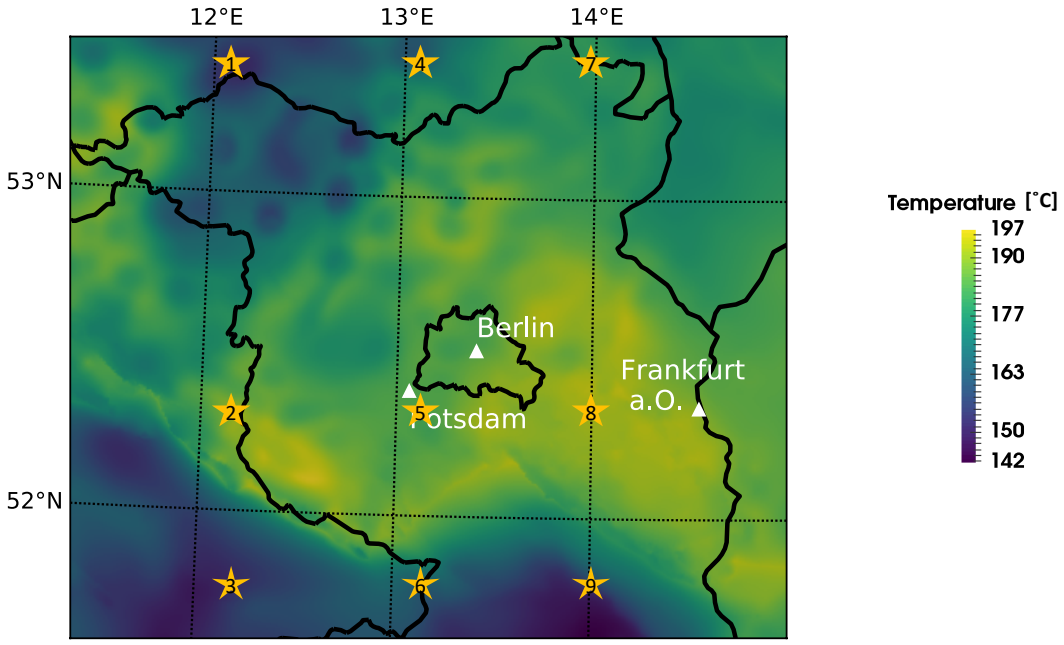


FIGURE 6.3: Prior temperature distribution of the Brandenburg model in a depth of 5 km. The yellow stars mark the positions for which we generate the Quantile-Quantile plots.

First, we present the posterior distributions in the entire Brandenburg model. The posterior standard deviations have their highest value within the sedimentary basin at a depth of about 30 to 35 km (see the black box of Fig. 6.5). Consequently, there are the highest uncertainties. They have a maximum value of 53 °C. Overall, we observe uncertainties ranging from 0 °C to 53 °C. We observe that the uncertainty decreases towards the boundaries and increases towards the center part of the model. The posterior mean temperature distribution has a steep gradient in the upper part of the model and a significantly less steep gradient in the lower part of the model. The temperatures range from 8 °C to 1300 °C (see the blue box of Fig. 6.6).

Now, we focus on the posterior distributions at a typical target depth for geothermal systems of 5 km. The posterior mean temperature ranges from 141 °C to 197 °C, and the posterior standard deviation from 8 °C to 18 °C. The highest uncertainty, in a depth of 5 km is north of the interface of the Tertiary-post-Rupelian and the Rupelian clay and south to the Zechstein - Sedimentary Rotliegend interface. The area is marked with an A in the blue box of Fig. 6.5. It has its highest peak southeast to the region, where salt structures majorly influence the posterior mean temperatures. Generally,

from the interface (marked with a B), the uncertainties increase towards the north and decrease towards the south of the model.

The highest posterior mean temperatures of over 190 °C are north to the interface of the Tertiary-post-Rupelian and the Rupelian clay (marked with a C). In contrast, the lowest posterior mean temperature values around 140 °C are south to this interface (see B in the blue box of Fig. 6.5). In general, the posterior mean temperature north to the interface decrease to the northern border of the model. Furthermore, in the north-west part of Brandenburg is a region of lower posterior mean temperatures located (area A in the blue box of Fig. 6.5). The reasons for this decreased posterior mean temperature will be discussed in Section 6.5.1.

Computational Cost

For the generation of the uncertainty quantification map, we need to calculate the temperatures for all parameters obtained through the MCMC analysis. Furthermore, we need to project the reduced space back to the full space and calculate the posterior mean temperatures and their posterior standard deviations. All these operations, in total, require about four minutes.

The reduction requires 273 basis functions for reaching the pre-defined error tolerance for the nondimensional model of $5 \cdot 10^{-4}$ (see the red box of Fig. 6.6). Note that the most accurate measurements have an accuracy of 10^{-1}. Consequently, the chosen error tolerance ensures that we do not introduce approximation errors above the measurement error. The reduced basis method leads to a speed-up of $1.0 \cdot 10^{5}$. Furthermore, we observe that the largest errors are occurring at the layer interfaces (see the red box of Fig. 6.6).

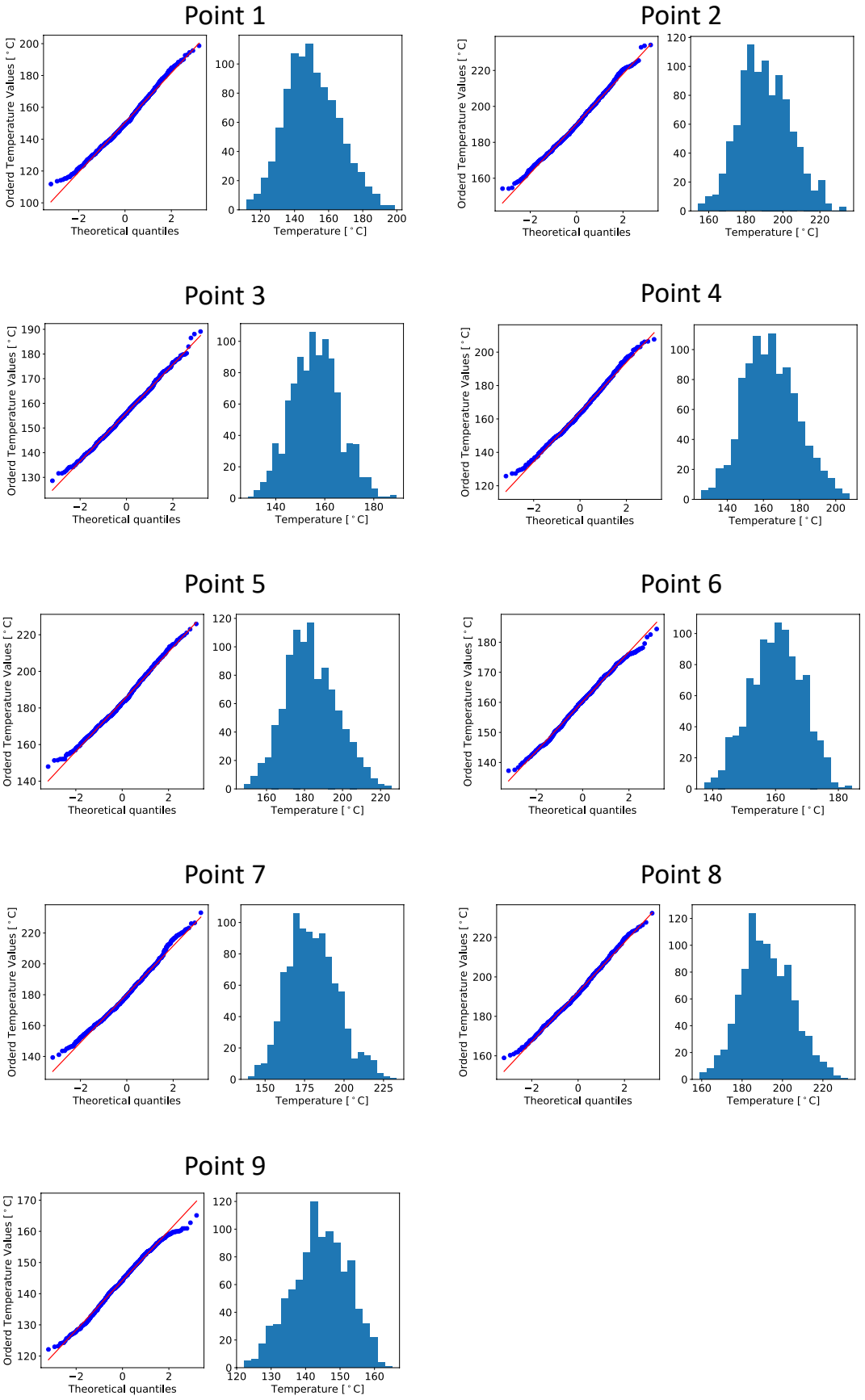


FIGURE 6.4: Quantile-Quantile plots and histograms of the temperatures for all parameters from the MCMC analysis for the Brandenburg model at nine points.

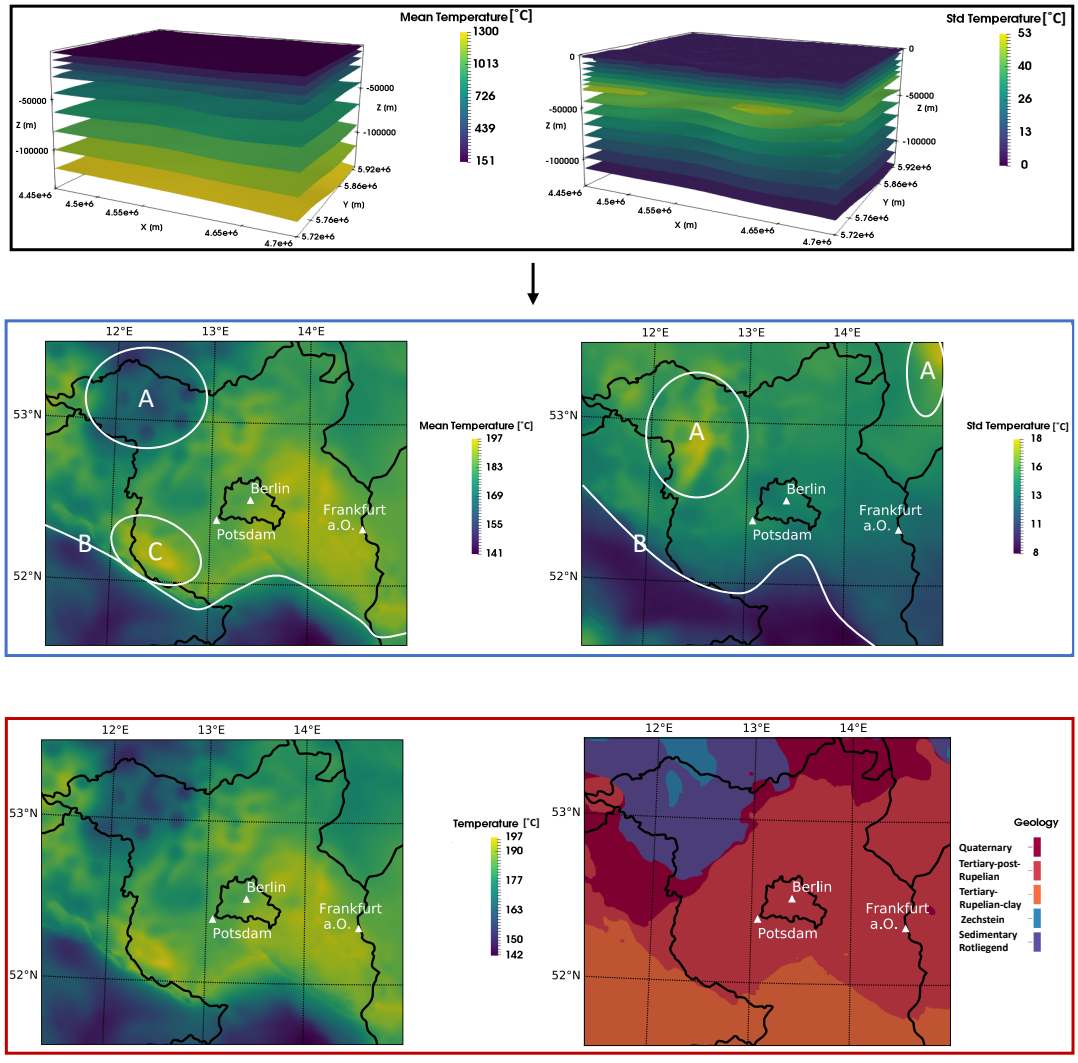


FIGURE 6.5: Black Box: Distribution of the posterior mean temperature and the posterior standard deviation of the entire Brandenburg model. Blue Box: Map of the posterior mean temperature and posterior standard deviation at the target depth of 5 km. Red Box: Prior temperature distribution and distribution of the geological layers at the target depth of 5 km.

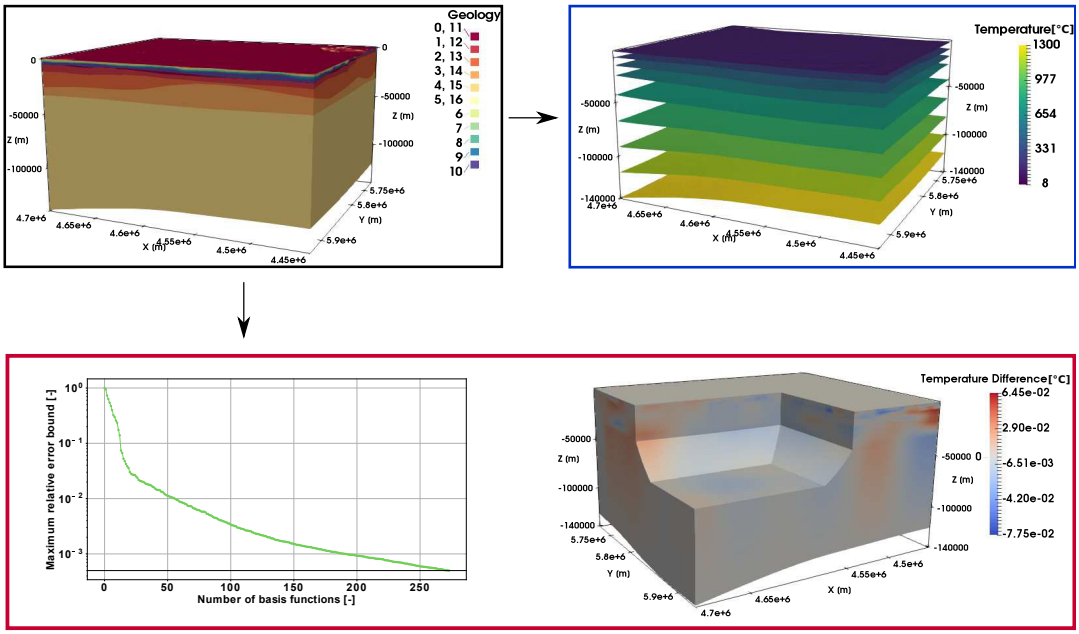


FIGURE 6.6: Black Box: Image of the Brandenburg model. 0: Quaternary, 1: Tertiary-post-Rupelian, 2: Tertiary Rupelian-clay, 3: Tertiary-pre-Rupelian-clay, 4: Upper Cretaceous, 5: Lower Cretaceous, 6: Jurassic, 7: Keuper, 8: Muschelkalk, 9: Buntsandstein, 10: Zechstein, 11: Sedimentary Rotliegend, 12: Permo-Carboniferous Volcanics, 13: Pre-permian, 14: Upper Crust, 15: Lower Crust, 16: Lithospheric Mantle. Blue Box: Prior temperature distribution for the Brandenburg model. Red Box: On the left side, the convergence of the maximum relative error bound for the entire parameter range is displayed. On the right side, the error between the full and reduced model for the prior parameters is shown.

6.4.3 Upper Rhine Graben

As for the model calibration, we use a global sensitivity study to reduce the number of parameters that need to be investigated. Hence, instead of 12 uncertain parameters, we have to analyze the uncertainty of four parameters only. All the other thermal properties are set to the values presented in Freymark et al. (2017). We allow the same variations as for the deterministic model calibrations, and we set the number of iterations for the MCMC run to 1,000,000 with a thinning of 1,000, and 10,000 burn-in-simulations. We use normally distributed priors, where the mean of each parameter corresponds to the fitted thermal conductivities provided by Freymark et al. (2017). The standard deviations are chosen such that the thermal conductivities stay in the ranges provided by Freymark et al. (2017). For all parameters, where we had no ranges, we assumed a parameter range of $\pm$ 50 %. The proposal standard deviation is 0.25 for the Cenozoic Volcancis, and the Doger/Lias/Keuper, Saxothuringian, Moldanubian layers. For the Cenozoic Rift Sediments, we use a proposal standard deviation of 0.1, and for the Cenozoic Folded Molasse, the Cenozoic Foreland Molasse, the Buntsandstein, the Jura Mountains, and the Odenwald we use 0.05. Furthermore, the proposal standard deviation for the Lithospheric Mantle is 1.0. For the standard deviation of the observation data, we use a value of 0.02, analog to the Brandenburg model. Hence, we put high trust in our measurements.

Thermal Conductivity

Analog to the Brandenburg model, we discuss first the posterior distribution of the thermal conductivities obtained by the MCMC analysis. Again, Fig. 6.7 shows the Quantile-Quantile plots of the individual thermal conductivities. As for the Brandenburg model, all parameters have a similar pattern. In the center part of the Quantile-Quantile plots, they follow a normal distribution, and towards the borders, they deviate from the normal distribution. The reason for this deviation is, as for the Brandenburg model, the usage of truncated normal distributions as our priors. For further discussion on this topic, we refer to 6.4.2. Still, normal distributions describe our parameters well. Hence, we discuss in the following only the posterior mean and standard deviations of the thermal conductivities.

First, we present the changes in the posterior mean thermal conductivities for all thermal conductivities. Table 6.2 and Fig. 6.8 presents the posterior mean and standard deviation of each of the four investigated thermal conductivities. Except for the posterior mean thermal conductivity of the Lithospheric Mantle, all posterior mean thermal conductivities have a similar value as their prior value. The posterior mean thermal conductivity of the Lithospheric Mantle is 24 % higher than the prior thermal conductivity for this layer. We also observe a slight increase in the posterior mean thermal conductivities of the Cenozoic Rift Sediments, in comparison to the prior thermal conductivities.

Now, we investigate the posterior standard deviations of the thermal conductivities. The uncertainty is the highest for the uppermost layer. The Cenozoic Volcanics have a posterior standard deviation of 22 %. The remaining thermal conductivities have similar posterior standard deviations of:

- 15 % for the Cenozoic Rift Sediments,
- 17 % for the Dogger/Lias/Keuper/Saxothuringian/Modanubian, and the Cenozoic Folded Molasse/Cenozoic Foreland Molasse/Buntsandstein/Jura Mountains/Odenwald, and

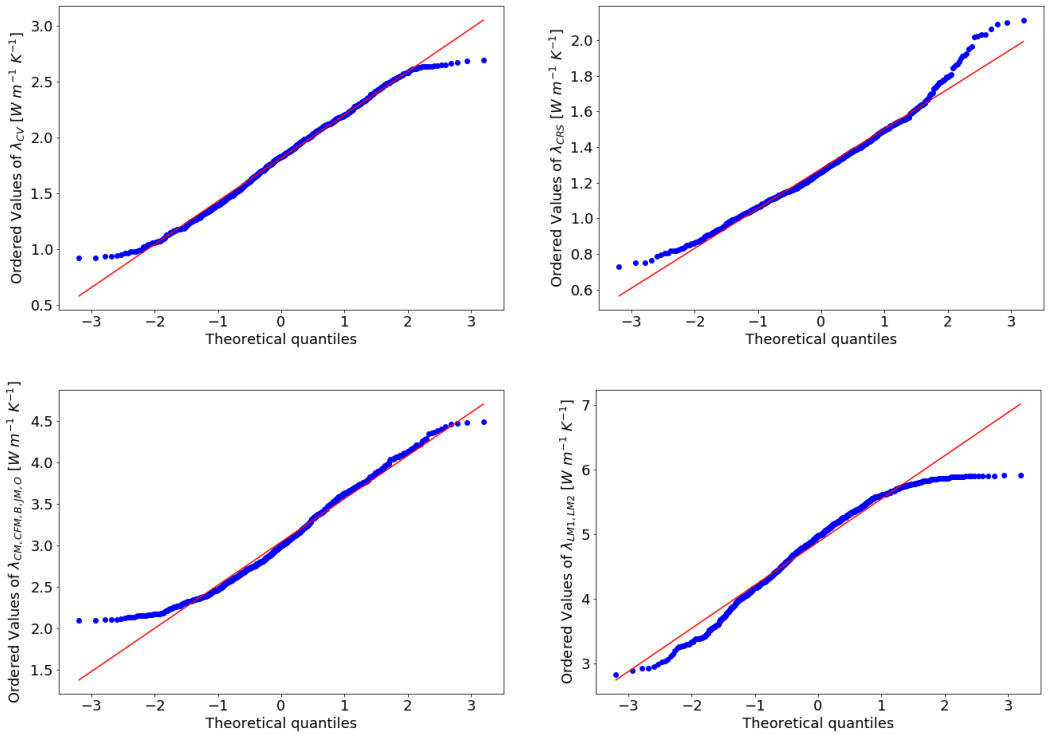


FIGURE 6.7: Quantile-Quantile plots for all thermal conductivities considered in the uncertainty quantification of the Upper Rhine Graben.

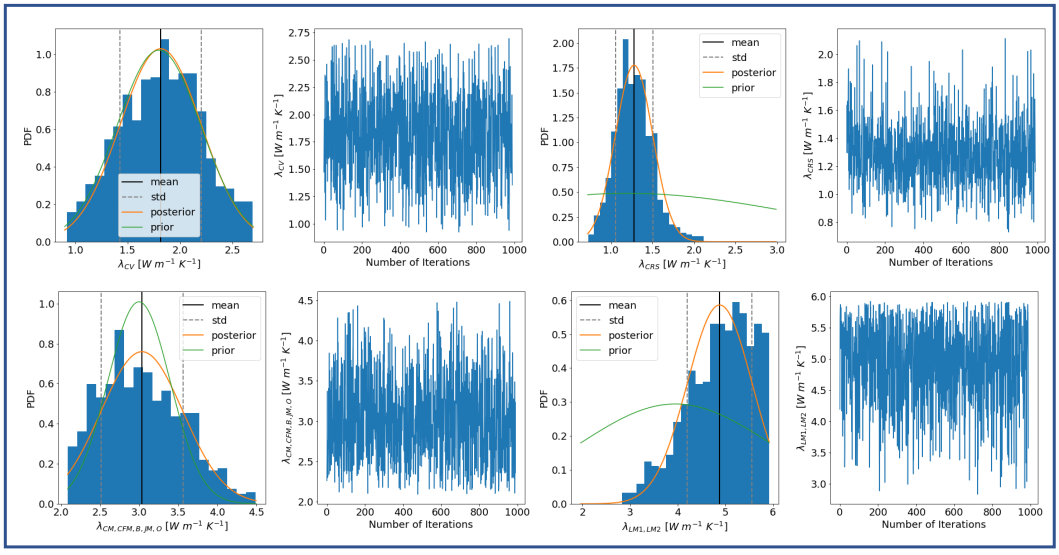


FIGURE 6.8: Prior and posterior parameter distributions of all four parameters and their respective traces.

- 14 % for the Lithospheric Mantle.

In case of the Cenozoic Volcanics and Dogger/Lias/Keuper/Saxothuringian/-Modanubian, and the Cenozoic Folded Molasse/Cenozoic Foreland Molasse/ Buntsandstein/Jura Mountains/Odenwald layer, we obtain similar prior and posterior standard deviations. In contrast, for both the Cenozoic Rift Sediments and the Lithospheric Mantle, we obtain a significantly lower posterior standard deviation than

our prior standard deviation. Hence, for those two parameters, we significantly reduce the uncertainty through the incorporation of the observation data.

TABLE 6.2: The thermal properties before and after the uncertainty quantification for the Upper Rhine Graben. The prior thermal properties are taken from Freymark et al. (2017). We denote all parameters that are not involved in the uncertainty quantification due to too low sensitivities with n/a. Note that UC stands for Upper Crust.

Layer	Training Parameter	Radiogenic Heat Production $[\mu W m^{-3}]$	Prior Thermal Conductivity $[W m^{-1} K^{-1}]$	Posterior Mean Thermal Conductivity $[W m^{-1} K^{-1}]$	Posterior Std Thermal Conductivity $[W m^{-1} K^{-1}]$
Cenozoic Volcanics	0	0.2	1.8	1.8	0.4
Cenozoic Rift Sediments	1	1.0	1.2	1.3	0.2
Tertiary salt	2	0.0	6.0	n/a	n/a
Alps	3	0.3	2.2	n/a	n/a
Cenozoic Folded Molasse	4	1.0	3.0	3.0	0.5
Cenozoic Foreland Molasse	4	1.0	3.0	3.0	0.5
Cretaceous	3	0.5	2.2	n/a	n/a
Malm	5	1.4	2.7	n/a	n/a
Dogger/Lias-/Keuper	6	1.6	2.5	n/a	n/a
Muschelkalk	7	1.2	2.0	n/a	n/a
Buntsandstein	4	1.0	3.0	3.0	0.35
Zechstein	8	0.8	2.3	n/a	n/a
Permo-Carboniferous	3	1.0	2.2	n/a	n/a

Jura Mountains	4	0.5	3.0	3.0	0.5
UC: Saxothuringian	6	2.5	2.5	n/a	n/a
UC: Odenwald	4	1.8	3.0	3.0	0.5
UC: Mid-German Crystalline High	9	1.8	2.4	n/a	n/a
UC: Nothern Phyllite Zone	5	3.0	2.7	n/a	n/a
UC: Rhenohercynian	5	1.0	2.7	n/a	n/a
UC: Moldanubian	6	2.6	2.5	n/a	n/a
UC: Alps	8	2.4	2.3	n/a	n/a
Lower Crust	10	0.5	2.1	n/a	n/a
Lithospheric Mantle 1	11	0.03	3.95	4.9	0.7
Lithospheric Mantle 2	11	0.03	3.95	4.9	0.7

Uncertainty Quantification Maps

We now use the parameter distributions obtained by the uncertainty quantification to generate predictive uncertainty maps. With these maps, we analyze the spatial variability of the uncertainty in temperature arising from the uncertainties of the thermal properties. Analog to the Brandenburg model, we need to choose a suitable representation of the temperature distribution and their uncertainties. Again, we plot the temperature for all parameter realizations provided by the MCMC analysis at nine points. These points are taken at high, low, and medium prior mean temperature locations. Fig. 6.9 shows all of them. The corresponding temperature distributions and their Quantile-Quantile plots are displayed in Fig. 6.10. All nine points show normally distributed temperatures. Hence, we discuss the uncertainty quantification maps in terms of their posterior mean and standard deviations.

First, we have a look at the posterior mean temperature distribution of the entire

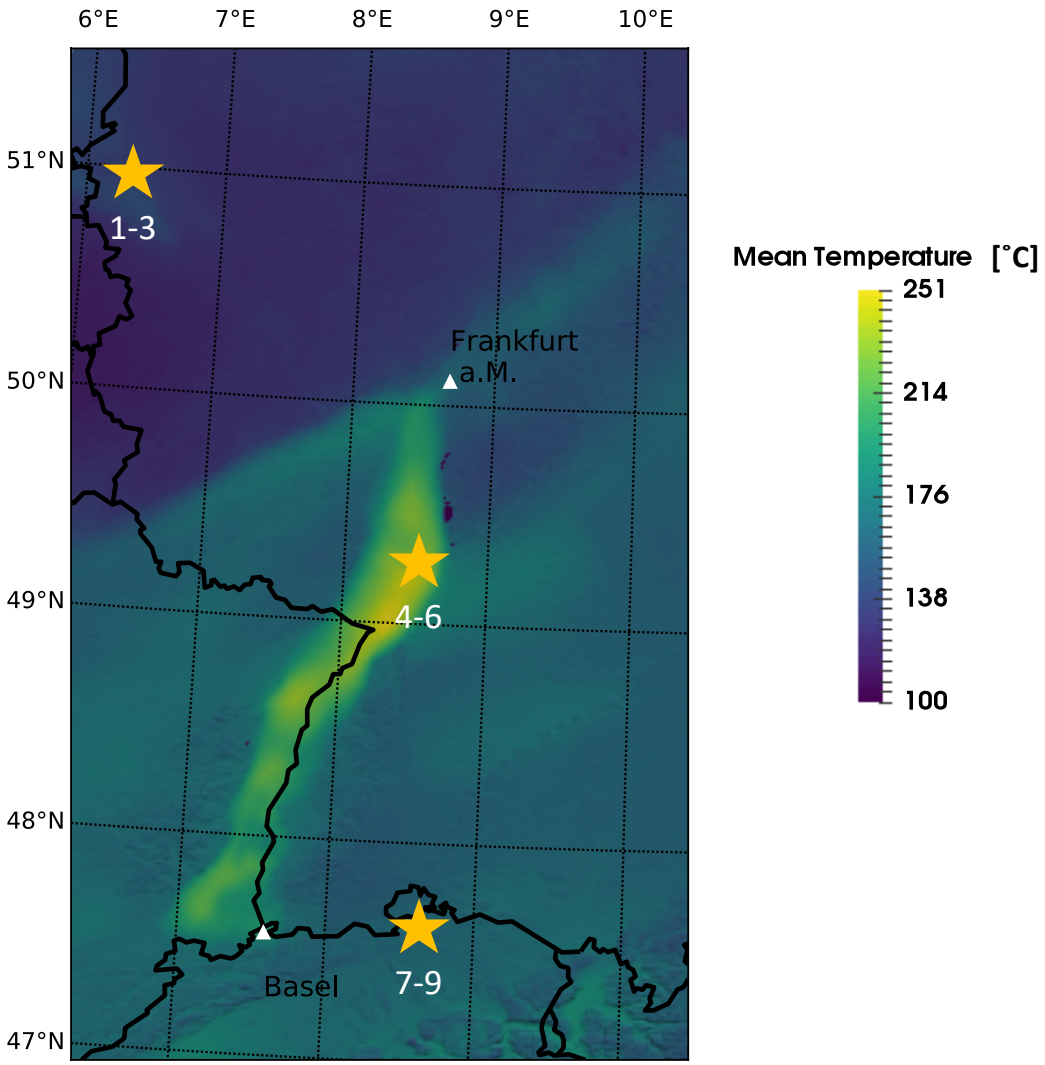


FIGURE 6.9: Posterior mean temperature distribution of the Upper Rhine Graben model in a depth of 5 km. The yellow stars mark the positions for which we generate the Quantile-Quantile plots at each position we take a point at 5 km, 6 km, and 7 km depth.

Upper Rhine Graben model (see Fig. 6.11). As expected, the posterior mean temperature ranges from -1 °C to 1300 °C and has a huge similarity to the prior temperature distribution of the Upper Rhine Graben model. Regarding the posterior standard deviation of the temperature of the Upper Rhine Graben model (see Fig. 6.12), we observe an uncertainty of 0 °C at the lower and upper boundary condition and an increasing uncertainty towards the center of the model. The uncertainty is reduced at the front left corner of the model. It has higher values in the Cenozoic Rift Sediment layer (marked with an A).

For geothermal exploration, we are interested in the spatial distribution of the temperatures at our target depth. Therefore, we further analyze the distribution of the posterior mean temperatures and their posterior standard deviations at a depth of five to seven km below the surface. For a depth of five km below the surface, we observe a posterior mean temperature with a maximum value of 251 °C (Fig. 6.13). This maximum temperature is reached in the central part of the Upper Rhine Graben (marked with an A). The Upper Rhine Graben, areas A and B, have significantly higher

temperatures than in the surrounding geological units. In the southern part of the Upper Rhine Graben, area C, we observe lower posterior mean temperature values. The Lower Rhine Graben (area D) is an area with increased posterior mean temperatures in comparison to its surroundings. Above the variscan domain boundary E, we obtain significantly lower posterior mean temperature values. Furthermore, we can identify further variscan domain boundaries in the posterior mean temperature distribution (interfaces F-H). At the eastern border of the Upper Rhine Graben in the Odenwald (area I), we see decreased posterior mean temperature values.

Note that the extremely low temperatures (yellow box) at the north-east side of the Upper Rhine Graben are an interpolation artifact and not a result of the physical model. To account for the topographic differences of the Upper Rhine Graben model, we slice with respect to the surface and not to the zero meter isoline in ParaView (Ayachit, 2015). However, for this calculation, ParaView produces a couple of points that are marked as invalid. These points contain no information regarding the geology and temperatures. Hence, ParaView assigns them with a value of zero. The assigned value has no relationship to the physical values. Nonetheless, all remaining values are marked as valid and can be used for further analyses.

We now present the distribution of the posterior standard deviations, with a maximum value of 22 °C, in a depth of 5 km below the surface. In the center part of the Upper Rhine Graben (area A of Fig. 6.14), we observe higher posterior standard deviations. Also, the remaining part of the Upper Rhine Graben (area B) has significantly higher uncertainties than its surroundings. In the southern part of the Upper Rhine Graben, we observe with area C a location of significantly lower uncertainty than the rest of the Upper Rhine Graben. The Odenwald (area D), east of the Upper Rhine Graben, marks an area with higher uncertainties. In the southwestern part of the model domain, we have an interface between the highest and lowest posterior standard deviations at this target depth. The area of high uncertainties corresponds to the Molasse Basin (area E), whereas the area of low uncertainties is associated to the Alps (area F). Furthermore, the Jura Mountains (area G) have similar uncertainties as the Molasse Basin. Additionally, we have an area of increased uncertainties in the Lower Rhine Graben (area H) and at the Westerwald, the Vogelsberg, and the Hesse Depression (area I). The Hesse area generally has low uncertainties. We discuss the reasons for the distribution of the uncertainties in Section 6.5.2.

The results for the deeper depth section are similar with a slight increase in both the posterior mean temperatures and the posterior standard deviations (Fig. 6.15 - 6.18).

Computational Cost

For all analyses, we required in total over a million forward simulations. Using the standard finite element method that would require more than 80 years computation time, not including any sampling times for the MCMC analysis. Using the RB method instead, we require less than 1.5 hours. In the previous section, we already performed various deterministic model calibrations incorporating the results of a global sensitivity analysis. For this analysis, we use the same offline-stage. Hence, we already have our reduced model and do not need to construct a new one. Therefore, the computation time reduces solely to the online execution times.

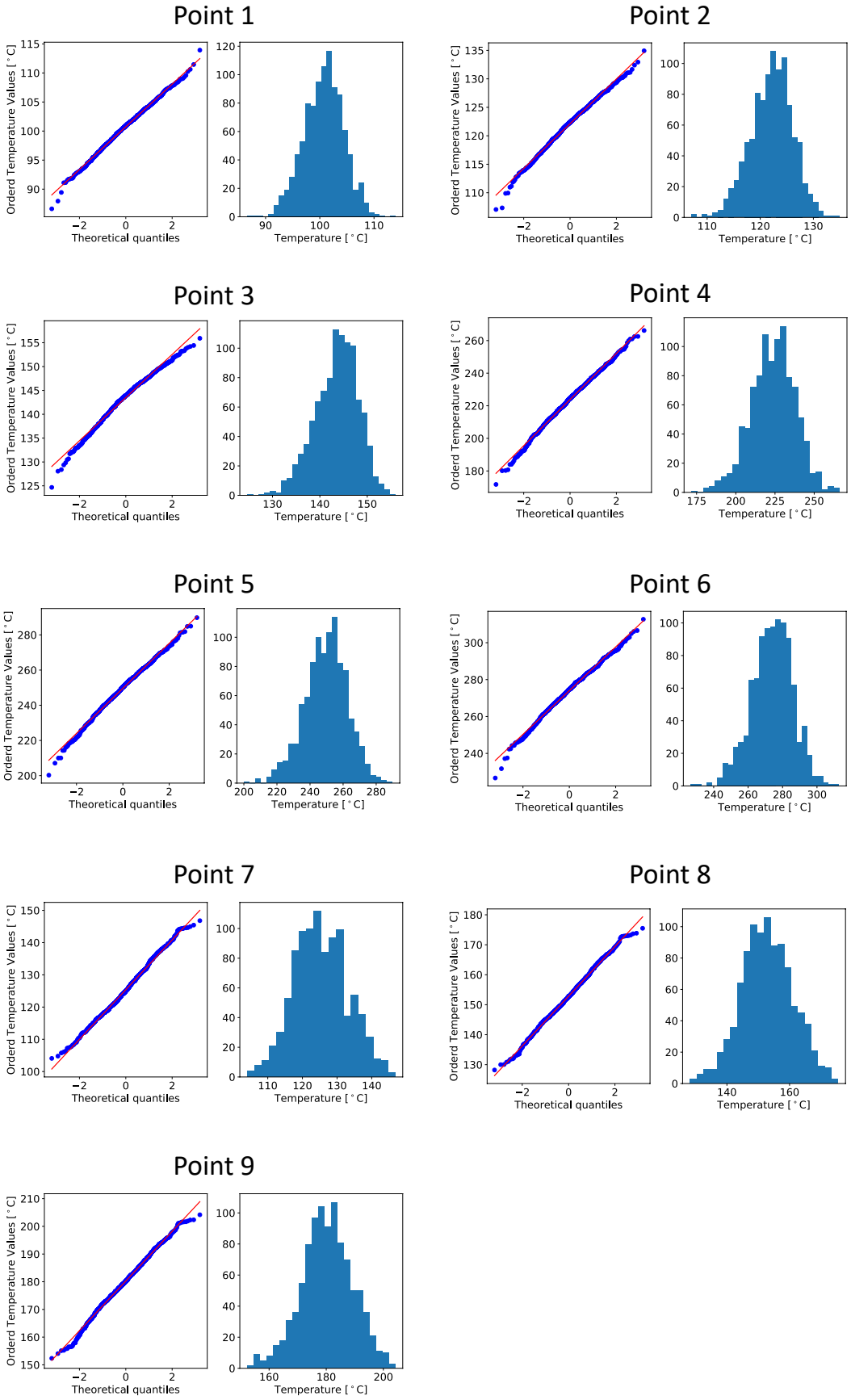


FIGURE 6.10: Quantile-Quantile plots and histograms of the temperatures for all parameters from the MCMC analysis for the Upper Rhine Graben model at nine points.

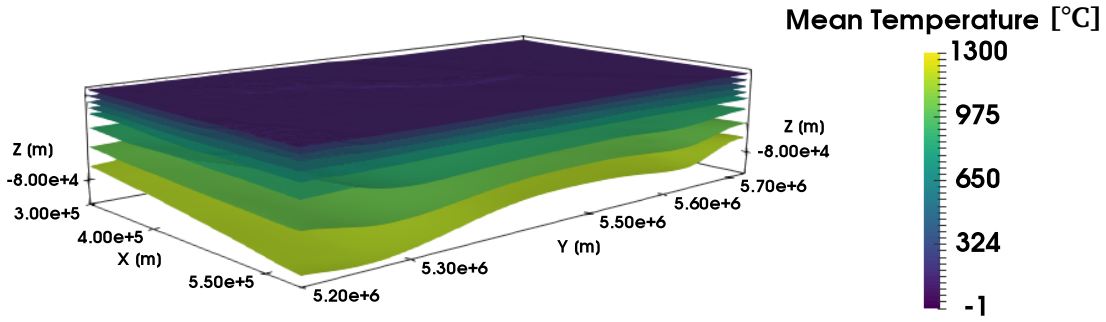


FIGURE 6.11: Map of the posterior mean temperature values for the entire Upper Rhine Graben model. Note that the parameters used for the generation of the posterior mean temperature map are obtained by using the MCMC method.

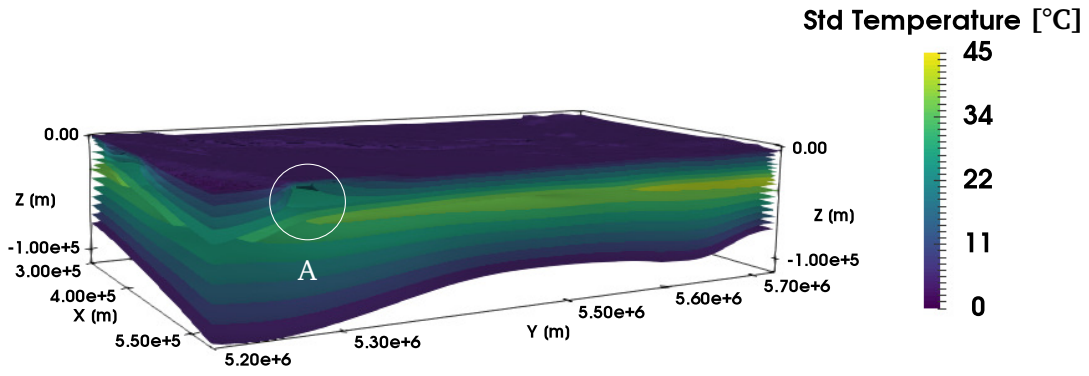


FIGURE 6.12: Map of the posterior standard deviation of the temperature values for the entire Upper Rhine Graben model. Note that the parameters used for the generation of the posterior standard deviation temperature map are obtained by using the MCMC method.

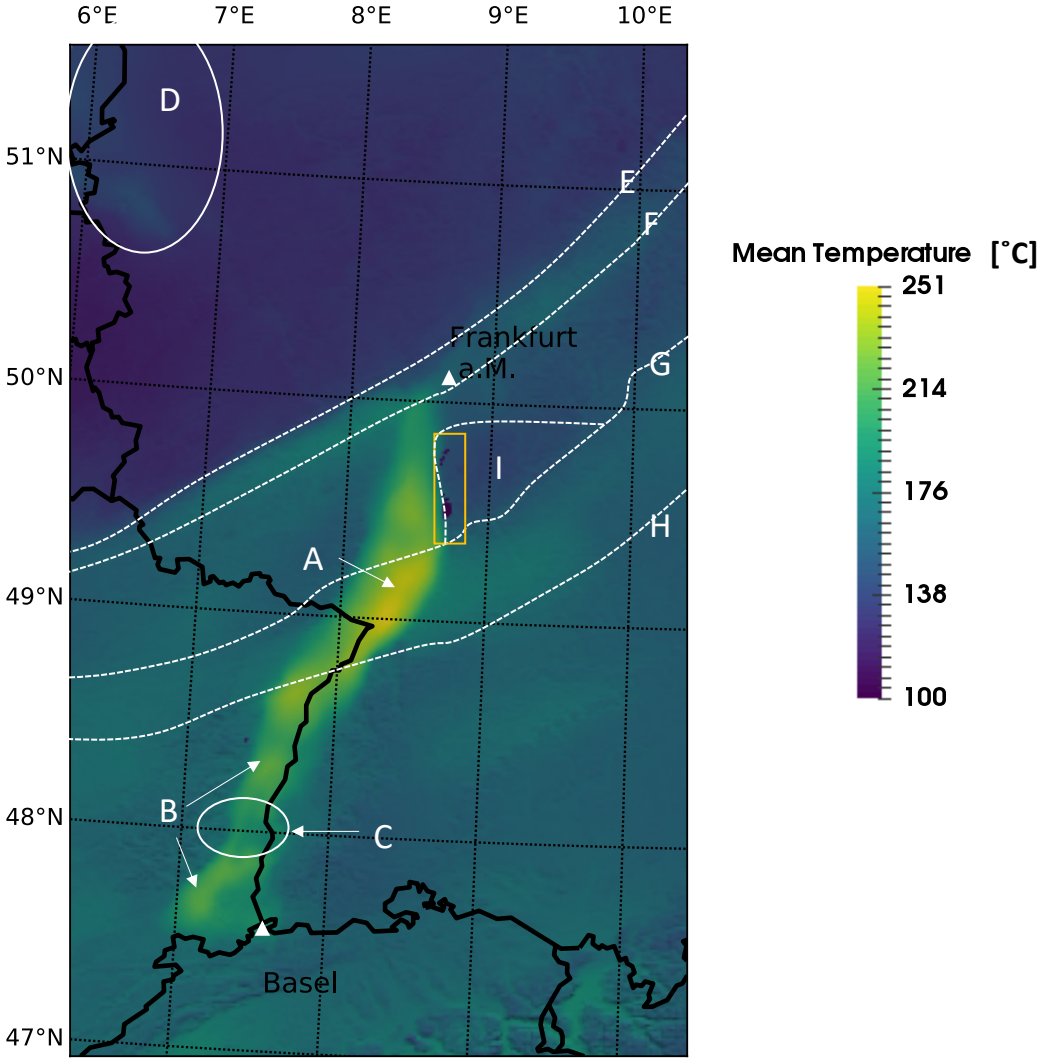


FIGURE 6.13: Map of the posterior mean temperature values for the Upper Rhine Graben model at a depth of 5 km below the surface. Note that the parameters used for the generation of the posterior mean temperature map are obtained by using the MCMC method.

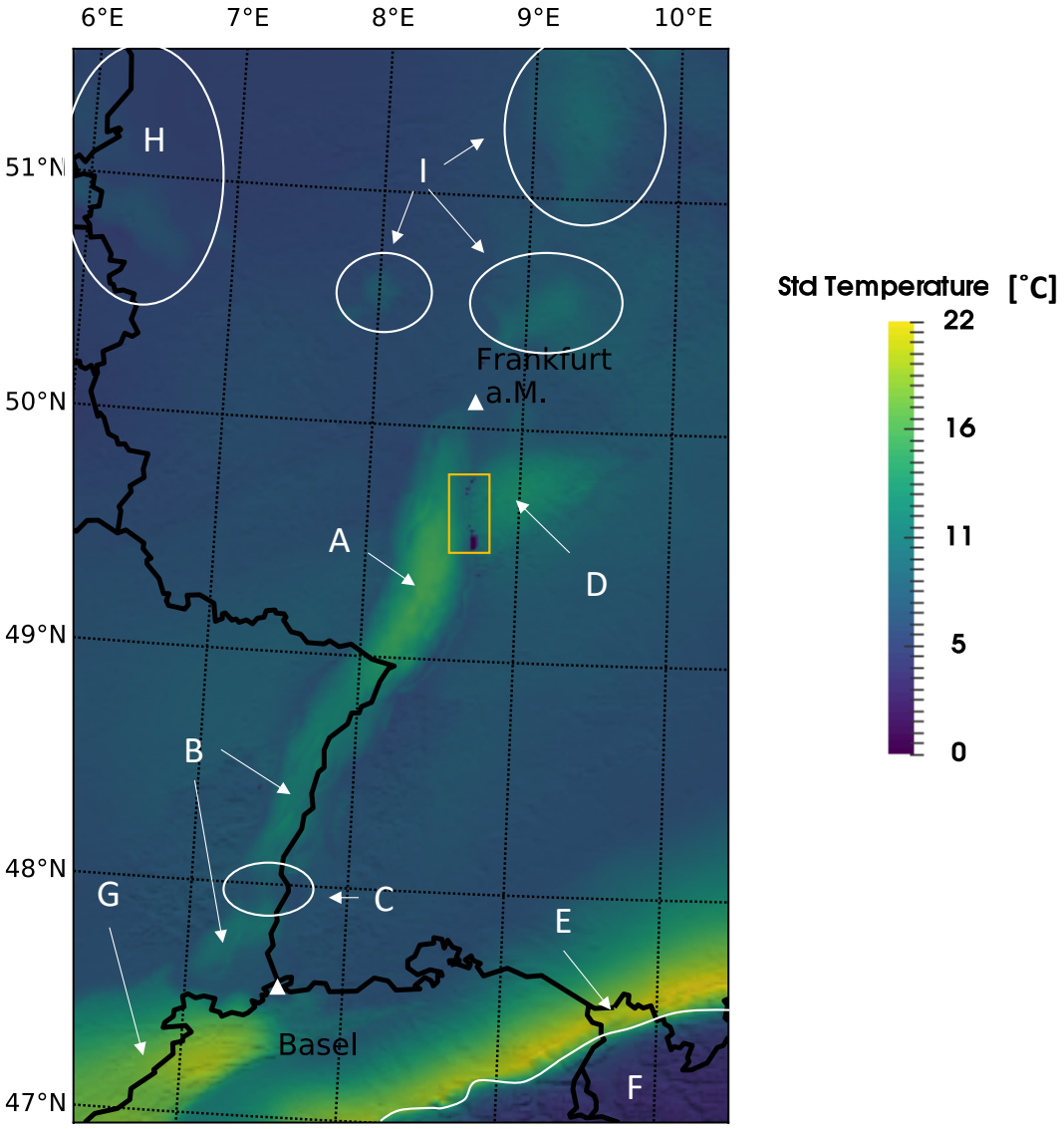


FIGURE 6.14: Map of the posterior standard deviation of the temperature values for the Upper Rhine Graben model at a depth of 5 km below the surface. Note that the parameters used for the generation of the posterior standard deviation temperature map are obtained by using the MCMC method.

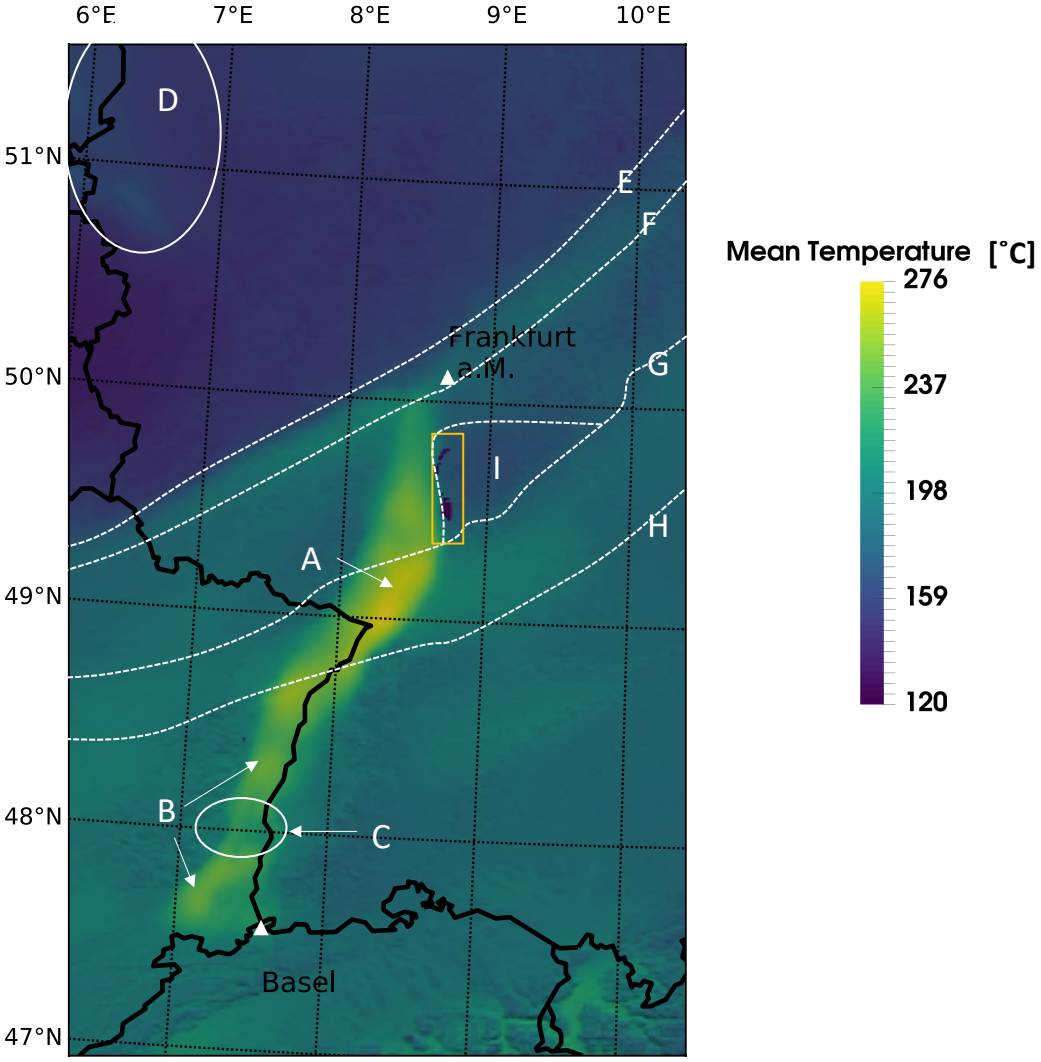


FIGURE 6.15: Map of the posterior mean temperature values for the Upper Rhine Graben model at a depth of 6 km below the surface. Note that the parameters used for the generation of the posterior mean temperature map are obtained by using the MCMC method.

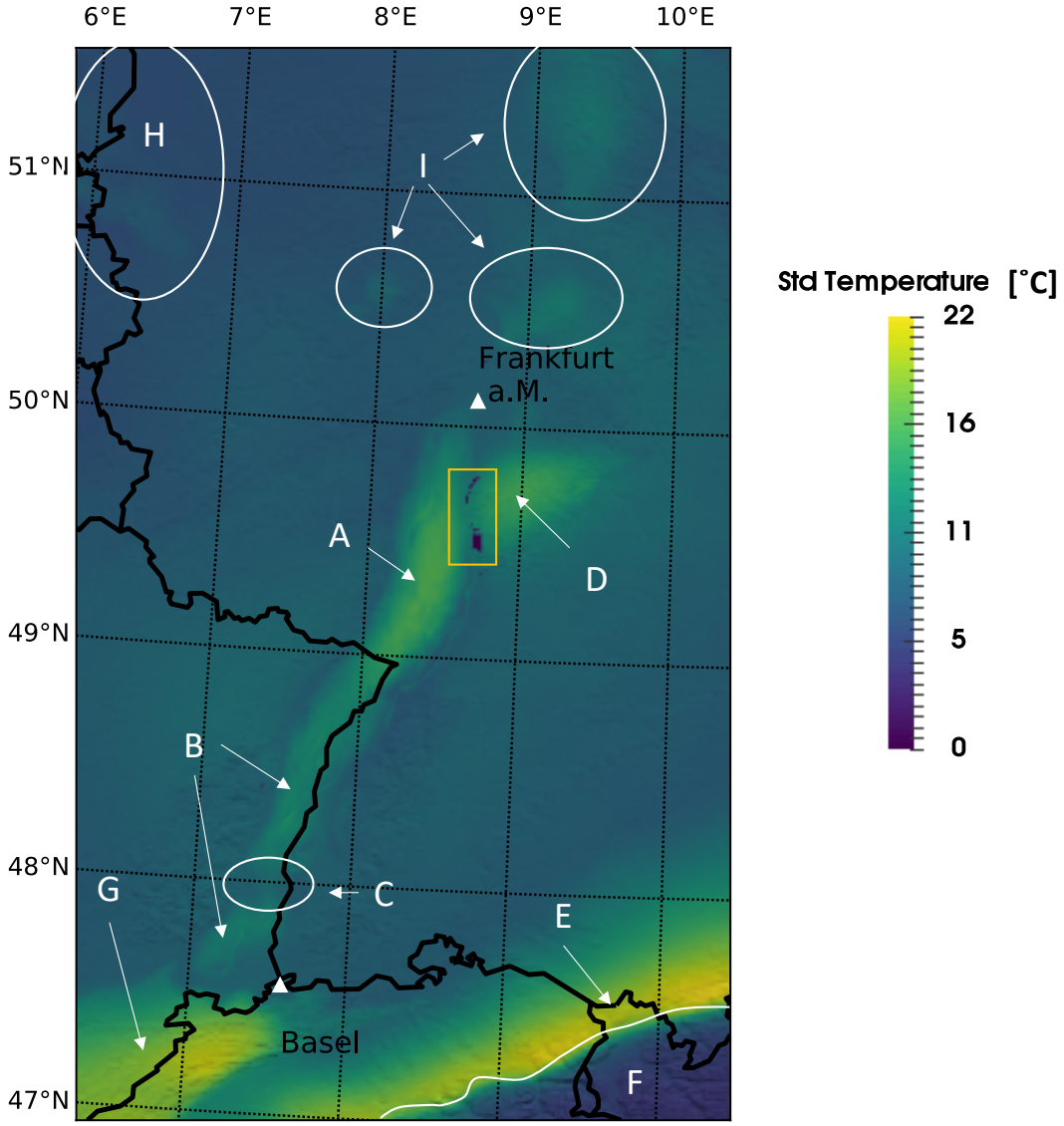


FIGURE 6.16: Map of the posterior standard deviation of the temperature values for the Upper Rhine Graben model at a depth of 6 km below the surface. Note that the parameters used for the generation of the posterior standard deviation temperature map are obtained by using the MCMC method.

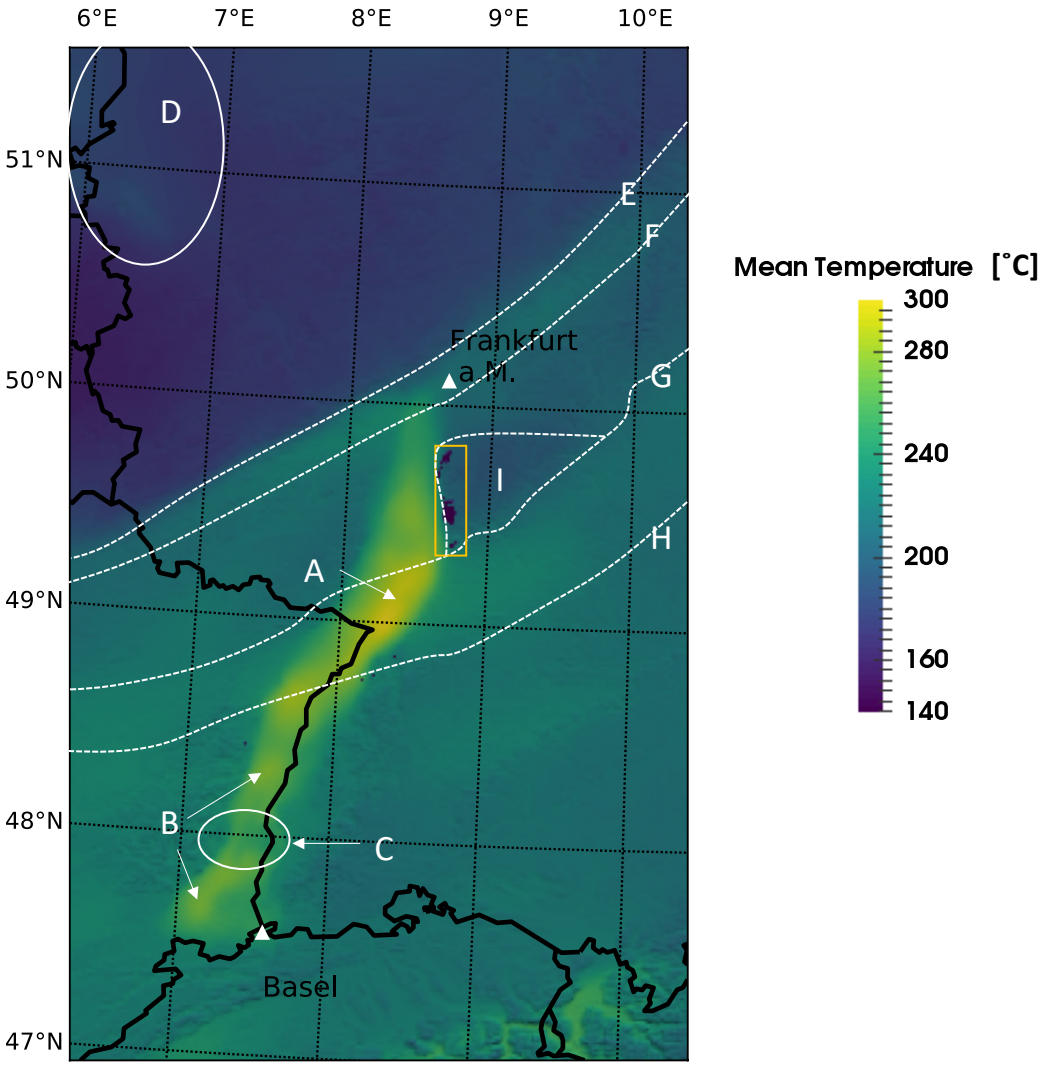


FIGURE 6.17: Map of the posterior mean temperature values for the Upper Rhine Graben model at a depth of 7 km below the surface. Note that the parameters used for the generation of the posterior mean temperature map are obtained by using the MCMC method.

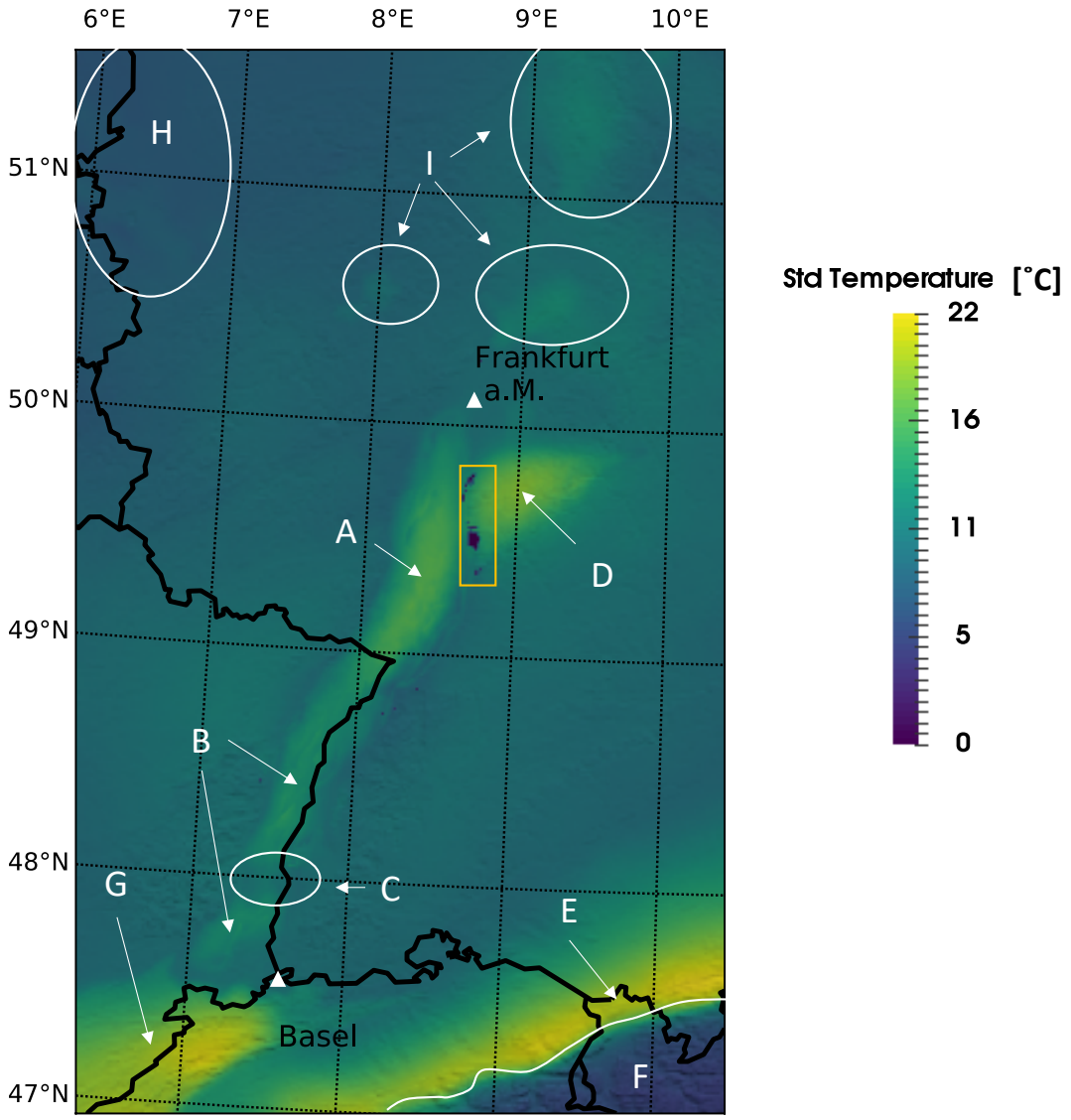


FIGURE 6.18: Map of the posterior standard deviation of the temperature values for the Upper Rhine Graben model at a depth of 7 km below the surface. Note that the parameters used for the generation of the posterior standard deviation temperature map are obtained by using the MCMC method.

6.5 Discussion

Analog to the Results Section, we first discuss the results of the Brandenburg model and afterwards the ones of the Upper Rhine Graben.

6.5.1 Brandenburg Model

In the following, we discuss the results from the uncertainties associated with the thermal conductivities, the spatial distribution of the uncertainties, and the reduced order model.

Thermal Conductivities

To discuss the uncertainties related to the thermal conductivities, we first focus on the posterior mean thermal conductivities. The posterior mean thermal conductivities of all layers show only a slight deviation from the prior thermal conductivities. This is not surprising since they are coming from previous model studies and are therefore already quite well adapted to the model. However, if we compare them to the measured thermal conductivities presented in Noack et al. (2012), we observe an apparent deviation.

Even though the posterior mean thermal conductivities are in a good agreement with the prior thermal conductivities, the need for uncertainty quantification becomes apparent through the posterior standard deviation. For all layers, we observe large posterior standard deviations for the thermal conductivity, meaning that we have high uncertainties for all layers. The uncertainty in the parameters is mainly influenced by the uncertainty of the observation data and by the upper boundary condition. In our study, we place a lot of trust in the data. Still, we allow variations from that data set since we are operating with partially corrected bottom-hole temperatures. We assume that the correction factor is not able to fully compensate for the perturbation of the temperature field during the drilling process, resulting in slightly uncertain observation data. The posterior standard deviation decreases by putting more trust in the observation data. Therefore, temperature observations that are performed when the temperature field is in equilibrium would significantly improve the certainty of the different thermal conductivities.

Except for the Lithospheric Mantle, all posterior mean thermal conductivities show an increase in comparison to the prior thermal conductivity. Since the layers above the salt show an increase in the posterior thermal conductivity and the layer below a decrease, that might be an indication that some salt structures were not resolved. The stochastic calibration demonstrates that a geothermal conduction problem adequately describes the sedimentary basin of Brandenburg. Furthermore, the small posterior standard deviation of the scaling parameter for the lower boundary condition shows that the boundary is placed far enough from the area of interest to avoid any interference.

Uncertainty Quantification Maps

Another reason for uncertainty quantification is the possibility to generate predictive uncertainty quantification maps. The generation of these maps is possible by using the RB method as a surrogate model and reveals important insights into the spatial distribution of the uncertainties. Other surrogate models would not allow the generation of predictive uncertainty maps since they are limited to the observation space.

So, after discussing the uncertainties related to the thermal conductivities, we focus on the uncertainties associated with the temperatures in the entire Brandenburg model. The distribution of these uncertainties seems to be, at first, contradictory to our expectations. Usually, one expects an increasing uncertainty with depth. We observe a decreasing uncertainty towards the boundaries and an increasing uncertainty towards the center part of the model instead. Both for the top and the bottom boundary condition, we apply Dirichlet boundary conditions, where the upper boundary condition has a value of 8 °C throughout all simulations. The lower boundary conditions vary by a factor of $\pm$ 10 %. We allow this variation to account for geometrical parameterization errors of the LAB. This is the reason why we observe decreasing uncertainties towards these boundary conditions because the values of the boundaries are relatively fixed within all simulations. The highest uncertainties are between 30 km and 35 km depth, where no interactions of the boundary conditions are observable.

We can also use the distribution of the uncertainties to see the influence of the respective boundary conditions. Although the LAB is at a depth varying from approximately 100 km to 140 km, the boundary significantly influences the model up to a depth of 80 km to 100 km. For our investigations, this is uncritical since our target depth is at 5 km depth. Nonetheless, this demonstrates that it is essential to have a vertical extent that is significantly larger than the target depth. The upper boundary condition is influencing the model to a depth of 10 km, meaning that the upper boundary condition significantly affects our target depth. This is not avoidable since the surface naturally defines the upper boundary. However, this is less critical than the influence of the lower boundary condition because we can determine the upper boundary with a much higher certainty than the lower. Nonetheless, it shows that it is crucial to characterize the upper boundary condition with great detail.

At the target depth, the highest uncertainties are in area A. Hence, they are north of the Tertiary-post-Rupelian and the Rupelian clay (interface B), and south of the Sedimentary Rotliegend and Zechstein interface. The reason is that the variations of the contrast in thermal conductivity are high at these interfaces. Note that the Rupelian clay has a posterior mean thermal conductivity of 1.93 W m^{-1} K^{-1} with a posterior standard deviation of 0.53 W m^{-1} K^{-1} and the Zechstein layer a posterior thermal conductivity of 3.60 ± 0.96 W m^{-1} K^{-1}. Furthermore, the highest uncertainties are adjacent to the region of the salt structures, further emphasizing the influence of the Zechstein layer on the uncertainties. At the target depth, we consider only the Rupelian clay and the Zechstein layer as uncertain and do not include other layers in the uncertainty quantification. The sensitivity analysis showed that the model is insensitive to these parameters. Consequently, the observed uncertainty is arising from the contrast in thermal conductivity between the Rupelian clay/Zechstein and the remaining layers.

The posterior mean temperatures at a depth of 5 km are higher north from the Tertiary-post-Rupelian and the Rupelian clay interface, marked with the letter B because the Tertiary-post-Rupelian has a lower thermal conductivity than the Rupelian clay. The colder posterior mean temperature values in the north-western part of the model (area A) are coming from the high thermal conductivity of the Zechstein layer. It is further emphasized by the round dome structures in the temperature distribution that are typical for salt. The posterior mean temperature after the stochastic model calibration only slightly deviates from the prior temperature distribution since the changes in the posterior mean thermal conductivity are also minor.

Reduced Order Model

Note that the usage of a physics-based machine learning approach has considerable advantages for geothermal and many other geophysical applications, which is caused by the sparsity of the observation data. The data sparsity makes data-driven approaches in many geophysical applications prohibitive. We train the physical model instead of data and are therefore able to overcome the problem with the data sparsity.

The offline stage requires, using two Intel Xeon Platinum 8160 CPUs (24 cores, 2.1 GHz, 192 GB of RAM), 5.4 hours. It requires 273 full solves for the basis generation. In comparison to the total number of evaluations of 1,030,990 performed in this section, the number of full solves for the basis construction therefore becomes negligible. Note that with the finite element method, we would need over 16 core-a for the same number of evaluations. Note also that the total number of evaluations only considers the final runs and no testing phases. Be aware that the total execution time of the MCMC method took about 4.5 hours with the RB method. However, performing 1,000,000 forward iterations with the RB method should require less than 1.4 hours. The sampling routine causes the difference in computation time. We would also see that for the finite element simulations, but because of their time-consuming nature, we cannot show this here. Nonetheless, note that the number of core-a given is in favor of the finite element method and would usually be significantly higher. Take further into account that the number can be decreased by using parallel infrastructures. However, most of the inverse processes used here are not parallelizable. Furthermore, note that we can perform the time-consuming offline phase during the preparation of a geothermal exploration. That results in fast analyses in the field, whereas we need to execute the finite element version during the exploration phase.

We presented an uncertainty quantification with the generation of uncertainty quantification maps at the basin-scale. This is computationally possible since we replace the finite element forward simulation by the reduced basis forward simulation. This results in a reduction of computation time from a couple of hundred seconds to a few milliseconds for a single forward simulation. Hence, we obtain a speed-up of five orders of magnitude. Therefore, we can efficiently perform both global sensitivity and MCMC analyses. Because we consider not only the deterministic but the stochastic temperature distribution, we are able to predict the temperatures more reliably. For the future, it would be interesting to incorporate these temperature uncertainties into the economic evaluation of potential geothermal wells. It would be furthermore interesting to investigate the effects of different observation data qualities on the uncertainty of the model temperature distributions.

6.5.2 Upper Rhine Graben

We first discuss the uncertainties for the various thermal conductivities. Afterwards, we discuss the spatial distribution of the uncertainties related to the temperatures. Finally, we elaborate on the computational cost and the approximation quality.

Thermal Conductivity

For the discussion of posterior thermal conductivities, we first focus on the posterior mean thermal conductivities. We observe for all layers except the Lithospheric Mantle similar prior and posterior mean thermal conductivities. Our prior mean thermal conductivity has been determined by Freymark et al. (2017) by a “trial-and-error” model calibration. Hence, the prior values are exceptionally well adapted to the model of

the Upper Rhine Graben. Nonetheless, the posterior mean thermal conductivity of the Lithospheric Mantle shows a clear deviation from its prior value. This deviation is caused by an inaccurate geometrical characterization of the LAB. This is in accordance with the results of the deterministic model calibration, and for further information, we refer to Chapter 4. Due to the small variations between the prior and posterior mean thermal conductivities, the question arises why do we need a computationally expensive uncertainty quantification?

We answer this question by focusing on the posterior standard deviations of the thermal conductivities. The posterior standard deviations of all layers are similar, ranging from 14 % to 22 %. For the Cenozoic Volcanics, the Dogger/Lias/Keuper/Saxothuringian/Modanubian, and the Cenozoic Folded Molasse/Cenozoic Foreland Molasse/Buntsandstein/Jura Mountains/Odenwald the deviations between the prior and posterior standard deviations are small. Hence, the provided observation data is not informative with respect to these layers. For decreasing the uncertainties in these areas, further temperature measurements at these locations are required. For both the Cenozoic Rift Sediments and the Lithospheric Mantle, we obtain significantly lower posterior standard deviations than our prior standard deviations. Consequently, the observation data helps us to decrease the uncertainties for those two layers.

The Upper Rhine Graben is subject to an ongoing discussion about the factors that determine its temperature distribution (Agemar et al., 2013, 2014; Freymark et al., 2017, 2019; GeORG-Projektteam, 2013; Stober and Bucher, 2015). The uncertainties of the Cenozoic Rift Sediments and the slight increase in the posterior thermal conductivity is an indicator of neglected convection effects. However, this increase stands in contrast to the decrease in thermal conductivity for this layer obtained by the deterministic model calibration. This emphasizes, the need to perform a hydrothermal simulation for the Upper Rhine Graben to further investigate and explain this discrepancy.

In contrast to a deterministic model calibration, we obtain not only one possible parameter realization but a vast number of them. This allows providing an estimate of the expected uncertainties not only for the thermal conductivities but also for the temperatures. Hence, we can use the results of the uncertainty quantification to generate predictive posterior temperature maps of the entire Upper Rhine Graben.

Uncertainty Quantification Maps

We now discuss these predictive temperature maps regarding the spatial distribution of the posterior standard deviation of the temperatures in the entire Upper Rhine Graben. For the volumetric distribution of the uncertainties, we observe zero uncertainties at the boundaries. This might seem a bit counter-intuitive since, from our geological knowledge, we would expect uncertainties that increase with depth. So, why do we obtain an uncertainty of zero °C at the lower boundary condition? One answers the question by having a closer look at the boundary conditions of the model. We applied both at the top and bottom of the model a Dirichlet boundary condition. Hence, we are fixing the temperatures at these surfaces. Consequently, all models obtained from the MCMC parameter distribution have precisely the same values at these surfaces, resulting in an uncertainty of zero. We can furthermore see that the lower boundary condition does not impact our target depths of five to seven km. However, the target depths show significant influences of the upper boundary conditions. A more detailed discussion about this phenomenon has been presented in Section 6.5.1 for the Brandenburg model, where we encountered the same phenomenon.

The uncertainties are with a maximum value of 45 °C very low, which is, as described above, a consequence of the used boundary conditions. Hence, for all further studies, we need to keep in mind that the uncertainties shown here are likely significantly underestimating the true uncertainties. However, using a Dirichlet boundary condition to describe the surface temperatures is standard procedure. We are not able to move the top boundary condition further away from the area of interest since we know the temperatures only at the surface. What we need to do in future studies is to incorporate the uncertainties, if available, from the surface temperature measurements. Furthermore, we need to account for climate interactions, which means that the current Dirichlet boundary condition has to be replaced by a Robin boundary condition. These studies have so far not been conducted on the basin scale due to the enormous computational cost that is involved. However, using the RB method and simple parameterizations of the climate effect, these analyses are possible.

We now focus the discussion on the posterior standard deviations of the temperatures at our target depth of 5 km to 7 km. We observe higher uncertainties in the Upper Rhine Graben (area A and B). This is partly related to the higher temperatures in the Upper Rhine Graben. However, the high posterior mean temperatures and the high posterior standard deviations do not correlate inside the entire Upper Rhine Graben. We know from the deterministic model calibrations that the conductive model fails to represent the temperature distribution in the Upper Rhine Graben. Hence, we expect the largest mismatches between the simulated and observed temperatures in this region. Furthermore, we have significantly fewer data available in the Upper Rhine Graben than in the Hesse area. Consequently, the MCMC method is better able to reduce the uncertainty in the Hesse area than inside the Upper Rhine Graben.

Area C, also located in the Upper Rhine Graben, has a lower posterior standard deviation. This area is characterized by Tertiary Salt yielding a lower posterior mean temperature. Furthermore, we did not consider the thermal conductivity of the Tertiary salt as uncertain. Combining these two aspects yields the lower posterior standard deviation.

Higher posterior standard deviation values clearly outline the Odenwald (area D). In contrast, the posterior mean temperatures for this area are lower than for the surrounding units. This has several reasons. First, the thermal conductivity of the Odenwald is considered as uncertain, whereas the thermal conductivity of the Saxothuringian and Mid-German Crystalline High are treated as known. Hence, the Odenwald has varying thermal conductivities, and the others have fixed values. This results in the higher posterior standard deviations for the temperatures in area D. Second, no temperature data is available in this area. Consequently, we are not able to reduce the uncertainty in this area.

Interesting is also that we obtain an interface between significantly higher and lower uncertainties (areas E and F). The Molasse Basin (area E) has the highest posterior standard deviations because both the Cenozoic Foreland and Folded Molasse were considered as uncertain. Again no data is available to reduce the prior standard deviations of the thermal conductivities yielding the high posterior standard deviations of the temperature. The lower posterior standard deviations of the Alps (area F) are partially associated with the lower posterior mean temperatures. The main reason for the low uncertainty is that we did not vary this thermal conductivity in the uncertainty quantification. We do not treat the thermal conductivity of the Alps as uncertain, because the global sensitivity study showed that the model is intensive to it. The low uncertainties in the Hesse areas result from the high amount of data available for this area.

The Jura Mountains (area G) have a similar posterior standard deviation as the Molasse Basin. Analog to the Molasse Basin, the reason for this high uncertainty, arises from the uncertainty of the thermal conductivity of the Jura Mountains and the data sparsity in this area.

For the Lower Rhine Graben (area H), we observe sightly higher posterior mean temperatures and standard deviations than for the surrounding units. The Cenozoic Rift Sediments are also present in the Lower Rhine Graben. We previously showed that the thermal conductivity of this layer is associated with a posterior standard deviation of 15 %. This uncertainty causes the higher posterior standard deviation in the Lower Rhine Graben and also the higher standard deviations in the south-west corner of the cross-section. The area has no observation points, which would again prohibit the uncertainty quantification from reducing the uncertainty in this area. Nonetheless, we observe for the Cenozoic Rift Sediments a significant reduction from the prior to posterior standard deviation. However, this reduction is caused by the data insight the Upper and not the Lower Rhine Graben.

Additionally, we observe higher posterior standard deviations in the area I, corresponding to the Westerwald, Vogelsberg, and Hesse Depression. The Cenozoic Volcanics characterize all three areas. The thermal conductivity of this layer is uncertain, and again due to the data sparsity, the MCMC analysis has not been able to reduce the uncertainty from our prior to posterior. This results in the higher standard deviations for these three regions.

We discuss now the posterior mean temperatures in a depth of 5 km to 7 km. The areas A to D, and the area I have been discussed above. The interfaces E-H correspond to the variscan domain boundaries. Here, E is the interface between the Rhenohercynian and the Northern Phyllite Zone. The Rhenohercynian has lower posterior mean temperatures than the Northern Phyllite Zone due to the lower radiogenic heat production rate. F is the interface between the Northern Phyllite Zone and the Mid-German Crystalline High. The Mid-German Crystalline High has a lower radiogenic heat production and a slightly lower thermal conductivity than the Nothern Phyllite Zone, yielding the slightly lower posterior mean temperatures. The interface G is the transition of the Mid-German Crystalline High and the Saxothuringian. The Saxothuringian has due to the higher radiogenic heat production rate higher posterior mean temperatures. The last variscan domain boundary H is the interface between the Saxothuringian and the Moldanubain. The Moldanubian has a minimal higher posterior mean temperature due to the slightly higher radiogenic heat production. The posterior mean temperature distributions are in accordance with the temperature distribution found in the literature (Freymark et al., 2017).

For geothermal explorations, one can conclude that the Upper Rhine Graben is the best-suited location, as already widely discussed by Agemar et al. (2013), Agemar et al. (2014), Tiefe Geothermie (2007), Illies (1972), Freymark et al. (2017), Pauwels et al. (1993), and Vidal et al. (2015). However, one should also note that it is, at the same time, also an area with high uncertainty. Considering, also, the likely highly underestimated uncertainties due to the chosen upper boundary condition, we need a further investigation of the model and the conduction of additional observation data to quantify the uncertainties better. In that context, it would be interesting to perform a design of experiment study to determine the location of new observations that are required to reduce the uncertainty of the temperature distribution.

6.5.3 Computational Cost and Approximation Accuracy

The computational cost of both the MCMC run and the generation of the uncertainty quantification method is extremely low. That is the case because we already constructed the reduced model for the previous sensitivity studies and model calibrations. It demonstrates the usefulness of the RB method in general. The reduced model, as used in this thesis, is physics preserving and is not tailored for a specific inverse method. Hence, we can use the same reduced model for various inverse problems. It is also possible to tailor the reduced model to specific inverse problems as done in, for instance, Grepl and Kärcher (2011), and Kärcher and Grepl (2014). This has the two main advantages: i) that we also obtain error bounds for the inversion, and ii) that the reduced basis is smaller.

In our specific example, we constructed the reduced model, such that each forward simulation is more accurate than the measurements. Therefore, in our case, the reduced and full model are equivalent, making the monitoring of the error bound for the inversion unnecessary. Even if we obtained the error bounds, including the inverse process, it would only provide the difference to the full but not to the true model. However, below the measurement error, we have no guarantee that our full model is a representation of the true model. Additionally, we need the reduced basis for several outer-loop processes. That is the reason why the non-tailored reduced model is more efficient for us since it allows the re-usage for various applications. It would not be directly possible with the tailored reduced model.

Chapter 7

Certified Reduced Basis Method for Groundwater Applications

Similar to Chapter 3, this chapter is a conceptual introduction to the RB method for transient geophysical applications. Therefore, we use the Fault model as a simple benchmark problem governed by a Darcy flow problem. Besides the application to the forward problem, we also present how we can incorporate parameter-dependent initial conditions and various coupling options for the DwarfElephant package.

7.1 Darcy's Law

Darcy's law is used to model fluid flow in porous media, and based on the conservation of momentum. Considering the pressure formulation, it has the following form:

$$q = -\frac{k}{\mu}(\nabla p - \rho g), \tag{7.1}$$

where q is the Darcy flux, k the permeability, μ the dynamic viscosity, p the pressure, ρ the fluid density, and g the gravitational acceleration. As for the geothermal problem, we consider, for efficiency reasons, the nondimensional Darcy's law. Therefore, we define the following nondimensional variables and parameters:

$$\nabla^* = \frac{\nabla}{l_{\text{ref}}}, \tag{7.2}$$

$$p^* = \frac{1}{p_{\text{ref}}}(p + p_0 - \rho g z), \tag{7.3}$$

where p_{ref} is the reference pressure, p_0 the atmospheric pressure, and z the depth of the model.

$$k^* = \frac{k}{k_{\text{ref}}}, \tag{7.4}$$

$$\mu^* = \frac{\mu}{\mu_{\text{ref}}}. \tag{7.5}$$

Here, k_{ref} is the reference permeability and μ_{ref} the reference dynamic viscosity. Using the nondimensional variables and parameters yields the following nondimensional Darcy's law.

$$\frac{q\ \mu_{\text{ref}}}{p_{\text{ref}}\ l_{\text{ref}}\ k_{\text{ref}}} = -\frac{k^*}{\mu^*}\nabla^* p^*. \tag{7.6}$$

If we additionally want to account for the specific storage S_s, and a source term Q, we obtain the following dimensional form:

$$S_s \frac{\partial p}{\partial t} - \nabla(\frac{k}{\mu}(\nabla p - \rho g)) - \frac{Q(z,t)}{\rho} = 0, \tag{7.7}$$

and the nondimensional equation can be expressed by:

$$\frac{S_s^* \; S_{s,\text{ref}} \; p_{\text{ref}} \; l_{\text{ref}}^2}{t_{\text{ref}}} \frac{\partial p^*}{\partial t^*} - \nabla^*(\frac{k^* \; k_{\text{ref}} \; p_{\text{ref}}}{\mu^* \; \mu_{\text{ref}}} \nabla p^*) - \frac{Q^*(z,t) \; Q_{\text{ref}}}{l_{\text{ref}} \; \rho \;\; \rho_{\text{ref}}} = 0, \tag{7.8}$$

where, we consider the the nondimensional specific storage as: $S_s^* = \frac{S_s}{S_{s,\text{ref}}}$, and the nondimensional source term as: $Q^*(z,t) = \frac{Q(z,t)}{Q_{\text{ref}}}$ with the reference specific storage $S_{s,\text{ref}}$ and the reference source term Q_{ref}.

If we solve for the hydraulic head h, instead of the pressure, and neglect both the source term and the specific storage, we can formulate Darcy's law as:

$$q = -\frac{k}{\mu} \nabla h, \qquad \text{with} \quad h = z + \frac{p}{\rho g}. \tag{7.9}$$

We obtain the nondimensional Darcy's law by using the above defined variable transformations plus:

$$h^* = \frac{h}{h_{\text{ref}}}, \tag{7.10}$$

where h_{ref} is the reference hydraulic head. The nondimensional Darcy's law has the same structure as the one using the pressure formulation:

$$\frac{q \; \mu_{\text{ref}}}{h_{\text{ref}} \; l_{\text{ref}} \; k_{\text{ref}}} = -\frac{k^*}{\mu^*} \nabla^* h^*. \tag{7.11}$$

7.2 Grid Search

Grid search is a method that will be used here to explore the parameter space. Before applying the method, one needs to define an upper and lower bound for all model parameters that should be explored. We partition each parameter with equal spacing between the upper and lower bound. This results in a grid over the parameter space as displayed in Fig. 7.1. We evaluate the response for every possible parameter combination. If the spacing is small enough, the grid search algorithm results in the best method of exploring the parameter space, but also in the most expensive one (Zabinsky, 2013).

7.3 Fault Model

As for the geothermal application, we first introduce the concepts of the RB method through a simple benchmark problem. This time, we investigate only the Fault model with the hydraulic head formulation of Darcy's law. We have a reference permeability of 10^{-12} m^2 and a reference hydraulic head of 300 m. On the left and right side of the model domain, we apply Dirichlet boundary conditions. The left side has a hydraulic head value of 300 m, and the right side of 400 m. This yields the nondimensional boundary conditions of 0 for the left side, and 0.1 for the right side. We listed the permeabilities in Tab. 7.1. The simulation parameters for both the FE and RB problem

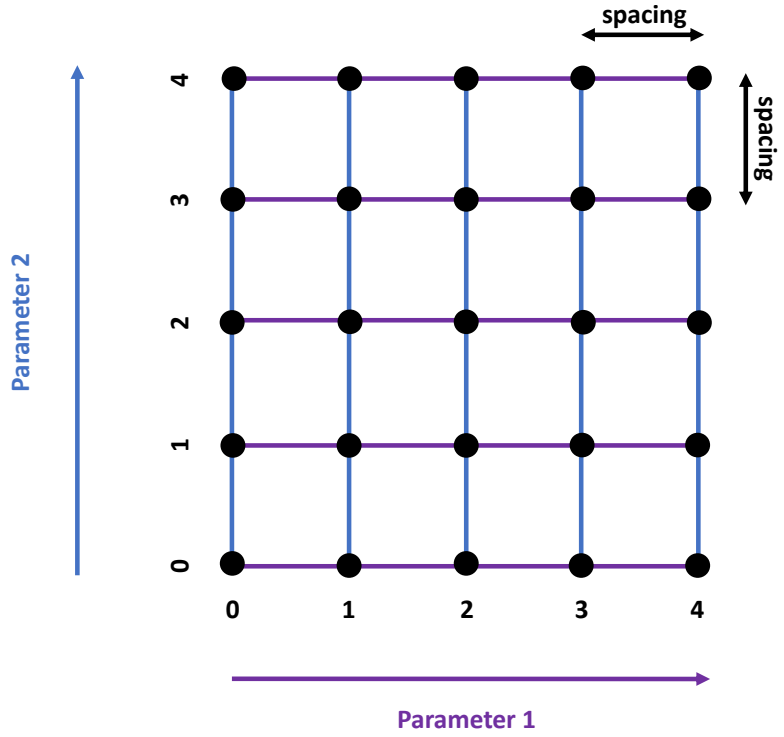


FIGURE 7.1: Exploration of the parameter space for the grid search algorithm.

are the same as listed in Tab. 3.1, except for the linear and non-linear solver tolerances. For all RB simulations, we use a linear and non-linear solver tolerance of 10^{-8}. Using the same solver tolerance for the FE model, we obtain non-converging simulations for the models with 1,040,325 and 1,969,077 degrees of freedom for the last time steps. Therefore, we change the linear and non-linear solver tolerance to 10^{-10} for these two FE models.

Furthermore, we consider the following decomposition for the bilinear form a:

$$a(w, v; k) = -\frac{1}{\mu} \sum_{q=0}^{4} k_q \int_{\Omega_q} \nabla w \, \nabla v \, d\Omega_q, \qquad \forall v, w \in X, \; \forall k \in D. \tag{7.12}$$

Here, w is the trial function, v the test function, the index "q" denotes the number of the training parameter, X the function space ($H_0^1(\Omega) \subset X \subset H_1(\Omega)$), Ω the spatial domain in $\mathbb{R}^3$, and D the parameter domain in $\mathbb{R}^5$. The decomposed linear form f is:

$$\begin{aligned} f(v; k) &= -\frac{1}{\mu} \sum_{q=0}^{2} k_q \int_{\Gamma} \nabla v \; g(x) \; d\Gamma, \qquad \forall v \in X, \; \forall k \in D, \\ \text{with g(x)} &= h_{\text{right}} \; x, \end{aligned} \tag{7.13}$$

where Γ is the boundary in $\mathbb{R}^3$, $g(x)$ the lifting function, h_{right} the hydraulic head at the right model domain boundary.

TABLE 7.1: Hydraulic and model properties for the Fault model. The hydraulic properties are taken as typical values for the respective geological units. Take into account that both the permeability, and the length are dimensionless ($k_{\text{ref}} = 10^{-12}$ m^2, and $l_{\text{ref}} = 1000$ m).

unit	permeability [m^2] / permeability [-]	extension in		
		x-direction [-]	y-direction [-]	z-direction [-]
top layer - section 1 (shale)	10^{-15} / 10^{-3}	0.00 - 0.34	0.00 - 1.00	0.66 - 1.00
top layer - section 2 (shale)	10^{-15} / 10^{-3}	0.40 - 0.70	0.00 - 1.00	0.60 - 1.00
top layer - section 3 (shale)	10^{-15} / 10^{-3}	0.76 - 1.00	0.00 - 1.00	0.66 - 1.00
middle layer - section 1 (sandstone)	10^{-13} / 10^{-1}	0.00 - 0.34	0.00 - 1.00	0.25 - 0.66
middle layer - section 2 (sandstone)	10^{-13} / 10^{-1}	0.43 - 0.70	0.00 - 1.00	0.33 - 0.60
middle layer - section 3 (sandstone)	10^{-13} / 10^{-1}	0.89 - 1.00	0.00 - 1.00	0.25 - 0.66
bottom layer - section 1 (shale)	10^{-15} / 10^{-3}	0.00 - 0.35	0.00 - 1.00	0.00 - 0.25
bottom layer - section 2 (shale)	10^{-15} / 10^{-3}	0.43 - 0.80	0.00 - 1.00	0.00 - 0.33
bottom layer - section 3 (shale)	10^{-15} / 10^{-3}	0.89 - 1.00	0.00 - 1.00	0.00 - 0.25
Fault 2	10^{-12} / 1	0.20 - 0.90	0.00 - 1.00	0.10 - 0.80
Fault 1 - section 1	10^{-12} / 1	0.20 - 0.55	0.00 - 1.00	0.45 - 0.80
Fault 1 - section 2	10^{-12} / 1	0.60 - 0.95	0.00 - 1.00	0.20 - 0.50

7.4 Results

We first present the results of the forward problem. Afterwards, we demonstrate how we can use parameter-dependent initial conditions within the RB method. Finally, we provide a comparison for the different coupling options of the DwarfElephant package,

where we point out the advantages and disadvantages for each coupling and possible application fields.

7.4.1 Darcy Flow Problem

For the Darcy flow problem, we consider only the Fault Model since we start with a hydraulic head equally distributed between 0 and 0.1 as an initial condition. So, for this scenario, we would not obtain any visible changes for the Base model.

We use the same model as for the elliptic (steady-state) case to evaluate the difference in the performance of the RB method for time-dependent and -independent problems. For the Fault model, we observe a speed-up that varies between 1.2 10^4 and 1.2 10^7 for an increase in degrees of freedom from 4,600 to 1,969,077 (Fig. 7.2). This is an increase in speed-up in comparison to the elliptic case of 388 to 10,560.

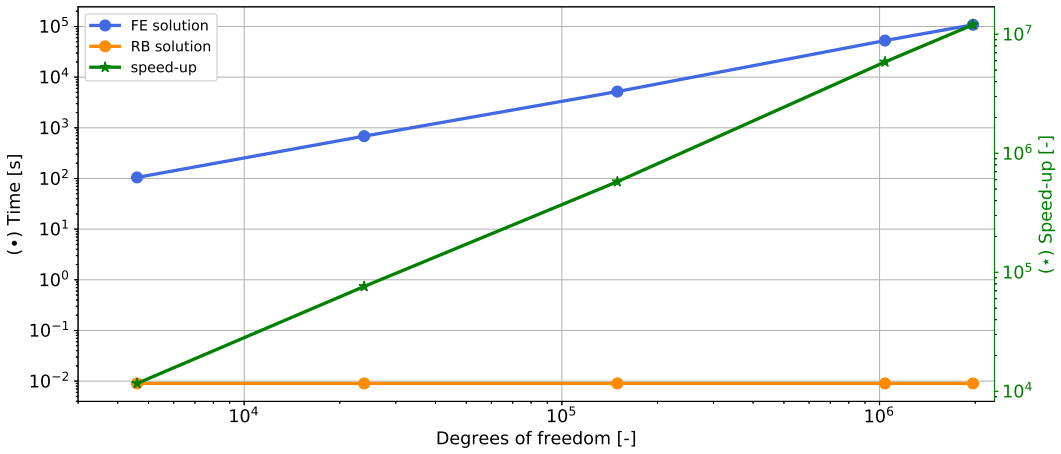
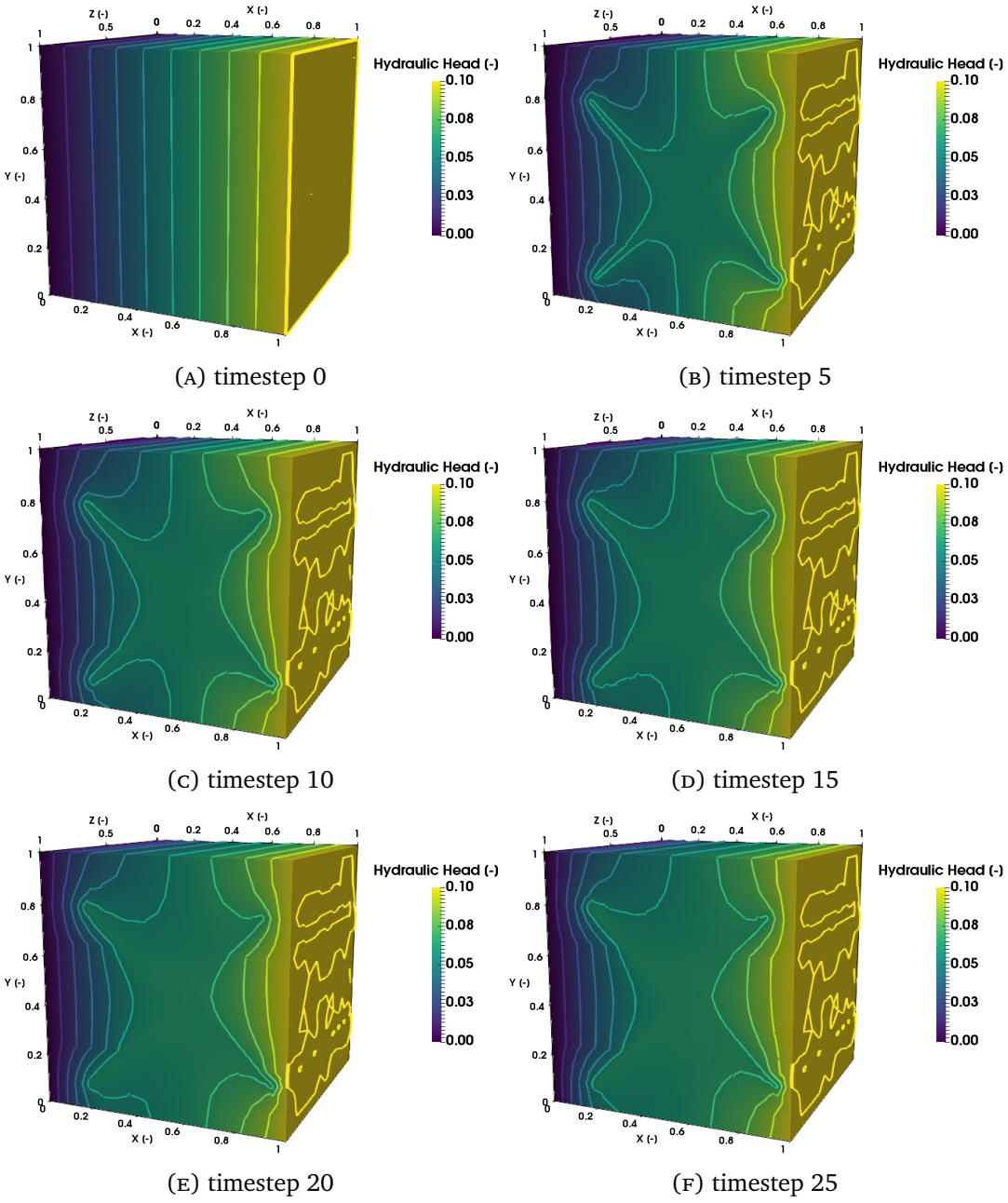


FIGURE 7.2: Runtime comparison between the full finite element solution –denoted in blue– and the RB solution –denoted in orange– for the Darcy flow problem using the Fault model with permeable faults with a solver tolerance of 10^{-10}. Furthermore, the blue line illustrates the speed-up.

The hydraulic head distribution, displayed in Fig. 7.3 and 7.4 for every fifth time step, is in accordance with our expectations. We observe higher hydraulic head values at and around the fault location, and no visible effects from the permeability contrast of the sand and shale layer. The hydraulic head has a minimum value of 0 (corresponds to 300 m) and a maximum value of 0.1 (400 m), which is matching the boundary conditions.

We set an error tolerance of 10^{-5} within the greedy algorithm. Fig. 7.5 and 7.6 show the difference between the full finite element and the reduced solution. Here, we observe the highest errors in the low permeable layer. Furthermore, the errors are similar except from the initial time step, where we have an error of zero everywhere. For the parameter combination presented here, we slightly exceed the given error tolerance.

As for the elliptic case, we evaluate the maximum relative error bound as a function of the number of basis functions. Fig. 7.7 shows the number of basis functions required to reach the error tolerance. We used the same random seed for all basis constructions to observe only the effects arising from the increasing mesh size. We observe that with large meshes, we get a slight increase in the number of basis functions of four to six basis functions.



(A) timestep 0 (B) timestep 5

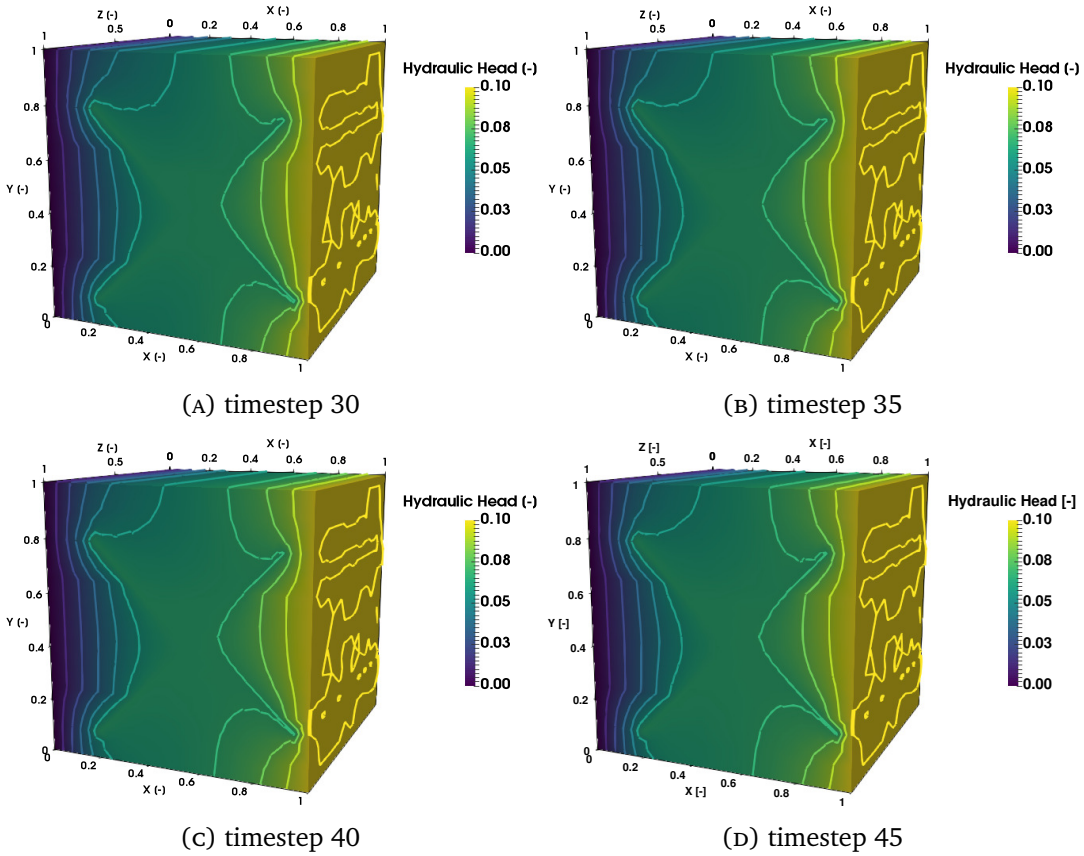
(C) timestep 10 (D) timestep 15

(E) timestep 20 (F) timestep 25

FIGURE 7.3: Hydraulic head distribution at every fifth time step. Note that both the mesh dimensions and the hydraulic head have been normalized for stability reasons, and are hence dimensionless.

Analog to the elliptic case, we can not observe a pattern in the evolution of the maximum relative error bound (Fig. 7.8). This demonstrates that the parameter complexity does not significantly increase with increasing degrees of freedom.

For the Fault model, we need up to five simulations, see Fig. 7.9, to be more efficient with the RB method than with the FE method. In comparison, we required for the elliptic case up to 50 simulations to be more efficient with the RB than the FE method. Note that we could no longer execute the offline stage of the Fault model with 1,040,325 and 1,969,077 degrees of freedom on a single core, due to memory limitations. They were instead executed using 48 cores. To ensure a fair comparison also, the FE simulations have been repeated using the same number of cores.



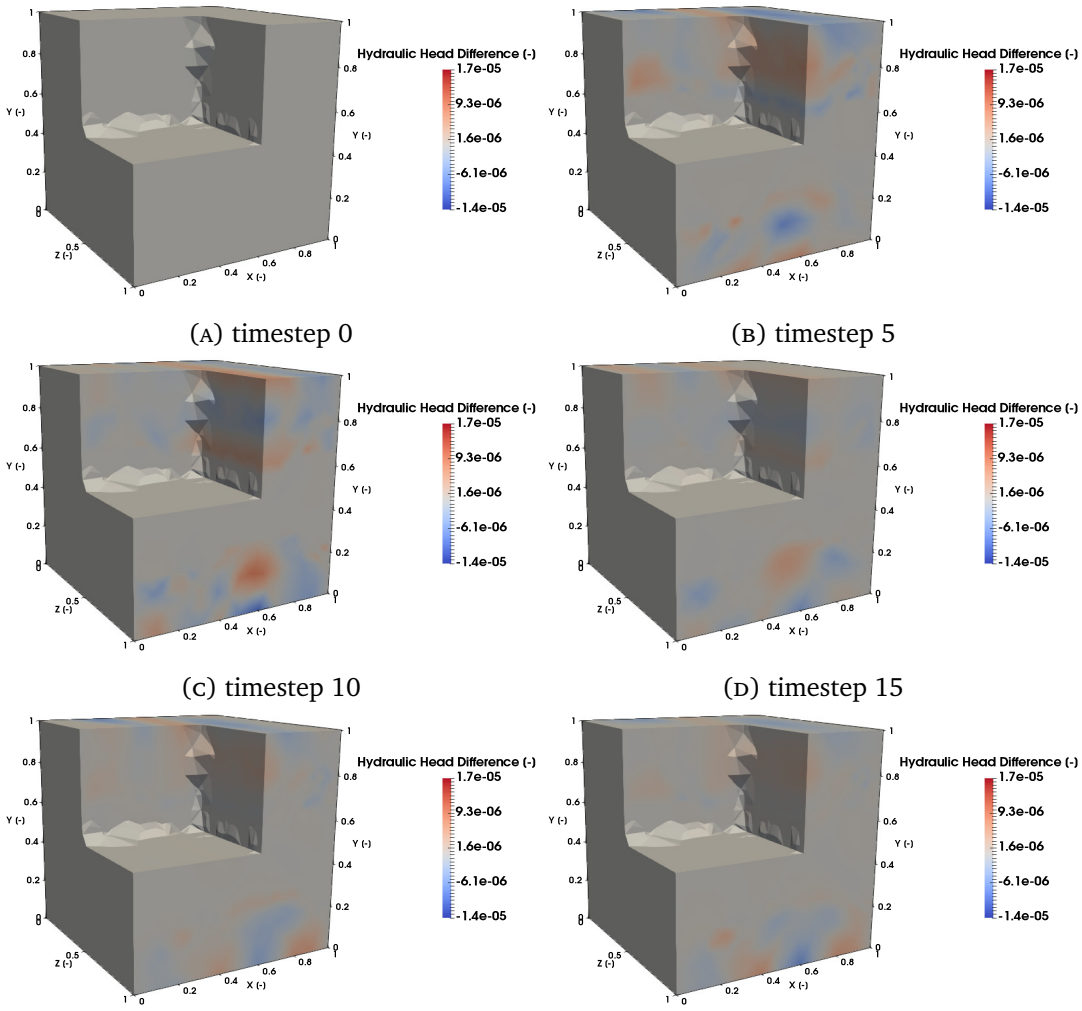
(A) timestep 30

(B) timestep 35

(C) timestep 40

(D) timestep 45

FIGURE 7.4: Continuation: Hydraulic head distribution at every fifth time step. Note that both the mesh dimensions and the hydraulic head have been normalized for stability reasons, and are hence dimensionless.

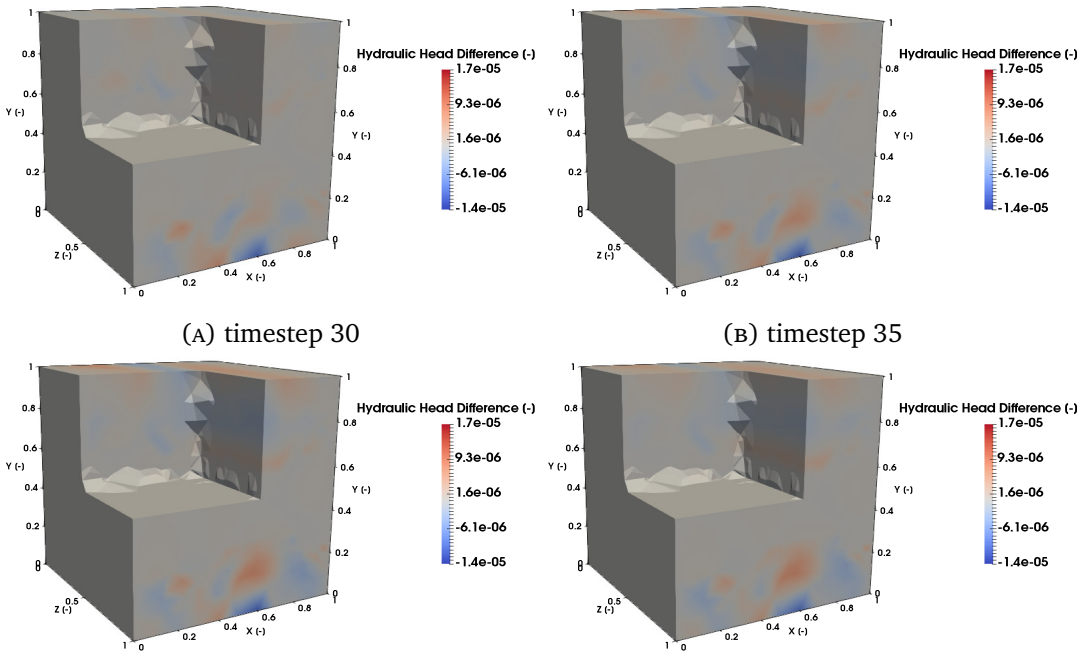


(A) timestep 0 (B) timestep 5

(C) timestep 10 (D) timestep 15

(E) timestep 20 (F) timestep 25

FIGURE 7.5: Difference plot of the hydraulic head distribution between the full finite element and the reduced basis solution for every fifth time step.



(A) timestep 30 (B) timestep 35

(C) timestep 40 (D) timestep 45

FIGURE 7.6: Continuation: Difference plot of the hydraulic head distribution between the full finite element and the reduced basis solution for every fifth time step.

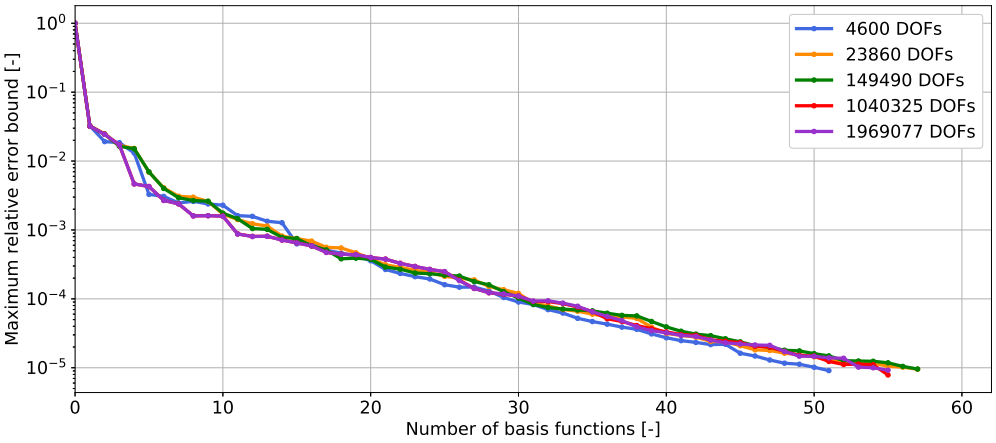


FIGURE 7.7: Convergence rate of the construction of the RB space for the Darcy flow problem using the Fault model for varying degrees of freedom and a relative error tolerance of 10^{-5} and solver tolerances of 10^{-10}.

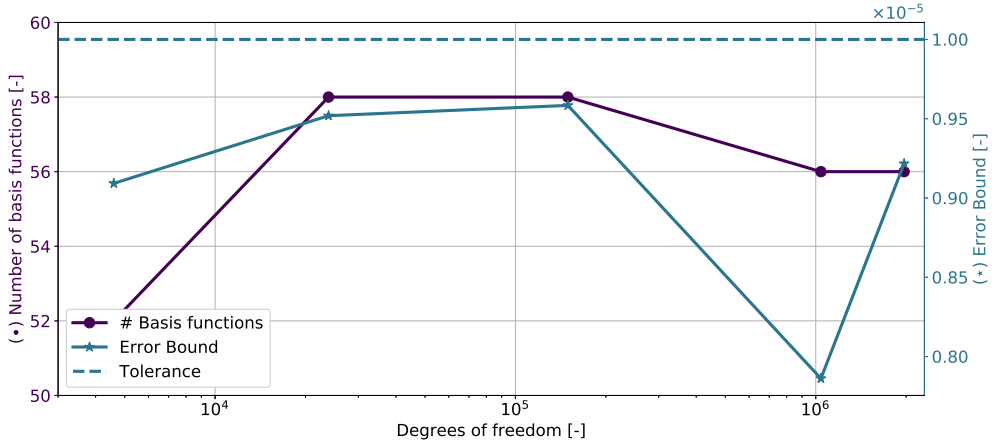


FIGURE 7.8: The evolution of the maximum relative error bound at the final basis function, and the number of basis functions required for reaching a relative tolerance of 10^{-5} by using solver tolerances of 10^{-10} for the Darcy flow problem using the Fault model.

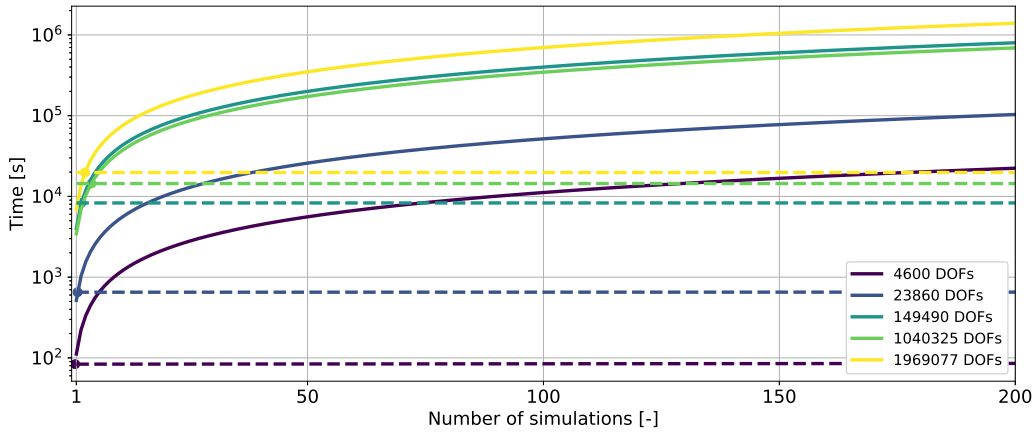


FIGURE 7.9: Comparison of the runtimes for the FE approach (solid lines) and the combined offline and online times of the RB approach (dashed lines) for varying degrees of freedom and solver tolerances of 10^{-10} for the Darcy flow problem using the Fault model over 200 simulations.

7.4.2 Parameter-Dependent Initial Conditions

Consider the following scenario: previous to our investigated time frame we have injected a pressure pulse. Unfortunately, the exact amount of pressure that entered the groundwater is not known. We only know that it is between a hydraulic head value of 20 m and 40 m. We now want to examine how this pressure pulse affects the hydraulic head distribution, dependent on the amount of pressure and the other model parameters, in our model. The pressure pulse can be modelled as a Gauss function of the following form:

$$A \ \exp(-\frac{(x-x_0)^2}{2\sigma_x^2} + \frac{(y-y_0)^2}{2\sigma_y^2}), \tag{7.14}$$

where A is the magnitude of the pressure, x_0 the x-coordinate of the center, y_0 the y-coordinate of the center, and σ_x and σ_y define the spread of the pressure pulse. We set the center of the pressure pulse to the middle of the surface of the model. Hence, x_0 and y_0 are equal to 500 m. For σ_x, we chose 80 m and for σ_y 750 m.

To evaluate the changes in the hydraulic head, we define 27 equally spaced observation points, as shown in Tab. 7.2. For many hydrological applications, the changes in hydraulic head must not exceed a certain threshold. To investigate which model parameters are crucial for the trespassing of the threshold, we perform a global sensitivity study with 10,000 samples per parameter yielding 140,000 forward simulations. For this sensitivity study, we investigate six model parameters: i) the permeabilities of the three geological layers, ii) the permeabilities of the two faults, and iii) the initial amount of pressure from the pressure pulse. We evaluate the sensitivity with respect to the absolute difference between the scenario with and without a pressure pulse. We only take the sensitivities into account if the change in hydraulic head is equal or above our pre-defined threshold of ten percent.

For this scenario, we need parameter-dependent initial conditions since the initial amount of pressure A is unknown. We use the same model and set up, as described in Section 7.3 and 7.4.1 with the initial pressure as an additional parameter.

Fig. 7.10 shows the results of the global sensitivity study. We observe that the model response is sensitive to the permeability of the bottom layer and the initial amount of pressure, where the influence of the initial amount of pressure is dominant. Both parameters have nearly exclusive first-order contributions. In the tested parameter ranges, the other permeabilities do not influence the changes of the hydraulic head for our base case. For the last two parameters, we obtain higher first-order than total-order contributions. This is theoretically not possible. It shows that for those two parameters, we need more parameter values. We need them to decrease the statistical error. Therefore, we would need to increase the number of parameters to, for instance, 100,000. In this case study, we did not further increase the number of samples per parameter since the last two parameters do not influence the model predictions.

Note that we have executed the 140,000 forward simulations in about 20 min. As a comparison performing the global sensitivity analysis with an on 48 cores parallelized, FE forward model would require over 15 core-a. That is a speed-up of over $2{\cdot}10^5$.

Now, we want to evaluate how the performance of the RB method changes due to the increased parameter space. The number of basis functions increases from 55 to 75 for the same relative error tolerance. The FE simulation of the problem requires 20.5 h, whereas the online stage requires 14 ms. The online execution time of the problem with an initial pressure pulse is slightly higher than the one without. The obtained speed-up is equal to $5.2{\cdot}10^6$, which is slightly smaller than the one obtained for the scenario without the initial pressure pulse. Both the offline and online execution times

TABLE 7.2: Coordinates of the observation points for the Fault model.

Observation Point	Coordinates (x|y|z)	Observation Point	Coordinates (x|y|z)	Observation Point	Coordinates (x|y|z)
0	(0.1|0.1|0.1)	9	(0.1|0.1|0.5)	18	(0.1|0.1|0.9)
1	(0.5|0.1|0.1)	10	(0.5|0.1|0.5)	19	(0.5|0.1|0.9)
2	(0.9|0.1|0.1)	11	(0.9|0.1|0.5)	20	(0.9|0.1|0.9)
3	(0.1|0.5|0.1)	12	(0.1|0.5|0.5)	21	(0.1|0.5|0.9)
4	(0.5|0.5|0.1)	13	(0.5|0.5|0.5)	22	(0.5|0.5|0.9)
5	(0.9|0.5|0.1)	14	(0.9|0.5|0.5)	23	(0.9|0.5|0.9)
6	(0.1|0.9|0.1)	15	(0.1|0.9|0.5)	24	(0.1|0.9|0.9)
7	(0.5|0.9|0.1)	16	(0.5|0.9|0.5)	25	(0.5|0.9|0.9)
8	(0.9|0.9|0.1)	17	(0.9|0.9|0.5)	26	(0.9|0.9|0.9)

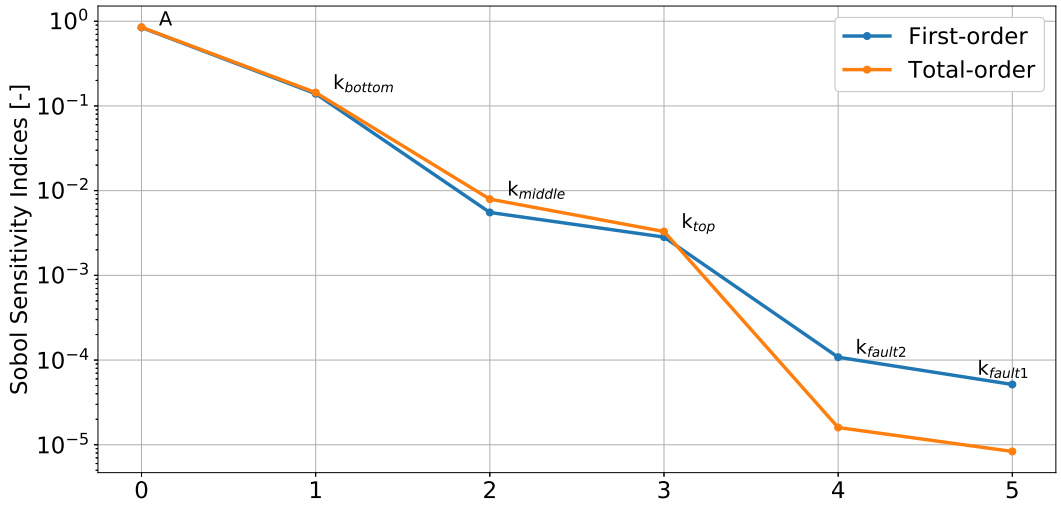


FIGURE 7.10: Global Sensitivity Analysis for the Fault model with an initial pressure pulse.

are higher for the case with an initial pressure pulse than without, also the FE simulation time increases. We observe an earlier pay-off and require now three simulations (instead of five) before the RB method is more efficient than the FE method.

If we compare the convergence of the maximum relative error bound with and without an initial pressure pulse then we observe a similar behavior (Fig. 7.11). The main difference occurs between the third and tenth basis function. There, we see a significant decrease in the error bound for the scenario without an initial pressure pulse and a slight increase for the case with an initial pressure pulse.

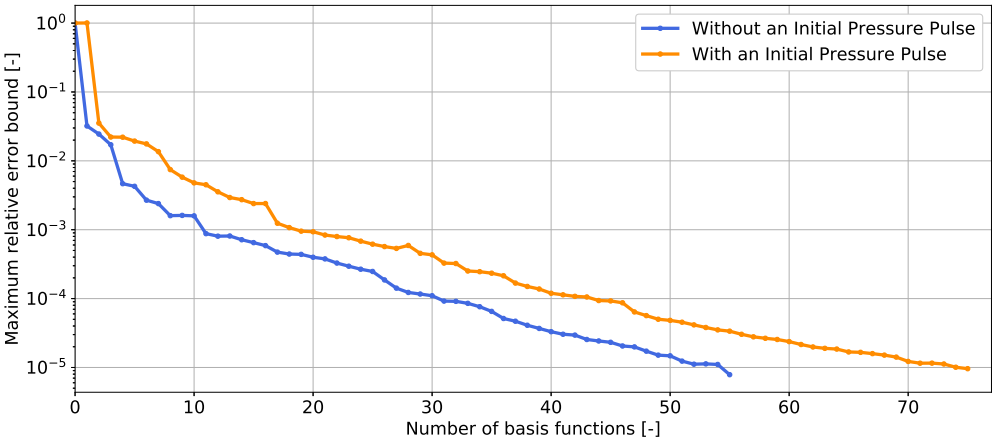
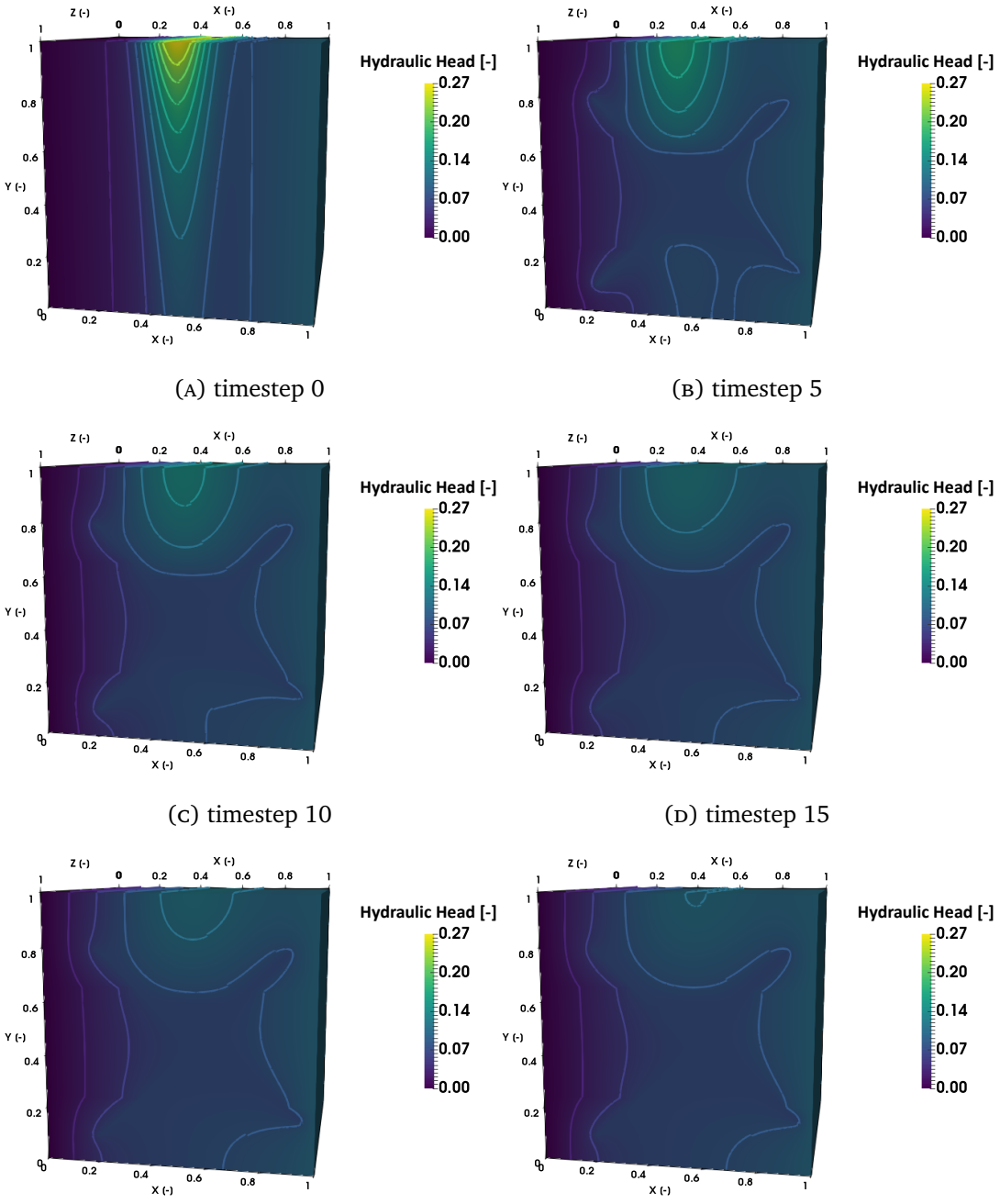


FIGURE 7.11: Convergence rate of the construction of the RB space for the Darcy flow problem using the Fault model with 1,040,325 degrees of freedom and a relative error tolerance of 10^{-5} for the scenario with and without an initial pressure pulse.

Considering the forward model (Fig. 7.12 and 7.13), we see that it follows our expectations. At the initial time step, we observe the pressure pulse, which leads to a significant increase in the hydraulic head. This pressure pulse diffuses over the simulation time since no additional pressure enters the groundwater.

For all time steps, we stay below the pre-defined error tolerance of 10^{-5} (see Fig. 7.14 and 7.15). As for the problem without an initial pressure pulse, no pattern in the error distribution is observable over time.

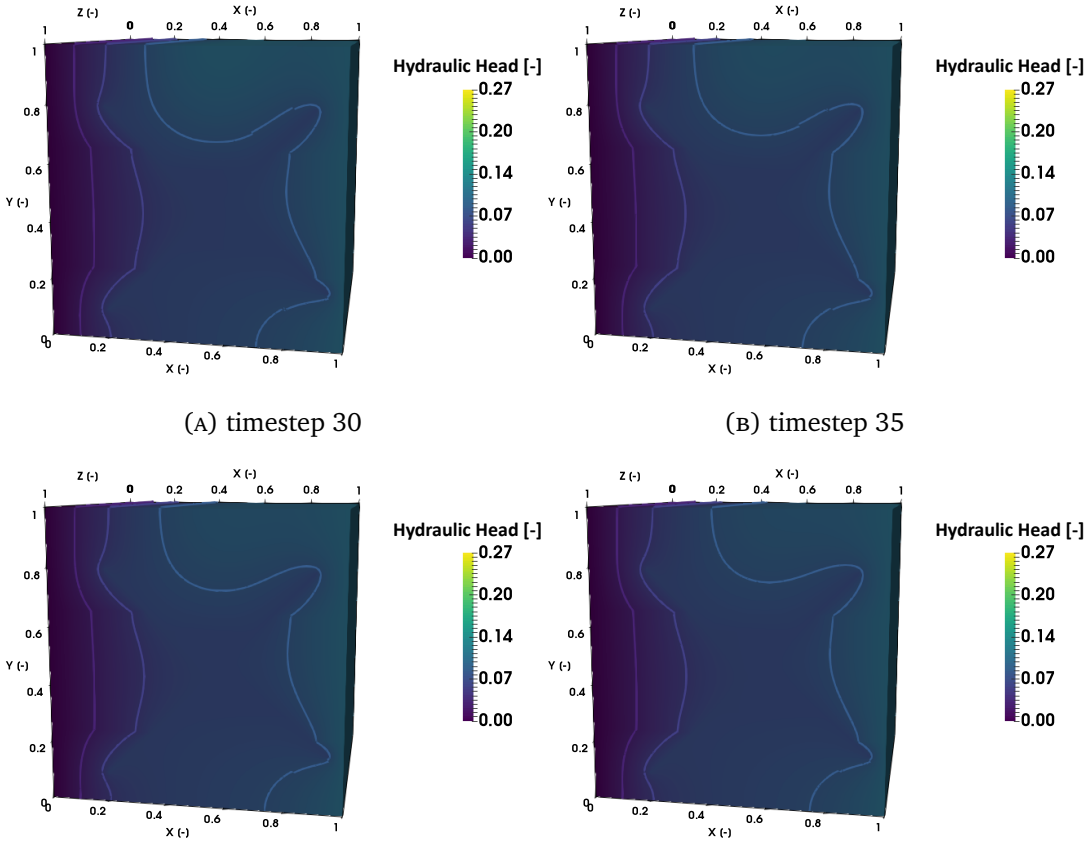


(A) timestep 0 (B) timestep 5

(C) timestep 10 (D) timestep 15

(E) timestep 20 (F) timestep 25

FIGURE 7.12: Hydraulic head distribution for the Fault model with an initial pressure pulse at every fifth time step. Note that both the mesh dimensions and the hydraulic head have been normalized for stability reasons, and are hence dimensionless.



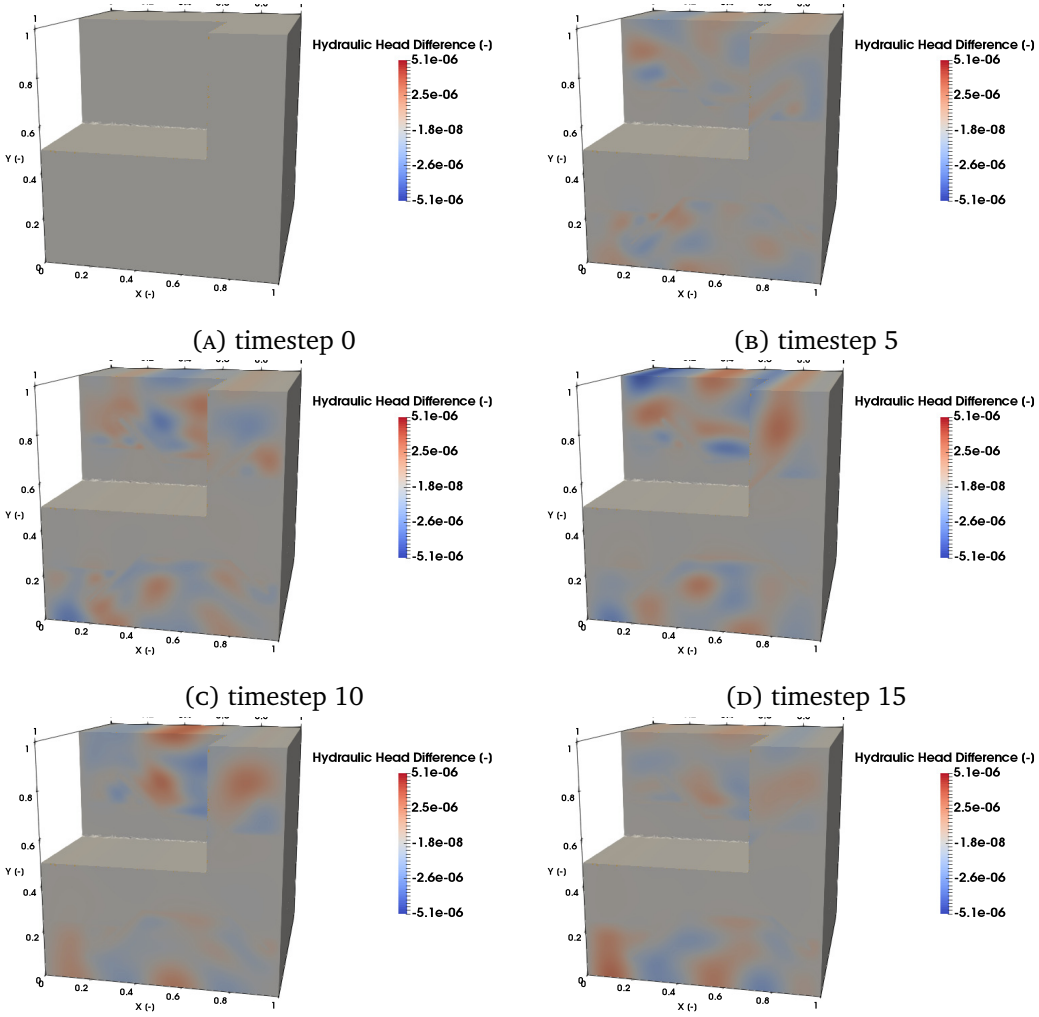
(A) timestep 30

(B) timestep 35

(C) timestep 40

(D) timestep 45

FIGURE 7.13: Continuation: Hydraulic head distribution for the Fault model with an initial pressure pulse at every fifth time step. Note that both the mesh dimensions and the hydraulic head have been normalized for stability reasons, and are hence dimensionless.

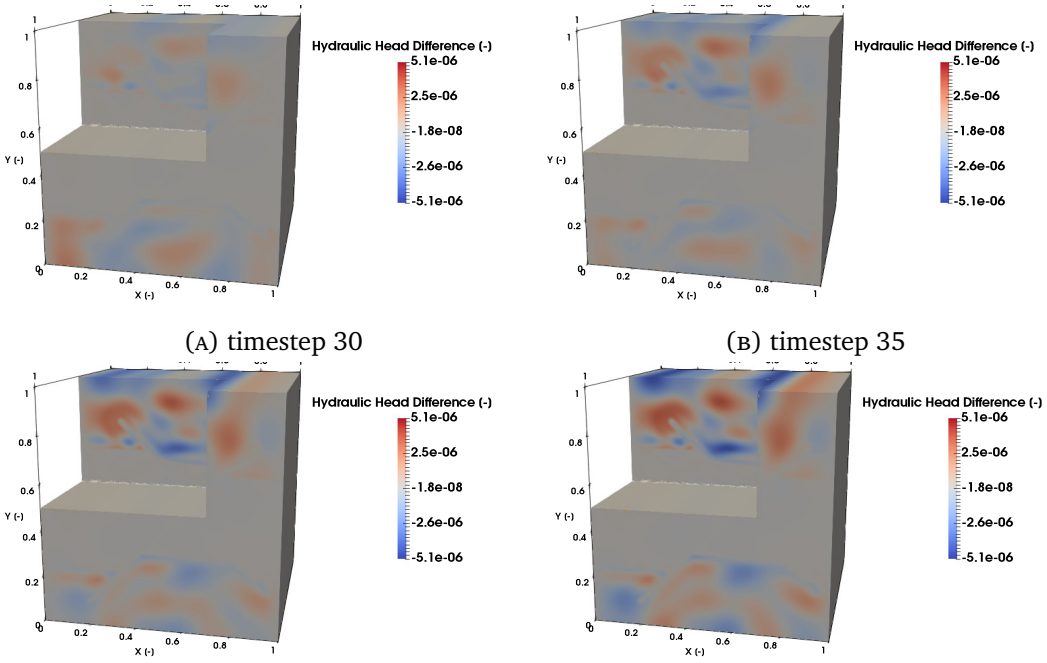


(A) timestep 0 (B) timestep 5

(C) timestep 10 (D) timestep 15

(E) timestep 20 (F) timestep 25

FIGURE 7.14: Difference plot of the hydraulic head distribution between the full finite element and the reduced basis solution for the Fault model with an initial pressure pulse at every fifth time step.



(A) timestep 30

(B) timestep 35

(C) timestep 40

(D) timestep 45

FIGURE 7.15: Continuation: Difference plot of the hydraulic head distribution between the full finite element and the reduced basis solution for the Fault model with an initial pressure pulse at every fifth time step.

7.4.3 Softwate Coupling

In this section, we want to demonstrate the benefits of the direct software coupling, available for the RB method, to libraries offering various tools for parameter estimation, such as the Dakota framework. In this work, we consider both the FE and RB method. For the RB method, we have direct and loose software couplings available, whereas for the FE method only a loose coupling is available. In this chapter, we want to compare the different software coupling options in terms of their efficiency. To be able to perform this comparison, we use the smallest mesh for the Fault model with 4,600 degrees of freedom. The larger meshes would make a comparison impossible due to their large compute time for the FE method. This section is purely meant to show how the RB method can be most efficiently used inside an inverse framework. Therefore, we use the grid search as our outer loop process. The grid search is the simplest many-query application which allows us to focus on the performance evaluation. We do not provide a geophysical interpretation of the results in this section. Instead, we focus solely on the performance aspect.

Figure 7.16 compares the run times of the various coupling options for a grid search that evaluates two, three, and four values per parameter. The Fault model has originally five parameters. However, by assuming the same permeability for the top and bottom layer and both faults, we reduce the parameter dimension from five to three.

For the FE method, we have only the loose coupling available, and with 49 min to 2.6 h it is the costliest option. For the RB method, we have three different coupling variants available: two loose couplings and one direct coupling. In order to retrieve point values, we have two options for the reduced model: either we retrieve the values from the full model or the reduced model. If we want to retrieve the values from the full model, we currently have to use the post-processing functionality of MOOSE. This forces us to use the loose coupling, resulting in an execution time of one to seven minutes. For the retrieval of the point values on the reduced model, we need to define an output of interest. This enables us to use besides the loose coupling also the direct coupling. The gird search using the loose coupling and the output of interest requires 20 s to 2 min, and the direct coupling with the output of interest 0.9 s to 1.7 s.

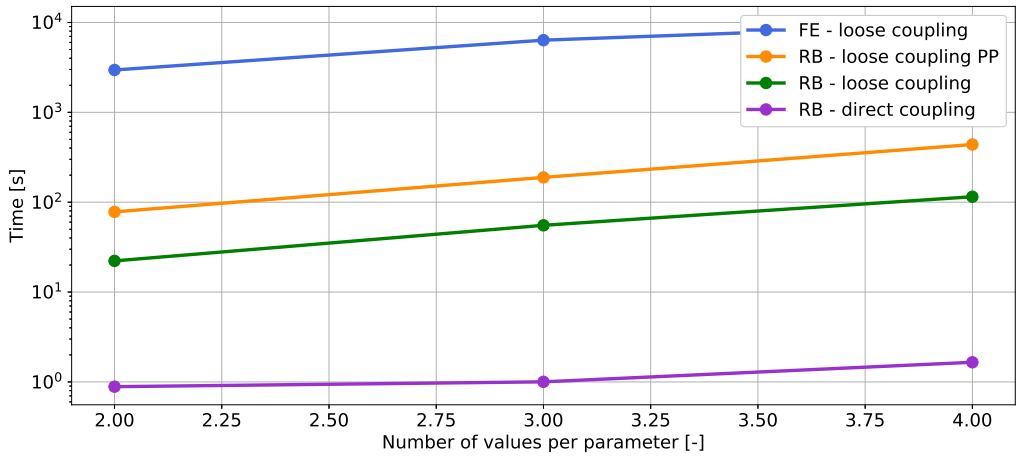


FIGURE 7.16: Comparison of the runtimes for the various coupling methods available for DwarfElephant and Dakota. The loose coupling using the FE method is denoted in blue, the loose coupling using the RB method and the postprocessors in orange, the loose coupling using the RB method and the output of interest in green, and the direct coupling using the RB method in purple.

Figure 7.17 shows that we obtain for the loose coupling using the RB method and the post-processing functionality a speed-up of 21 to 37. For the loose coupling using the output of interest the speed-up increases to 82 to 133, and for the direct coupling it increases even further to 3,342 to 6,334. In Section 7.5.3, we are going to discuss

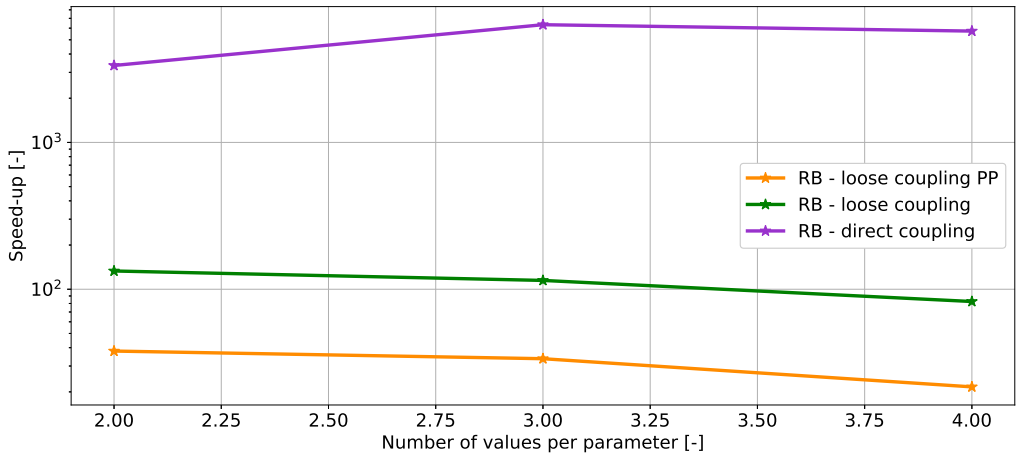


FIGURE 7.17: Comparison of the speed-ups for the various coupling methods available for DwarfElephant and Dakota. Shown are the speed-ups of the loose coupling using the RB method and the postprocessors (orange), the loose coupling using the RB method and the output of interest (green), and the direct coupling using the RB method (purple).

for which scenario which coupling method is best.

7.5 Discussion

As in the Result Section, we first discuss the results of the forward model. This is followed by a discussion of the parameter-dependent initial conditions and the discussion of the various coupling options.

7.5.1 Darcy Flow Problem

We want to compare the RB performance for elliptic and parabolic geophysical problems. Therefore, we performed the Darcy flow problem on the same meshes of the Fault model as the geothermal problem. If we compare the speed-ups of the elliptic and parabolic PDEs then we see an increase in speed-up of a factor of 388 to 1,0560. The reason for this increase is that the RB method reduced both the spatial and temporal degrees of freedom. Hence, we get in addition to the speed-up in “space” also the speed-up in “time”.

We evaluate the computational costs of the Darcy flow (parabolic) and the geothermal problem (elliptic) in the following. Comparing the offline times to the execution time of a single FE solution, we observe smaller difference for the Darcy than for the geothermal problem.

During the offline stage, we have to solve the full problem a couple of times to generate our reduced basis. However, we can reuse many objects as, for instance, the mesh, the preconditioner, the matrix, and vector parameter-independent parts. This re-usage saves a lot of compute time. Consequently, the offline stage, of the Darcy flow

problem requires for most of the Fault model meshes less compute time than two FE solves.

This also leads to the much earlier pay-off of the RB method. For this specific example, we need up to five simulations for the RB method to be more efficient than the FE method. Keep in mind that both the FE and the RB simulations for the Fault model with 1,040,325 and 1,969,077 degrees of freedom were executed using 48 cores instead of a single core. Therefore, the run times of the FE simulations of both the Fault model with 149,490 and 1,040,325 degrees of freedom are nearly identical. The models using more cores have a slightly later pay-off. The different parallel scaling behaviors of the FE and RB model might cause this.

The size of the reduced basis depends on the underlying parameter complexity and not on the spatial and temporal degrees of freedom. That is again the reason why we observe increasing speed-ups with increasing degrees of freedom. That the parameter complexity is not significantly changing from mesh to mesh can also be seen by the sizes of the reduced bases for the various meshes. This size is only slightly increasing from 51 to 57.

For one case, the offline stage is even faster than a single FE solve. This seems theoretically impossible and would be impossible if both simulations could be executed under precisely the same conditions. However, both simulations have been computed on the RWTH Cluster at comparable times. Due to different working loads on the cluster, the computation time can significantly differ. Therefore, the execution of the offline stage can be faster than a single FE solve if the working load during the FE solve is higher on the cluster.

With the reduced model, we reach an accuracy of 10^{-5}. Similar to the geothermal problem, this accuracy is adaptable to the measurement error. Again, we can construct approximations that are more accurate than our measurements. Hence, also for the Darcy flow problem, we are able to generate reduced models that are as accurate as our full models in terms of model predictions. The errors that we observe are the highest in the top and bottom layer of the model and the lowest in the middle layer. This is not surprising since we have two orders of magnitude higher permeabilities in the top and bottom of the model. We can also see that the approximation in time works well since we obtain similar errors over time.

If we compare the online times of the Darcy flow and the geothermal problem for the Fault model, then we observe that those for the Darcy flow problem stay constant. The ones for the geothermal problem increase slightly with increasing degrees of freedom. We execute the online stage of the geothermal problem in MOOSE and the one of the Darcy flow problem with the Python version of the online stage. This is the reason for the discrepancy in the online time behavior. Due to the current setup in MOOSE, it is not possible to execute the online stage alone yielding this scaling behavior.

7.5.2 Parameter-Dependent Initial Conditions

The amount of pressure of the pressure pulse and permeability of the lowest shale layer dominantly influence the change in the hydraulic head with respect to the base case (without an initial pressure pulse). Both parameters have nearly exclusively first-order contributions. This means the parameters themselves influence the model response. We do not observe significant parameter correlations. That the amount of initial pressure is the driving parameter in the model matches our expectations. Interesting is that only the permeability of the bottom layer has an influence. In order to explain this, we have to look at the allowed variation ranges. The ranges for the permeabilities comprise one order of magnitude. Hence, for all parameter combinations, the faults

have the highest permeability and the shale layers the lowest. Keep in mind that we have two Dirichlet boundary conditions, meaning the ratio of the permeabilities is the crucial factor. Due to the chosen variation range, the permeabilities of the faults do not influence the results. This will likely change if we have a fault with a low permeability.

We acquire a larger reduced basis for the scenario with an initial pressure pulse than without, because of the higher underlying parameter complexity. Remember that the size of the reduced basis, assuming a fine enough spatial and temporal discretization, dependents solely on the parameter complexity. We have one additional parameter (the initial amount of pressure) for this model yielding the increase in basis functions from 55 to 75. This also explains the slight increase in the online execution time, which is dependent on the dimension of the reduced basis. Also, the FE simulation time increases; however, the increase in percent is smaller than the one of the online time yielding the smaller speed-up. For the calculation of the pay-off, we used 48 cores again and obtained an earlier pay-off which is contradictory to the smaller speed-up. The different workloads of the cluster cause this discrepancy.

7.5.3 Software Coupling

Before we start to discuss the application fields of the various software coupling method, we want to discuss the accuracy of the FE and RB method. We are using the Fault model with 4,600 degrees of freedom. This model was already discussed in detail in the previous Section 7.5.1; hence, we do not discuss it in detail here. Just note that we were able to construct a reduced model that has a higher accuracy than our measurements. Consequently, for our specific example, the full and reduced model are equivalent.

Comparing only the run times and speed-ups, we directly see that the direct coupling using the RB method is the fastest. This has several reasons:

1. For the direct coupling, we extract the point values from the reduced model, which is way more efficient than extracting it from the full model.

2. A direct coupling has, in contrast to a loose coupling, no file reading and writing operations. This is, for our example, not very important because we extract only three-point values. However, the more data we have to transfer via files, the more influence it has.

3. We are using the reduced model that is (as previously shown) several orders of magnitude faster than the full model.

The major difference in the execution times between direct and loose software coupling is that for the loose coupling, we have to start an entire MOOSE application, including unnecessary processes as mesh reading.

If we compare the execution time of the direct coupling to the execution time of the loose coupling using the FE method, we observe that the speed-up is 2.6 times higher than we would expect. Furthermore, the speed-up is increasing for more forward simulations. This discrepancy is caused by using a loose coupling for the FE method. The loose coupling has the disadvantage that for each forward, solve we need to start a new MOOSE simulation. This also, means that, for instance, the mesh is read in for every forward simulation, an unnecessary step. The focus, of this thesis, is, however, the reduced basis method. Because of this focus, and the time-consuming nature of constructing a direct coupling between a MOOSE application and Dakota for the FE method, we did not further improve the coupling for the FE method. Also, note that we need to construct this direct coupling for every library we want to use. The direct

coupling using the RB method, on the other hand, can be, with minor adaptations, used for any library that provides a Python interface.

We already know that the direct coupling using the RB method leads to the shortest execution time. So, why do we offer additional coupling methods for the RB method? The loose coupling using the RB method and extracting the point values on the reduced model is only available for debugging purposes. In actual applications, it does not provide any benefit. It produces the same outcome as the direct coupling with a much higher execution time (factor of 30 to 70), due to the repetitive mesh reading.

The loose coupling using the RB method and the post-processing functionality has, however, a practical usage. Extracting the point values on the reduced model requires the knowledge of the position of every measurement point before we construct the reduced model. During, for instance, geothermal explorations it can happen that later on, additional measurement points are defined. Due to the time-consuming construction of the reduced basis, it is, in that specific case, more efficient to extract the desired values from the full model – obtained by projecting the reduced space back to the full space. This is the major advantage of the reduced model in comparison to other surrogate models. Currently, only a loose coupling for the post-processing is available, making this option less efficient. Theoretically, we could obtain execution times of about a few seconds. In the future, it is planned to extend the available loose coupling to a direct coupling for better efficiency.

We would expect that the speed-up between the loose RB coupling and the FE loose coupling is independent of the number of forward simulations. This is not observable since we cannot ensure the same conditions on the working station and the same software performance. The mismatches to our expectations are, hence, caused by different workload conditions.

For the speed-up between the direct RB coupling and the loose FE coupling, we expect an increase in speed-up with larger numbers of forward simulations. That is the reason because direct coupling avoids unnecessary repetitive processes, such as mesh reading. For two and three values per parameter, we observe this behavior but not for four values per parameter. Again, the difference in the workload causes this discrepancy. For revealing the dependency between the number of forward simulations and the various speed-ups, thorough tests are required. This is computationally expensive due to the large FE compute times. It has not been performed in this thesis because the main focus of this section is to present the various coupling methods and briefly introduce their main advantages and disadvantages.

To conclude, the direct coupling of the RB method enables an efficient coupling to a broad range of external software packages and libraries by relying on the widely used computing language Python. This coupling allows to perform efficiently outer loop processes such as global sensitivity analyses for both time-dependent and -independent problems.

Chapter 8

Global Sensitivity Analysis for the Investigation of Fault Zone Behavior

After introducing the concept of the RB method for transient geophysical applications such as groundwater flow problems, we focus in this chapter on the real-case applications in this field. Konrad et al. (2019) investigate the visibility of fault zones in pumping tests. The base of their study is a local sensitivity analysis. Here, we show how we extend this local sensitivity analysis to a global one. This chapter is the preparation for the publication Konrad et al. (2020). This is the reason, why we focus in this chapter on the aspects of the RB method and global sensitivity analysis for transient real-case geophysical applications. The chapter is not meant to provide detailed insights into the specific application of the Malm Fault Zone. These details will follow in the publication Konrad et al. (2020).

8.1 Malm Fault Zone

The model of the Malm Fault Zone (Fig. 8.1) is located in southern Germany in the region around Munich. This region is of high interest for geothermal exploration, and long term pumping tests accompany the drilling of the geothermal wells. The pumping test results for this region are in disagreement with our expectations. The area is characterized by fault zones that should result in a bilinear flow behavior at the wells. However, the fault zones are invisible at most of these wells.

To explain the mismatch, Konrad et al. (2019) have constructed a simplified, but still realistic model of the region. This model contains one fault zone, one well, and the surrounding rock matrix. It has a spatial extent of 30 km in the x- and y-direction, and of 534 m in the z-direction. The model is discretized using tetrahedrons, resulting in an unstructured mesh with 154,199 degrees of freedom. As the physical model, we consider a transient Darcy flow problem (Eq. 7.7 and 7.8). The model has no boundary conditions, and an initial condition of $p_0 - \rho g(z - GW)$, where GW is the groundwater level.

Table 8.1 summarizes all rock and fluid properties that are varied for the later analyses. Additionally, we fix the permeability of the well to 10^{-7} m^2 and the specific storage to $2{\cdot}10^{-15}$ Pa^{-1}. The simulations are performed using a hypre pre-conditioner, and a linear and nonlinear solver tolerance of 10^{-8}. For the time stepping, we follow the insights gathered by Konrad et al. (2019) and use an initial time step size of 1 s that grows after each timestep with a factor of 1.1. The growth rate was reduced from 1.6 to 1.1 since the RB method yields a much faster forward simulation. The authors of Konrad et al. (2019) plan to further analyze the influence of the size of the time steps

on the model response. That is the reason why we chose a much smaller growth rate than initially proposed in Konrad et al. (2019). The sink term $Q(t)$ is time-dependent and implemented as a Dirac function. The time dependency is a box function, meaning that between a pre-defined start and end time, we have a constant extraction rate, and outside this time interval no extraction. For more detailed information regarding the model and the geological region, we refer to Konrad et al. (2019). The original study of Konrad et al. (2019) has been executed with the MOOSE application GOLEM (Cacace and Jacquey, 2017). In contrast to Konrad et al. (2019), we consider dimensionless parameters.

For the construction of the reduced model, we use a relative training tolerance of 10^{-3} (adjusted to the typical measurement error), a maximum number of basis functions of 200 and 1000 training samples. The affine decomposition of the bilinear form a and m is the same as in Eq. 2.43 and Eq. 2.45, respectively. The linear form f has the following decomposition:

$$f(v; pr) = pr \int_{\Gamma} v \; Q(t) \; d\Gamma, \qquad \forall v \in X, \; \forall pr \in D, \tag{8.1}$$

where pr is the production rate and $Q(t)$ the time-dependent sink term.

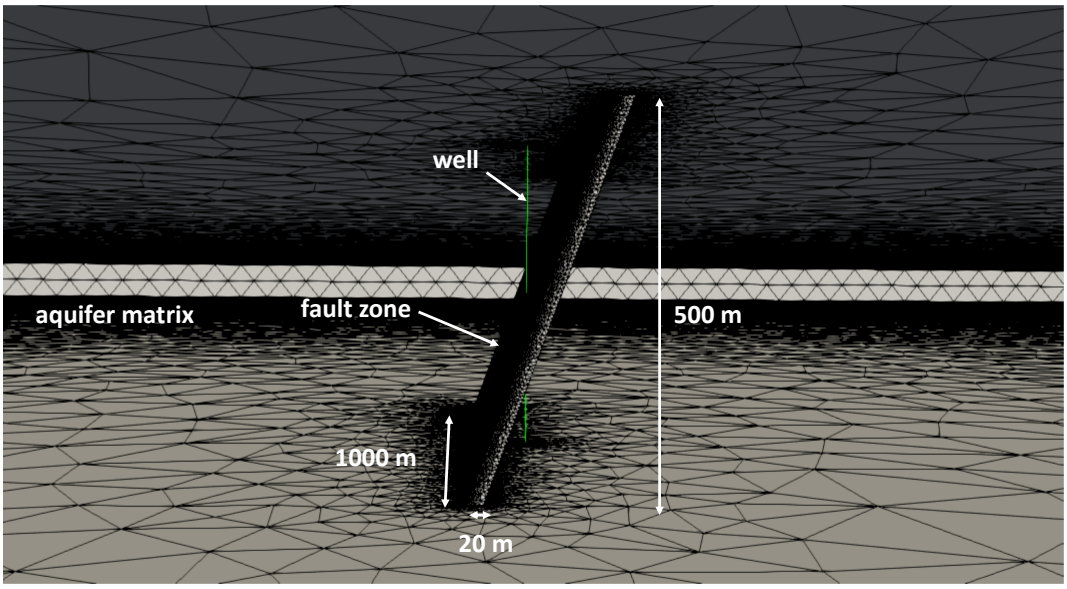


FIGURE 8.1: Image of the Malm Fault Zone and the unstructured mesh constructed by Konrad et al. (2019). We use the model with a fault zone width of 20 m.

8.2 Results

The Malm Fault Zone model has been studied previously by Konrad et al. (2019). For their analysis, a sensitivity study is crucial. So, far only a local sensitivity study, namely a grid search, has been performed. However, this has the disadvantage that in the tested setup, they have assumed that the model parameters are independent of each other, an assumption that is not true. Furthermore, the analysis required over 30,000 function evaluations leading to an execution time on a high-performance infrastructure of over two months. Considering that they could test only seven values per

TABLE 8.1: Hydraulic properties that are varied for further analyses for the Malm Fault Zone model. The parameters are taken from Konrad et al. (2019).

unit	training parameter	property	lower bound	upper bound
matrix	0	permeability	10^{-17} m^2	10^{-12} m^2
fault zone	1	permeability	10^{-14} m^2	10^{-9} m^2
all	2	fluid viscosity	10^{-4} Pa s	$30 \cdot 10^{-4}$ Pa s
matrix	3	specific storage	$20 \cdot 10^{-12}$ Pa^{-1}	$16 \cdot 10^{-10}$ Pa^{-1}
fault zone	4	specific storage	$20 \cdot 10^{-12}$ Pa^{-1}	$16 \cdot 10^{-10}$ Pa^{-1}
-	5	production rate	20 l s^{-1}	90 l s^{-1}

parameter, it is likely that the parameter space is not adequately explored. However, due to the large execution time, a more detailed exploration is not possible.

To overcome both the shortcomes of the local sensitivity study and the huge amount of computing resources required, we use the RB as a surrogate model for the FE method. Therefore, we follow the requirements for the parameter study described in Konrad et al. (2019). Hence, we allow in total to vary six model parameters. The model parameters and their variation range are listed in Tab. 8.1. In accordance with the measurement accuracy in the field, we define a relative error tolerance of 10^{-3} for the Greedy algorithm. In contrast to Konrad et al. (2019), we use a global sensitivity analysis to learn about possible correlations between the model parameters.

Note that the production rate is time-dependent and that we have a varying time step size. The original libMesh distribution of the RB method does not support both features. Therefore, we extended the functionality of the package within the Dwarf-Elephant application.

We investigate the sensitivity of the model parameters to the total model response over all time steps. For this quantity of interest, only three parameters have a notable influence on the model response: the matrix permeability, the production rate, and the dynamic fluid viscosity (Fig. 8.2). The major contributions of all three parameters are first-order contributions, and their higher-order contributions are significantly smaller. The strongest correlation is between the matrix permeability and the production rate. Note that the correlations involving the matrix permeability are at least one order of magnitude higher than the remaining parameter correlations. The remaining three parameters (specific storage of the matrix and the fault zone, and permeability of the fault zone) have nearly no influence on the model response. Their major contributions to the model response are higher-order contributions.

In order to reach the pre-defined error tolerance, we require 112 basis functions (Fig. 8.3). This leads to an online execution time of 0.2 s. The FE method yields a compute time of 68 min. Hence, we obtain a speedup of $2{\cdot}10^4$. Note that if we repeat the analysis performed by Konrad et al. (2019), we could reduce the computation time from around two months to less than two hours. The global sensitivity analysis presented here required 140,000 simulations resulting in a computation time of 7,5 h. Note both the offline stage and the FE simulations have been executed on a MacBook Pro, using four Intel Core i7 processors (2.5 GHz, 16 GB of RAM).

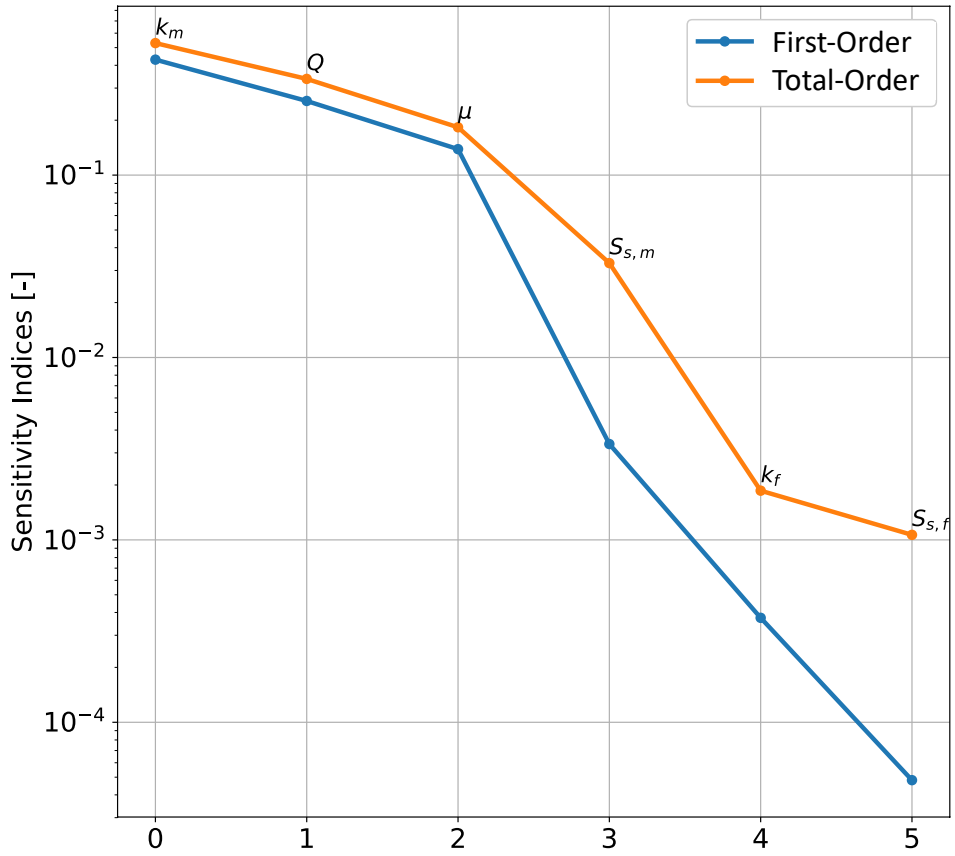


FIGURE 8.2: Global sensitivity analysis of the Malm Fault Zone Model constructed after the local sensitivity analysis performed in Konrad et al. (2019).

8.3 Discussion

The sensitivity analysis of the Malm Fault Zone model showed that the influence of the parameters themselves dominates the model response. However, the parameter correlations are not negligible, emphasizing the need for global sensitivity analyses (Fig. 8.2). A local sensitivity analysis would not have returned the parameter correlation, which would have resulted in an overestimation of all model parameters.

We also observed that the driving factor of the model is the matrix permeability. This is surprising since we know from hydraulic studies that fault zones have a major influence on the hydraulic behavior of a model. However, we allowed variations in the permeability of the fault zone between 10^{-14} and 10^{-9} for a fault zone width of 20 m. For higher permeabilities and a large fault zone, the influence of the permeability of the fault zone would increase significantly. Nonetheless, for this model setup, the model is matrix driven.

The second highest influence is arising from the production rate. This is only natural since the production rate is the sink term in the model that initializes a change in the pressure distribution. Therefore, it is logical that the strongest correlation is between the matrix permeability and the production rate. The permeability divided by the dynamic fluid viscosity determines how fast or slow the pressure is distributed

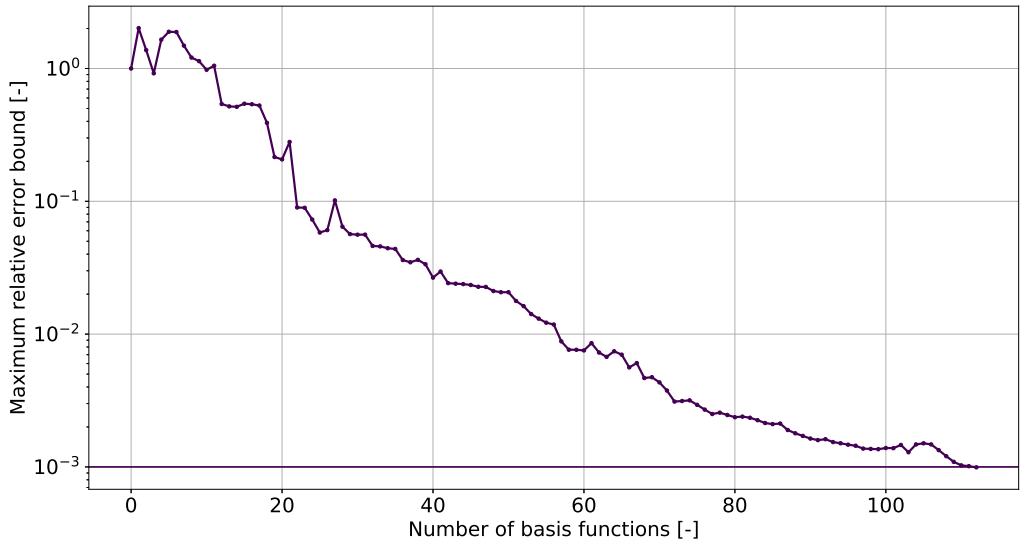


FIGURE 8.3: Convergence of the maximum relative error bound for the Malm Fault Zone model. The error tolerance is set to 10^{-3}.

in the model, which explains the strong influence of the fluid viscosity on the model response.

The convergence of the maximum relative error bound shows a stagnation of the error bound for the first 10 basis functions. The jumps at later stages is a sign that the parameter complexity is very high. We have six parameters that we vary over a variation range of several orders of magnitude. From the sensitivity analysis, we know that only three parameters are influencing the model. This is an indication that reducing the number of parameters would not significantly improve the convergence behavior. We observe, in some cases, an increasing error after the addition of a new basis function. This seems to be counterintuitive. How can adding information lead to worse results? The Greedy algorithm only locally chooses the optimal basis function. However, what is optimal for one part of the model domain might not be optimal for another. Hence, we might decrease the approximation quality in another region. Consequently, we obtain an approximation with a higher error than before. However, one can reduce this issue by narrowing down the parameter ranges that one needs to investigate.

Although the RB method can reduce the model, with the desired accuracy, we see challenges for board parameter ranges. This model demonstrates that we have to think about combining the method with other methods. These methods are domain decomposition (Beattie et al., 2006; Liao and Willcox, 2015; Toselli and Widlund, 2006) and hp refinement (Eftang et al., 2011). A more detailed description of this aspect follows in Section 11.

The reduced basis is probably larger than it needs to be. Despite that, we are still able to reduce the compute time by several orders of magnitude, making the method well suited for this case study. Nonetheless, keep in mind that if we either increase the accuracy of the reduction or the variation ranges, we will most likely reach its limits.

We presented that the RB method is efficiently applicable to real-case groundwater applications. All transient groundwater applications face the challenge of determining

the initial condition of the pressure state. Therefore, it would be interesting to investigate for the model of the Malm Fault Zone how sensitive the model parameters are to the initial conditions. Furthermore, we would like to investigate how data assimilation can help to determine the initial condition for this model and how we can use the RB method to make the process more efficient.

Chapter 9

Reduced Basis Method for Sequential Data Assimilation

In the previous Chapters 7 and 8, we discussed the benefits of the RB method for the forward problem of groundwater applications, for both synthetic and real-case applications. Now, we want to shift the focus to inverse methods. In transient groundwater flow problems, one typical problem is to determine the initial condition of a model from the given measurement data. To overcome this issue, we apply data assimilation techniques. In this chapter, we illustrate how one can use the RB method most efficiently for data assimilation methods, such as the Ensemble Kalman Filter. Therefore, we use a simple benchmark example since this chapter aims at introducing the various concepts and their respective challenges.

9.1 Introduction

In the previous sections, we introduced various parameter estimation techniques. Now, we shift the focus from the model parameters to the model variables. The climate and groundwater community widely use data assimilation to commonly determine the initial conditions (i.e., Barker et al., 2004; Dennis et al., 2015; Hendricks Franssen and Kinzelbach, 2008; Huang et al., 2009; Li et al., 2012; Lorenc et al., 2000; Panzeri et al., 2014; Rawlins et al., 2007; Rodell et al., 2007; Zaitchik et al., 2008; Zhou et al., 2011). In data assimilation, we distinguish between variational and sequential data assimilation. Variational data assimilation, as the 4D-Var method, is a continuous approach, where we consider the entire measurement time frame. Via the minimization of a cost function, we obtain the state estimate, similar to the previously described optimization methods (Courtier, 1997; Le Dimet and Talagrand, 1986; Lorenc, 1981, 1986; Navon, 2009; Sasaki, 1970). Applications of variational data assimilation are currently restricted to climate (Barker et al., 2004; Dennis et al., 2015; Huang et al., 2009; Lorenc et al., 2000; Rawlins et al., 2007), and oceanographic applications (Bennett, 1992). Sequential data assimilation, such as the Ensemble Kalman Filter, is a discontinuous approach using only small portions of the measurement time frame. At each measurement, one performs a statistical correction of the state variable (Evensen, 1994).

Numerical models are used to forecast the state of a system, for instance, the hydraulic head distribution in an aquifer. Data assimilation provides a way to combine these forecasts with measurements. In contrast to the other inverse processes presented in this thesis, data assimilation aims at determining the state and not the parameters. Data assimilation is often used to estimate the initial conditions or the boundary conditions of a system. In this thesis, we focus on the sequential data assimilation, and there on the Ensemble Kalman Filter (ENKF). We use it to estimate

the initial conditions for our groundwater applications. To reduce the computational cost of the method, we propose to use the RB as a surrogate for the FE solutions. Pagani et al. (2017) present how they apply the RB method for the ENKF. However, they focus on the mathematical aspects of this approach and do not discuss the problems arising from typical geophysical applications. Additionally, their case studies consider the aspect of parameter estimation only. In Geosciences data assimilation is, however, typically used for state estimation problems or a combination of state and parameter estimation and not only for parameter estimation (Hendricks Franssen and Kinzelbach, 2008).

9.2 Data Assimilation – Ensemble Kalman Filter

In the following, we briefly introduce the concepts of the ENKF. For a more detailed description of the filter, we refer to Evensen (1994). The implementation of the ENKF follows Evensen (2003).

The ENKF was first introduced by Evensen (1994) and later clarified by Burgers et al. (1998). It is a modification of the Kalman Filter. By using a Monte Carlo approximation, we obtain a suboptimal estimator. That avoids evolving the covariance matrix of the probability distribution of the state, which would be a very costly operation. The ENKF represents the distribution instead by a certain number of realizations, which one refers to as ensembles. The filter consists of three stages (Evensen, 1994, 2003; Gillijns et al., 2006).

1. The first stage deals with the representation of the error statistics in the forecast step. The Kalman Filter requires the true state. This state is, however, in reality not known. Therefore, the Ensemble Kalman Filter approximates the true state with the mean overall ensembles $\overline{\Psi}$, assuming that for a sufficiently large number of ensembles, we converge to the true state. The error matrix is then constructed around the ensemble mean (Eq. 9.1, and 9.2). Thus, we interpret the ensemble mean as our best estimate of the true state and the ensemble spread as the error to the best estimate (Evensen, 1994, 2003; Gillijns et al., 2006).

$$P_e^f \approx \overline{(\Psi^f - \overline{\Psi^f})(\Psi^f - \overline{\Psi^f})^T}, \tag{9.1}$$

$$P_e^a \approx \overline{(\Psi^a - \overline{\Psi^a})(\Psi^a - \overline{\Psi^a})^T}, \tag{9.2}$$

   where Ψ^f and Ψ^a are the model forecast and analysis and $\overline{\Psi}$ denotes the mean.

2. The second stage is the so-called analysis step. Here, the classical Kalman Filter gain (Eq. 9.3) is applied to analyze and update the state estimate. Note that instead of using the covariance matrix, as for the classical Kalman Filter, we use the approximation of the error covariance. Another difference to the Kalman Filter is that the Ensemble Kalman Filter uses perturbed observations instead of the true observations. This allows considering the uncertainty of the individual measurements (Evensen, 1994, 2003; Gillijns et al., 2006).

$$K_e = P_e^f H^T (H P_e^f H^T + R_e)^{-1}, \qquad \text{with} \quad R_e = \overline{\epsilon \epsilon^T}. \tag{9.3}$$

   Here, H denotes the measurement matrix, mapping the observations to the model, R_e the ensemble covariance matrix of the measurements, and ϵ the errors of the measurements.

3. The third and last stage is the prediction of the error statistics, again in the forecast step. During this stage, the model parameters are sampled from a normal distribution (Evensen, 1994, 2003; Gillijns et al., 2006).

Note that in this thesis, we use only the Ensemble Kalman Filter with a state update. We do not consider the dual update for state and parameter. For more information about the dual update filter, we refer to Hendricks Franssen and Kinzelbach (2008).

9.3 Box Model

For illustrating the benefits and disadvantages of the RB method as a surrogate model for the Ensemble Kalman Filter, we constructed a two-dimensional box model, which we refer to as “Box model". The Box model has a higher permeable obstacle in the middle, as displayed in Fig. 9.1. It has a reference length of 1,000 m, the same as for the Base and Fault model. For the reference permeability we choose 10^{-12} m^2 (i.e., corresponding to the largest permeability in the model). The reference hydraulic head is 300 m. The Box model consists of an outer sandy gravel layer with a nondimensional permeability of 7.0 10^{-2} (7.0 10^{-14} m^2) and an inner marl, sand layer with a permeability of 10^{-12} (10^{-12} m^2). It has a nondimensional extent in the x-direction of 1 (1,000 m), and of 0.5 (500 m) in the y-direction. Here, the inner marl, sand layer is placed 0.375 (375 m) away from the left model boundary – with an extent of 0.25 (250 m) in the x-direction, and 0.175 (175 m) from the lower boundary – with an extent of 0.15 (150 m) in the y-direction. For the generation of the unstructured triangular mesh, we used Gmsh. Using a global mesh size factor of 0.01, we obtain a mesh with 6,794 degrees of freedom and 13,284 elements.

Both on the left and the right side of the model, we apply Dirichlet boundary conditions to fix the hydraulic head on both sides. On the left model boundary, the hydraulic head is equal to 300 m and on the right to 400 m. Hence, we have an increase of 100 m in hydraulic head over a length of 1000 m. This corresponds to a hydraulic gradient of 0.1. Since we consider nondimensional variables only, the nondimensional boundary values are 0 and 0.1 for the left, and right model boundary, respectively.

We use the same preconditioner as for the Fault and Base model (see Tab. 3.3), but a smaller linear and nonlinear solver tolerance of 10^{-10}. For the time step size, we use a value of 0.072 (5 d), and we simulate for 25 days. The allowed variation of the sandy gravel layer ranges from 7.0 10^{-3} to 0.7 and for the marl, sand layer from 0.5 to 1.5. The number of training samples is 200, and the maximum number of basis functions 100, with a relative training tolerance of 10^{-5}.

For the Ensemble Kalman Filter, we define two sets of observations, the first one is close to the higher permeable layer and denoted as nearby observations. The second set is further away and referred to as faraway observations. Both sets are defined in Tab. 9.1 and 9.2, and displayed in Fig. 9.2.

9.4 Results

In the following, we first present the results from the forward problem of the Box model. Afterwards, we focus on the Ensemble Kalman Filter, where we first present the results from the finite element version of the filter, and then the ones from the reduced basis version.

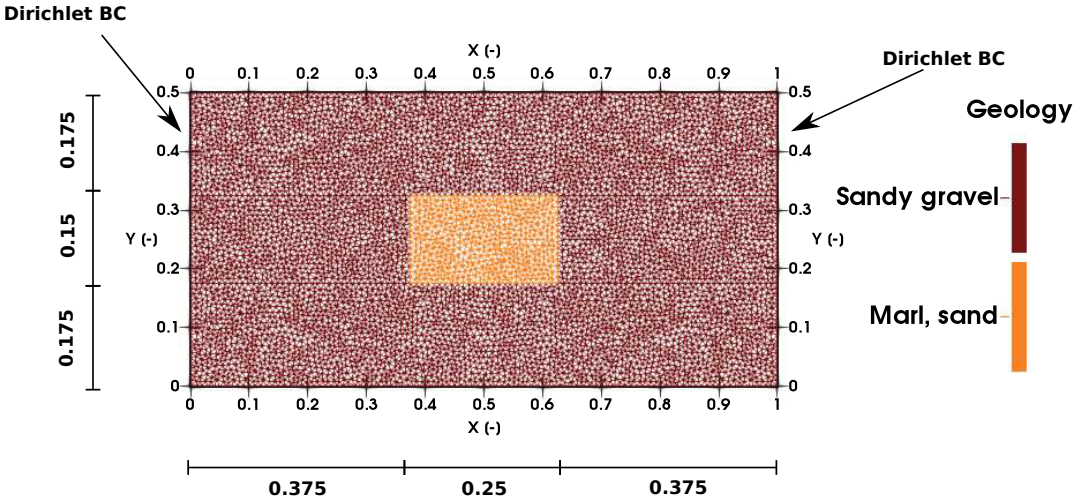


FIGURE 9.1: Mesh representation of the Box model consisting of two geological layers. The model has Dirichlet boundary conditions at the left and right side of the model domain. Note that the mesh is dimensionless for performance issues.

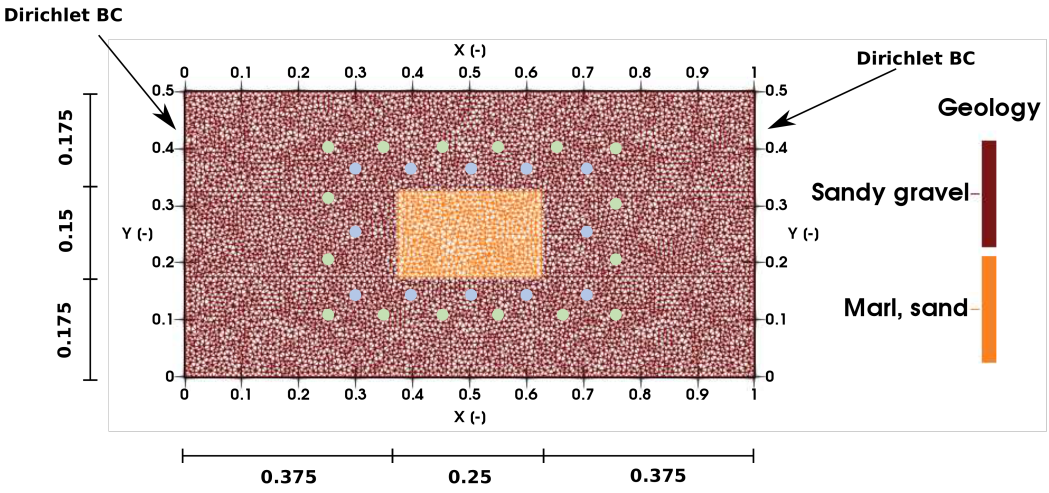


FIGURE 9.2: Distribution of the observation points for the Box model. The first set of observation points are denoted in light blue and the second in light green.

TABLE 9.1: Coordinates of the first set of observation points for the Box model.

Observation Point	Coordinates (x|y)
First Set	
0	(0.3|0.35)
1	(0.4|0.35)
2	(0.5|0.35)
3	(0.6|0.35)
4	(0.7|0.35)
5	(0.7|0.25)
6	(0.7|0.15)
7	(0.6|0.15)
8	(0.5|0.15)
9	(0.4|0.15)
10	(0.3|0.15)
11	(0.3|0.25)

TABLE 9.2: Coordinates of the second set of observation points for the Box model.

Observation Point	Coordinates (x|y)
Second Set	
0	(0.25|0.4)
1	(0.35|0.4)
2	(0.45|0.4)
3	(0.55|0.4)
4	(0.65|0.4)
5	(0.75|0.4)
6	(0.75|0.3)
7	(0.75|0.2)
8	(0.75|0.1)
9	(0.65|0.1)
10	(0.55|0.1)
11	(0.45|0.1)
12	(0.35|0.1)
13	(0.25|0.1)
14	(0.25|0.2)
15	(0.25|0.3)

9.4.1 Forward Model

In the following, we present the results for Darcy flow forward simulation of the Box model that we use later on to generate the observation data needed for the Ensemble Kalman Filter.

The hydraulic head distribution, displayed in Fig. 9.3 is in accordance with our expectations. We start with a hydraulic head gradient between 0 (corresponds to 300 m) and 0.1 (400 m). At timestep 6, which corresponds to time 0.432 (5 d), we observe a perturbation in the hydraulic head distribution around the marl and sand layer. In this area, we observe an increase in hydraulic head. This perturbation slightly grows in the next timestep and remains nearly constant throughout the remaining simulation time.

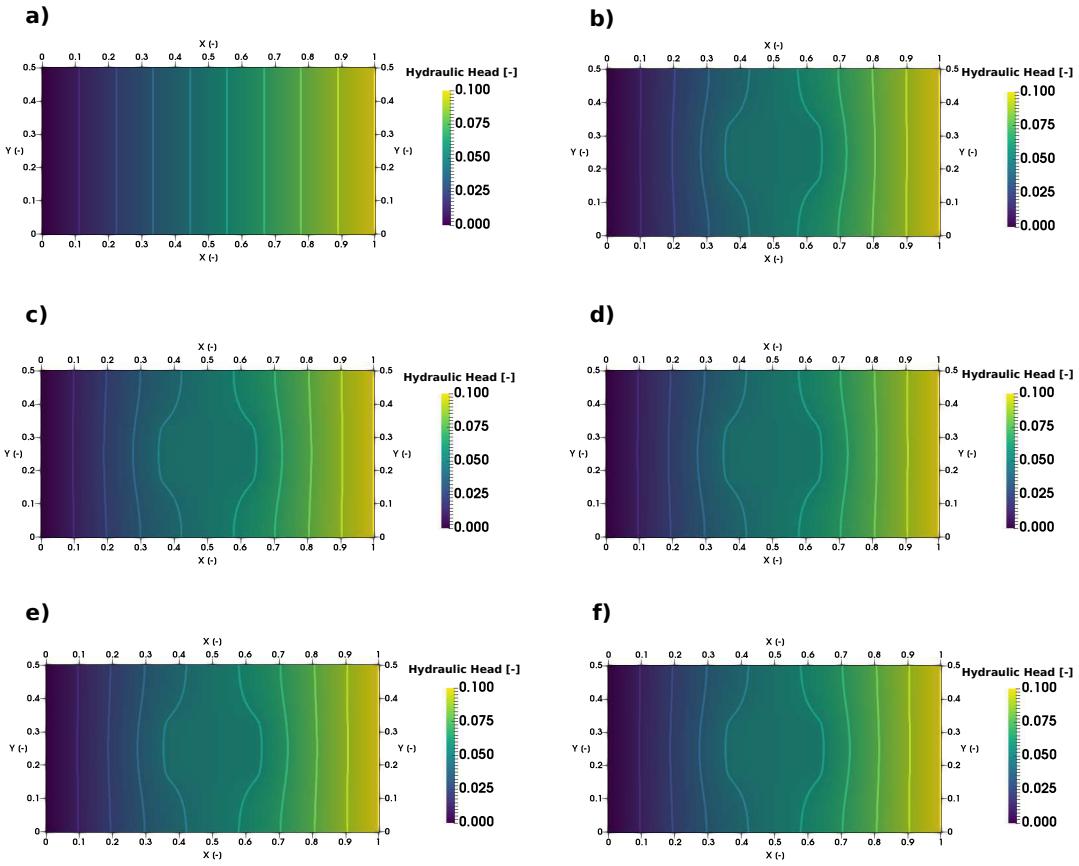


FIGURE 9.3: Hydraulic head distribution at timestep a) 0 b) 6 c) 12 d) 18 e) 24 f) 30. Note that both the mesh and the hydraulic head have been normalized for numerical reasons and are hence dimensionless. The model has a reference hydraulic head of 300 m, and a reference length of 1,000 m.

The FE solution requires 24 s for the execution, whereas the online stage of the RB method requires only 16 ms, yielding a speedup of over 1,600. The offline stage of the RB method requires 29 s generating a reduced basis with 21 basis functions. Already, two simulations are enough for the RB method to be more efficient than the FE method.

The difference between the FE and RB solution (Fig. 9.4), shows that the error is below the desired threshold of 10^{-5} for all timesteps. The largest error is with $-1.565 \cdot 10^{-7}$ nearly two magnitudes smaller than our error tolerance. For the initial

condition and at the Dirichlet boundaries, the error is zero. We observe a slightly increasing error with time, and the largest errors occur close to the lateral boundaries. The pattern of the error distribution in space is nearly symmetrical for all timesteps. The convergence plot of the relative maximum error bound (Fig. 9.5) shows that the given error tolerance is reached not only for the specific parameter combination of the true model but also for all other parameter combinations within the training range.

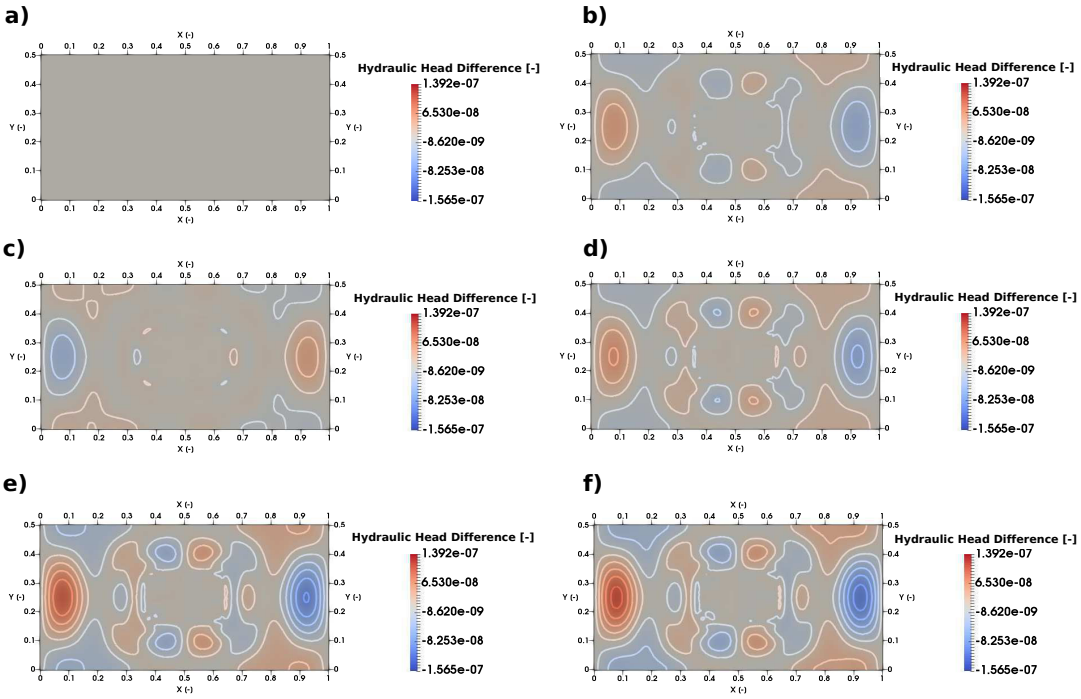


FIGURE 9.4: Difference plot between the FE and RB solution for the hydraulic head distribution at timestep a) 0 b) 6 c) 12 d) 18 e) 24 f) 30. Note that both the mesh and the hydraulic head have been normalized for numerical reasons and are hence dimensionless. The model has a reference hydraulic head of 300 m, and a reference length of 1,000 m.

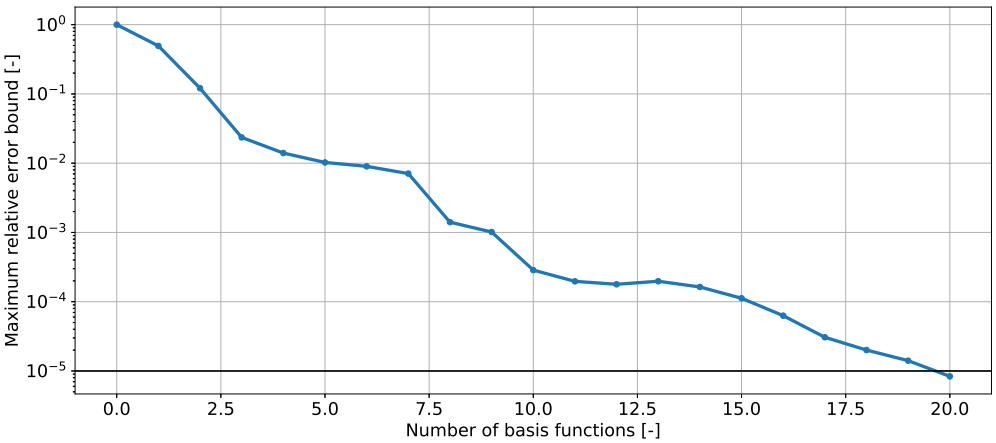


FIGURE 9.5: Convergence rate of the construction of the RB space for the Box model with a relative error tolerance of 10^{-5} and solver tolerances of 10^{-8}.

9.4.2 Ensemble Kalman Filter – Finite Element Version

In the following, we consider various scenarios for the Ensemble Kalman Filter. First, we test whether the implementation of the Ensemble Kalman Filter yields the expected results and how well it recovers the truth initial state. Therefore, we analyze the influence of the position of the observation points and the influence of the boundary conditions. Furthermore, we need to slightly modify the Ensemble Kalman Filter to enable a combination with the RB method. In this chapter, we also present the errors introduced by this modification.

As the reference configuration for the Ensemble Kalman Filter, we use the first set of observation points listed in Tab. 9.1 and shown in Fig. 9.2. We define a measurement error of $5 \cdot 10^{-4}$. The parameters are perturbed using a normal distribution with a standard deviation of 0.002 for the lower permeable layer, and 0.08 for the higher permeable layer. All configurations are tested for the ensemble sizes 50, 100, 150, and 200. Furthermore, we assume that the truth boundary conditions are known.

Fig. 9.6 shows the data assimilation results for the observation points 2, 5, 8, and 11. The results for all ensemble sizes are similar for all observation points. The difference between the estimated and truth state is the lowest for an ensemble size of 200. The estimated hydraulic head values at the observation point 5 (right side of the model domain), and observation point 11 (left side of the model domain) are very close to the truth hydraulic head values. In contrast, the estimated hydraulic head values at observation point 2 (top of the model domain), and observation point 8 (bottom of the observation point) show a slightly higher deviation from the truth hydraulic head values. For all observation points and all ensemble sizes, we observe no changes in the hydraulic head after the third time step, which is in accordance with the truth hydraulic head values. We furthermore observe that the spread of the ensembles is small for the observation points 5 and 11 and negligible for the other two.

Influence of the Observations Points

We first investigate the influence of the position of the observation points on the estimated initial condition. Therefore, we repeat the analyses of the Ensemble Kalman Filter with the second set of observation points (see Tab. 9.2, and Fig. 9.2). Fig. 9.7 shows the estimated, and truth hydraulic head values at the observation points 3, 6, 11, and 14. For all observation points and ensemble sizes, we observe an increase in the spread width of the ensembles. Again, the estimated hydraulic head values are similar for all ensemble sizes. From time step three onwards, we obtain the same estimated hydraulic head values as for our reference analysis. For the second set of observation points, we obtain a closer match between the estimated and truth hydraulic head values for all observation points. Especially the misfit for the observation points at the top and bottom of the model domain significantly decreases in comparison to the first set of observation points.

In Fig. 9.8, we show the estimate of the initial conditions for the first and second set of observation points using an ensemble size of 200. Note that the first set is denoted as "nearby observations" because the observation points are close to the higher permeable layer. The second set of observations is referred to as "faraway observations" because their distances to the higher permeable layer are greater than those of the first set of observation points. For the nearby observations, we see that the increase in hydraulic head due to the higher permeable layer is detected but underestimated. For the same time frames, we observe for the faraway observations a decrease in hydraulic head,

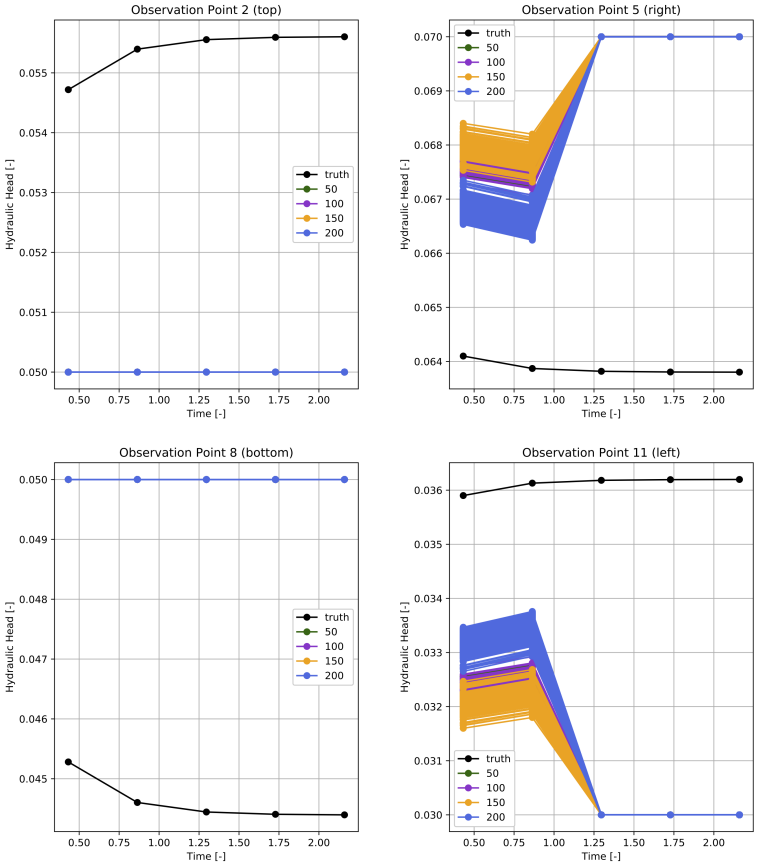


FIGURE 9.6: Comparison of the estimated and truth hydraulic head values for ensemble sizes of 50 (green), 100 (violet), 150 (orange), and 200 (blue) over time at the observation points 2, 5, 8, and 11 using the nearby observation points and certain boundary conditions. The truth solution is denoted in black.

which would indicate a lower permeable layer. From the third time step onward both realizations estimate the same initial conditions and do not detect the obstacle in the middle of the model.

For an ensemble size of 200, we show in Fig. 9.9 the difference between the estimated and truth initial conditions obtained by the Ensemble Kalman Filter using the nearby observation point (first column), and the faraway observation points (second column). Furthermore, we show in the third column the difference between the two realizations of the Ensemble Kalman Filter. In accordance with our previous observations, the difference to the truth model is the same for the nearby and faraway observations from the third time step onwards. Hence, the difference between those two is zero. For the first time step, we observe higher differences for the faraway observations than the nearby observations. The same is true for the second time step. For all time steps, the error distribution is symmetrical. In the right half of the model domain, we underestimate the hydraulic head values, and in the left model domain, we overestimate the hydraulic head values.

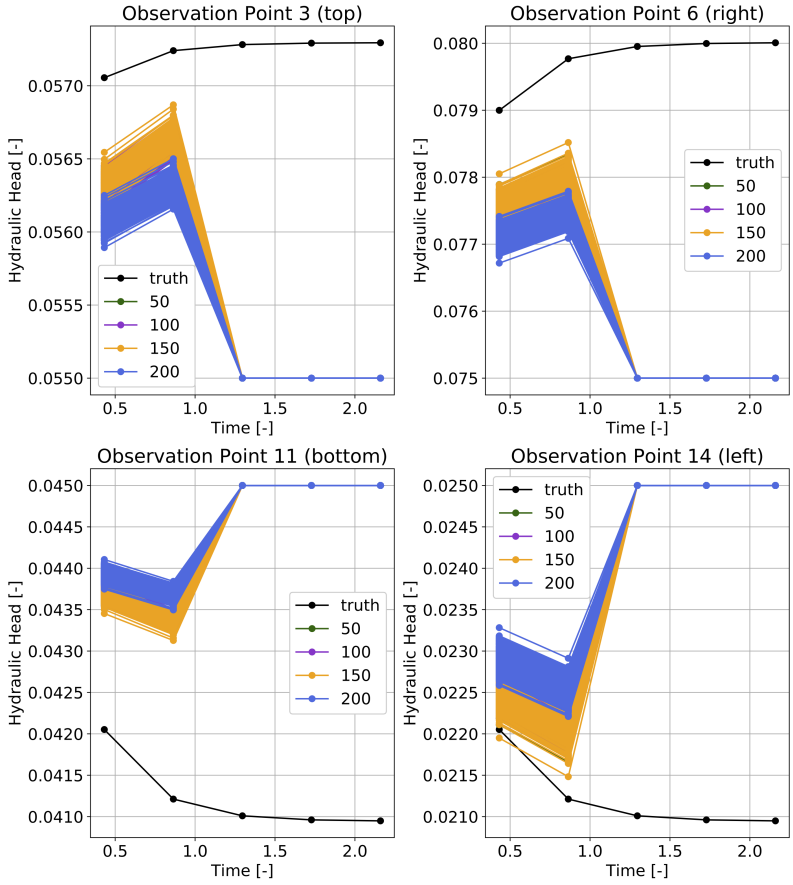


FIGURE 9.7: Comparison of the estimated and truth hydraulic head values for ensemble sizes of 50 (green), 100 (violet), 150 (orange), and 200 (blue) over time at the observation points 3, 6, 11, and 14 using the faraway observation points and certain boundary conditions.

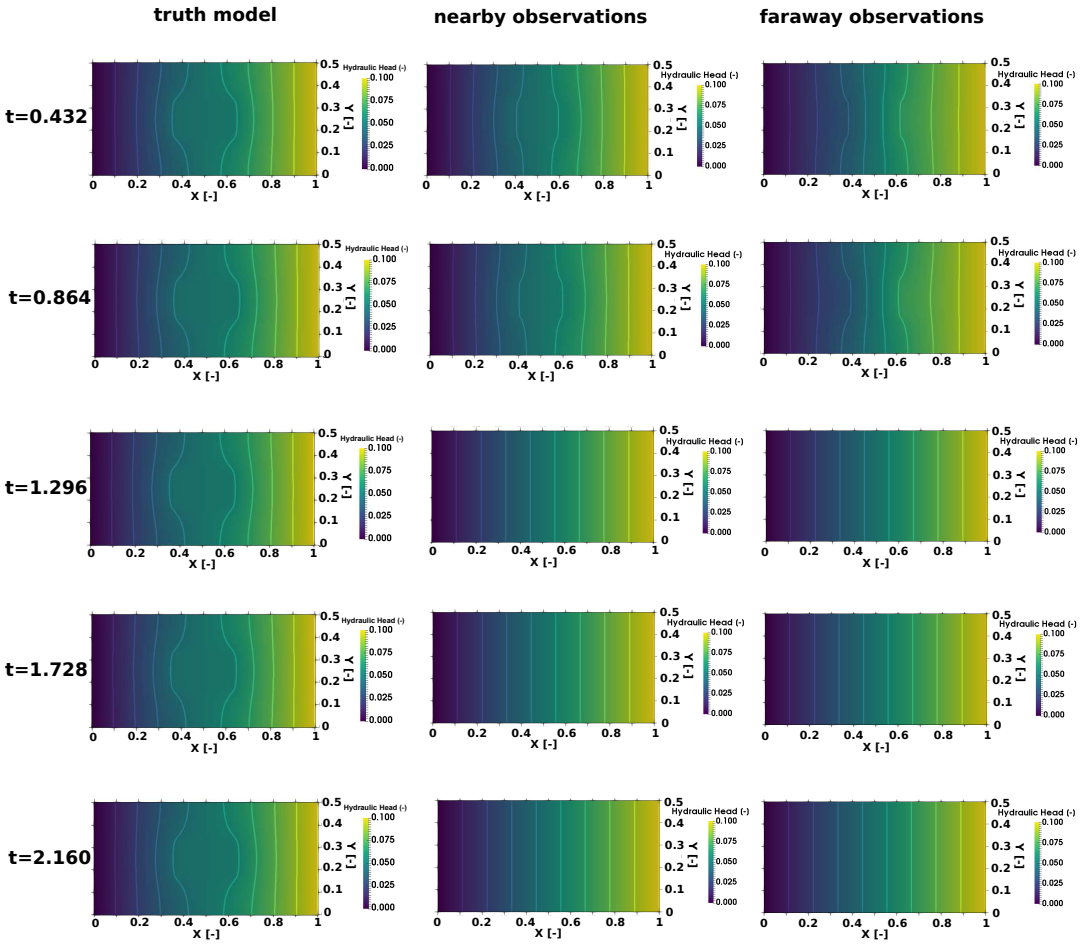


FIGURE 9.8: Estimated initial conditions for the truth model (first column), the nearby observation points (second column), and the faraway observation points (third column) with certain boundary conditions at time 0.432, 0.864, 1.296, 1.278, and 2.160.

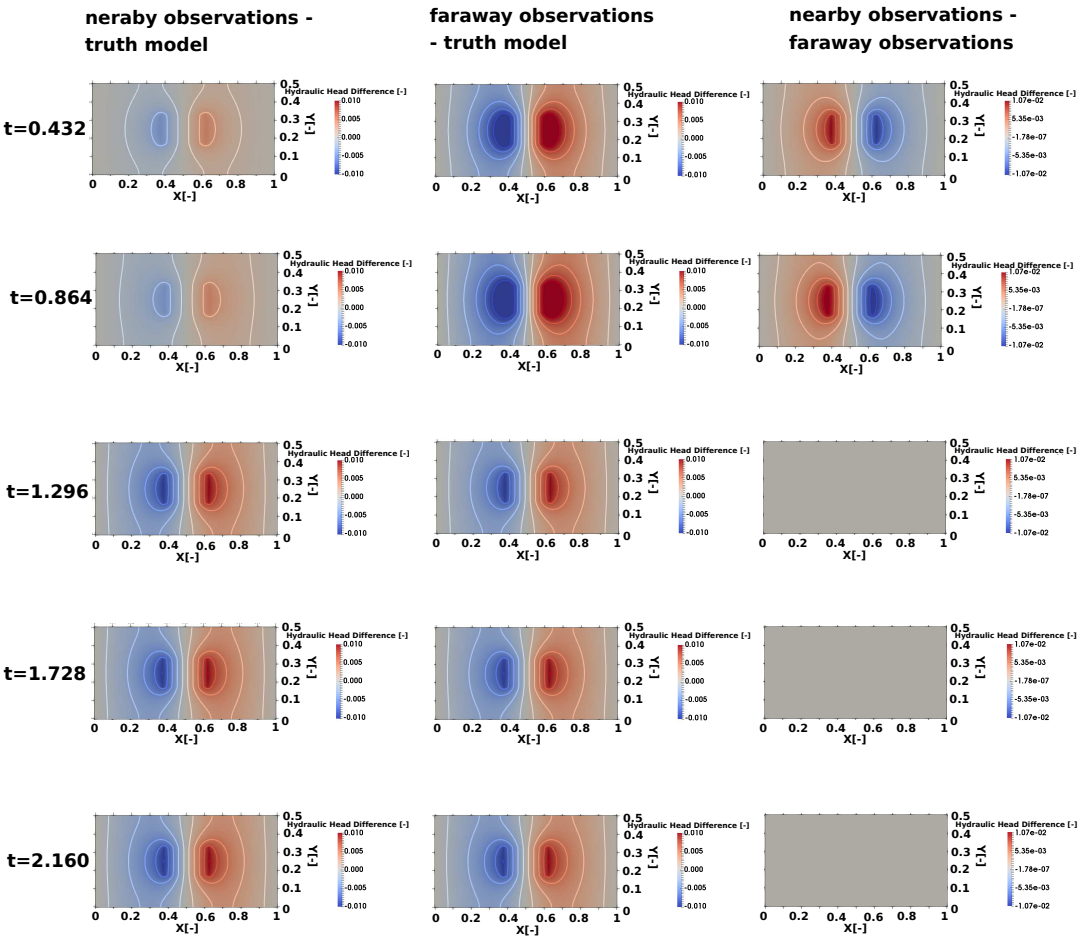


FIGURE 9.9: Difference between the estimated initial conditions and the truth initial conditions for the nearby observation points (first column), the faraway observation points (second column) at time 0.432, 0.864, 1.296, 1.278, and 2.160. At the same time steps, the difference in the hydraulic head distribution between the nearby and faraway observation points with certain boundary conditions are shown in the third column.

Influence of the Boundary Conditions

As a next step, we investigate the influence of the uncertainty of the boundary condition on the estimated hydraulic head distribution. So far, we used a lifting function to remove the influence of the boundaries on the model and to ensure that the Ensemble Kalman Filter does not modify the boundary conditions. Now, we treat the boundary conditions as uncertain and no longer use a lifting function. In that way, we have now an Ensemble Kalman Filter that also modifies the boundary conditions.

In Fig. 9.10, we see the hydraulic head values for the observation points 2, 5, 8, and 11 over time. In comparison to our reference model, we see an increase in the spread width for the observation points in the entire model domain. We obtain similar estimated hydraulic head values for all ensemble sizes at all observation points. The difference between the estimated and truth hydraulic head values is significantly higher than those observed for the reference model. Other than that, we observe the same behavior as for the reference model.

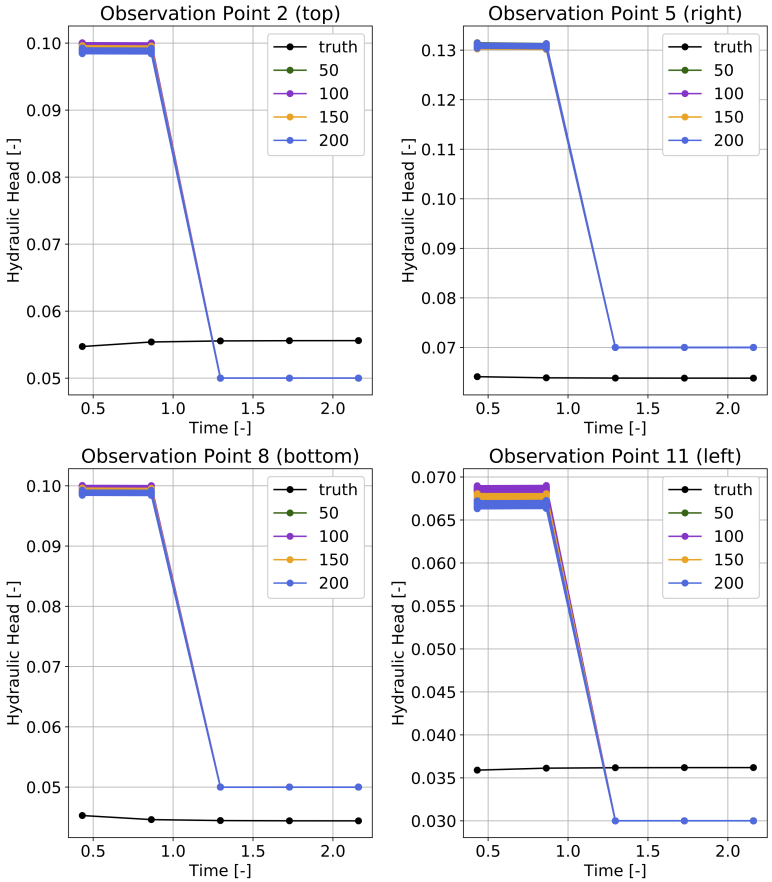


FIGURE 9.10: Comparison of the estimated and truth hydraulic head values for ensemble sizes of 50 (green), 100 (violet), 150 (orange), and 200 (blue) over time at the observation points 2, 5, 8, and 11 using the nearby observation points and uncertain boundary conditions.

Fig. 9.11 shows the truth initial conditions (first column), the estimated initial conditions assuming known boundary conditions (second column), and the estimated initial conditions assuming unknown boundary conditions (third column). For the case of known boundary conditions, we retrieve the symmetric pattern of the hydraulic head

distribution. However, for the first two time steps, we underestimate the change in hydraulic head caused by the permeability contrast. In case of the unknown boundary conditions, we seem to lose the symmetric pattern of the hydraulic head distribution. On the left model domain side, we overestimate the hydraulic head values, whereas, on the right side, we underestimate them. Note that the errors are higher on the right model domain side than on the left. This behavior is again only observed for the first two time steps. From time step three onwards, we observe for both realizations that the filter is not able to recover the initial state. Hence, we obtain for the case of the known boundary conditions a hydraulic head gradient between 0 and 0.1, the pure boundary influence, and for the realization assuming unknown boundaries, a zero solution.

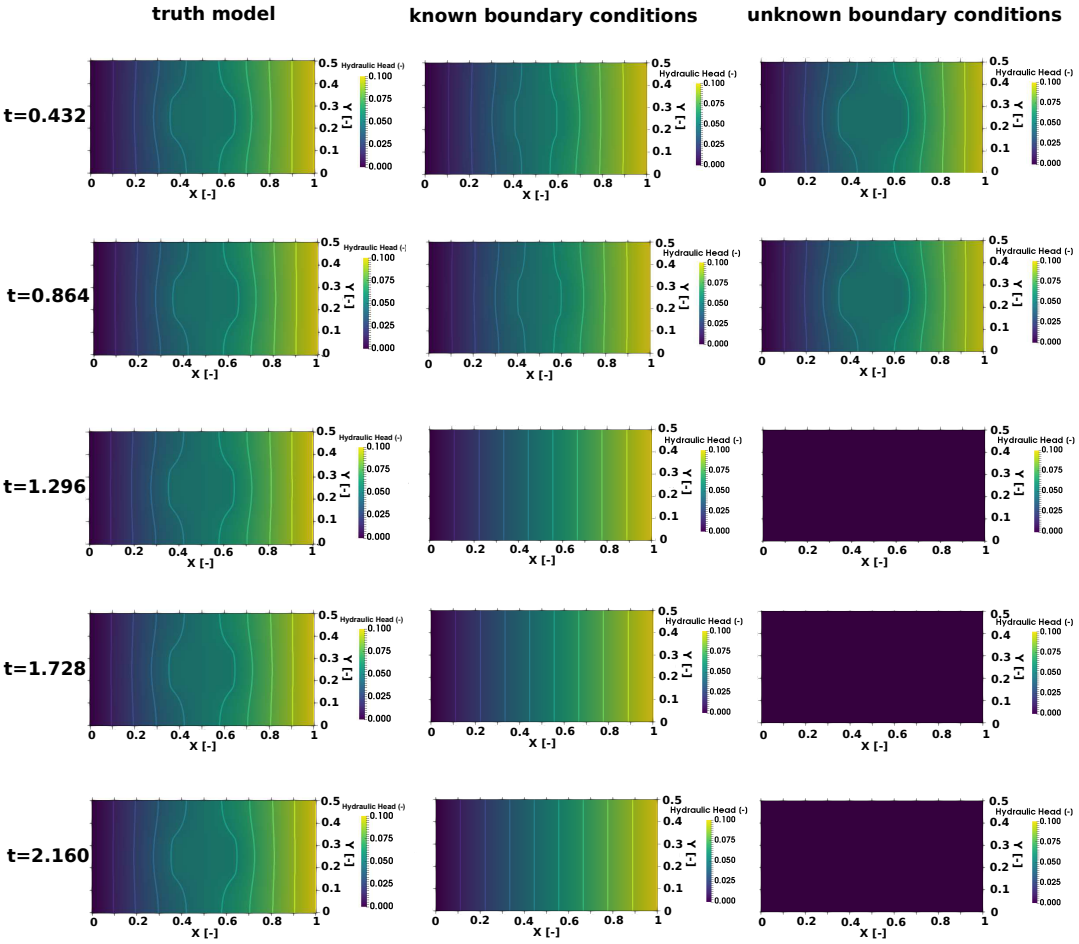


FIGURE 9.11: Estimated initial conditions for the truth model (first column), the nearby observation points with certain boundary conditions (second column), and the nearby observation points with uncertain boundary conditions (third column) at time 0.432, 0.864, 1.296, 1.728, and 2.160.

Fig. 9.12 shows the differences between the model assuming known boundary conditions and the truth model (first column), the model assuming uncertain boundary conditions and the truth model (second column), and the model with known and uncertain boundary conditions (third column). We discussed the difference between the model assuming known boundary conditions and the truth model above. Therefore, we focus the presentation of the results on the second and third column of Fig. 9.12. We observe for the first two time steps relatively small errors, where the right part of the model yields slightly higher errors than the left side. However, from the

third time step onwards, we get huge errors since the ENKF is not able to recover the initial state. Note that due to the huge error differences for the first and last time steps, we require two different color bars.

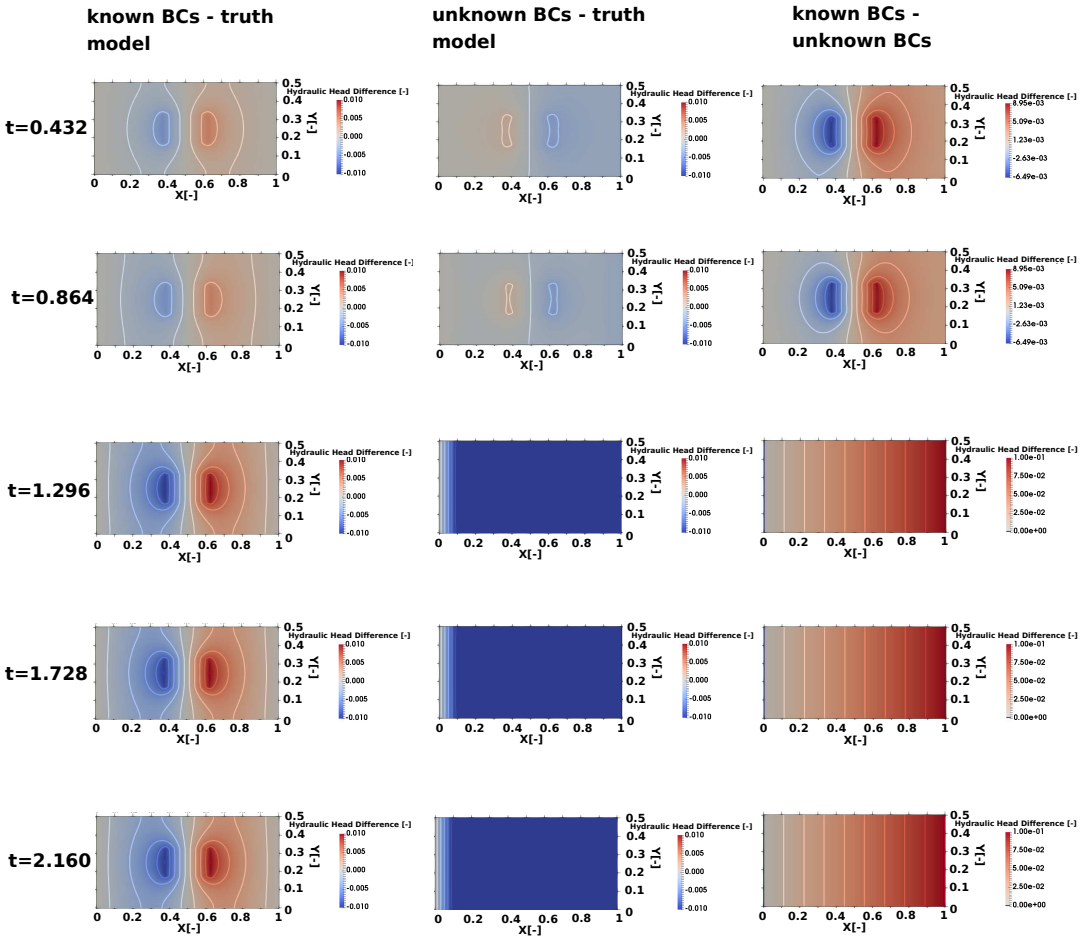


FIGURE 9.12: Difference between the estimated initial conditions and the truth initial conditions for the nearby observation points with certain boundary conditions (first column), and uncertain boundary conditions (second column) at time 0.432, 0.864, 1.296, 1.728, and 2.160. At the same time steps, the difference in the hydraulic head distribution between the nearby observation points with certain and uncertain boundary conditions is shown in the third column. Note that due to the huge error differences for the first and last time steps, we require two different colorbars.

9.4.3 Ensemble Kalman Filter – Reduced Basis Version

In this thesis, we aim to demonstrate the advantages of the RB method for various geophysical forward and inverse problems. In that regard, we evaluate the performance of the RB method in combination with the ENKF. If we use the RB method with the classical set up of the ENKF (Fig. 9.13), we would require one offline stage per ensemble per forecast step. In that case, the FE method is more efficient than the RB method. To overcome the issue, we slightly modify the ENKF. Instead of propagating the corrected initial condition, for each ensemble, after each update step, we propagate the mean of all initial conditions (see Fig. 9.14). In the following, we first investigate the errors introduced by the Filter modification using the finite element method, and then the approximations errors introduced by the reduced basis method.

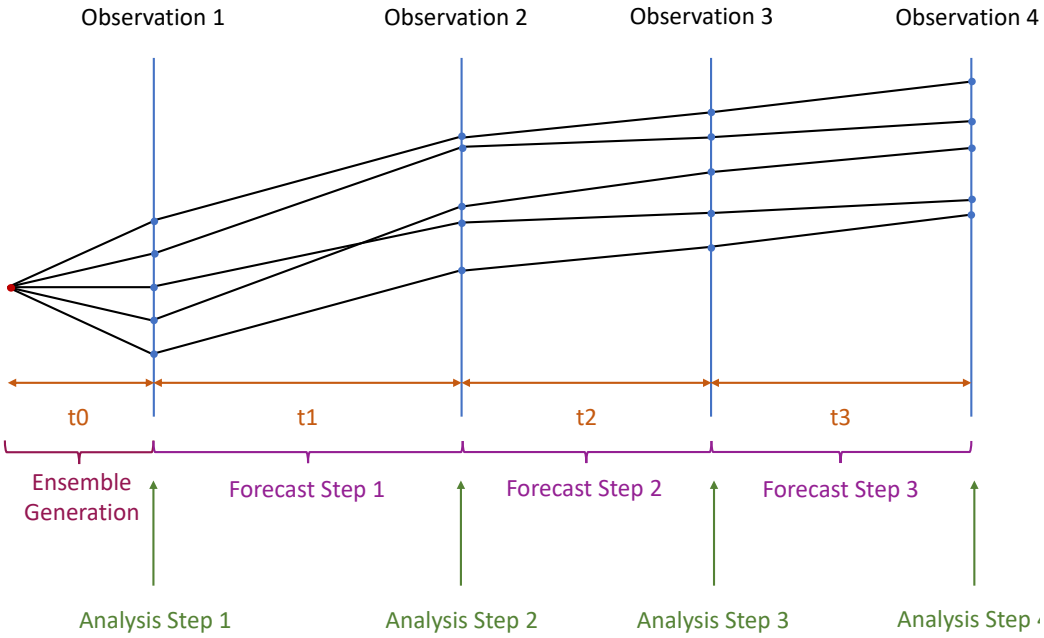


FIGURE 9.13: Schematic representation of the Ensemble Kalman Filter.

Modification of the Ensemble Kalman Filter

In this section, we investigate how large the errors are by using the modified instead of the classical ENKF. To avoid any approximation errors, we use the FE method for this comparison. Fig. 9.15, and 9.16 show no visible difference between the two versions of the ENKF. Therefore, we focus on the results in Fig. 9.17. The differences are only visible by looking at the third column, where the difference between the original, and the modified ENKF are displayed. For the first time step, and from the third time step onwards, we get the same initial state for both filters. At the second time step, we get a symmetric error distribution. In the lower permeable layer, we underestimate the hydraulic head with respect to the original filter in the left part of the model domain. We overestimate the hydraulic head in the right part of the model. For the higher permeable layers, we observe the reversed pattern, and at the model boundaries, the interfaces, and in the center of the model, we get the same hydraulic head values for both filters. Note that the errors are four orders of magnitude smaller than the observed hydraulic head values (with subtracted lifting function). For the

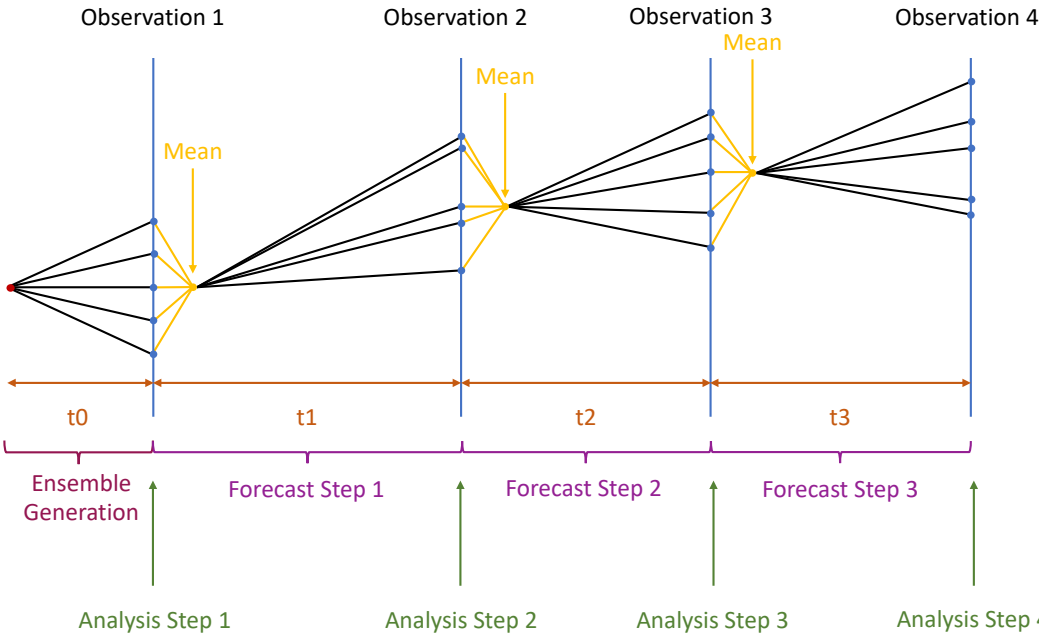


FIGURE 9.14: Schematic representation of the modified Ensemble Kalman Filter. Note that the mean of the initial conditions is taken at the analysis step, and not later on. We moved the mean slightly to the right only for visualization purposes.

third to fifth time step, we obtain the same error distribution. On the left model side, we underestimate the hydraulic head values, and on the right side, we overestimate them. The error increased by two orders of magnitude in comparison to the second time step. However, we are still one order of magnitude below the error produced by the data assimilation itself.

Reduced Basis Method

After evaluating the errors caused by the modification of the ENKF that is required to efficiently use the reduced model as a surrogate for the full FE model, we investigate the approximation quality. Therefore, we compare the assimilated hydraulic head distribution at every update step of the finite element and reduced basis version of the modified ENKF. Fig. 9.18 shows the results, and Tab. 9.3 the simulation properties. The errors for the first step are the highest and range from $-2 \cdot 10^{-8}$ to $2 \cdot 10^{-8}$, where the error tolerance of the forward solve is $1 \cdot 10^{-5}$. The execution of the Ensemble Kalman Filter, with 200 ensemble members, using the FE method takes 2.3 h, and 59 s with the RB method including the offline stage. Hence, we obtain a speed-up of 140.

Additionally, we increased the number of ensembles, in order to see whether a higher number of ensembles yields better state estimation or if an ensemble size of 200 is sufficient to estimate the state. Fig. 9.19 summarizes the results and shows similar hydraulic head values over all time steps at all observation points.

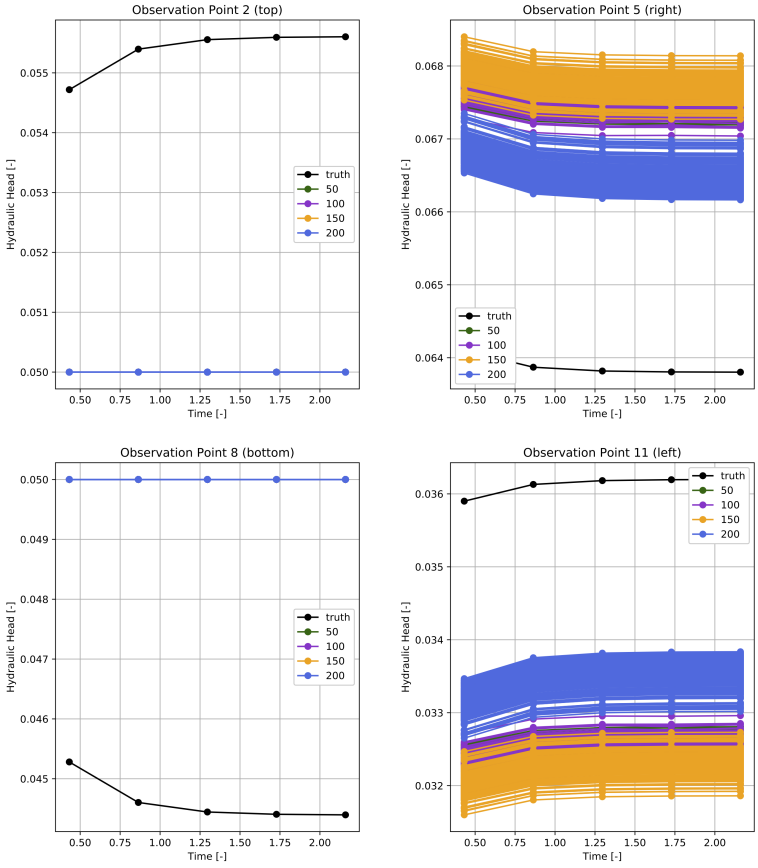


FIGURE 9.15: Comparison of the estimated and truth hydraulic head values for ensemble sizes of 50 (green), 100 (violet), 150 (orange), and 200 (blue) over time at the observation points 2, 5, 8, and 11 using the nearby observation points, certain boundary conditions, and the modified Ensemble Kalman Filter.

TABLE 9.3: Simulation parameters for the forward simulation of the Box model, which is used for the Ensemble Kalman Filter, for both the FE and the RB method.

Simulation parameter	Value
FE method	
Solver type	Newton
Linear solver tolerance	10^{-5}
Nonlinear solver tolerance	10^{-8}
Preconditioner type	hypre
Hypre type	boomerang

ksp_gmres_restart	101
RB method	
Linear solver tolerance	10^{-10}
Nonlinear solver tolerance	10^{-10}
Rel. training tolerance	10^{-5}
Parameter range (outer box)	$7{\cdot}10^{-3}$ to 0.7
Parameter range (inner box)	0.5 to 1.5
Number of parameters	2
Number of training samples	500
Maximum number of basis functions	10

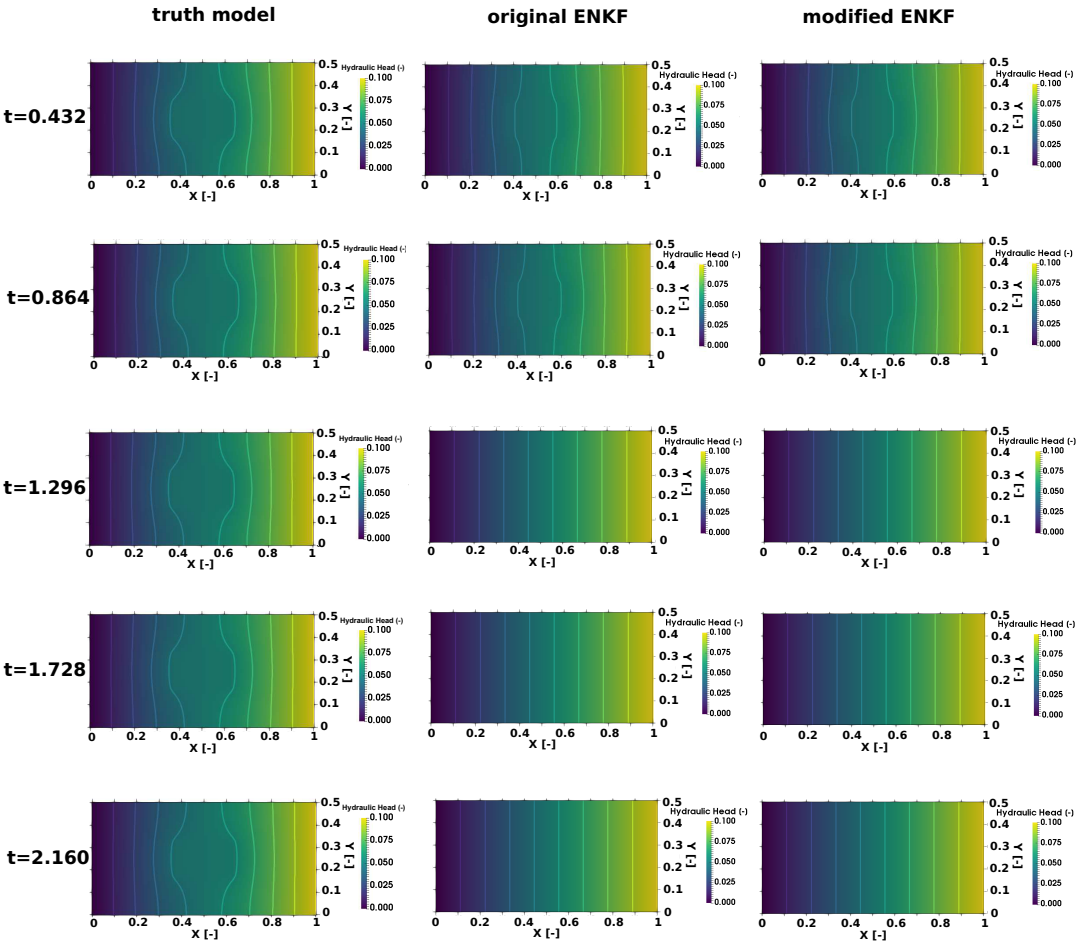


FIGURE 9.16: Estimated initial conditions for the truth model (first column), the nearby observation points with certain boundary conditions using the original ENKF (second column), and the nearby observation points with certain boundary conditions using the modified ENKF (third column) at time 0.432, 0.864, 1.296, 1.728, and 2.160.

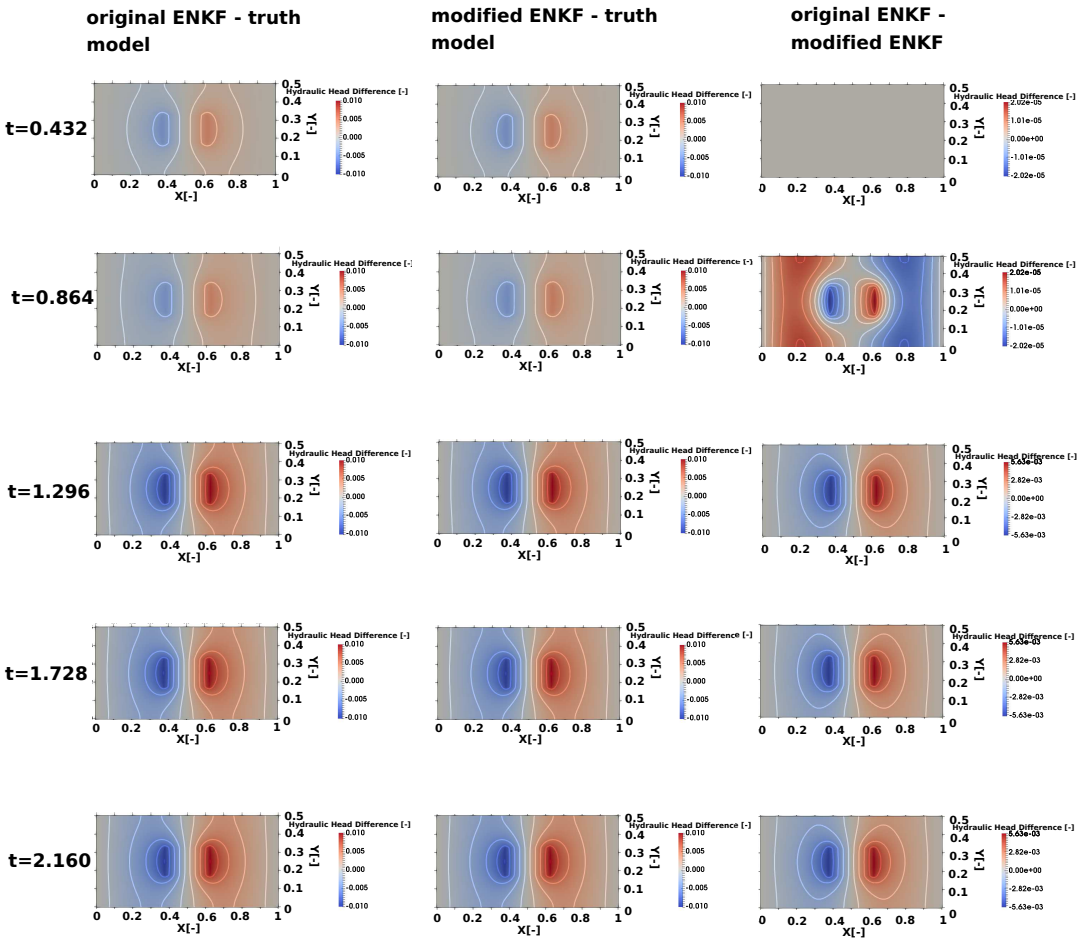


FIGURE 9.17: Difference between the estimated initial conditions and the truth initial conditions for the nearby observation points with certain boundary conditions using the original ENKF (first column), and the modified ENKF (second column) at time 0.432, 0.864, 1.296, 1.728, and 2.160. At the same time steps, the difference in the hydraulic head distribution between the original and modified ENKF are shown in the third column.

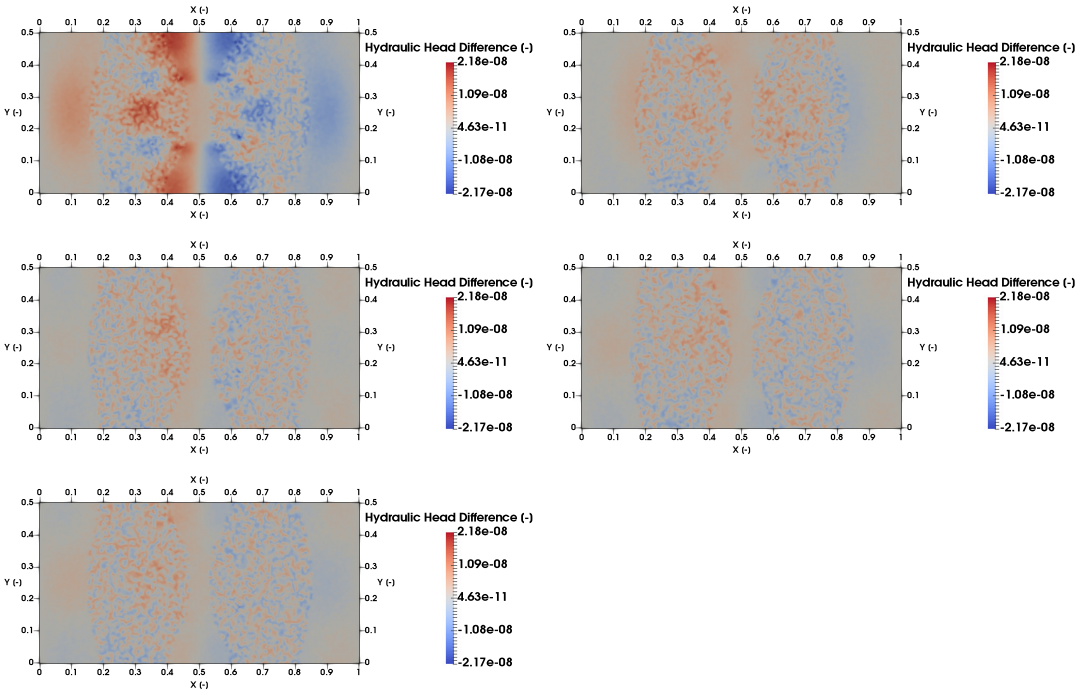


FIGURE 9.18: Difference of the estimated hydraulic head distribution between the reduced and full version of the modified Ensemble Kalman Filter for an error tolerance of 10^{-5}.

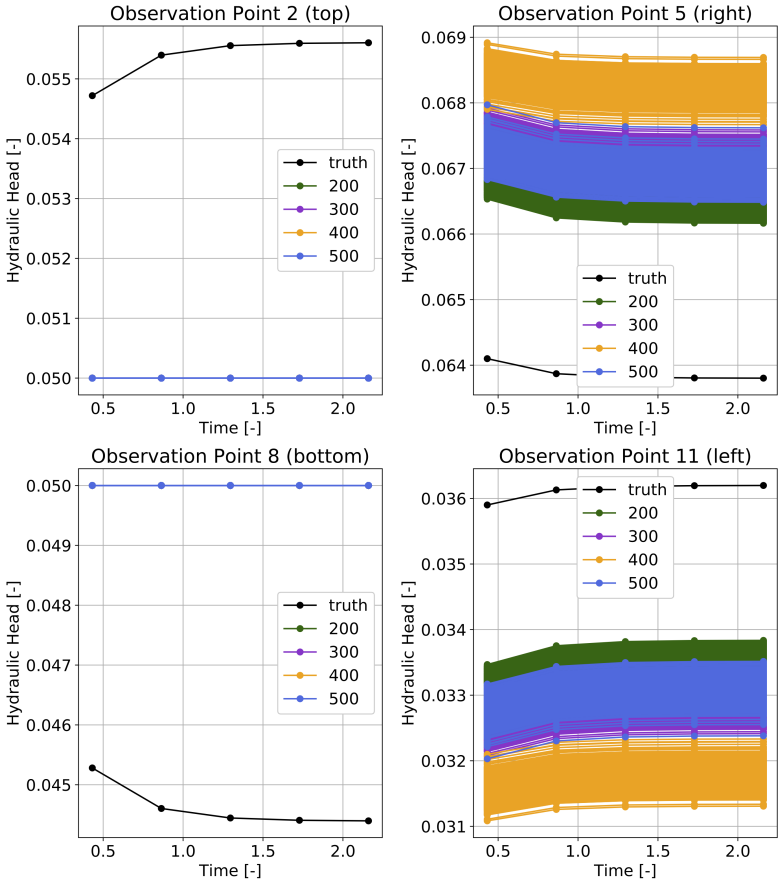


FIGURE 9.19: Comparison of the estimated and truth hydraulic head values for ensemble sizes of 200 (green), 300 (violet), 400 (orange), and 500 (blue) over time at the observation points 2, 5, 8, and 11 using the nearby observation points, certain boundary conditions, and the modified Ensemble Kalman Filter.

9.5 Discussion

Analog to the Result Section, we discuss the results of the forward model first. The discussion of the FE and RB filter versions follows. There, we especially point out the advantages and restrictions of the RB version.

9.5.1 Forward Model

For the Box model, we obtained a speed-up of over 1,600. As for the Fault model, we observe that the speed-ups for the parabolic PDEs are significantly higher than those for the elliptic case. In comparison, the Base model had for a similar mesh size a speed-up of under 100. The reduction in both space and time causes the increased speed-up (as explained for the Fault model).

The generation of the reduced basis takes only slightly longer than one FE solve because, for instance, the mesh, and the preconditioners are reusable. Hence, we can save a lot of time. That also implies that we obtain a much earlier pay-off than for the elliptic problems. For the Box model, we only need two forward simulations in order to be more efficient with the RB method than with the FE method. Note that this pay-off would shift to a higher number of forward simulation if we spend more time optimizing the repetitive execution of the FE problem. As we explained for the Darcy flow problem using the Fault model.

In the following, we discuss the FE version of the Ensemble Kalman Filter, and how we can benefit from the speed-up gained through the RB method.

9.5.2 Ensemble Kalman Filter – Finite Element Version

The spread of the ensembles represents the amount of uncertainty in the system. For the reference model, where we have observations close to the higher permeable layer, and where we assume known boundary conditions, we have a very low spread. Consequently, the amount of uncertainty is for the reference model low. The spread increases if we use the further away observations in the data assimilation, or if we assume that the boundaries are uncertain. So, for both changes, we increase the uncertainty in the system. For the configuration using uncertain boundary conditions, this is logical since the addition of another source of uncertainty should increase the overall uncertainty of the system. The spread increases also for the "far away" observations since the changes in the hydraulic head values are smaller than those for the closer observations. The smaller the changes in the state, the more problematic the ENKF becomes in recovering the state, which becomes apparent through the increase in uncertainty.

That is also the reason why no realization of the Ensemble Kalman Filter is able to recover the true hydraulic head distribution from the third time step onwards. The difference to the true hydraulic head is smaller for the case of known boundary conditions. Their influence is added later on over the lifting function and is therefore not modified by the filter. Hence, we see only the influence of the boundary conditions for the cases of known boundary conditions and a zero state for the case of unknown boundary conditions. That represents the trivial solution for a system where no changes occur.

Similarly, the state estimates are better at the observation points left and right from the higher permeable layer than above and below it. Also, here we have smaller changes in the hydraulic head above and below the higher permeable layer than left and right from it.

For the ENKF realization with the faraway observation points, we observed closer matches to the true state as for the reference model at the observation points. However, the differences to the reference model are relatively small, and a direct comparison is not possible at the observation points since they are at different locations for both realizations. Therefore, we focus the discussion on the mean of the entire state. The mean of the estimated state for the faraway observations shows overall higher difference to the true model than the one with the closer observations. Furthermore, the state estimate of the version with the faraway observations corresponds to one caused by a lower permeable inner box. The version with the closer observations recovers the higher permeable layer, although it underestimates the hydraulic head values. We can conclude that the ENKF performs better for closer observations. This matches our expectations since for the closer observations, we obtain higher changes in the hydraulic head.

Also, the mean estimated state of the case of unknown boundary conditions is further away from the true state than the one with known boundary conditions. This is not surprising since we add another source of uncertainty. Interestingly, the symmetry of the hydraulic head distribution seems to get lost for the case of unknown boundary conditions. Note that this is not true and just caused by the visualization. For a better comparison of the different estimated states, we chose the same color bar for every realization. In case of the unknown boundary conditions, we apply the filter also to the boundary itself. The update step of the ENKF is purely statistical. Hence it does not preserve the physical properties of the model. Therefore, we obtain hydraulic head values below 0.1 at the right boundary. We adjust the color bar to a maximum value of 0.1, leading to the apparently asymmetric estimated state.

9.5.3 Ensemble Kalman Filter – Reduced Basis Version

The modification of the Ensemble Kalman Filter introduces errors from the second time steps onward. The first time step is still the same since we take the mean of the estimated states after the update step. For the second time step, we get an error in the magnitude of 10^{-5}, which are negligible in comparison to typical measurement errors. For the third to fifth time step, we obtain errors in the magnitude of 10^{-3}, which is significantly higher than for the previous time step. This is caused by the fact that we add to the already existing error, additional errors. The error does not increase after the third time step since the filter fails to recover the initial conditions and hence keeps the state estimate from the third time step. Also, for the last time steps, the error is still significantly lower than those caused by the data assimilation itself.

However, this will not hold in general. For the tested example, we obtain a very small spread. Hence the mean is a good approximation. If we have examples with a higher uncertainty, we will still get a good estimate of the mean state but will lose the information about the uncertainty through using the modified filter. Therefore, it is important to investigate whether the impulse response can be used to solve the problem. In that case, we convolve the initial conditions with a Dirac function. In that way, we can rewrite the initial condition as a linear combination of Dirac functions such that only the parameter-dependent impulse response needs to be approximated (Grepl, 2005).

We have seen that for this simple benchmark problem already 50 ensemble members are enough to obtain a good estimate of the hydraulic head distribution. Due to the small offline times, the RB method is already more efficient than the FE method for that relatively small number of forward simulations. For 50 ensemble members,

we get a speed-up of 41, including the offline stage that speed-up increases to 140 for 200 ensembles.

To conclude, we illustrate that the RB method is an ideal method for parameter estimation methods but can also suitable for data assimilation problems if we are interested in the mean state only. Both parameter estimation and data assimilation problems are not limited to the fields of geothermal and Darcy flow applications but are also used in other geophysical fields.

Chapter 10

Extension to Other Geophysical Applications

In this thesis, we considered the conductive heat transfer and the Darcy flow problem as geophysical applications. Here, we want to show how the RB method can be beneficial for other geophysical fields, such as hydrothermal and electrical resistivity tomography applications.

10.1 Hydrothermal

So far, we have considered only the changes in temperature due to a conductive heat transfer, and the changes in pressure. Now, we want to focus on the coupled processes of both. In this section, we conceptually show how the RB method can also be applied to hyperbolic problems by using concepts such as entropy production.

10.1.1 Forward Problem

We consider an advection-diffusion problem, where we assume an isotropic medium with negligible radiative effects and viscous dissipation. Furthermore, we have a local thermal equilibrium of the solid and fluid phases (Nield and Bejan, 2017).

$$\begin{aligned} \frac{\partial}{\partial t}\left\{(\rho c)_m T\right\} + \nabla\left(\rho_f c_f \; q \; T\right) - \nabla\cdot\left(\lambda_m \nabla T\right) = 0, \\ \text{with } (\rho c)_m = (1-\phi)\rho_s c_s + \phi\rho_f c_f, \\ \text{and } \lambda_m = (1-\phi)\lambda_s + \phi\lambda_f. \end{aligned} \tag{10.1}$$

Here, t is the time, ρ the density, c the specific heat capacity, T the temperature, q the Darcy flux, λ the thermal conductivity, and ϕ the porosity. The subscript m, s, and f denote the porous medium, the solid, and the fluid, respectively. We want to investigate the effect of natural convection; hence, we need to take the density as a function of temperature into account, yielding the following expression (Nield and Bejan, 2017):

$$\rho_f = \rho_0\left\{1 - \alpha\left(T - T_0\right)\right\}, \tag{10.2}$$

where ρ_0 denotes the reference fluid density for a given reference temperature T_0, and α is the thermal expansion coefficient. Additionally, we simplify the relationship by taking the Boussinesq assumption into account (Nield and Bejan, 2017):

$$B = \alpha\Delta T. \tag{10.3}$$

This assumption states that all fluid and solid properties are considered as constant and that the density variation is solely arising from the body force (Nield and Bejan, 2017).

For efficiency reasons, we again use the nondimensional form of the hyperbolic PDE (Eq. 10.1). We use the nondimensionalization presented in Huang (2018) that was inspired by Fowler and Fowler (1997). The nondimensional parameters and variables are (Huang, 2018):

$$\begin{aligned} T &= T_0 + (\Delta T)T^*, \\ \rho_f &= \rho_0 \left(1 - BT^*\right), \\ p &= p_0 - \rho_0 g y + [p]p^*, \\ q &= [q]q^* \\ t &= [t], \\ \nabla &= \frac{1}{l_{\text{ref}}}\nabla^*, \\ \partial_t &= \frac{1}{[t]}\partial_t^*, \\ \text{with } [p] &= \sqrt{\frac{\mu d B g \lambda_m}{c_f k}}, \\ [q] &= \sqrt{\frac{k B g \lambda_m}{\mu c_f l_{\text{ref}}}}, \\ \text{and } [t] &= \frac{(\rho c)_{m_0} l_{\text{ref}}^2}{\lambda_m}. \end{aligned} \tag{10.4}$$

Inserting these nondimensional parameters and variables into the mass conservation equation ($\nabla \cdot q = 0$), the momentum conservation (Eq. 7.1), and the energy conservation (Eq. 10.1) yields the following nondimensional equations (Huang, 2018):

$$\begin{aligned} \nabla \cdot q^* &= 0, \\ q^* &= -\nabla^* p^* + \sqrt{Ra} T^* \hat{j}, \\ T^{*,k} + \sqrt{Ra} q^* \cdot \nabla^* T^* - \nabla^2 T^* &= 0, \\ \text{with } Ra &= \frac{\rho_0 g \alpha k l_{\text{ref}} \Delta T}{\mu \kappa}, \\ \text{and } \kappa &= \frac{\lambda_m}{\rho_0 c_f}. \end{aligned} \tag{10.5}$$

Here, Ra is the Rayleigh number, $\hat{j}$ the unit vector, and k the thermal diffusivity.

10.1.2 Entropy Production

Using the certified RB method for geothermal applications has, currently, the disadvantage that we are limited to conductive heat transfer problems because of the nonexistent efficient error bounds for the convective part of the heat transfer equation. We present in this thesis how the entropy production can be applied to capture the convective effects into an effective thermal conductivity. This can then be used to simplify the heat transfer again to a purely conductive problem using the effective thermal conductivity instead of the physical one (Bejan, 1979, 2013; Manzano et al., 2016; Siavashi and Jamali, 2016; Börsing et al., 2017). This enables us to use the certified RB method also for convective geothermal problems. In this section, we want to briefly introduce the concept of entropy production and the effective thermal conductivity.

The volumetric total entropy production S_{total} can be expressed as the sum of

the irreversible heat transfer S_{therm}, and the fluid flow friction S_{visc} assuming a fluid-saturated porous medium (Bejan, 2013):

$$\begin{aligned} \frac{dS''_{\text{total}}}{dt} &= \dot{S}''_{\text{total}} = \dot{S}''_{\text{therm}} + \dot{S}''_{visc}, \\ \text{with } \dot{S}''_{\text{therm}} &= \frac{\lambda_m}{\overline{T}^2}(\nabla T)^2, \\ \overline{T} &= \frac{T_{\text{top}} + T_{\text{bot}}}{2}, \\ \text{and } \dot{S}''_{\text{visc}} &= \frac{\mu_f}{k\overline{T}}q^2 + \frac{\mu_f}{\overline{T}}\Phi. \end{aligned} \tag{10.6}$$

Here, T_{top} is the temperature at the top of the model, T_{bottom} the temperature at the bottom of the model, and Φ the viscous dissipation function. The second term of the fluid flow friction is only important for non-Darcy flow. In this thesis, we consider only Darcy flow. Hence, we neglect the second term. For the nondimensionalization, we use again the nondimensional parameters and variables of Huang (2018) (Eq. 10.4) yielding the following nondimensional expression for the entropy production (Huang, 2018):

$$\dot{S}_{\text{total}} = \int_\Omega \dot{S}''_{\text{total}} d\Omega = \underbrace{\frac{\lambda_m \Delta T^2}{\overline{T}^2 l_{\text{ref}}^2} \int_\Omega (\nabla^* T^*)^2 \, d\Omega}_{\int_\Omega \dot{S}''_{\text{therm}} d\Omega} + \underbrace{\frac{\alpha \Delta T g \lambda_m}{\overline{T} c_f l_{\text{ref}}} \int_\Omega (q^*)^2 \, d\Omega}_{\int_\Omega \dot{S}''_{\text{visc}} d\Omega}, \tag{10.7}$$

Accordingly, the dimensionless entropy generation number can be written as (Bejan, 2013; Huang, 2018):

$$N_{\dot{S}} = \frac{l_{\text{ref}}^2 \overline{T}^2}{\lambda_m \Delta T^2} \frac{\dot{S}}{V} = \frac{1}{V} \left(\int_\Omega (\nabla^* T^*)^2 \, d\Omega + \frac{\alpha l_{\text{ref}} \overline{T} g}{\Delta T c_f} \int_\Omega (\boldsymbol{q}^*)^2 \, d\Omega \right), \tag{10.8}$$

where V is the volume of the model domain.

Using the entropy production, we can now derive an effective thermal conductivity. From $\dot{S}''_{\text{therm}}$ (Eq. 10.6) follows:

$$\sqrt{\frac{\dot{S}''_{\text{therm}} \, \overline{T}^2}{\lambda_m}} = \nabla T. \tag{10.9}$$

Inserting Eq. 10.9 into the relationship $q = -\lambda_{\text{eff}} \nabla T$, we can define the effective thermal conductivity λ_{eff} as:

$$\lambda_{\text{eff}} = -\frac{q}{\overline{T}} \sqrt{\frac{\lambda_m}{\dot{S}''_{\text{therm}}}}. \tag{10.10}$$

10.1.3 Results

In this section, we want to demonstrate how the entropy production can be used to characterize the convective behavior in a hydrothermal model (Fig. 10.1) and substitute it into a conductive model over the effective conductivity. This section is meant as an introduction to the concept that is the reason why the analyses here are quite limited. We use the previously presented Base model with 2,378 degrees of freedom. We

are well aware that this mesh size is inappropriate to resolve the convection cell without numerical instabilities in the earlier timesteps. However, to introduce the concept and test the method, we require fast FE simulations, and these can be only obtained by coarse meshes. We further restrict the analyses to the forward model and only briefly outline how this could be used in, for instance, model calibrations. More detailed analyses for the forward and inverse problem are the subject of future studies.

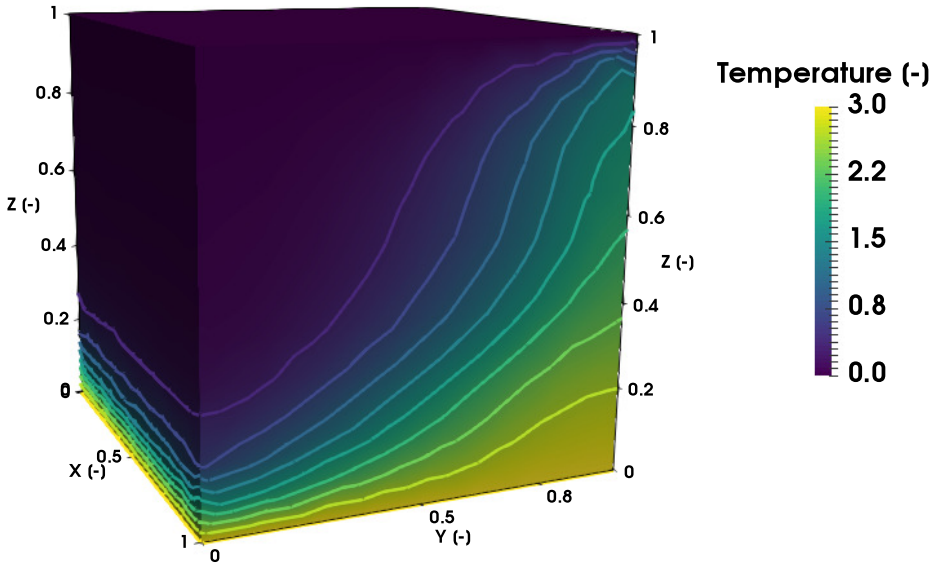


FIGURE 10.1: Hydrothermal temperature distribution of the Base model at the last time step of the finite element solution.

So far, we assumed for both the conductive and convective model homogenous and isotropic model parameters. Hence, we have, for instance, one thermal conductivity per geological unit. We want to incorporate the convective behavior of the model by using Eq. 10.10. All parameters in this equation except the thermal conductivity of the medium are space dependent. In a first attempt, we take the average of each of the space-dependent parameters over each unit such that we end up with one effective conductivity value for each geological unit.

Furthermore, we run the hydrothermal simulation long enough to reach equilibrium. Then, we take the value of the effective conductivities for the last time step and use these for the steady-state conductive simulation. In order to evaluate the quality of the various models, we look at the average temperature misfit of the entire Base model. Using a purely conductive model results in an average misfit of 0.73 (17.3 °C). Taking the effective conductivity instead of the physical one reduces this misfit by 12 % to an average misfit of 0.52 (15.2 °C). However, if we look at the temperature distribution (Fig. 10.2), we still get the distribution of a purely conductive model since we neglect all space dependencies.

So, as a next step, we extend the analysis by considering the spatial dependence of the entropy production. Therefore, we extract the entropy along the y-axis in the middle of each layer, and at the front of the model. Hence, we obtain the entropy values along the three, in Fig. 10.3 shown, black lines. These entropy values are then used to calculate the effective conductivities (Fig. 10.4). We fit the effective thermal conductivity of the top and middle layers with an exponential function and the bottom layer with a second-degree polynomial function (Tab. 10.1). For the top

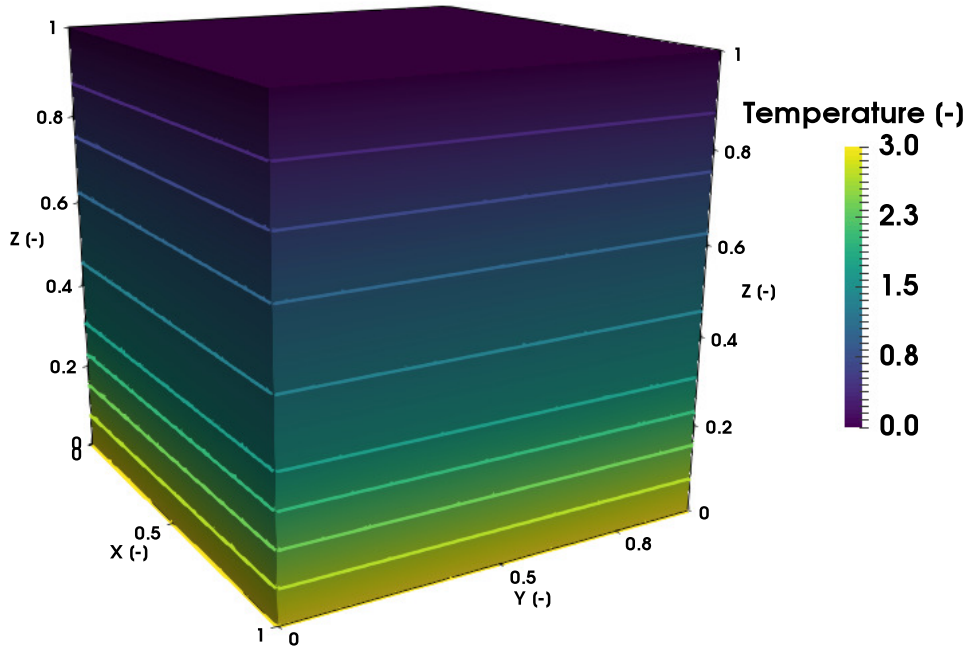


FIGURE 10.2: Temperature distribution of the Base model using one space independent effective conductivity per layer.

and bottom layer, we obtain good fits, but for the middle layer, the exponential function can not fully capture the trend of the effective conductivity. Using this fitted effective conductivities within our conductive model decreases the error by 26 % in comparison to the original conductive model with the physical thermal conductivity. Hence, we decrease the average temperature misfit from 0.73 (17.3 °C) to 0.27 (12.7 °C).

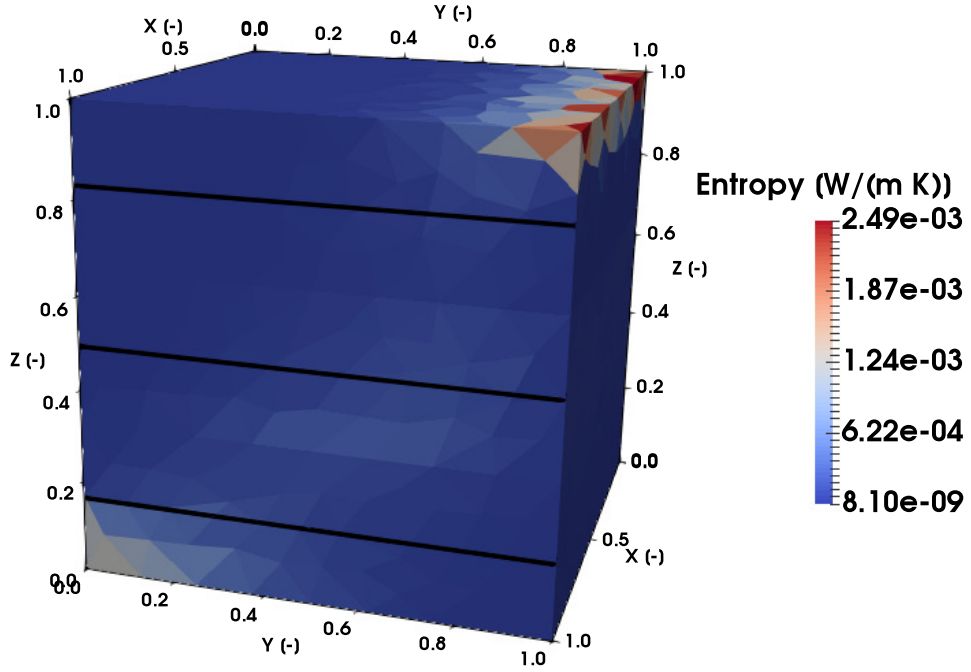


FIGURE 10.3: Entropy distribution for the hydrothermal FE simulation of the Base model. Additionally, we show the lines (denoted in black), which we use to extract the entropy values that are later on used for the fitting of the effective conductivity.

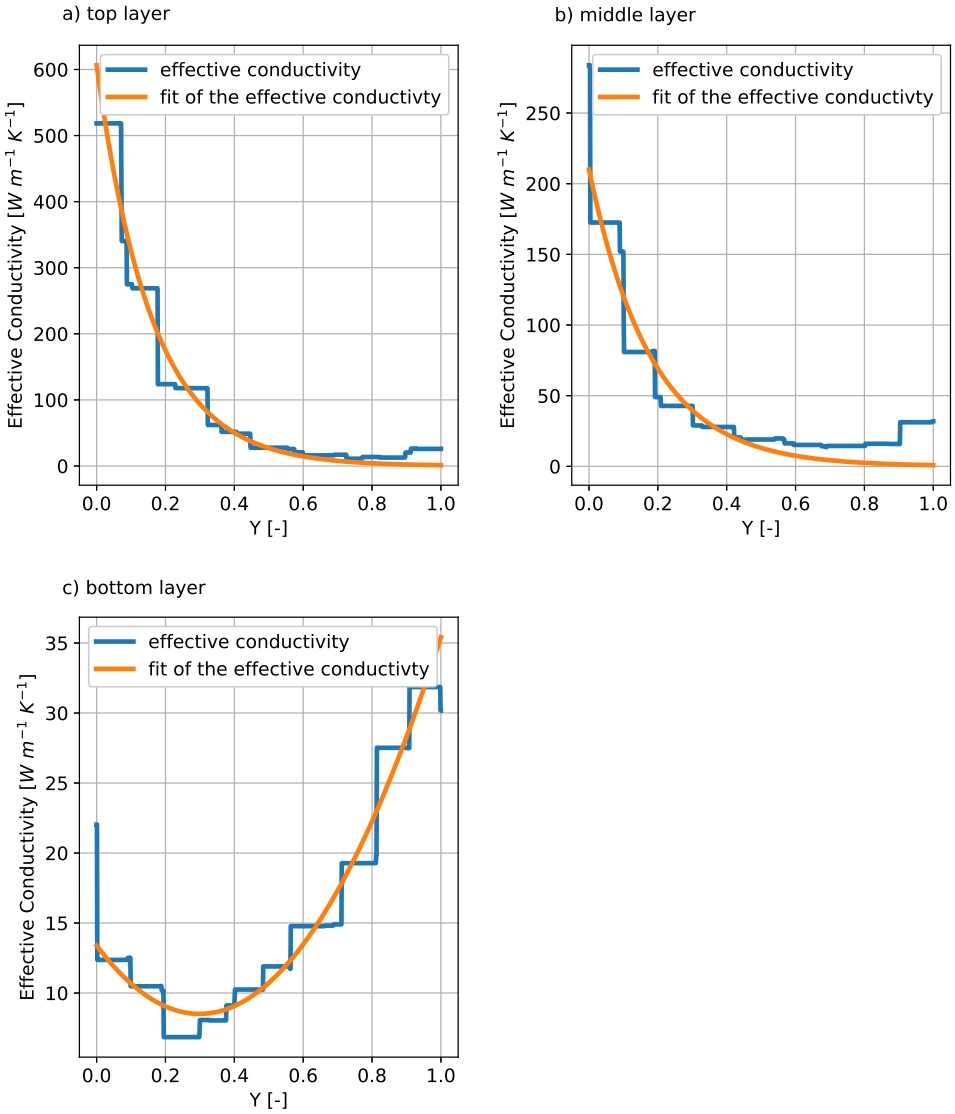


FIGURE 10.4: Values of the effective conductivity (denoted in blue) using the extracted entropy values and their fits (denoted in orange) for a) the top shale layer b) the middle sandstone layer, and c) the bottom shale layer.

Furthermore, the temperature distribution shows that we can recover the general pattern of the convection cell (Fig. 10.5). In Fig. 10.6, we show the difference between the conductive simulation using the fitted effective conductivities and the hydrothermal simulation. We see that in the upper right part of the model we are underestimating the temperatures, and in the lower right part, we are overestimating them by one to one and a half, which corresponds to 20 °C to 25 °C (Fig. 10.7).

These conductive models with the effective conductivity are governed by an elliptic PDE. Hence, we are able to use the RB method for these models. Currently, we use a single scaling parameter per layer as our parameter-dependent part. Hence, for the exponential function, we vary the amplitude a. For the polynomial function, we allow a variation of the coefficients a, b, and c where we fix the ratio between them. As for the geothermal Base model, we use a relative error tolerance of 10^{-5}. In order to reach this tolerance, we require 13 basis functions, so ten more than for the original conductive model. The spatial distribution of the approximation error shows that for every part of the model we stay below the threshold value and that there are no regions that are significantly worse approximated than others (Fig. 10.6).

TABLE 10.1: Functions and corresponding coefficients for the fit of the effective conductivity for three layers of the Base model. We denote the parameters that are not applicable with n/a.

Layer	Function	Coefficients		
		a	b	n
Top Shale Layer	$a\ \exp(b\ y)$	606.4	-6.2	n/a
Middle Sandstone Layer	$a\ \exp(b\ y)$	210.1	-5.6	n/a
Bottom Shale Layer	$ax^2 + bx + n$	54.7	-32.7	13.4

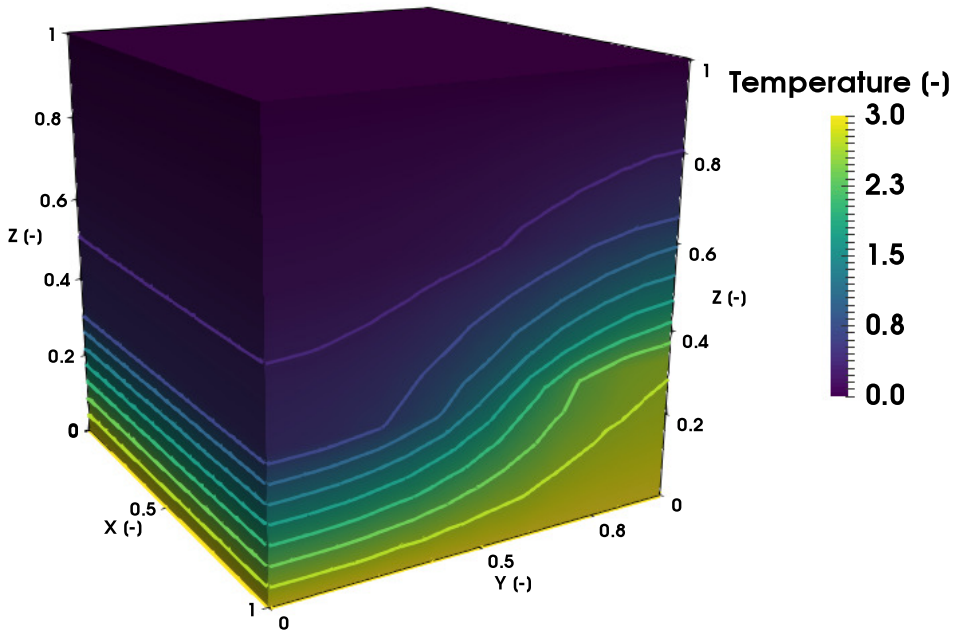


FIGURE 10.5: Temperature distribution of the Base model using a space-dependent effective conductivity.

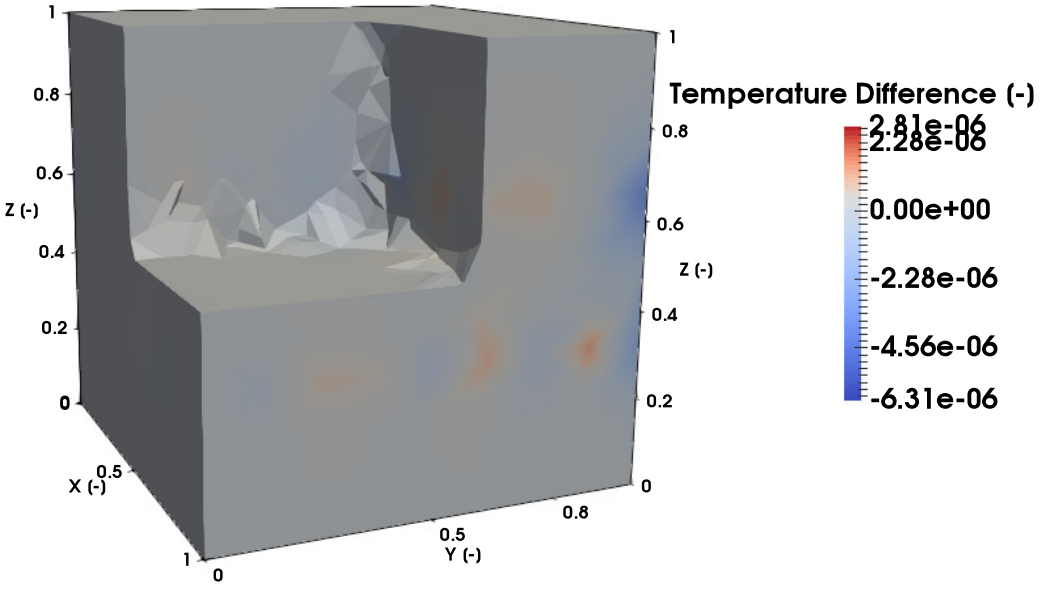


FIGURE 10.6: Temperature difference between the full and reduced model for the Base model.

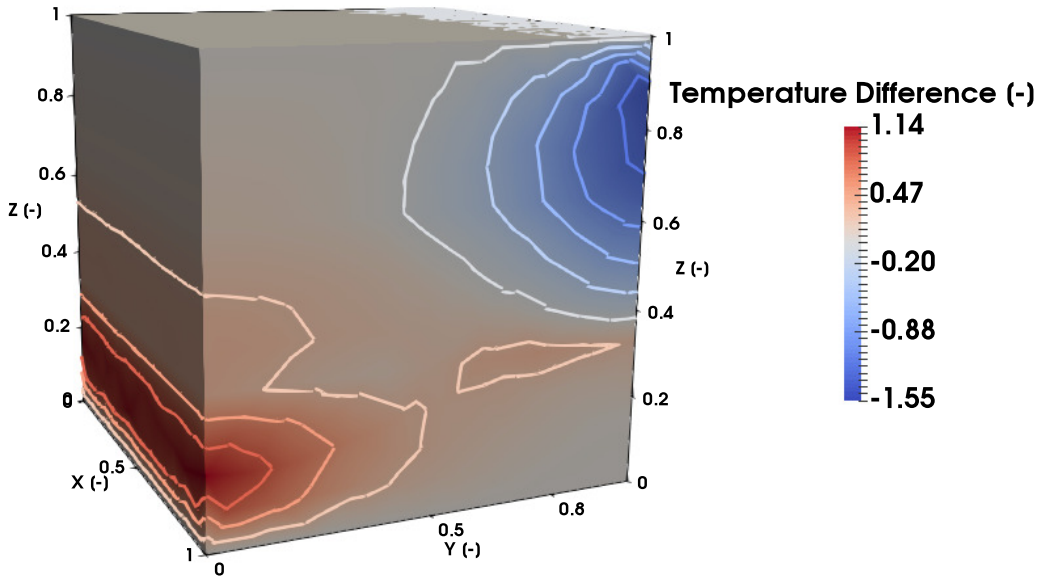


FIGURE 10.7: Temperature difference between the conductive simulation using the fitted effective conductivities and the hydrothermal simulation for the Base model.

10.1.4 Discussion

Using the entropy production to determine an effective conductivity that can be used to capture the behavior of convective models inside conductive models seems very promising. Although using only simple 1D exponential and polynomial functions, we are able to reduce the conductive model errors significantly. Furthermore, we already obtain the basic shape of the convection cell. The main reason why the 1D approximation works so well is the simplicity of the model. For more complex models, we need to use 2D or 3D approximations. Especially, promising is the error reduction from using a single parameter to using a space-dependent conductivity. The fits presented here are a first simple attempt to characterize the entropy. We expect a further reduction of the error with more complex functions.

Note that the full hydrothermal finite element simulation requires one min, whereas the reduced simulation requires only 1 ms, which is a speed-up of $6{\cdot}10^4$. This speed-up is three orders of magnitude higher than the one from the purely conductive model presented in Section 3.2.1. We have seen that the speed-up of the conductive models increases by several orders of magnitude for basin-scale models. If the speed-up for our hydrothermal studies behaves similarly, which is to be expected since the governing equation is similar, we can expect for real-case applications speed-ups of the order of 10^6 or higher. This would allow solving inverse problems for hydrothermal applications.

Model calibrations could be used to improve the fit both to the original hydrothermal model and the observations. This could be done, as here suggested, by allowing a scaling factor for the amplitude of the exponential function and the coefficients of polynomial functions. Note that it is also possible to have a different scaling factor for each of these coefficients. Hence, we do not need to keep the ratio between them fixed.

We intended to introduce the effective conductivity in this section. Already with the current fitting, the method is of direct use. The newly corrected models are not able to fully capture the complexity of the hydrothermal simulation, but they give a first good impression of the model, which we can use for more detailed analyses. In the future, we want to extend the concept by using better fits and additional kernels that can be used to compensate for model errors.

However, we need to think about ways to generalize the concept. The shape of the convection cell is parameter dependent. Depending on the chosen parameters, the number of convection cells can vary, something that we are currently not accounting for. One way to address this challenge is to use geometrical parameters. Additionally, it is possible to add Gauss functions to introduce additional heat sources that are able to represent sources of high entropy production.

Also note that the errors are likely smaller for real-case models since the sediment thickness of the convective layer in relation to the other layers is significantly smaller. However, this has to be tested in future studies.

10.2 Electrical Resistivity Tomography

So far, we applied the RB method to geothermal problems concerning both the conductive heat transfer and the groundwater flow. However, the RB method is also attractive for other geophysical methods. To demonstrate that we show in the following how the RB method can be applied for electrical resistivity tomography (ERT) applications to allow the usage of, for instance, global sensitivity analyses. We focus mainly on the numerical aspects of this application. The physical interpretation will be discussed in

von Bülow et al. (2020) in detail. This chapter contains the preparational work for von Bülow et al. (2020).

10.2.1 Governing Equations

In this section, we briefly introduce the concepts of electrical resistivity tomography (ERT). ERT, also called DC Electromagnetics, belongs to the geoelectric methods. All of them use the natural variation in the electrical properties of the subsurface to determine subsurface structures (Lowrie, 2007). Due to the high contrast in electrical resistivity between water and sediment, ERT is often used for the detection or monitoring of groundwater (Brunet et al., 2010; Lowrie, 2007; Michot et al., 2003; Zhou et al., 2001). Furthermore, it is used in archaeological applications and for the monitoring of landslides (Leucci and Greco, 2012; Lowrie, 2007; Perrone et al., 2014).

ERT applications are interested in the resistivity ρ or the electrical conductivity ($\frac{1}{\rho}$) of the individual subsurface layers. The resistance R can be determined using Ohm's law (Lowrie, 2007):

$$U = RI, \tag{10.11}$$

where U is the potential, and I the current. The resistance itself is not a material property only the resistivity. The resistance itself is dependent on the volume of the body, the intrinsic material property is called resistivity (Lowrie, 2007):

$$\rho = R\frac{A}{l}. \tag{10.12}$$

Here, l is the length and A the cross-sectional area of the sample. Both active and passive geoelectrical surveys are conducted in practice. We consider in this thesis only the active case. Hence, we introduce only the concepts of the active geoelectrical survey. For information regarding the passive set-up, we refer to Lowrie (2007). In ERT commonly a four-electrode configuration is used, where two electrodes are used to inject the current in the subsurface (current electrodes), and two electrodes are used two measure the potential (potential electrodes). Thus, each electrode pair builds up an electrical dipole. ERT surveys do not only use four electrodes but an array of electrodes to explore the area of interest. Different depth levels can be investigated by changing the offset of the electrodes. The injected current and the measured potential difference is then used to determine the apparent resistivity ρ_a (Lowrie, 2007):

$$\rho_a = 2\pi\frac{\Delta U}{I}\left[\left(\frac{1}{r_{AC}} - \frac{1}{r_{CB}}\right) - \left(\frac{1}{r_{AD}} - \frac{1}{r_{DB}}\right)\right]^{-1}. \tag{10.13}$$

Here, r denotes the distance between two electrodes, and the subscripts A, and B the position of the current electrodes, and the subscripts C, and D the position of the potential electrodes. The most common four electrode configurations are the Dipole-Dipole, the Wenner, and the Schlumberger configuration. The Dipole-Dipole configuration keeps the two Dipoles separated from each other with a fixed offset. The Wenner configuration keeps the same distance between all electrodes. We can combine this in modern multi-channel measurement systems with electrical sounding elements. Geoelectrical sounding is traditionally used to receive 1D-profiles of the subsurface by keeping the current electrodes fixed at a position while the potential electrodes are placed outside. During the measurements, the offset increases. In this thesis, we focus on the electrical sounding.

We restrict the introduction to the forward problem. In this thesis, we do not consider the inverse process to determine the geometry of the model. For information regarding the inversion of ERT data, we refer to (Günther et al., 2006).

The ERT forward problem is a diffusion dominated problem, where we solve for the potential U (Rücker et al., 2006):

$$\nabla \cdot (\sigma \nabla U) = -\nabla \cdot j \text{ in } \Omega \subset \mathbb{R}^3, \qquad \text{with } j = -\sigma \nabla U. \tag{10.14}$$

Here, σ is the electrical conductivity. The current density j is obtained using Maxwell ($E = -\nabla U$, with E being the electrical field). We consider point electrodes. Consequently, the current density at the source position is described as a Dirac function (Rücker et al., 2006):

$$\nabla \cdot j = I \cdot \delta \left(r - r_s \right), \tag{10.15}$$

where r is the current position, and r_s the distance to the source position. Choosing the following nondimensional parameters and variables:

$$\begin{aligned} \nabla^* &= \frac{\nabla}{l_{\text{ref}}}, \\ U^* &= \frac{U}{U_{\text{ref}}}, \\ \sigma^* &= \frac{\sigma}{\sigma_{\text{ref}}}, \\ I^* &= \frac{I}{I_{\text{ref}}}, \\ \\ r^* &= \frac{r}{l_{\text{ref}}}, \end{aligned} \tag{10.16}$$

yields the following nondimensional equation:

$$\sigma^* (\nabla^*)^2 U^* = -\frac{I^* I_{\text{ref}}}{l_{\text{ref}}\, \sigma_{\text{ref}}\, U_{\text{ref}}} \delta(r^* - r_s^*), \tag{10.17}$$

with the reference current I_{ref}, length l_{ref} conductivity σ_{ref}, and potential U_{ref}.

10.2.2 Trench Model

In order to demonstrate how the RB method can be applied to Electrical Resistivity Tomography (ERT), we use the trench model (Fig. 10.8) constructed after a trench behind the EonERC building (Mathieustr. 6 in Aachen). The model has been generated by von Bülow (2020) using the meshing software Gmsh. It is a three-dimensional unstructured tetrahedron mesh with 8,496 nodes and 41,124 elements. In the x-direction the model extents 300 m, in the y-direction 50 m, and -40 m in the z-direction. The area of interest extends 4 m in the x-direction, and 1 m in the y-direction. The model itself is much larger since the ERT approach assumes an infinite layer. To satisfy this assumption the models are significantly extended in each direction to avoid any interference from the boundary conditions during the forward calculation. The parameter mesh for the inversion contains coarser mesh cells and exists only in the area of data points. This additional part of the model is called “world”.

The area of interest is further subdivided into three geological units: the top layer, the finesand layer, and the graven layer. Both the top and the finesand layers have a thickness of 0.3 m, whereas the graven layer has a thickness of 0.7 m. Table 10.2 lists

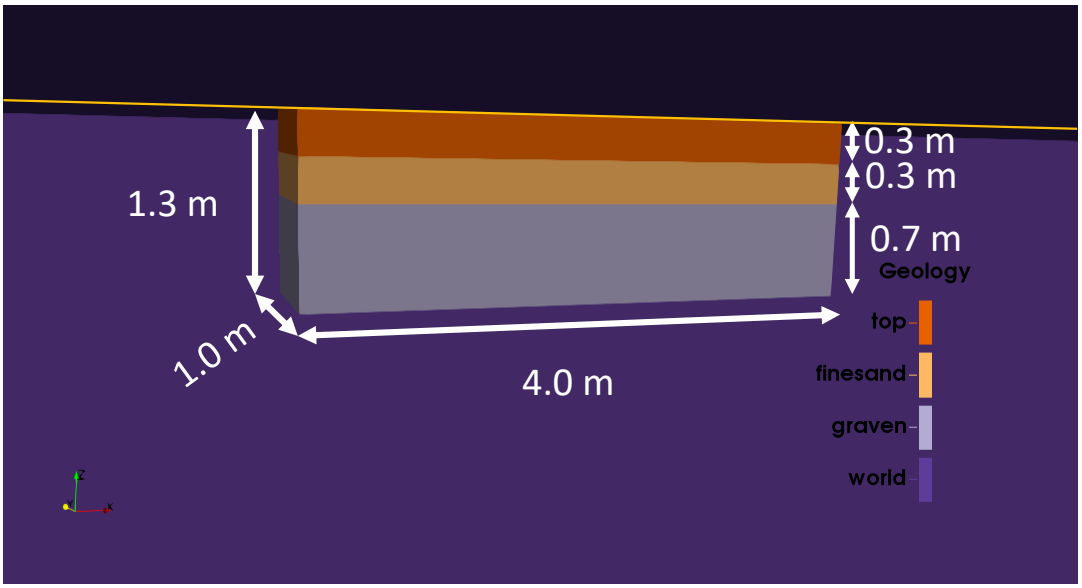


FIGURE 10.8: Zoom into the Trench model with the focus on the area of interest which consists of three geological units: the top layer, the finesand layer, and the graven layer. The yellow line indicates the electrode positions. For the exact electrode positions, we refer to Appendix D.

the electrical properties and their variation range. We apply a homogeneous (zero) Neumann boundary condition at the top of the model domain, and for all other boundaries, we use homogeneous Dirichlet boundary conditions.

We perform an ERT sounding with a spacing of 0.2 m and a current of 1 A, which results in varying source and potential electrode positions. In total, 40 electrodes are involved along a 2D-profile. The profile starts 1.9 m before the trench, crosses it over the complete length of 4 m and ends 1.9 m after its end. A list of all source positions and their respective potential electrode positions are presented in Appendix D.

We choose a reference electrical conductivity of $\frac{1}{280}\ \Omega^{-1}\ \mathrm{m}^{-1}$, corresponding to the highest resistivity within the model. The reference length is 300 m, which is equal to the maximum spatial extent of the model, and a reference potential of 10 V to obtain potential values at the potential electrodes that do not significantly exceed 1. Additionally, we set the reference current to 1 A, corresponding to the injected current.

For both the FE and RB simulations, we use a linear and non-linear solver tolerance of 10^{-8} and a Newton solver. The preconditioner is the same as for the Base and Fault model (Section 3.1.1). For the RB simulations, we allow that all four electrical resistivities can be varied. We set the maximum number of basis functions to 100 and the size of the training set to 300. For the relative error tolerance, we use a value of 10^{-4}, which is adjusted to the measurement error. The measurement accuracy of ERT measurements in the field is at most up to a percent accurate. With the chosen error tolerance, we are still one order of magnitude more accurate than the measurements (keep in mind that the potential reference is 10 V). Through this, we ensure that our full and reduced model are equivalent. The affine decomposition is analog to the decomposition presented in Section 2.1.3. Note that in case of the Trench model, Q_a is equal to four and that the parameter-dependent part is $\frac{1}{\rho}$ instead of λ.

TABLE 10.2: Electrical properties that are varied for further analyses for the Trench model. The initial parameters are taken from von Bülow (2020).

unit	training parameter	initial electrical resisitivty $[\Omega m]$	lower bound $[\Omega m]$	upper bound $[\Omega m]$
top	0	15	7.5	22.5
finesand	1	280	140	420
graven	2	50	25	75
world	3	50	25	75

10.2.3 Results

The purpose of this section is to demonstrate how the RB method can be used for ERT applications. Our focus is on the numerical aspects of this study and not on the physical aspects. In the following, we present how the RB method has to be used for the ERT forward simulations and how the reduced solutions can be used in inverse processes. For this purpose, we perform a global sensitivity study, where we slightly simplify the quantity of interest to demonstrate the working flow better. The physical aspects of this study will be presented and discussed in von Bülow et al. (2020), later on.

The current mesh is too coarsely meshed around the source positions, as shown in von Bülow (2020). This has the consequence that the steep gradient caused by the two source electrodes (represented as Dirac functions) cannot be fully captured. Hence, we underestimate the electrical potential close to the source positions. The electrical potential in the far-field is, however, well approximated with the coarse discretization (von Bülow, 2020). We purposely use a coarse mesh to ensure both fast FE solutions and fast construction times of the reduced model. This is fine in our case since we are interested in the numerical aspect. For a physical study, this would not be sufficient. A study using a finer mesh and a detailed physical interpretation of the shown model is presented in von Bülow et al. (2020).

We perform a global sensitivity study to investigate the influence of the different electrical resistivities on the potential distribution at the potential electrodes. Note that this study is meant to demonstrate how the RB method can be used in the context of global sensitivity studies, and other inverse processes, for ERT applications. For this reason, we slightly simplify the quantity of interest. We investigate the sensitivities of the resistivities for the individual source positions with respect to the sum of the absolute potential value at the potential electrodes. During the measurement routine, each source position is fixed while we measure multiple current positions. The different source positions have differing numbers of potential electrodes associated with it. In order to avoid that the source positions with more potential electrodes get a higher weight in the total sum, we divide the sum of the absolute potential values by the number of potential electrodes. In ERT applications, it would make more sense to take the potential differences between the potential electrodes instead of the absolute

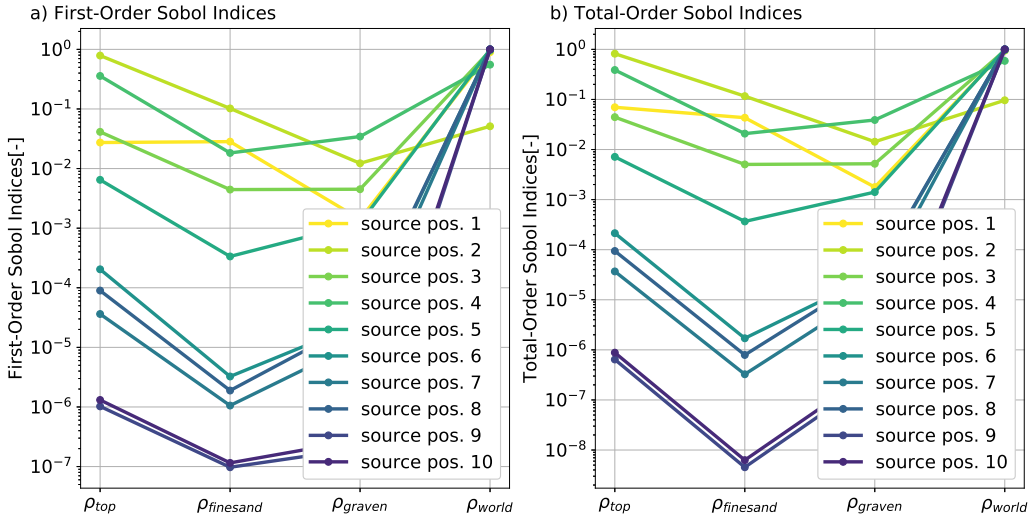


FIGURE 10.9: Results for the global sensitivity analysis for all source positions individually. Shown are on the left side the first-order indices and on the right side the total-order indices.

values. This will be done in von Bülow (2020). Here, we focus on the aspect of the RB method and keep the physical side as simple as possible to better illustrate the combination of the RB method and ERT applications.

If we look at the first- and total order Sobol indices of the different source positions (Fig. 10.9), we see some general patterns:

1. The first- and total order influence of the resistivity of the world is the highest one for all source positions except the second one. For the second source position, the highest first- and total order contribution arises from the resistivity of the top layer.

2. The resistivity of the top layer causes the second highest total-order contribution for all source positions except the second one. Here, the second highest influence arises from the finesand layer. The same pattern is visible for the first-order contribution with the difference that for the first source position, the contribution from the top layer is slightly lower than the one from the finesand layer.

3. For the first source positions the finesand layer has higher first- and total order contributions as the graven layer. For the latter source positions, the graven layer has a stronger influence.

4. For the latter source positions, we observe nearly only influences from the resistivity of the world.

For the average over all source positions (Fig. 10.10), we obtain a clear ranking, for both the first- and total-order sensitivity indices: $\rho_{\text{world}} > \rho_{\text{top}} > \rho_{\text{finesand}} > \rho_{\text{graven}}$. Also, note that we observe nearly exclusive first-order contributions.

For the ERT sounding, we have in total of ten different positions for the source electrode pair. We construct a reduced model for each of these scenarios. Figure 10.11 presents the convergence of the maximum relative error bound for all source positions. We see that for all source positions, we reach the error tolerance after seven to seventeen basis functions. The trend of all convergences lines is the same. However, we observe that the convergence is the slowest for the second source position and the fastest for the fourth source position. The FE simulations require between 5 s and

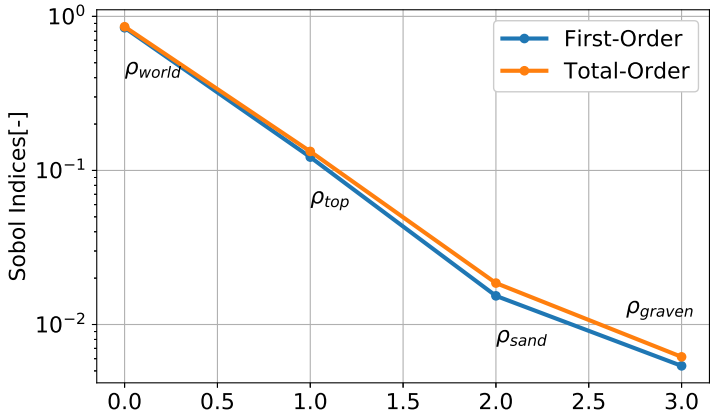


FIGURE 10.10: Results for the global sensitivity analysis over all source positions. Shown are the first-order indices (orange) and the total-order indices (blue).

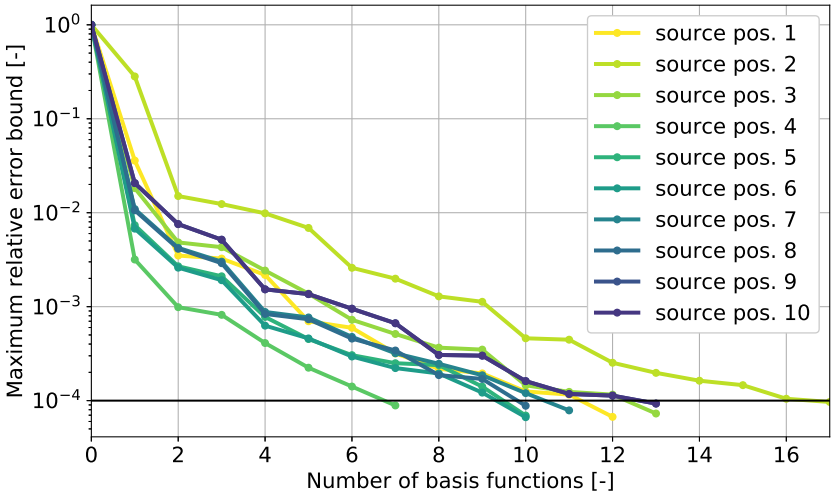


FIGURE 10.11: Convergence of the maximum relative error bound for all source positions with a relative error tolerance of 10^{-4}.

6 s, whereas the online times range between 2 ms and 3 ms. Consequently, we obtain speed-ups between $1.7{\cdot}10^3$ and $2.7{\cdot}10^3$. The construction of the reduced basis requires between 13 s and 18 s. This means that we require three to four simulations in order to be more efficient with the RB method than with the FE method.

10.2.4 Discussion

We observed that for most source electrode pair positions, the highest influence arises from the resistivity of the world. The enormous spatial extent of this layer is responsible for this. For the second source position, the top layer has the highest influence. If we look at the potential electrode positions for this source location, we see that it has the highest number of potential electrodes in the trench in relation to the total number of potential electrodes. All electrodes are positioned at the surface. Consequently, most of the current is transported through the top layer. Hence, we get for the second source location the strong influence from the top layer resistivity. This is also the reason why the top layer resistivity causes the second biggest influence on the remaining source locations.

Similarly, we obtain generally higher influences from the finesand than the graven layer for the earlier source positions, although the graven layer has more than twice the sediment thickness of the finesand layer because the finesand layer is closer to the electrodes. The latter source positions show the reverse behavior. For these source positions, we have more potential electrodes outside the trench or at the border of the trench. Note that both sensitivities are extremely low. For the latter source positions less current is penetrating through the trench. Hence, in combination with the higher sediment thickness, this leads to the stronger influence of the graven layer.

The latter source locations are placed further and further away from the trench, and the top, finesand and graven layer are only presented within the trench. This is the reason why the current nearly only penetrates through the world, meaning we obtain in the sensitivity analysis mostly influences from the resistivity of the world.

We have also seen that we have nearly exclusively first-order contributions. Hence, we have no significant correlations between the different resistivities.

Although the mesh dimension is relatively small and the FE simulation with 5 s to 6 s very fast, we get speed-ups of three orders of magnitude. This demonstrates the potential of the RB method for ERT applications, especially since much larger meshes are expected for 3D ERT applications as we tested it here. Therefore, the speed-ups should increase for these larger meshes because we expect that the increase in parameter complexity is much smaller than the increase in the mesh dimension. This will become especially important for the comparison of the Dirac function and the potential split. Even if the Dirac function requires way larger meshes, we could consider this approach if it yields more accurate results because we can compensate for the higher simulation times by using the RB method.

We need very few basis functions to approximate the full solution yielding small offline stage execution times. The fast offline stages make the RB method for already four simulations more efficient than the FE method. Consequently, already for deterministic inverse processes, it makes sense to use the RB solution as a surrogate model. Keep in mind that the global sensitivity study with 1,000,000 function evaluations requires 2 min compute time with the RB method, whereas the same study using the FE method would take more than 60 days. This demonstrates that the RB method is also a suitable surrogate model for stochastic ERT inversions.

In contrast to Rücker et al. (2006), we use a Dirac function to represent the source/sink terms (as formulated in the PDE). Rücker et al. (2006) perform instead a splitting of the potential since a Dirac function is numerically harder to stabilize. The Dirac function has the disadvantage that it needs very fine elements close to the injection points to ensure that we capture the steep gradient. In the future, we need to make a cost-efficiency comparison of the two approaches to analyze, which method is best suited for the forward simulations.

We presented in this chapter how the RB method can be extended to other geophysical applications fields. We have seen throughout the entire thesis that the RB method is a very promising method for geophysical applications. Many possible extensions are possible to even better adapt the method for the needs of the geophysical community. In the following, we want to outline how we plan to extend the method in future projects.

Chapter 11

Outlook

We have seen that the RB method is an extremely promising tool for the dimension reduction of geoscientific problems, allowing us to perform inverse analyses in a way that is impossible with the current state-of-the-art. Nonetheless, the RB method has some shortcomings, which we would like to address and overcome in future projects. In the following, we describe which future studies are interesting for hydrothermal, data assimilation, electrical, and electromagnetic applications.

First, we want to discuss four aspects that are of general interest for all these applications:

1. In this thesis, we never varied the geometry of the model. However, the reduced basis method is able to account also for geometrical parameters (Hesthaven et al., 2016; Rozza et al., 2007). By using these parameters, it is possible to change, for instance, the depth of our geological layers and their shapes. This offers a way to include also structural uncertainties in the prediction of the state variables. Hence, we could investigate, in detail, the effects of structural uncertainties in combination with the rock property uncertainties on the state variables. Using the RB method, instead of the FE method, has the advantage that we avoid re-meshing. That this is possible, for geoscientific applications, has been demonstrated by Zlotnik et al. (2015), who varied the shape of a salt dome using the proper general decomposition method. Geometrical parameterizations are currently investigated and implemented for the DwarfElephant package, using a medical application, by Vaidya (2020).

   We follow a first attempt to investigate the effects of structural uncertainties for groundwater problems in Jaite (2019). Here, Jaite (2019) investigates these influences by generating probabilistic structural models in GemPy, translating them into unstructured meshes using CGAL (The CGAL Project, 2019), and using them as a basis for finite element Darcy flow simulations within DwarfElephant. This work also serves as a foundation to link GemPy to DwarfElephant, allowing a full consideration of the geoscientific simulation cycle.

2. Geoscientific applications have in general high-dimensional parameter spaces, especially if we take more and more coupled processes into account. We would like to exploit how either or both domain decomposition methods (Beattie et al., 2006; Liao and Willcox, 2015; Toselli and Widlund, 2006) and the hp RB method (Eftang et al., 2010, 2011) can be used to overcome the issue. The general idea of the hp RB method is that instead of constructing one reduced basis, the parameter space is hierarchically split into smaller subdomains. Hence, we end up with several small RB spaces instead of a single large RB space (Eftang et al., 2010) yielding faster online execution times. The method is presented in case of the RB method for the elliptic case in Eftang et al. (2010), and the parabolic

case in Eftang et al. (2011). Furthermore, it has been exploited for the empirical interpolation (EIM) method in Eftang and Stamm (2012). The method is currently exploited and implemented for the EIM method for a medical application, in Vaidya (2020).

3. Another interesting aspect would be to investigate the possibility of multi-fidelity methods (Peherstorfer et al., 2018) for basin-scale geoscientific applications. These methods allow combining low fidelity, such as the here presented RB models, with high fidelity models, such as coupled hydrothermal applications (Peherstorfer et al., 2018). This would be extremely beneficial, for hydrothermal, and other hyperbolic studies where we cannot use the RB method, in the here presented format.

4. One critical question in Geosciences is always how to reduce uncertainties. So, in the end, we need to solve the question of where to place additional measurements to reduce the uncertainties obtained by previous studies. These measurements can be tremendously expensive. This is the reason why we want to use design of experiment methods to determine these additional measurement points. In design of experiment methods, we distinguish between deterministic methods, such as the central composite, Box-Behnken, and full/fractional factorial design, and probabilistic methods as, for instance, orthogonal array, Monte Carlo (random), and Latin hypercube sampling (Adams et al., 2017). Under the name optimal selection of measurements, Aretz-Nellesen et al. (2019) propose a similar concept for 3D-Var, a variational data assimilation technique. Design of experiment methods have been used by Ansari and Hughes (2016), Ansari et al. (2018), Khalajzadeh et al. (2011), and Quinao and Zarrouk (2014) for geothermal studies. We want to explore how the RB method can be most efficiently used as a surrogate model to perform the studies on a basin-scale and to allow in addition to the classical deterministic sampling methods the usage of stochastic methods.

11.1 Hydrothermal Applications

In this thesis, various synthetic and real-case applications for both the conductive heat transfer and Darcy flow problems have been presented and discussed. For geothermal applications also the convective part of the heat transfer equations plays a crucial role. Considering both the conductive and convective part of the heat transfer yields a hyperbolic PDE. For this type of PDE, no error bounds are available so far. There is some work going on regarding RB for transport problems (Brunken et al., 2019). However, these error bounds are not applicable for the heat transfer equation and are currently not offline-online decomposable, which makes them inefficient, especially for large scale problems. In the future, we would like to investigate the possibilities to use the RB method for hydrothermal applications. We like to persecute two ways: i) Using the entropy production to reduce the hyperbolic to an elliptic PDE to enable the usage of error bounds (as briefly presented in Section 10.1). ii) Replacing the error bounds by hierarchical error estimators (as presented in, for instance, Hain et al. (2018)) to use the hyperbolic PDE itself.

1. **Entropy Production:** In this thesis, we presented how the concept of entropy production can be used to overcome the issue by introducing the effective thermal conductivity (see Section 10.1.2). However, the investigation was performed

on a simple benchmark example only. Furthermore, the parameterization of the effective thermal conductivity has been restricted to a single value per geological unit or simple one-dimensional exponential and polynomial functions per unit. In further studies, we would like to include higher-order polynomials for the description of the effective thermal conductivity. This would allow us to capture the spatial variability of the entropy production better. Hence, we could improve the value of the effective thermal conductivity. We think that this would significantly reduce the errors introduced by the physical simplification of the problem. Furthermore, we would like to apply the concept also to real-case basin-scale applications, such as the Upper Rhine Graben, where conductive heat transfer is likely to occur. This would be a great advantage for regional hydrothermal investigations since more detailed physical modeling approaches could purely focus on the non-conductive areas.

2. **Hierarchical error estimators:** As mentioned, we want to explore the possibility of using hierarchical error estimators instead of error bounds to construct a reduced basis for the hyperbolic PDE. This has the advantage that we can obtain a significantly faster forward model that allows the application of inverse processes, such as automated model calibration, global sensitivity analyses, and uncertainty quantification (as presented for the conductive heat transfer). Using error estimators instead of error bounds also means that we lose the guarantee that the approximation error stays below a certain threshold. Considering that with the FE method even simple parameter studies are prohibitive (Brazell et al., 2010; Khaitan and Gupta, 2012) and that other surrogate methods such as Kriging and polynomial chaos expansion do not provide error bounds either (Miao et al., 2019; Mo et al., 2019; Navarro et al., 2018), this is for geothermal applications acceptable. We might lose a bit of accuracy through the approximation. However, due to the expected speed-up, we should be able to perform extensive parameter studies yielding a substantial improvement of the existing geothermal models.

   Nevertheless, Freymark et al. (2019) showed that the increasing number of parameters through a hydrothermal simulation makes it increasingly difficult to determine the individual model parameters. This is further intensified by the, in geothermal applications common, data sparsity. Therefore, the approach via the entropy production becomes not only from a computational aspect but also from a physical point increasingly attractive. Also, note that in case of the approach via the hierarchical error estimators, we would most likely need to combine the RB method with either domain decomposition or use the hp RB method to account for the high-dimensional parameter space.

Furthermore, it would be interesting to investigate how the upper temperature boundary condition can be, in general, better described to obtain more reliable estimates of the uncertainties. In that regard, we want to incorporate the interactions of the present climate, and consequently, replace the Dirichlet boundary condition with a Robin boundary condition. This no longer fixes the temperature to a specific value. Hence, the amount of heat available in the system is no longer fixed throughout the entire simulation. In the long term, it would also be interesting to account for paleoclimate effects. This would require a transient simulation. The higher computational cost can be compensated by using the RB method instead of standard finite element simulations. However, we remain with a major challenge. Direct paleotemperature data are not available. They have to be derived from indirect measurements. Hence,

we face the major challenge of how to provide the necessary input data for these simulations. A first idea might be to use trends in the development of the past temperature to incorporate the paleoclimate into our models.

Another interesting aspect would be to investigate the effects of the climate change. This study would focus again on the upper boundary condition. We would like to investigate what effect the changing evaporation and transpiration have on the groundwater level at basin-scale. Both parameters are uncertain. Therefore, it would be interesting to investigate both the sensitivities of the model parameters and determine the uncertainties.

We have seen that also the lower boundary condition is not optimally described. Although this does not influence the uncertainties at our target depth, it often yields non-physically high thermal conductivities of the Lithospheric Mantle. We already applied a scaling factor to the boundary to compensate for possible geometrical inaccuracies of the LAB. However, we conclude that a single scaling factor is not enough to compensate for the model errors. Therefore, it would make sense to set the temperature at the lower boundary condition not to a single value but, for instance, to a polynomial function. One idea, to determine the form of this function would be to use data assimilation methods such as 3D-Var (as presented in Aretz-Nellesen et al. (2019) for a benchmark problem). Furthermore, several geometrical parameterizations of the LAB are available for the Upper Rhine Graben model. Hence, one idea is to repeat the model calibration (either deterministic or probabilistic) and explore which description best fits the observed temperature. Another possibility is to use the current parameterization and optimize the shape in accordance with the observations by using geometrical parameters within the reduction (Hesthaven et al., 2016; Rozza et al., 2007).

11.2 Data Assimilation

For the data assimilation part, we used a modification of the Ensemble Kalman Filter to overcome the issue with the initial conditions for the offline phase. Currently, we are forced to take the mean of the updated states as the initial condition for the next forecast step. Furthermore, we need to construct a new reduced basis after each analysis step. In the future, we would like to investigate if and how the impulse response, presented in Grepl (2005), can be used to overcome these issues. The idea of the impulse response is to rewrite the initial condition as a convolution of the initial condition and a Dirac function. This means that the initial condition can be expressed as a linear combination of Dirac functions, and allows an approximation of the parameter-dependent part of the initial conditions only. For further details regarding this concept, we refer to Grepl (2005).

So far, only the RB method, in combination with sequential data assimilation, has been investigated for geoscientific problems. We would like to extend it to variational data assimilation. Sequential data assimilation methods, such as the Ensemble Kalman Filter, use only small portions of the time by considering only the time period until the next observation. Variational data assimilation is, in contrast, a continuous approach, where the entire time frame is considered. In order to obtain an estimate of the state variational data assimilation methods minimize a cost function. In our future work, we like to consider two types of variational data assimilation methods: 3D-Var and 4D-Var. The difference between the two is that 4D-Var considers in addition to the three spatial dimensions also the time, making it to an alternative of the Ensemble Kalman Filter (Navon, 2009). Variational data assimilation has been studied intensively in

numerical weather forecast by, for instance, Barker et al. (2004), Huang et al. (2009), Lorenc et al. (2000), and Rawlins et al. (2007) but is fairly unknown for groundwater modeling. We want to combine the RB method with variational data assimilation for groundwater flow, and heat transfer problems.

1. **3D-Var:** How the RB method can be used for 3D-Var data assimilation problems is presented in Aretz-Nellesen et al. (2019). So far, it has been only tested for simplified benchmark examples. We plan to use it for real-case geothermal applications to demonstrate the usefulness of the method for better determining boundary conditions. We demonstrated that geothermal applications require very deep models to avoid effects from the lower boundary condition. We have seen that both the Upper Rhine Graben and the Brandenburg model tend to get an unphysically high thermal conductivity for the Lithospheric Mantle, which is the result of geometrical inaccuracies of the LAB. To compensate for these errors, we would like to assimilate the lower boundary condition with the 3D-Var method.

2. **4D-Var:** Also, a 4D-Var algorithm using the RB method is available (Kärcher et al., 2018). Again, the method has so far only been used for simple benchmark problems. Darcy flow problems, as presented in this thesis, would greatly benefit from the method since one major issue with time-dependent problems is the determination of the initial conditions. Therefore, we would like to test the method in the context of real-case basin-scale applications.

   Additionally, it would be interesting to compare the results and the performance of the 4D-Var method with the classical Ensemble Kalman Filter.

Independent of the question of whether to use sequential or variational data assimilation, we face another issue with data assimilation. Many data assimilation applications want to estimate the initial conditions of geostatistical fields (i.e., Hendricks Franssen and Kinzelbach, 2008; Li et al., 2012; Zhou et al., 2011). For these applications, we obtain a permeability value per cell, yielding a very high-dimensional parameter space. This high-dimensional parameter space makes the current available RB-data assimilation combinations inefficient. We want to investigate if and how the method of pilot-points for groundwater applications (i.e., Alcolea et al., 2006; Certes and de Marsily, 1991; de Marsily et al., 1984; Doherty, 2003; LaVenue and Pickens, 1992; LaVenue et al., 1995; RamaRao et al., 1995) can be used to significantly reduce the parameter space in order to enable an efficient use of the RB method. This would allow a much faster data assimilation and way larger ensemble sizes. The method of pilot-points has been already used for the Ensemble Kalman Filter for finite difference simulations by Keller et al. (2019).

11.3 Electrical Resistivity Tomography and Maxwell

We briefly presented the benefits of the RB method for ERT applications in this thesis. Therefore, we used a simple synthetic model. In the future, we would like to extend this to real-case applications. Some preparatory work, as the automated stepping of source and receiver electrodes, has been realized in a student project (Aceves et al., 2019). The students compared the MOOSE MultiApp system with a conventional Python script leading to the conclusion that the MultiApp system already provides a parallelized version of the stepping. This makes the MultiApp system beneficial over the conventional Python script. Nonetheless, we need to address further issues as how

to avoid the repetitive mesh reading process. How the already existing implementation can be used for real-case ERT applications is currently investigated by von Bülow (2020). Furthermore, in the thesis of von Bülow (2020), it is planned to investigate how the FE and RB forward simulations, provided by DwarfElephant, can be used to improve the inversion of ERT data.

Building up on top of the already available ERT implementation, we would like to exploit the RB method for the Magnetotelluric and Controlled-Source Electromagnetics. Both applications are using the Maxwell equation as a basis and are hence coupled electromagnetic problems. Transferring the Maxwell equations to the frequency domain yields a parabolic PDE for which the certified RB method (meaning with error bounds) can be applied. The RB method for Maxwell problems has been already intensively studied by (Chen et al., 2010; Hammerschmidt et al., 2015; Hess and Benner, 2013a,b, 2014; Hess et al., 2015). However, it has not been studied for geophysical applications, which would greatly benefit from the reduction due to the huge computational cost associated with these problems (Siripunvaraporn, 2012).

Chapter 12

Overall Discussion

We want to review to what extent the scientific aims of this thesis have been addressed:

1. **If the reduced basis method is a suitable approach to obtain accurate simulation results while considering the full scale and complexity of geophysical models.**

   We presented in this thesis that the reduced basis method is an extremely promising technique to considerably speed-up numerical forward simulations. For the real-case basin-scale models of the Upper Rhine Graben and Brandenburg, we obtained speed-ups ranging from 10^4 to 10^6. To further emphasize this, we can reduce the compute time from tens to hundreds of core years to just a couple of core hours, while maintaining a high accuracy. The models so far have an approximation accuracy that is higher than the measurement accuracy. Consequently, the reduced and full model are, in the here presented cases, equivalently valid for the model predictions.

   Especially, the groundwater flow applications are promising since we saw that parabolic problems have several orders of magnitude larger speed-up as the elliptic applications for the same models. Furthermore, we have also seen that the RB method can be very successfully applied to other geophysical problems such as hydrothermal and ERT applications.

   Although the RB method is very efficient for the space and time reduction, it is not efficient for the reduction of the parameter space. Therefore, we are limited to a small number of parameters that we can vary because otherwise, the offline times become unfeasible. There, are various ways on how to address this issue ranging from pilot point methods (i.e., Alcolea et al., 2006; Certes and de Marsily, 1991; de Marsily et al., 1984; Doherty, 2003; LaVenue and Pickens, 1992; LaVenue et al., 1995; RamaRao et al., 1995) over domain decomposition (Beattie et al., 2006; Liao and Willcox, 2015; Toselli and Widlund, 2006) to hp refinement methods for RB (Eftang et al., 2010, 2011). It is, however, clear that we have to address this issue in future studies.

   The hydrothermal applications revealed another limitation of the RB method. Currently, error bounds are only available for elliptic and parabolic PDEs but not for hyperbolic PDEs. However, many geophysical applications such as geothermal, seismic, and electromagnetic applications are based on hyperbolic PDEs. For geothermal applications, we demonstrated in this thesis, using a simple benchmark example, how the entropy production can be used to overcome the issue. Nonetheless, further investigations and especially the transitions to real-case models are required to explore the effectiveness of this approach.

   Another possibility is to replace the error bounds by estimators. In this case, we lose our guarantee on the approximation error but are, therefore, able to

perform extensive parameter estimation studies. However, this yields a much higher dimensional parameter space, which is extremely difficult to explore due to the data sparsity of geothermal applications, as shown in Freymark et al. (2019). Taking this into account makes the entropy approach very attractive.

For both electromagnetic and seismic applications, one can first transfer the PDEs to the frequency domain and then employ the reduced basis method. This has been already demonstrated by Chen et al. (2010), Hammerschmidt et al. (2015), Hess and Benner (2013a), Hess and Benner (2013b), Hess and Benner (2014), and Hess et al. (2015) for electromagnetic problems using simple benchmark models. Again, another idea is to use error estimators instead of error bounds.

2. **How the consideration of global sensitivity studies can be used to improve automated model calibration, by, for instance, incorporating parameter correlations.**

   Global sensitivity analyses allow determining the parameter correlations, in contrast to local studies. Taking the knowledge of the parameter correlations into account significantly reduces the parameter space that needs to be investigated.

   Especially, the case study of the Upper Rhine Graben demonstrated the great benefits of incorporating global sensitivity studies into model calibration schemes. For this case study, only the combination of the global sensitivity study and the model calibration led to reliable and robust results. This allows using the scheme for a reliable identification of model errors, induced by, for instance, unaccounted physical processes. Furthermore, it allows compensating for these model errors by, for instance, improving the geological or physical model.

   Of course, the main advantage of employing the reduced basis method is the speed-up of the individual forward simulations. This makes global sensitivity studies and probabilistic model calibrations possible, which are with the standard finite element method prohibitive. However, the benefits of the method for geoscientific applications go far beyond "simply" speed. Through using the reduced basis method, we can perform far more intensive parameter studies leading to an improved understanding of our models. In that regard, we were able to demonstrate that cutting-off the models at a low depth, as commonly done in the field of geothermal simulations, yields entirely uninformative models. This has been revealed by employing a global sensitivity analysis in combination with an automated model calibration.

3. **Evaluating the feasibility of uncertainty quantification for the entire state (i.e., temperature and pressure distribution) of large-scale complex geophysical systems.**

   The reduced basis method allows us to predict the state variables, and is not, like many other surrogate models, such as Gaussian processes, polynomial chaos expansion, and response surfaces, restricted to the observation space (Baş and Boyacı, 2007; Bezerra et al., 2008; Frangos et al., 2010; Khuri and Mukhopadhyay, 2010; Miao et al., 2019; Mo et al., 2019; Myers et al., 2016; Navarro et al., 2018). Hence, we can use the method as a surrogate model within the Markov Chain Monte Carlo method. Execution times of less than five hours for a complex sedimentary basin, while maintaining a high accuracy, illustrates the feasibility of uncertainty quantification for large-scale geophysical systems. For both case studies of the Upper Rhine Graben and Brandenburg, we were able to both demonstrate the feasibility and usefulness of the concepts.

Furthermore, the usage of the RB method allowed us to construct predictive uncertainty quantification maps, which is not possible with the above mentioned surrogate models. We discuss that aspect in the following.

We further explored the following aims:

1. **If model calibrations are necessary and how they improve our geological models.**

   The case studies of both the Upper Rhine Graben and the Brandenburg model emphasized the need for model calibrations for geoscientific applications. Although we started with a good initial guess of the parameters – by using the parameter values from previous "trial and error" model calibrations – we could significantly reduce the mismatch of simulated and observed temperatures. Furthermore, automated model calibrations have, in contrast to "trial-and-error" model calibrations, the advantage of being reproducible.

2. **If it is possible to generate predictive 2D and 3D uncertainty quantification maps of basin-scale models.**

   We successfully generate predictive 2D, and 3D uncertainty quantification maps for both the Upper Rhine Graben and the Brandenburg model. In both cases, this helps to obtain a better understanding of the spatial distribution of the uncertainties. These maps are extremely valuable for geothermal exploration since they mark not only the areas with high temperatures but also high uncertainties. They also help to address the next aim.

3. **How the boundary conditions influence our model predictions.**

   We showed that applying a Dirichlet boundary condition for the top surface – also common practice – results in a significant underestimated uncertainty at typical geothermal target depths.

   Additionally, we also showed the importance of the lower boundary condition. As previously mentioned, the here presented investigations, demonstrated that reducing the spatial domain by reducing the depth of the model yields entirely boundary-driven models. This has the consequence that also the error of the lower boundary has a major influence on the overall model.

4. **Whether the RB method is suitable for sequential data assimilation problems.**

   We saw that the RB method is very well applicable to parameter estimation studies but not to sequential data assimilation. We presented a modification of the Ensemble Kalman Filter that allows applying the RB method also for sequential data assimilation efficiently. However, this modification is restricted to only certain applications. For problems, where we expect a large ensemble spread, the modification will likely lead to a significant underestimation of the ensemble spread. For applications, where we are only interested in the mean values, the modification will perform well and will, through the RB method, allow a much faster evaluation with larger ensemble sizes. In contrast to the algorithm proposed in Pagani et al. (2017), we are able to obtain state estimates. Using the impulse response approach, presented by Grepl (2005), might solve the issue, but this requires further investigations. Nonetheless, the RB method can be used for data assimilation by using variational instead of sequential data assimilation methods. This has been already presented, for benchmark problems, in Kärcher et al. (2018) and in Aretz-Nellesen et al. (2019).

Beyond the original scientific aims of this thesis, we can demonstrate the following aspects:

1. By using the empirical interpolation method, we can use the presented RB method also for non-linear problems, which is not possible with the other mentioned surrogate models (Frangos et al., 2010). How the RB method can be used for non-linear problems and that it performs superior to a direct interpolation of the output is presented in Grepl (2012). This makes the RB method to a much more flexible approach than most other surrogate models.

2. The usage of the RB method allows us not only to get a better understanding of our geological models but also provides us with a tool to incorporate more information. We saw that the upper boundary has to be further investigated and that both current and possible paleoclimate effects have to be taken into account. That would be a major challenge with the finite element method. However, we should not forget that this leads to very well-known challenges since direct paleotemperature data is not available and have to be derived from indirect measurements (Clauser, 1984; Demezhko et al., 2007; Shen and Beck, 1992; Vasseur et al., 1983).

3. Although the RB method is more difficult to set up, in comparison to a standard finite element problem, we can use the gained knowledge by having a better understanding of the structure of our problem to improve not only the reduced but also the full model. In this work, we use the RB method with clear separation of the offline and online stage. Meaning that we need to pay a priori a high cost to construct our surrogate model but are then able to perform very fast online solves. This allows, in turn, to perform even complex and probabilistic inverse methods directly in the field. Therefore, we are directly able to adjust, for instance, our measurement setup in the field.

All investigations in this thesis could be carried out because we have integrated the reduced basis method into the MOOSE Framework, by developing the DwarfElephant application, and relying on the libMesh rbOOmit package. Thanks to this integration, we can provide a high-level access to the RB method. This allows an easy interchange of different geological models, for the given PDEs.

Furthermore, it enables a simple transfer of the implementation of other MOOSE applications used in the field of Geosciences, such as Golem and Redback, due to several base classes. These base classes take care of the RB related operations such that we keep the changes between a standard finite element and the reduced basis simulation to a minimum for the user.

The DwarfElephant package is an open-source, high-performance package, having the advantage that everyone is able to use it. Additionally, the scaling behavior introduced by the MOOSE framework allows using the same code on a normal working station and on a high-performance infrastructure. This allows optimal test conditions for the various applications. Note that we use the package for geoscientific applications, but it is not restricted to these applications. Due to the given setup, it can be directly used for other application that is governed by the same PDEs, and easily extended to other PDEs, allowing a wide range of applications. As an example Vaidya (2020) is using and further developing the package for medical applications.

To conclude, the reduced basis method has great potential for geoscientific applications and has already led to promising and interesting results. It opens new application fields, especially for large-scale models. With these new applications, many new challenges arise, which we like to address in the future.

Appendix A

DwarfElephant

A.1 Introduction

For this tutorial, we are using an application, namely the DwarfElephant package, to the Multiphysics Object-Orientated Simulation Environment (MOOSE) (Alger et al., 2019; Permann et al., 2020), implementing the reduced basis (RB) method. The RB method is a model order reduction technique. Meaning the method aims at significantly reducing the dimensionality of the finite element (FE) problems. This is realized via a separation resulting in an offline and online stage. During the offline phase, all expensive pre-computations are performed allowing a fast execution of the online stage. Therefore, the online stage can be effectively used in, for instance, parameter studies, which require many forward simulations. For further details refer to Section A.2.

MOOSE, primarily developed by the Idaho National Laboratory, offers a built-in parallelization. The framework builds upon the libMesh (Kirk et al., 2006) and PETSc (Balay et al., 2017) libraries. The RB implementation mainly uses the rbOOmit package provided by libMesh. A built-in parallelization is especially advantageous since constructing parallelized problems for HPC infrastructures is often anything but trivial. For further details regarding the software packages, refer to Section A.5.

A.2 The Reduced Basis Method

Note, that the following explanation about the RB method is very brief for further details, especially regarding the mathematical background, refer to Prud'homme et al. (2002), Quarteroni et al. (2015), Hesthaven et al. (2016), and for more information about the method in a geoscientific context we refer to Degen et al. (2020c).

In general, the RB method is independent of the discretization scheme; however, since MOOSE is an FE solver, we explain the background of the RB approach by considering an FE problem that can be defined by a parametric coercive partial differential equation (PDE). We further assume throughout the entire tutorial that the PDE is affine parameter-dependent, i.e., it can be expressed as a sum of terms that are compromised of a parameter-dependent and -independent part. For a thermal conduction problem, the diffusive part of the equation is always the same, only the thermal conductivity is varying for, for instance, different rock types. That does not mean that both parts can be evaluated independently of each other. Hence, the thermal diffusion ($\nabla^2 T$) is the parameter-independent part of the PDE, whereas the thermal conductivity (λ) is the parameter-dependent part. In contrast to the Dirichlet boundary conditions, Neumann and Robin boundary conditions appear in the integral form and are generally dependent on our parameters. In our specific example, however, we consider a parameter-independent Neumann boundary condition only.

The principal idea of the RB approach is to take advantage of the decomposition by implementing an offline-online procedure. During the offline stage, all the expensive pre-computations are performed. The cost of these pre-computations is dependent on the high-dimensional FE solution because several solves of the full FE problem are required to build the low-order approximations, which will be explained in the following. Hence, it is advisable to perform the offline stage for large-scale models, where the solution of a single FE problem is already extremely time-consuming, on HPC infrastructures if available.

In the offline-stage, performed once, we solve for the RB functions. Hence, the computational cost depends on $\mathcal{N}$. In the online-stage, performed many times – for each new μ, the computational cost is of the dimension N. Considering $N \ll \mathcal{N}$, this means that the online stage has a much lower computational cost because of the largely reduced dimensionality of the problem (Prud'homme et al., 2002; Veroy and Patera, 2005; Quarteroni et al., 2015; Hesthaven et al., 2016). In case of our geothermal conduction example, we can perform the online-stage for any new thermal conductivity that falls within the range of the training set.

A.3 Problem Description

In the following we will consider a geothermal conduction problem as given in (A.1):

$$-\lambda\nabla^2 T = 0, \tag{A.1}$$

where λ is the thermal conductivity and T the temperature. In our case, we consider a steady-state problem without sources and sinks, thus ending up with an elliptic PDE.

For performance and convergence reasons we take only dimensionless parameters and variables into account. Therefore, we nondimensionalize the length by (A.2):

$$l^* = \frac{l}{l_{\text{ref}}}. \tag{A.2}$$

Throughout the entire tutorial the stars or asterisks denote the dimensionless quantities; the quantities without any asterisks refer to the dimensional quantities and the subscript "ref" denotes the reference parameter used for the nondimensionalization. Here, l_{ref} is 1000 m.
The thermal conductivity is nondimesionalized by applying (A.3):

$$\lambda^* = \frac{\lambda}{\lambda_{\text{ref}}}. \tag{A.3}$$

The reference thermal conductivity is, in our example, 1.05 W m^{-1} K^{-1} (i.e., the smallest thermal conductivity).

Finally, the nondimensionalisation of the temperature is performed with (A.4):

$$T^* = \frac{T - T_{\text{ref}}}{T_{\text{ref}}}, \tag{A.4}$$

where the reference temperature T_{ref} is taken as 10 °C. All model and simulation parameters are found in Table A.1 and A.2. The mesh is uniformly refined and has in total 1,658 nodes and 8,127 elements (Fig. A.1).

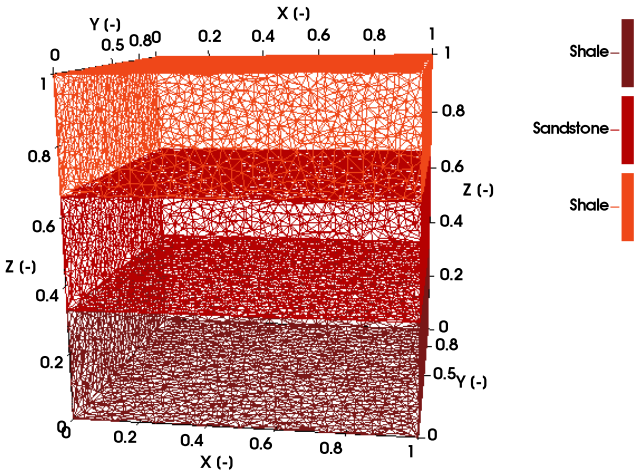


FIGURE A.1: Mesh representation of a three-layer model. The model has a Dirichlet boundary condition at the top and a Neumann boundary condition at the bottom of the model. Note, that the mesh is dimensionless for performance reasons.

TABLE A.1: Thermal and model properties for the synthetic model (Midttømme et al., 1998). Take into account that both the thermal conductivity and the length are dimensionless (λ_{ref} = 1.05 W m^{-1} K^{-1} and l_{ref} = 1,000 m).

unit	thermal conductivity [W m^{-1} K^{-1}] / thermal conductivity [-]	extension in		
		x-direction [-]	y-direction [-]	z-direction [-]
top layer (shale)	1.05 / 1.00	0.00 - 1.00	0.00 - 1.00	0.66 - 1.00
middle layer (sandstone)	2.50 / 2.38	0.00 - 1.00	0.00 - 1.00	0.33 - 0.66
bottom layer (shale)	1.05 / 1.0	0.00 - 1.00	0.00 - 1.00	0.00 - 0.33

TABLE A.2: Simulation parameters for the reduced basis method of the synthetic model.

Simulation parameter	Value
Solver type	Newton
Linear solver tolerance	10^{-8}
Nonlinear solver tolerance	10^{-8}
Preconditioner type	hypre
Hypre type	boomerang
ksp_gmres_restart	101
Rel. training tolerance	10^{-5}
Parameter range (Layer 1 and 3)	0.5 to 5.0
Parameter range (Layer 2)	0.5 to 7.0
Number of parameters	3
Number of training samples	20
Maximum number of basis functions	20

A.4 Prerequisites and First Steps

Please, be aware that throughout the entire tutorial we are assuming that the MOOSE software package is correctly installed in the directory projects/moose. For further information on how to install MOOSE refer to Alger et al. (2019).

As the first step download the DwarfElephant package from the Git Repository (`https://github.com/denise91/DwarfElephant`) into the folder projects/DwarfElephant. Afterwards compile the package with the make command. Note, if you want to use the transient RB implementation you have to ensure that the SLEPc library (Hernandez et al., 2005) is enabled during the libMesh build.

After successfully setting up the DwarfElephant package we can start with our first RB example. Therefore create an empty input file with the name FirstRBProblem.i within the folder projects/DwarfElephant. The complete input file of this tutorial is also stored in the RBFirstProblem.i file.

A.5 Implementation

In this section, we will explain the implementation of the RB method within MOOSE. In order to use this code package, it is necessary that the problem can be described as a linear PDE.

A.5.1 Mesh, Variables, and Global Params

The DwarfElephant package has no additional mesh classes. Consequently, the usage of the Mesh and MeshModifier classes does not differ from MOOSE. Hence, for further explanation regarding the Mesh refer to the MOOSE examples and tutorials. Note that for efficiency reasons for both the solver and the method, we consider only dimensionless meshes.

As for the mesh classes, the here presented package does not offer any additional variable classes. Therefore, refer to the MOOSE examples and tutorials for further information about the usage of this feature.

Analog to the mesh and variable classes, the DwarfElephant package does not add any additional features to the GlobalParam class. For further information about the global parameters please refer to the MOOSE examples and tutorials.

A.5.2 Kernels

Kernels are in MOOSE responsible for implementing the PDEs, so they define the "physics" of a problem. In contrast to the standard finite element (FE) method, the RB method needs the PDE decomposed into a parameter-dependent and independent part. Additionally, each parameter that should be variable during the online stage requires its own contribution to the stiffness matrix. Therefore, we implemented the DwarfElephantRBKernel. This Kernel class implements additionally an automatic splitting of the stiffness matrix into the mesh subdomains. However, the IDs of the individual stiffness matrices can be directly defined over the input file and hence any kind of decomposition is possible. Note that the DwarfElephantRBKernel does not implement any specific PDE. This Kernel has the same role as the Kernel provided by MOOSE. Hence, if you want to implement a PDE and want to use the RB method for the simulations you have to ensure that your own Kernel inherits from the DwarfElephantRBKernel instead of the Kernel.

A.5.3 BCs

A similar scenario as for the Kernels is encountered for the boundary conditions. If the boundaries have a parameter-dependent part an affine decomposition has to be performed. MOOSE offers two types of boundary conditions: nodal and integrated boundary conditions. Nodal boundary conditions are forced boundary conditions (i.e., Dirichlet boundary conditions), in contrast, integrated boundary conditions are boundary conditions that appear in the weak form of the PDE (i.e., Neumann boundary conditions). Be aware that due to the implementation libMesh is using, which destroys the affine decomposition for Dirichlet boundary conditions, non-zero Dirichlet boundary conditions problematic. Hence, it is important to nondimensionalize the PDE in such a way that the Dirichlet boundary condition is equal to zero. This leads also to an improved solver tolerance.

If you want to implement your own BC, it has to inherit from the DwarfElephantRBNodalBC or the DwarfElephantRBIntegratedBC class. Both of these boundary condition classes are responsible for performing the necessary RB related operations.

A.5.4 Materials

As already mentioned the RB method requires a decomposition into a parameter-dependent and independent part. Materials are in general parameter-dependent. Hence, in most of the cases, it is not useful to define a Material class for the RB method. The parameter-dependent parts of the PDE will be later defined in a separate block. Nonetheless, the package has a few Material classes, which are used for validation purposes within the FE method.

A.5.5 Problem, Executioner and UserObjects

For the RB method, the assembly procedure had to be modified from the FE version MOOSE provides. Additionally, we have to perform the solve directly over the libMesh class RBConstruction and are accordingly bypassing the NonlinearSystem class from MOOSE. Therefore, a new Problem (DwarfElephantRBProblem), System (DwarfElephantSystem) and Executioner (DwarfElephantRBExecutioner) class are implemented. Note that the Executioner of the RB Problem is the same for the steady-state and transient case because the time-stepping has to be included in the basis construction. However, this tutorial covers the steady-state case only.

To use the RB method in MOOSE two types of UserObjects are required. The UserObject DwarfElephantInitializeRBSystemSteadyState/DwarfElephantInitializeRBSystemTransient and DwarfElephantOfflineOnlineStageSteadyState/DwarfElephantOfflineOnlineStageTransient. The first UserObject, which has to be executed on initial, is responsible for initializing the RB System for the steady state and transient case, respectively. Here, all necessary parameter for the RB construction have to be defined for more information about these parameters we refer to Section A.6.7.

The second UserObject performs both the offline and online stage, where both can also be performed separately. Be aware that this UserObject has to be executed on timestep_end because for this UserObject all matrices and vectors must have been assembled. The necessary parameters are related to the storing options and the online stage, again we refer to Section A.6.7 for further details. Keep in mind that the number of parameters is problem specific and need to be defined at the beginning of each new problem.

A.5.6 PostProcessors

For the PostProcessors, we offer two ways either on the full or reduced model. In order to do the postprocessing on the full model you can use the PostProcessors provided by MOOSE but you have to execute them on the time_step 'custom'. However, performing the postprocessing on the full instead of the reduced model is not advisable in terms of efficiency. Hence, we strongly advise using the second option for the postprocessing. Keep in mind that the current implementation only supports postprocessing steps on the reduced model that are affinely decomposable (meaning linearly dependent on the variable). If you want to perform the postprocessing on the reduced model you have to decompose the operation into a parameter-dependent and -independent part (analog to the Kernel and BC). Currently, the package contains a reduced order version of the MOOSE PostProcessors ElementalVariableValue, NodalVariableValue, and PointValue. Note, that for implementation reasons these PostProcessors (DwarfElephantRBElementalVariableValue, DwarfElephantRBNodalVariableValue, and DwarfElephantRBPointValue) are defined as UserObjects. Furthermore, keep in mind that in the RB literature the post processed values are referred to as the output of interest.

A.5.7 Outputs

All MOOSE Output classes can be used in the same way for the RB and FE Problem. We just offer an additional Output class DwarfElephantDakotaOutput to output PostProcessor values in a format that is compatible to the Dakota framework (Adams et al., 2017). In case you are using the RB please method set the boolean use_rb to true.

A.6 A Geothermal Conduction Problem

In this section, we will explain the usage of the RB method within MOOSE through the above described thermal conduction problem. Note, that although we are using in this tutorial a geophysical problem the DwarfElephant package is physics independent and is therefore not restricted to geophysical problems.

A.6.1 Mesh

For our example please copy the provided mesh file (RB_mesh_3layers.e) into the DwarfElephant folder and load the file with the following command into your simulation. The mesh capabilities are entirely from the MOOSE Framework. Hence, for further information, we refer to Alger et al. (2019).

```
[Mesh]
    file = RB_mesh_3layers.e
[ ]
```

A.6.2 Variables

We define the temperature as our variable of interest since we are considering a geothermal conduction problem for our example simulation. Again, we did not add any functionality to this class and therefore we advise to visit Alger et al. (2019) for further details.

```
[Variables]
    [./temperature]
    [../]
```

```
[ ]
```

A.6.3 GlobalParams

For this example we define the variable and the initial_rb_userobject as global parameters. Also the functionality of the GlobalParams is the one from MOOSE (check Alger et al. (2019) for more information).

```
[GlobalParams]
    variable = temperature
    initial_rb_userobject = initializeRBSystem
[]
```

A.6.4 Kernels

As previously mentioned we have to separately implement the parameter-dependent and independent part. Therefore, we need to define only the diffusion part in the Kernel Block. If nothing else is defined, then the number of generated stiffness matrices corresponds to the number of subdomains. If that is not desired you have to set the boolean matrix_seperation_according_to_subdomains to false.

```
[Kernels]
    [./RBConduction]
        type = DwarfElephantRBDiffusion
    [../]
[]
```

A.6.5 BCs

The default settings to not consider any splitting in accordance with the mesh for the boundaries. In case you need to define several load vectors use the ID_Fq identifier. Per default, the ID is set to zero. In our example, we define a zero Dirichlet BC for the top and a Neumann BC for the bottom of the model. Both boundaries are stored in the load vector zero.

```
[BCs]
    [./top]
        type = DwarfElephantRBDirichletBC
        boundary = 2
        value = 0.00
    [../]

    [./bottom]
        type = DwarfElephantRBNeumannBC
        boundary = 1
        value = 3.71
    [../]
[]
```

A.6.6 Problem and Executioner

As mentioned above we need a separate Problem and Executioner class. Please define them in your input file as:

TABLE A.3: Input parameters for the RB simulation.

Parameter	Definition
N_max	maximal number of basis functions
n_training_samples	number of samples in the training set
parameter_names	name of the parameters (for reusability please use mu_0 to mu_n)
parameter_min_values	lower bound of the parameter range
parameter_max_values	upper bound of the parameter range
store_basis_functions	defines whether the basis functions are stored in the offline data folder
online_mu	definition of the online parameters

```
[Problem]
    type = DwarfElephantRBProblem
[]

[Executioner]
    type = DwarfElephantRBExecutioner
    solve_type = 'Newton'
    l_tol = 1.e-8
    nl_rel_tol = 1.e-8}
[]
```

A.6.7 UserObjects

In this section, we define the parameter-dependent part of the problem. In order to run the simulation for this tutorial ensure that in the DwarfElephantRBClassesSteadyState.h file (projects\ DwarfElephant\include\userobjects) the line under "FEProblemBase & fe_problem;" is equal to DwarfElephantRBP1T3EqualF1O1SteadyStateExpansion _rb_theta_expansion; To execute the offline-online procedure for the steady state and transient case the UserObjects DwarfElephantOfflineOnlineStageSteadyState and DwarfElephantOfflineOnlineStageTransient are needed. In this example we will use the steady state classes. The definition of the input parameters are found in Tab. A.3 :

```
[UserObjects]
    [./initializeRBSystem]
        type = DwarfElephantInitializeRBSystemSteadyState
        execute_on = 'initial'
        N_max = 20
        n_training_samples = 100
        rel_training_tolerance = 1.e-5
        parameter_names = 'mu_0 mu_1 mu_2'
        parameter_min_values = '1.0 1.0 1.0'
```

```
        parameter_max_values = '5.15 7.15 5.15'
    [../]

    [./performRBSystem]
        type = DwarfElephantOfflineOnlineStageSteadyState
        store_basis_functions = true
        online_mu = '1.05 2.5 1.05'
        execute_on = 'timestep_end'
    [../]
[]
```

Note: Since the first UserObject has the task to initialize the RB System it has to be executed on the initial time step. The other UserObject has to be executed at the end of the timestep because the matrices and vectors have to be assembled first.

A.6.8 Outputs

For the Output class the standard MOOSE classes can be used.

```
[Outputs]
    exodus = true
    execute_on = 'timestep_end'
    [./console]
        type = Console
        outlier_variable_norms = false
    [../]
[]
```

A.6.9 Run

Now, all parameters are set. Go to the folder projects/DwarfElephant/ (cd projects/DwarfElephant) and compile the code with the make command. Afterwards run the example with ./DwarfElephant-opt -i RBFirstProblem.i. The resulting temperature distribution is stored in the file RBFirstProblem_out.e and should look like Fig. A.2.

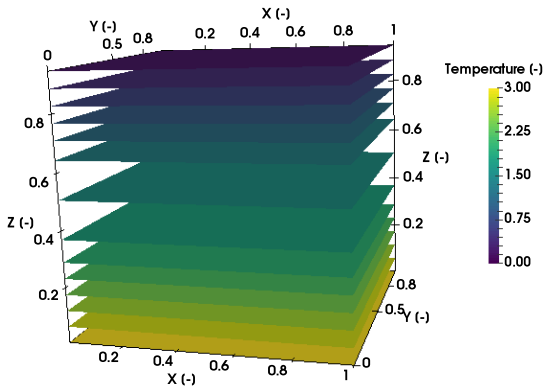


FIGURE A.2: Temperature distribution for a three-layer model. Note that both the mesh dimensions and the temperature have been normalized for numerical reasons and are hence dimensionless. The model has a reference temperature of 10 °C, a reference length of 1,000 m and solver tolerances of 10^{-8}.

Appendix B

Loose Coupling to Dakota - A Tutorial

Currently, we only support a loose coupling between the software packages MOOSE (Alger et al., 2019; Permann et al., 2020) and Dakota (Adams et al., 2017) for the finite element simulations. The focus of the DwarfElephant package is on the RB method therefore, we did not develop a direct coupling of the software for the finite element simulations. We assume for this tutorial that both the DwarfElephant package and Dakota are correctly installed.

For performing coupled analyses with the MOOSE and Dakota package we need three different files: i) the Dakota input file ii) the driver script iii) a template for the MOOSE input file. In the following tutorial, we will present all three files, assuming the usage of the MOOSE application DwarfElephant.

B.1 Prepare the Dakota inputfile

The Dakota input file has to be modified in the interface section only. The interface type has to be “fork” and the “analysis_drivers” has to be assigned to the file name of the used driver script. Then the parameter file name, in our case “params.in” and the results file, in our case “results.out” have to be defined.

B.1.1 Example File

```
# Example of the Dakota Input File

environment
  tabular_data
    tabular_data_file = 'Dakota.dat'

method
  multidim_parameter_study
    partitions = 5 5

model
  single

variables
  continuous_design = 2
   lower_bounds       2.0       2.9
   upper_bounds       4.0       3.9
   descriptors        "cond"      "flux"

interface
  analysis_drivers = 'sample_driver_script'
```

```
    fork
    parameters_file = 'params.in'
    results_file = 'results.out'

responses
  response_functions = 1
  no_gradients
  no_hessians
```

B.2 Prepare the Driver Script

B.2.1 Pre-Processig

Dakota writes the parameters, defined in the Dakota input file, into the file "params.in". These input parameters are then inserted into the marked places of the MOOSE template file ("moose.template") to generate the file moose.i, which serves as an input file for the FE forward simulations. Note that $1 denotes the params.in file from Dakota and $2 denotes the results.out file generated from DwarfElephant, and returned to Dakota. The following line is used to insert the input parameters and generate the MOOSE input file.

```
    dprepro $1 moose.template moose.i
```

"dprepro" is a function provided by Dakota to allow a faster interfacing of external software packages with Dakota.

B.2.2 Simulation

Once the MOOSE input file is generated, we need to execute the forward simulation with the given parameters. Therefore, we have to change the directory to the main folder of the DwarfElephant package, where the executable is found. Afterwards, we need to run the code from the command line. The following two lines of code are responsible for that:

```
    cd absolute/path/to/DwarfElephant
    ./DwarfElephant-opt -i /absolute/path/to/folder/containing/Dakota/
        inputfile/moose.i >> /absolute/path/to/folder/containing/Dakota/
        inputfile/console.txt 2>&1
```

Note that you have to adjust the paths to your local system. Furthermore, we write the console output of the forward simulations to file. This is an optional setting.

B.2.3 Post-Processing

Unless you want to use the file_tag option from Dakota all post-processing steps are done by the DwarfElephant package. The file_tag option is required if you want to execute Dakota in parallel.

B.2.4 Exampe File

```
# Example of the driver script for coupling the FE simulation of
# DwarfElephant with Dakota

# ----------------
# PRE-PROCESSING
```

```
#-----------------
# Pass the input parameters of Dakota, and generate the MOOSE input file.
dprepro $1 moose.template moose.i

# -----------
# SIMULATION
# -----------
# Execution of the forward simulation.

cd absolute/path/to/DwarfElephant
./DwarfElephant-opt -i /absolute/path/to/folder/containing/Dakota/
    inputfile/moose.i >> /absolute/path/to/folder/containing/Dakota/
    inputfile/console.txt 2>&1

# -----------------
# POST-PROCESSING
# -----------------
#add this line if you want to use the file_tag option
#mv name_defined_in_moose.template $2
```

B.3 Prepare the MOOSE template

In this tutorial, we consider a geothermal conduction problem with, out of simplicity, a one-layer model, where we want to vary the thermal conductivity and the heat flux. The thermal conductivity is denoted with “cond”, and the heat flux with “flux”. Note that the names insight the curled brackets have to coincide with the ones from the Dakota input file.

We return the temperature value at the observation point 1 to Dakota for further analyses. Therefore, we have to define an output class of the type “DwarfElephant-DakotaOutput”. For this class, it is necessary to define the post-processor output that has to be passed to Dakota and the path to the folder with the Dakota input file.

B.3.1 Example File

```
[Mesh]
 type = GeneratedMesh
 dim = 3
 xmin = 0
 xmax = 1
 ymin = 0
 ymax = 1
 zmin = 0
 zmax = 1
[]

[Variables]
  [./temperature]
  [../]
[]

[GlobalParams]
  variable = temperature
[]

[Kernels]
  [./Conduction]
    type = DwarfElephantFEThermalConduction
```

```
    thermal_conductivity = {x1}
    norm_value = 2.5
  [../]
[]

[BCs]
[./RBtop]
  type = DirichletBC
  boundary = 3
  value = 0.00
[../]

[./RBbottom]
  type = NeumannBC
  boundary = 1
  value = {x2}
[../]
[]

[Executioner]
  type = Steady
  solve_type = 'Newton'
  l_tol = 1.0e-8
  nl_rel_tol = 1.0e-8
[]

[Postprocessors]
  [./pressure_observation_1]
    type = PointValue
    point = '0.2 0.4 0.2'
  [../]
[]

[Outputs]
  [./exodus]
    type = Exodus
  [../]
  [./DakotaOutput]
    type = DwarfElephantDakotaOutput
    postprocessor = 'pressure_observation_1'
    execute_on = 'initial timestep_end'

    # Important replace "pathToFolderWithDakotaInputFile" with the
        relative path to the folder containing the Dakota input file
    file_base = pathToFolderWithDakotaInputFile/results
  [../]
[]
```

B.4 Execute the Analysis

First, prepare the input files according to the previous sections. Then start the Dakota input file with the following line, as you would do it for any other Dakota input file.

```
dakota -i dakota.in -o dakota.out >> dakota.stdout
```

Appendix C

Direct Coupling - A Tutorial

At the current stage, the direct coupling has been successfully used within the Python libraries SALib (Herman and Usher, 2017), SciPy (Jones et al., 2014), and PyMC (Patil et al., 2010). Furthermore, we provide a direct coupling to Dakota (Adams et al., 2017) over their Python interface. Note, that for using the Python interface in Dakota, the library has to be compiled from source code. Within this tutorial, we assume a basic understanding of the RB method. In case you want to learn more about the method itself we refer to Prud'homme et al. (2002); Veroy et al. (2003); Hesthaven et al. (2016), and for further explanations regarding the method in a geoscientific context we refer to Degen et al. (2020c).

The idea of the direct coupling is to enable an easy use of the RB model as a surrogate model within inverse processes. We perform the offline stage of the RB method within the DwarfElephant package (Degen et al., 2020c), a MOOSE application (Alger et al., 2019; Permann et al., 2020) since this stage is computationally demanding, and requires for large-scale models high-performance infrastructures. For the further tutorial, we assume that the offline stage has been already executed and that the offline data was stored in a folder "offline_data", which is placed in the current working directory. For more information regarding the generation of the offline data, we refer to Appendix A. Furthermore, we are using the same physical problem (from Appendix A) for this tutorial.

C.1 Python Libraries

The affine decomposition is for each model different, therefore the imported online_stage class contains not the model-specific decomposition. This decomposition is defined in the model class in our example. There we define the lower bound, and the parameter-dependent parts for the stiffness matrix, and the load vector.

Next, we have to define the number of A_q, F_q, and output of interests. Finally, we need to specify the online parameters. So, in our case the thermal conductivities, which we want to use during the online stage. Note, that in this tutorial we are only performing one online stage to illustrate how to use the direct coupling. In real-case applications, we would perform several of these online solves.

To execute the online stage, we have to first read the offline data. Afterwards, we can perform the online stage itself.

C.1.1 Example File

```
1 import online_stage as online
2
3 # Define the model specific decomposition
4 class model():
```

```
    def stability_lower_bound(self, online_mu_parameters):
        return min(online_mu_parameters)

    def theta_A (self,online_mu_parameters):
        return [online_mu_parameters[0]]

    def theta_F (self,online_mu_parameters):
        return [online_mu_parameters[1]]
    def theta_O (self,online_mu_parameters):
        return [1.0]

# Define the model specific input parameters for the reduced model
path_offline_data = "offline_data"
q_a = 1
q_f = 1
q_l = 1
n_outputs = 1
online_mu_parameters = np.array([2.5 3.0])

# Read the offline Data
[online_N,RB_Aq,RB_Fq, RB_Oq] = OnlineStage().read_offline_data(
    path_offline_data,
                                                    q_a, q_f, q_l,
                                                        n_outputs)

# Perform the online solve
RB_outputs = model().rb_solve_without_error_bound(online_mu_parameters,q_a
    ,
                                                q_f, q_l, n_outputs,
                                                    online_N,
                                                RB_Aq, RB_Fq, RB_Oq)
```

C.2 Dakota Coupling

For the direct coupling, over the Python interface of Dakota, to Dakota, we rely on the same Python implementation of the online stage with the additional loading of the file "theta_objects". In this file, we define the model-specific decomposition. An example is given below. Additionally, we need one extra method to allow a communication between Dakota and our Python model. This method is in our example called "Dwarf-Elephant_online_stage" it takes the arguments from Dakota as an input and with the keyword "cv" the continuous variables are accessible. In addition to the Python interfacing file, we have to specify the interface in the Dakota input file with the following lines:

```
interface
    python
        analysis_drivers = 'example_direct_coupling_dakota:
            DwarfElephant_online_stage'
            numpy
```

Note that the file "example_direct_coupling_dakota" has to be in the folder from which you start the analysis with Dakota.

C.2.1 Example File

```
import online_stage_dakota as online
import numpy as np
```

```
#**********************PART TO MODIFY**********************************#
q_a = 1
q_f = 1
q_l = 1
n_outputs = 1
path_offline_data = "offline_data"
[online_N,RB_Aq,RB_Fq, RB_Oq] = online.read_offline_data(path_offline_data,
                                                         q_a, q_f, q_l, n_outputs)
#**********************PART TO MODIFY**********************************#

def DwarfElephant_online_stage(**kwargs):
    # get online parameters
    x = kwargs['cv']

    online_mu_parameters = [x[0], x[1]]

    RB_output = online.rb_solve_without_error_bound(online_mu_parameters, q_a,
                                                    q_f, q_l, n_outputs, online_N,
                                                    RB_Aq, RB_Fq, RB_Oq)

    return RB_output
```

```
######################################################################
# Model specific
def stability_lower_bound(online_mu_parameters):
    return min(online_mu_parameters)

def theta_A (online_mu_parameters):
    return [online_mu_parameters[0]]

def theta_F (online_mu_parameters):
        return [online_mu_parameters[1]]

def theta_O (online_mu_parameters):
    return [1.0]
######################################################################
```

Appendix D

Electrode Positions

TABLE D.1: Electrode coordinates for a spacing of 0.2 m. The data is taken from von Bülow (2020).

source electrodes				potential electrodes			
ID	X [m]	Y [m]	Z [m]	ID	X [m]	Y [m]	Z [m]
0	1.4	0.0	0.0	0	0.2	0.0	0.0
1	1.6	0.0	0.0	1	0.4	0.0	0.0
2	2.0	0.0	0.0	2	0.6	0.0	0.0
3	2.4	0.0	0.0	3	0.8	0.0	0.0
4	2.6	0.0	0.0	4	1.0	0.0	0.0
5	3.4	0.0	0.0	5	1.2	0.0	0.0
6	3.0	0.0	0.0	6	1.4	0.0	0.0
7	3.2	0.0	0.0	7	1.6	0.0	0.0
8	4.6	0.0	0.0	8	1.8	0.0	0.0
9	4.2	0.0	0.0	9	2.0	0.0	0.0
10	4.8	0.0	0.0	10	2.2	0.0	0.0
11	5.6	0.0	0.0	11	2.4	0.0	0.0
12	6.2	0.0	0.0	12	2.6	0.0	0.0
13	6.4	0.0	0.0	13	2.8	0.0	0.0

TABLE D.1: Electrode coordinates for a spacing of 0.2 m. The data is taken from von Bülow (2020).

source electrodes				potential electrodes			
ID	X [m]	Y [m]	Z [m]	ID	X [m]	Y [m]	Z [m]
14	6.8	0.0	0.0	14	3.0	0.0	0.0
				15	3.2	0.0	0.0
				16	3.4	0.0	0.0
				17	3.6	0.0	0.0
				18	3.8	0.0	0.0
				19	4.0	0.0	0.0
				20	4.2	0.0	0.0
				21	4.4	0.0	0.0
				22	4.6	0.0	0.0
				23	4.8	0.0	0.0
				24	5.0	0.0	0.0
				25	5.2	0.0	0.0
				26	5.4	0.0	0.0
				27	5.6	0.0	0.0
				28	5.8	0.0	0.0
				29	6.0	0.0	0.0
				30	6.2	0.0	0.0
				31	6.4	0.0	0.0
				32	6.6	0.0	0.0

TABLE D.1: Electrode coordinates for a spacing of 0.2 m. The data is taken from von Bülow (2020).

source electrodes				potential electrodes			
ID	X [m]	Y [m]	Z [m]	ID	X [m]	Y [m]	Z [m]
				33	6.8	0.0	0.0
				34	7.0	0.0	0.0
				35	7.2	0.0	0.0
				36	7.4	0.0	0.0
				37	7.6	0.0	0.0
				38	7.8	0.0	0.0

TABLE D.2: Electrode positions for a spacing of 0.2 m. The data is taken from von Bülow (2020).

source electrodes	potential electrodes
0, 1	0-5, 8-14
2, 3	0-8, 13-21
4, 5	0-11, 13, 17-26
6, 7	3-13, 16-23
5, 8	0-15, 23-38
9, 8	9-19, 24-32
8, 10	14-21, 24-31
10, 11	12-22, 28-38
12, 13	22-29, 32-38
13, 14	20-30, 34-38

List of Figures

1.1 Illustrating the benefits of the RB method for geophysical inverse problems using the case study of the Upper Rhine Graben. 2
1.2 Mesh representation of a) the Base model consisting of three parallel layers (11, 269 degrees of freedom) and b) the Fault model consisting of three layers and two faults (23,860 degrees of freedom). Both models have a Dirichlet boundary condition at the top and a Neumann boundary condition at the bottom of the respective model. Note that the mesh is dimensionless for performance reasons. 6
1.3 Location of the case studies of the Upper Rhine Graben, Brandenburg, and the Malm Fault Zone. The letters indicate the various countries. For the Upper Rhine Graben and Brandenburg model, we show the posterior standard deviation maps at a depth of 5 km. For more details regarding these maps including the colorbar we refer to Sections 6.4.3 and 6.4.1. 7

2.1 Schematic representation of possible parameter variations: rock properties (green), boundary conditions (orange), positions of layer interfaces (blue). 11
2.2 Schematic representation of the RB method. The snapshots are denoted in green, the exact solution in red, the approximation and the solution manifold in blue and the space and error bound in black. Note, that the error bound is an approximation to the truth error. The error bound is for our examples larger than the truth error. 14
2.3 Mesh representation of the Base model consisting of three parallel layers. The model has a Dirichlet boundary condition at the top and a Neumann boundary condition at the bottom of the respective model. Note, that the mesh is dimensionless for performance reasons. 19

3.1 Mesh representation of a) the Base model consisting of three parallel layers (11, 269 degrees of freedom) and b) the Fault model consisting of three layers and two faults (23,860 degrees of freedom). Both models have a Dirichlet boundary condition at the top and a Neumann boundary condition at the bottom of the respective model. Note that the mesh is dimensionless for performance reasons. 28
3.2 Mesh representation of the Perth Basin model with 1,771,561 degrees of freedom. The model has a Dirichlet boundary condition at the top and a Neumann boundary condition at the bottom of the model. Note that the mesh is dimensionless for performance reasons. 32

3.3 Runtime comparison between the full finite element solution –denoted in blue– and the RB solution –denoted in orange– for the a) Base model and b) Fault model with permeable faults with a solver tolerance of 10^{-8}. Additionally, in b) the red dashed line denotes the FE solution and the yellow dashed line the RB solution for a tolerance of 10^{-9}. Furthermore, the blue line illustrates the speed-up with a solver tolerance of 10^{-8} and the violet dashed line the speed-up with a solver tolerance of 10^{-9}. . . . 35
3.4 Temperature distribution for a) the Base model, and b) Fault model with permeable faults. Note that both the mesh dimensions, and the temperature have been normalized for numerical reasons, and are hence dimensionless. Both models have a reference temperature of 10 °C, a reference length of 1,000 m and solver tolerances of a) 10^{-8} and b) 10^{-9}. 36
3.5 Difference plot between the full finite element and the reduced basis solution for the a) Base model and b) Fault model with permeable faults. The relative error tolerance was set for all cases to 10^{-5}, and the solver tolerances to a) 10^{-8} and b) 10^{-9}. 36
3.6 Convergence rate of the construction of the RB space for the a) Base model and b) Fault model for varying degrees of freedom and a relative error tolerance of 10^{-5} and solver tolerances of a) 10^{-8} and b) 10^{-9}. . . 36
3.7 Comparison of the runtimes for the FE approach (solid lines) and the combined offline and online times of the RB approach (dashed lines) for varying degrees of freedom for a) the Base model and b) the Fault model over 50 and 200 simulations, respectively. 37
3.8 The evolution of the maximum relative error bound at the final basis function, and the number of basis functions required for reaching a relative tolerance of 10^{-5} for the a) Base model and b) Fault model. . . 38
3.9 H_0^1 norm for the a) Base model and b) Fault model for varying degrees of freedom and a relative error tolerance of 10^{-5}. 38
3.10 Temperature distribution for the Perth Basin model. Note that both the mesh dimensions and the temperature have been normalized for numerical reasons and are hence dimensionless. The model has a reference temperature of 22 °C, a reference length of 70,000 m and solver tolerances of 10^{-8}. 39
3.11 Difference plot between the full finite element and the reduced basis solution for the Perth Basin model. The relative error tolerance was set to 10^{-3} and the solver tolerances to 10^{-8}. The L2-norm of the FE solution is $5.559 \cdot 10^4$. 39
3.12 Convergence rate of the greedy construction of the RB space for the Perth Basin model for a relative error tolerance of 10^{-3} and solver tolerances of 10^{-8} over a set of 100 training samples. 39
3.13 Performance analysis for the a) Base model (1,512,161 degrees of freedom) and b) Fault model (1,969,077 degrees of freedom). Shown are the strong scalability (purple lines) and the parallel efficiency (blue lines). The performance test has been done on Intel Xeon E5-2680 v3 Haswell cores (2.5 GHz). 42

4.1 Simplified basin-scale model for illustrating the concepts of the methods presented in this chapter. 49

4.2 We present a comparison of the local and global sensitivity analysis for the benchmark problem. The local sensitivity indices are denoted in green, the first-order contributions (global sensitivity analysis) in blue, and the total-order contributions (global sensitivity analysis) in orange. 51
4.3 Comparison of the distribution of the weights for the automated model calibration. On the left side is the distribution derived from expert knowledge, and on the right side the weighting scheme using the distance matrix. 55
4.4 Local sensitivity analysis, without weights (green), with user-defined weights (orange), and with automated weights (blue) for the reduced order model varying only the thermal conductivities. 59
4.5 We show a cross-section of the Upper Rhine Graben with the local sensitivities of the different geological layers. The local sensitivities without considering weights are denoted in black, the ones with user-defined weights in white, and the ones with automatic weights in purple. . . . 59
4.6 Local sensitivity analysis, without weights (green), with user-defined weights (orange), and with automated weights (blue) for the reduced order model varying the thermal conductivity of the Cenozoic Rift Sediments and the radiogenic heat productions of the Upper Crust. 60
4.7 Global sensitivity analysis, without weights (green), with user-defined weights (orange), and with automated weights (blue) for the reduced order model varying only the thermal conductivities. 64
4.8 We show a cross-section of the Upper Rhine Graben with the global sensitivities of the different geological layers. The global sensitivities without considering weights are denoted in black, the ones with user-defined weights in white, and the ones with automatic weights in purple. 64
4.9 Global sensitivity analysis, without weights (green), with user-defined weights (orange), and with automated weights (blue) for the reduced order model varying the thermal conductivity of the Cenozoic Rift Sediments and the radiogenic heat productions of the Upper Crust. 65
4.10 Spatial distribution of a) the user-defined weights b) the misfit in temperature for the calibration without weights c) the misfit in temperature after the calibration with user-defined weights, and d) the misfit in temperature for the initial parameter distribution. In the background of all images are the annual regional average surface temperature values plotted to outline the region of the Upper Rhine Graben. 66
4.11 Image of the geology, and the initial temperature distribution of the Upper Rhine Graben model (for the layer IDs refer to Tab. 4.3). 69
4.12 On the left side, the temperature distribution of the observed data and the initial parameter distribution are shown. The right image shows the observed data and the temperature distribution of the parameters after the trust region reflection (TRF) calibration without weights. 70
4.13 On the left side, the observed data and the temperature distribution after the trust region reflective (TRF) calibration with user-defined weights are shown. The right image shows the temperature distribution of the observed data, the initial parameter distribution, and the parameter distribution after the calibration with and without weights. Only for the temperature distribution after the calibration with user-defined weights, we observe a good agreement of the temperature values for all parts of the model. 70

4.14 Convergence of the relative maximum error bound for both reduced order models. The error bound is relative to the approximation of the "truth" (FE solution). For that reason, it is a measure of the approximation quality. 71
4.15 Representation of the fluid flow in the Upper Rhine Graben that explains the lower effective thermal conductivity for the Cenozoic Rift Sediments (modified after Freymark et al. (2019)). 75

5.1 Exploration of the parameter space for the Rosenbrock function using the conjugate gradient method. 80
5.2 Exploration of the parameter space for the Rosenbrock function using a pattern search algorithm. 81
5.3 Graphical representation of the evolutionary algorithm. 82
5.4 Exploration of the parameter space for the Rosenbrock function using an evolutionary algorithm. 83
5.5 Geology of the Brandenburg LAB model – 0: Quaternary, 1: Tertiary, 2: Upper Cretaceous, 3: Lower Cretaceous, 4: Jurassic, 5: Keuper, 6: Muschelkalk, 7: Buntsandstein, 8: Zechstein, 9: Sedimentary Rotliegend, 10: Permo-Carboniferous Volcanics, 11: Pre-permian, 12: Upper Crust, 13: Lower Crust, 14: Lithospheric Mantle. 84
5.6 Geology of the Brandenburg 6 km model – 0: Quaternary, 1: Tertiary-post-Rupelian, 2: Tertiary Rupelian-clay, 3: Tertiary-pre-Rupelian-clay, 4: Upper Cretaceous, 5: Lower Cretaceous, 6: Jurassic, 7: Keuper, 8: Muschelkalk, 9: Buntsandstein, 10: Zechstein, 11: Basement. 84
5.7 Geology of the Brandenburg combined model – 0: Quaternary, 1: Tertiary-post-Rupelian, 2: Tertiary Rupelian-clay, 3: Tertiary-pre-Rupelian-clay, 4: Upper Cretaceous, 5: Lower Cretaceous, 6: Jurassic, 7: Keuper, 8: Muschelkalk, 9: Buntsandstein, 10: Zechstein, 11: Sedimentary Rotliegend, 12: Permo-Carboniferous Volcanics, 13: Pre-permian, 14: Upper Crust, 15: Lower Crust, 16: Lithospheric Mantle. . 85
5.8 Global Sensitivity analysis for the Brandenburg LAB model. We show the first- (blue) and total-order contributions (orange). 88
5.9 Global Sensitivity analysis for the Brandenburg combined model. We show the first- (blue) and total-order contributions (orange). 89
5.10 Global Sensitivity analysis for the Brandenburg 6 km model. We show the first- (blue) and total-order contributions (orange). 89
5.11 Observed temperature distribution (blue), temperature distribution for the initial parameter distribution (orange), and after the calibration (green) for the Brandenburg LAB model. 90
5.12 Observed temperature distribution (blue), temperature distribution for the initial parameter distribution (orange) and after the calibration: with a Kriging boundary condition (green), with the boundary condition from the calibrated model (red), with the boundary condition from the uncalibrated model and a bad initial guess (purple), and with a good initial guess (brown) for the Brandenburg 6 km model. 90
5.13 Observed temperature distribution (blue), temperature distribution for the initial parameter distribution (orange), and after the calibration (green) for the Brandenburg combined model. 91
5.14 Resulting temperature difference between the initial and calibrated parameter distributions for the Brandenburg LAB model. 92

5.15 Resulting temperature difference between the initial and calibrated parameter distributions for the Brandenburg 6 km model. 92
5.16 Resulting temperature difference between the initial and calibrated parameter distributions for the Brandenburg combined model. 93
5.17 Initial temperature distribution of the Brandenburg LAB model. 93
5.18 Initial temperature distribution of the Brandenburg 6 km model. 94
5.19 Initial temperature distribution of the Brandenburg combined model. . 94
5.20 Convergence of the maximum relative error bound for the Brandenburg 6 km model (denoted in purple), the Brandenburg combined model (denoted in green), and the Brandenburg LAB model (denoted in blue). We are using an error tolerance of $1{\cdot}10^{-3}$ for the Brandenburg 6 km model, and an error tolerance of $5{\cdot}10^{-4}$ for the Brandenburg combined and LAB model. 97
5.21 Displayed are the observed temperature data (blue) and the simulated temperatures using the initial parameter distribution (orange). 98
5.22 Displayed are the observed temperature data (blue) and the simulated temperatures using the initial parameter distribution (orange), and the parameter distribution obtained by the conjugate gradient method (green). 98
5.23 Displayed are the observed temperature data (blue) and the simulated temperatures using the parameter distribution obtained by the conjugate gradient method (green), and the parameter distribution obtained by the pattern search method (purple). 99
5.24 Displayed are the simulated temperatures using the parameter distribution obtained by the conjugate gradient method (green), and the parameter distribution obtained by the pattern search method (purple). 99
5.25 Displayed are the observed temperature data (blue) and the simulated temperatures using the parameter distribution obtained by the conjugate gradient method (green), and the parameter distribution obtained by the evolutionary algorithm (black). 100
5.26 Displayed are the simulated temperatures using the parameter distribution obtained by the conjugate gradient method (green), and the parameter distribution obtained by the evolutionary algorithm(black). 100
5.27 Displayed are the temperature misfits between the simulated temperatures obtained by the conjugate gradient method (green), pattern search method (purple), evolutionary algorithm (black), the initial parameter distribution (orange) and the observed temperatures. 101

6.1 Quantile-Quantile plots for all thermal conductivities considered in the uncertainty quantification of the Brandenburg model. 110
6.2 Prior and posterior parameter distributions of all six parameters and their respective traces. 111
6.3 Prior temperature distribution of the Brandenburg model in a depth of 5 km. The yellow stars mark the positions for which we generate the Quantile-Quantile plots. 113
6.4 Quantile-Quantile plots and histograms of the temperatures for all parameters from the MCMC analysis for the Brandenburg model at nine points. 115

6.5 Black Box: Distribution of the posterior mean temperature and the posterior standard deviation of the entire Brandenburg model. Blue Box: Map of the posterior mean temperature and posterior standard deviation at the target depth of 5 km. Red Box: Prior temperature distribution and distribution of the geological layers at the target depth of 5 km. 116
6.6 Black Box: Image of the Brandenburg model. 0: Quaternary, 1: Tertiary-post-Rupelian, 2: Tertiary Rupelian-clay, 3: Tertiary-pre-Rupelian-clay, 4: Upper Cretaceous, 5: Lower Cretaceous, 6: Jurassic, 7: Keuper, 8: Muschelkalk, 9: Buntsandstein, 10: Zechstein, 11: Sedimentary Rotliegend, 12: Permo-Carboniferous Volcanics, 13: Prepermian, 14: Upper Crust, 15: Lower Crust, 16: Lithospheric Mantle. Blue Box: Prior temperature distribution for the Brandenburg model. Red Box: On the left side, the convergence of the maximum relative error bound for the entire parameter range is displayed. On the right side, the error between the full and reduced model for the prior parameters is shown. 117
6.7 Quantile-Quantile plots for all thermal conductivities considered in the uncertainty quantification of the Upper Rhine Graben. 119
6.8 Prior and posterior parameter distributions of all four parameters and their respective traces. 119
6.9 Posterior mean temperature distribution of the Upper Rhine Graben model in a depth of 5 km. The yellow stars mark the positions for which we generate the Quantile-Quantile plots at each position we take a point at 5 km, 6 km, and 7 km depth. 122
6.10 Quantile-Quantile plots and histograms of the temperatures for all parameters from the MCMC analysis for the Upper Rhine Graben model at nine points. 124
6.11 Map of the posterior mean temperature values for the entire Upper Rhine Graben model. Note that the parameters used for the generation of the posterior mean temperature map are obtained by using the MCMC method. 125
6.12 Map of the posterior standard deviation of the temperature values for the entire Upper Rhine Graben model. Note that the parameters used for the generation of the posterior standard deviation temperature map are obtained by using the MCMC method. 125
6.13 Map of the posterior mean temperature values for the Upper Rhine Graben model at a depth of 5 km below the surface. Note that the parameters used for the generation of the posterior mean temperature map are obtained by using the MCMC method. 126
6.14 Map of the posterior standard deviation of the temperature values for the Upper Rhine Graben model at a depth of 5 km below the surface. Note that the parameters used for the generation of the posterior standard deviation temperature map are obtained by using the MCMC method. 127
6.15 Map of the posterior mean temperature values for the Upper Rhine Graben model at a depth of 6 km below the surface. Note that the parameters used for the generation of the posterior mean temperature map are obtained by using the MCMC method. 128

6.16 Map of the posterior standard deviation of the temperature values for the Upper Rhine Graben model at a depth of 6 km below the surface. Note that the parameters used for the generation of the posterior standard deviation temperature map are obtained by using the MCMC method. 129
6.17 Map of the posterior mean temperature values for the Upper Rhine Graben model at a depth of 7 km below the surface. Note that the parameters used for the generation of the posterior mean temperature map are obtained by using the MCMC method. 130
6.18 Map of the posterior standard deviation of the temperature values for the Upper Rhine Graben model at a depth of 7 km below the surface. Note that the parameters used for the generation of the posterior standard deviation temperature map are obtained by using the MCMC method. 131

7.1 Exploration of the parameter space for the grid search algorithm. . . . 141
7.2 Runtime comparison between the full finite element solution –denoted in blue– and the RB solution –denoted in orange– for the Darcy flow problem using the Fault model with permeable faults with a solver tolerance of 10^{-10}. Furthermore, the blue line illustrates the speed-up. . . 143
7.3 Hydraulic head distribution at every fifth time step. Note that both the mesh dimensions and the hydraulic head have been normalized for stability reasons, and are hence dimensionless. 144
7.4 Continuation: Hydraulic head distribution at every fifth time step. Note that both the mesh dimensions and the hydraulic head have been normalized for stability reasons, and are hence dimensionless. 145
7.5 Difference plot of the hydraulic head distribution between the full finite element and the reduced basis solution for every fifth time step. 146
7.6 Continuation: Difference plot of the hydraulic head distribution between the full finite element and the reduced basis solution for every fifth time step. 147
7.7 Convergence rate of the construction of the RB space for the Darcy flow problem using the Fault model for varying degrees of freedom and a relative error tolerance of 10^{-5} and solver tolerances of 10^{-10}. 147
7.8 The evolution of the maximum relative error bound at the final basis function, and the number of basis functions required for reaching a relative tolerance of 10^{-5} by using solver tolerances of 10^{-10} for the Darcy flow problem using the Fault model. 148
7.9 Comparison of the runtimes for the FE approach (solid lines) and the combined offline and online times of the RB approach (dashed lines) for varying degrees of freedom and solver tolerances of 10^{-10} for the Darcy flow problem using the Fault model over 200 simulations. 148
7.10 Global Sensitivity Analysis for the Fault model with an initial pressure pulse. 150
7.11 Convergence rate of the construction of the RB space for the Darcy flow problem using the Fault model with 1,040,325 degrees of freedom and a relative error tolerance of 10^{-5} for the scenario with and without an initial pressure pulse. 151

7.12 Hydraulic head distribution for the Fault model with an initial pressure pulse at every fifth time step. Note that both the mesh dimensions and the hydraulic head have been normalized for stability reasons, and are hence dimensionless. 152
7.13 Continuation: Hydraulic head distribution for the Fault model with an initial pressure pulse at every fifth time step. Note that both the mesh dimensions and the hydraulic head have been normalized for stability reasons, and are hence dimensionless. 153
7.14 Difference plot of the hydraulic head distribution between the full finite element and the reduced basis solution for the Fault model with an initial pressure pulse at every fifth time step. 154
7.15 Continuation: Difference plot of the hydraulic head distribution between the full finite element and the reduced basis solution for the Fault model with an initial pressure pulse at every fifth time step. . . . 155
7.16 Comparison of the runtimes for the various coupling methods available for DwarfElephant and Dakota. The loose coupling using the FE method is denoted in blue, the loose coupling using the RB method and the postprocessors in orange, the loose coupling using the RB method and the output of interest in green, and the direct coupling using the RB method in purple. 156
7.17 Comparison of the speed-ups for the various coupling methods available for DwarfElephant and Dakota. Shown are the speed-ups of the loose coupling using the RB method and the postprocessors (orange), the loose coupling using the RB method and the output of interest (green), and the direct coupling using the RB method (purple). 157

8.1 Image of the Malm Fault Zone and the unstructured mesh constructed by Konrad et al. (2019). We use the model with a fault zone width of 20 m. 162
8.2 Global sensitivity analysis of the Malm Fault Zone Model constructed after the local sensitivity analysis performed in Konrad et al. (2019). . 164
8.3 Convergence of the maximum relative error bound for the Malm Fault Zone model. The error tolerance is set to 10^{-3}. 165

9.1 Mesh representation of the Box model consisting of two geological layers. The model has Dirichlet boundary conditions at the left and right side of the model domain. Note that the mesh is dimensionless for performance issues. 170
9.2 Distribution of the observation points for the Box model. The first set of observation points are denoted in light blue and the second in light green. 170
9.3 Hydraulic head distribution at timestep a) 0 b) 6 c) 12 d) 18 e) 24 f) 30. Note that both the mesh and the hydraulic head have been normalized for numerical reasons and are hence dimensionless. The model has a reference hydraulic head of 300 m, and a reference length of 1,000 m. 172
9.4 Difference plot between the FE and RB solution for the hydraulic head distribution at timestep a) 0 b) 6 c) 12 d) 18 e) 24 f) 30. Note that both the mesh and the hydraulic head have been normalized for numerical reasons and are hence dimensionless. The model has a reference hydraulic head of 300 m, and a reference length of 1,000 m. 173

9.5 Convergence rate of the construction of the RB space for the Box model with a relative error tolerance of 10^{-5} and solver tolerances of 10^{-8}. . . 173
9.6 Comparison of the estimated and truth hydraulic head values for ensemble sizes of 50 (green), 100 (violet), 150 (orange), and 200 (blue) over time at the observation points 2, 5, 8, and 11 using the nearby observation points and certain boundary conditions. The truth solution is denoted in black. 175
9.7 Comparison of the estimated and truth hydraulic head values for ensemble sizes of 50 (green), 100 (violet), 150 (orange), and 200 (blue) over time at the observation points 3, 6, 11, and 14 using the faraway observation points and certain boundary conditions. 176
9.8 Estimated initial conditions for the truth model (first column), the nearby observation points (second column), and the faraway observation points (third column) with certain boundary conditions at time 0.432, 0.864, 1.296, 1.278, and 2.160. 177
9.9 Difference between the estimated initial conditions and the truth initial conditions for the nearby observation points (first column), the faraway observation points (second column) at time 0.432, 0.864, 1.296, 1.278, and 2.160. At the same time steps, the difference in the hydraulic head distribution between the nearby and faraway observation points with certain boundary conditions are shown in the third column. 178
9.10 Comparison of the estimated and truth hydraulic head values for ensemble sizes of 50 (green), 100 (violet), 150 (orange), and 200 (blue) over time at the observation points 2, 5, 8, and 11 using the nearby observation points and uncertain boundary conditions. 179
9.11 Estimated initial conditions for the truth model (first column), the nearby observation points with certain boundary conditions (second column), and the nearby observation points with uncertain boundary conditions (third column) at time 0.432, 0.864, 1.296, 1.728, and 2.160.180
9.12 Difference between the estimated initial conditions and the truth initial conditions for the nearby observation points with certain boundary conditions (first column), and uncertain boundary conditions (second column) at time 0.432, 0.864, 1.296, 1.728, and 2.160. At the same time steps, the difference in the hydraulic head distribution between the nearby observation points with certain and uncertain boundary conditions is shown in the third column. Note that due to the huge error differences for the first and last time steps, we require two different colorbars. 181
9.13 Schematic representation of the Ensemble Kalman Filter. 182
9.14 Schematic representation of the modified Ensemble Kalman Filter. Note that the mean of the initial conditions is taken at the analysis step, and not later on. We moved the mean slightly to the right only for visualization purposes. 183
9.15 Comparison of the estimated and truth hydraulic head values for ensemble sizes of 50 (green), 100 (violet), 150 (orange), and 200 (blue) over time at the observation points 2, 5, 8, and 11 using the nearby observation points, certain boundary conditions, and the modified Ensemble Kalman Filter. 184

9.16 Estimated initial conditions for the truth model (first column), the nearby observation points with certain boundary conditions using the original ENKF (second column), and the nearby observation points with certain boundary conditions using the modified ENKF (third column) at time 0.432, 0.864, 1.296, 1.728, and 2.160. 186
9.17 Difference between the estimated initial conditions and the truth initial conditions for the nearby observation points with certain boundary conditions using the original ENKF (first column), and the modified ENKF (second column) at time 0.432, 0.864, 1.296, 1.728, and 2.160. At the same time steps, the difference in the hydraulic head distribution between the original and modified ENKF are shown in the third column. 187
9.18 Difference of the estimated hydraulic head distribution between the reduced and full version of the modified Ensemble Kalman Filter for an error tolerance of 10^{-5}. 188
9.19 Comparison of the estimated and truth hydraulic head values for ensemble sizes of 200 (green), 300 (violet), 400 (orange), and 500 (blue) over time at the observation points 2, 5, 8, and 11 using the nearby observation points, certain boundary conditions, and the modified Ensemble Kalman Filter. 189

10.1 Hydrothermal temperature distribution of the Base model at the last time step of the finite element solution. 196
10.2 Temperature distribution of the Base model using one space independent effective conductivity per layer. 197
10.3 Entropy distribution for the hydrothermal FE simulation of the Base model. Additionally, we show the lines (denoted in black), which we use to extract the entropy values that are later on used for the fitting of the effective conductivity. 197
10.4 Values of the effective conductivity (denoted in blue) using the extracted entropy values and their fits (denoted in orange) for a) the top shale layer b) the middle sandstone layer, and c) the bottom shale layer. 198
10.5 Temperature distribution of the Base model using a space-dependent effective conductivity. 199
10.6 Temperature difference between the full and reduced model for the Base model. 200
10.7 Temperature difference between the conductive simulation using the fitted effective conductivities and the hydrothermal simulation for the Base model. 200
10.8 Zoom into the Trench model with the focus on the area of interest which consists of three geological units: the top layer, the finesand layer, and the graven layer. The yellow line indicates the electrode positions. For the exact electrode positions, we refer to Appendix D. 204
10.9 Results for the global sensitivity analysis for all source positions individually. Shown are on the left side the first-order indices and on the right side the total-order indices. 206
10.10 Results for the global sensitivity analysis over all source positions. Shown are the first-order indices (orange) and the total-order indices (blue). 207
10.11 Convergence of the maximum relative error bound for all source positions with a relative error tolerance of 10^{-4}. 207

A.1 Mesh representation of a three-layer model. The model has a Dirichlet boundary condition at the top and a Neumann boundary condition at the bottom of the model. Note, that the mesh is dimensionless for performance reasons. xxiii
A.2 Temperature distribution for a three-layer model. Note that both the mesh dimensions and the temperature have been normalized for numerical reasons and are hence dimensionless. The model has a reference temperature of 10 °C, a reference length of 1,000 m and solver tolerances of 10^{-8}. xxx

List of Tables

3.1 Thermal and model properties for both synthetic models (Midttømme et al., 1998). Take into account that both the thermal conductivity and the length are dimensionless (with normalization factor λ_{ref} = 1.05 W m^{-1} K^{-1} and l_{ref} = 1000 m). 29
3.2 Mesh properties for both synthetic models. Listed are the refinement, the number of nodes, and the number of elements for the various meshes. We denote the parameters that are not applicable with n/a. . . 30
3.3 Simulation parameters for the finite element and the reduced basis method of both synthetic models. We denote the parameters that are not applicable with n/a. 31
3.4 Thermal properties for the Perth Basin (Wellmann and Reid, 2014). Take into account that both the thermal conductivity and the length are dimensionless (normalization factor λ_{ref} = 4.3 W m^{-1} K^{-1} and l_{ref} = 70,000 m). 33
3.5 Listed are the simulation parameters for the Perth Basin model for both the FE and the RB method. 33

4.1 The thermal properties before and after the calibration for the Upper Rhine Graben without considering the results of a sensitivity study. The initial thermal properties are taken from Freymark et al. (2017). Note that UC stands for Upper Crust, ub for upper bound, and lb for lower bound. 56
4.2 The thermal properties before and after the calibration for the Upper Rhine Graben with considering the results of the local sensitivity study. The initial thermal properties are taken from Freymark et al. (2017). We denote all parameters that are not involved in the model calibration, due to too low sensitivities, with n/a. Note that UC stands for Upper Crust, ub for upper bound, and lb for lower bound. 61
4.3 The thermal properties before and after the calibration for the Upper Rhine Graben considering the results of the global sensitivity study. The initial thermal properties are taken from Freymark et al. (2017). We denote all parameters that are not involved in the model calibration, due to too low sensitivities, with n/a. Note that UC stands for Upper Crust. 67

5.1 Thermal Properties of all models for and after the automated model calibration. The initial thermal properties are from Noack et al. (2012, 2013). We denote all parameters that are not involved in the model calibration, due to too low sensitivities or that are not applicable for the specific model version, with n/a. Additionally, the training parameter ID is provided. 95

5.2 Thermal properties of the Brandenburg combined model before and after the model calibrations. The initial thermal properties are from Noack et al. (2012, 2013). We denote all parameters that are not involved in the model calibrations, due to too low sensitivities, with n/a. Additionally, the training parameter ID is provided. 101

6.1 Thermal Properties of the Brandenburg model before and after the uncertainty quantification. The prior thermal properties are from Noack et al. (2012, 2013). We denote all parameters that are not involved in the uncertainty quantification, due to too low sensitivities, with n/a. Additionally, the affine decomposition for the model reduction is provided.111
6.2 The thermal properties before and after the uncertainty quantification for the Upper Rhine Graben. The prior thermal properties are taken from Freymark et al. (2017). We denote all parameters that are not involved in the uncertainty quantification due to too low sensitivities with n/a. Note that UC stands for Upper Crust. 120

7.1 Hydraulic and model properties for the Fault model. The hydraulic properties are taken as typical values for the respective geological units. Take into account that both the permeability, and the length are dimensionless (k_{ref} = 10^{-12} m^2, and l_{ref} = 1000 m). 142
7.2 Coordinates of the observation points for the Fault model. 150

8.1 Hydraulic properties that are varied for further analyses for the Malm Fault Zone model. The parameters are taken from Konrad et al. (2019). 163

9.1 Coordinates of the first set of observation points for the Box model. . . 171
9.2 Coordinates of the second set of observation points for the Box model. 171
9.3 Simulation parameters for the forward simulation of the Box model, which is used for the Ensemble Kalman Filter, for both the FE and the RB method. 184

10.1 Functions and corresponding coefficients for the fit of the effective conductivity for three layers of the Base model. We denote the parameters that are not applicable with n/a. 199
10.2 Electrical properties that are varied for further analyses for the Trench model. The initial parameters are taken from von Bülow (2020). 205

A.1 Thermal and model properties for the synthetic model (Midttømme et al., 1998). Take into account that both the thermal conductivity and the length are dimensionless (λ_{ref} = 1.05 W m^{-1} K^{-1} and l_{ref} = 1,000 m). xxiii
A.2 Simulation parameters for the reduced basis method of the synthetic model. xxiv
A.3 Input parameters for the RB simulation. xxix

D.1 Electrode coordinates for a spacing of 0.2 m. The data is taken from von Bülow (2020). xxxix
D.1 Electrode coordinates for a spacing of 0.2 m. The data is taken from von Bülow (2020). xl
D.1 Electrode coordinates for a spacing of 0.2 m. The data is taken from von Bülow (2020). xli

D.2 Electrode positions for a spacing of 0.2 m. The data is taken from von Bülow (2020). xli

Bibliography

Aceves, U., Gupta, P., and Montero, A. (2019). Geophysical Inversion of ERT Data. Student project, RWTH Aachen University, Aachen.

Adams, B. M., Ebeida, M., Eldred, M., Geraci, G., Jakeman, J., Maupin, K., Monschke, J., Swiler, L., Stephens, J., Vigil, D., Wildey, T., Bohnhoff, W., Dalbey, K., Eddy, J., Frye, J., Hooper, R., Hu, K., Hough, P., Khalil, M., Ridgway, E., and Rushdi, A. (2017). DAKOTA, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 6.6 User's Manual. *Sandia National Laboratories, Tech. Rep. SAND2014-4633*.

Agemar, T., Brunken, J., Jodocy, M., Schellschmidt, R., Schulz, R., and Stober, I. (2013). Untergrundtemperaturen in Baden-Württemberg [Subsurface temperatures in Baden-Württemberg, Germany.]. *Zeitschrift der Deutschen Gesellschaft für Geowissenschaften*, 164(1):49–62.

Agemar, T., Weber, J., and Schulz, R. (2014). Deep Geothermal Energy Production in Germany. *Energies*, 7(7):4397–4416.

Alcolea, A., Carrera, J., and Medina, A. (2006). Pilot points method incorporating prior information for solving the groundwater flow inverse problem. *Advances in water resources*, 29(11):1678–1689.

Alger, B., Andrš, D., Carlsen, R. W., Gaston, D. R., Kong, F., Lindsay, A. D., Miller, J. M., Permann, C. J., Peterson, J. W., Slaughter, A. E., and Stogner, R. (2019). MOOSE Web page. `https://mooseframework.org`.

Ansari, E. and Hughes, R. (2016). Response surface method for assessing energy production from geopressured geothermal reservoirs. *Geothermal Energy*, 4(1):15.

Ansari, E., Hughes, R., and White, C. D. (2018). Modeling a new design for extracting energy from geopressured geothermal reservoirs. *Geothermics*, 71:339–356.

Antoulas, A. C. (2005). *Approximation of large-scale dynamical systems*, volume 6. Siam.

Aretz-Nellesen, N., Grepl, M. A., and Veroy, K. (2019). 3D-VAR for Parametrized Partial Differential Equations: A Certified Reduced Basis Approach. *arXiv preprint arXiv:1905.05817*.

Ayachit, U. (2015). *The ParaView Guide: A Parallel Visualization Application*. Kitware, Inc.

Balay, S., Abhyankar, S., Adams, M. F., Brown, J., Brune, P., Buschelman, K., Dalcin, L., Eijkhout, V., Gropp, W. D., Kaushik, D., Knepley, M. G., May, D. A., McInnes, L. C., Rupp, K., Smith, B. F., Zampini, S., Zhang, H., and Zhang, H. (2017). PETSc Web page. `http://www.mcs.anl.gov/petsc`.

Ballarin, F., Sartori, A., and Rozza, G. (2017). RBniCS-reduced order modelling in FEniCS. `http://mathlab.sissa.it/rbnics`.

Barker, D. M., Huang, W., Guo, Y.-R., Bourgeois, A., and Xiao, Q. (2004). A three-dimensional variational data assimilation system for MM5: Implementation and initial results. *Monthly Weather Review*, 132(4):897–914.

Baroni, G. and Tarantola, S. (2014). A General Probabilistic Framework for uncertainty and global sensitivity analysis of deterministic models: A hydrological case study. *Environmental Modelling & Software*, 51:26–34.

Barrault, M., Maday, Y., Nguyen, N. C., and Patera, A. T. (2004). An empirical interpolation method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672.

Baş, D. and Boyacı, I. H. (2007). Modeling and optimization I: Usability of response surface methodology. *Journal of food engineering*, 78(3):836–845.

Bauer, J. F., Meier, S., and Philipp, S. L. (2015). Architecture, fracture system, mechanical properties and permeability structure of a fault zone in Lower Triassic sandstone, Upper Rhine Graben. *Tectonophysics*, 647:132–145.

Bayer, U., Scheck, M., and Koehler, M. (1997). Modeling of the 3D thermal field in the northeast German basin. *Geologische Rundschau*, 86(2):241–251.

Beattie, C. A., Borggaard, J., Gugercin, S., and Iliescu, T. (2006). A domain decomposition approach to POD. In *Proceedings of the 45th IEEE Conference on Decision and Control*, pages 6750–6756. IEEE.

Bejan, A. (1979). A Study of Entropy Generation in Fundamental Convective Heat Transfer. *Journal of Heat Transfer*, 101(4):718–725.

Bejan, A. (2013). *Convection Heat Transfer*. John wiley & sons.

Békési, E., Lenkey, L., Limberger, J., Porkoláb, K., Balázs, A., Bonté, D., Vrijlandt, M., Horváth, F., Cloetingh, S., and van Wees, J.-D. (2018). Subsurface temperature model of the Hungarian part of the Pannonian Basin. *Global and planetary change*, 171:48–64.

Benner, P., Gugercin, S., and Willcox, K. (2015). A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems. *SIAM review*, 57(4):483–531.

Bennett, A. F. (1992). *Inverse methods in physical oceanography*. Cambridge university press.

Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., Turian, J., Warde-Farley, D., and Bengio, Y. (2010). Theano: a CPU and GPU math expression compiler. In *Proceedings of the Python for scientific computing conference (SciPy)*, volume 4. Austin, TX.

Bertsimas, D. and Tsitsiklis, J. N. (1997). *Introduction to Linear Optimization*, volume 6. Athena Scientific Belmont, MA.

Bezerra, M. A., Santelli, R. E., Oliveira, E. P., Villar, L. S., and Escaleira, L. A. (2008). Response surface methodology (rsm) as a tool for optimization in analytical chemistry. *Talanta*, 76(5):965–977.

Binev, P., Cohen, A., Dahmen, W., DeVore, R., Petrova, G., and Wojtaszczyk, P. (2011). Convergence Rates for Greedy Algorithms in Reduced Basis Methods. *SIAM journal on mathematical analysis*, 43(3):1457–1472.

Börsing, N., Wellmann, J. F., Niederau, J., and Regenauer-Lieb, K. (2017). Entropy production in a box: Analysis of instabilities in confined hydrothermal systems. *Water Resources Research*, 53(9):7716–7739.

Box, G. E. (1979). Robustness in the Strategy of Scientific Model Building. In *Robustness in statistics*, pages 201–236. Elsevier.

Branch, M. A., Coleman, T. F., and Li, Y. (1999). A Subspace, Interior, and Conjugate Gradient Method for Large-Scale Bound-Constrained Minimization Problems. *SIAM Journal on Scientific Computing*, 21(1):1–23.

Brazell, O., Messenger, S., Abusalbi, N., and Fjerstad, P. (2010). Multicore Evaluation and Performance Analysis of the ECLIPSE and INTERSECT Reservoir Simulation Codes. In *Oil and Gas High Performance Computing Workshop*.

Brunet, P., Clément, R., and Bouvier, C. (2010). Monitoring soil water content and deficit using Electrical Resistivity Tomography (ERT)–A case study in the Cevennes area, France. *Journal of Hydrology*, 380(1-2):146–153.

Brunken, J., Smetana, K., and Urban, K. (2019). (Parametrized) First Order Transport Equations: Realization of Optimally Stable Petrov–Galerkin Methods. *SIAM journal on scientific computing*, 41(1):A592–A621.

Buchmann, T. J. and Connolly, P. T. (2007). Contemporary kinematics of the Upper Rhine Graben: a 3D finite element approach. *Global and Planetary Change*, 58(1-4):287–309.

Burgers, G., Jan van Leeuwen, P., and Evensen, G. (1998). Analysis Scheme in the Ensemble Kalman Filter. *Monthly weather review*, 126(6):1719–1724.

Cacace, M. and Blöcher, G. (2015). MeshIt - a software for three dimensional volumetric meshing of complex faulted reservoirs. *Environmental Earth Sciences*, 74(6):5191–5209.

Cacace, M. and Jacquey, A. B. (2017). Flexible parallel implicit modelling of coupled thermal–hydraulic–mechanical processes in fractured rocks. *Solid Earth*, 8(5):921–941.

Cannavó, F. (2012). Sensitivity analysis for volcanic source modeling quality assessment and model selection. *Computers & geosciences*, 44:52–59.

Certes, C. and de Marsily, G. (1991). Application of the Pilot Point Method to the Identification of Aquifer Transmissivities. *Advances in Water Resources*, 14(5):284–300.

Chang, K.-H. (2014). *Design Theory and Methods using CAD/CAE: The Computer Aided Engineering Design Series*. Academic Press.

Chen, P. and Schwab, C. (2015). Sparse-grid, reduced-basis Bayesian inversion. *Computer Methods in Applied Mechanics and Engineering*, 297:84–115.

Chen, P. and Schwab, C. (2016). Sparse-grid, reduced-basis Bayesian inversion: Nonaffine-parametric nonlinear equations. *Journal of Computational Physics*, 316:470–503.

Chen, Y., Hesthaven, J. S., Maday, Y., and Rodríguez, J. (2010). Certified reduced basis methods and output bounds for the harmonic maxwell's equations. *SIAM Journal on Scientific Computing*, 32(2):970–996.

Clauser, C. (1984). A climatic correction on temperature gradients using surface-temperature series of various periods. *Tectonophysics*, 103(1-4):33–46.

Cloke, H., Pappenberger, F., and Renaud, J.-P. (2008). Multi-Method Global Sensitivity Analysis (MMGSA) for modelling floodplain hydrological processes. *Hydrological Processes: An International Journal*, 22(11):1660–1674.

Courtier, P. (1997). Dual formulation of four-dimensional variational assimilation. *Quarterly Journal of the Royal Meteorological Society*, 123(544):2449–2461.

Cui, T., Marzouk, Y. M., and Willcox, K. E. (2015). Data-driven model reduction for the Bayesian solution of inverse problems. *International Journal for Numerical Methods in Engineering*, 102(5):966–990.

Curtis, A. and Lomax, A. (2001). Prior information, sampling distributions, and the curse of dimensionality. *Geophysics*, 66(2):372–378.

de la Varga, M., Schaaf, A., and Wellmann, F. (2019). Gempy 1.0: open-source stochastic geological modeling and inversion. *Geoscientific Model Development*.

de Marsily, G., Lavedan, G., Boucher, M., and Fasanino, G. (1984). Interpretation of interference tests in a well field using geostatistical techniques to fit the permeability distribution in a reservoir model. *Geostatistics for natural resources characterization, Part*, 2:831–849.

Degen, D., Veroy, K., Freymark, J., Scheck-Wenderoth, M., and Wellmann, F. (2020a). Global Sensitivity Analysis to Optimize Basin-Scale Conductive Model Calibration - Insights on the Upper Rhine Graben.

Degen, D., Veroy, K., Scheck-Wenderoth, M., and Wellmann, F. (2020b). How Well Do We Know Our Boundary Conditions? *JGR: Solid Earth*. (in preperation).

Degen, D., Veroy, K., and Wellmann, F. (2020c). Certified reduced basis method in geosciences.

Degen, D., Veroy, K., and Wellmann, F. (2020d). Uncertainty Quantification for Basin-Scale Conductive Models. *Earth and Space Science Open Archive ESSOAr*.

Demezhko, D. Y., Ryvkin, D., Outkin, V., Duchkov, A., and Balobaev, V. (2007). Spatial distribution of Pleistocene/Holocene warming amplitudes in Northern Eurasia inferred from geothermal data. *Climate of the Past*, 3(3):559–568.

Dennis, M. R., Glendinning, P., Martin, P. A., Santosa, F., and Tanner, J. (2015). *The Princeton Companion to Applied Mathematics*. Princeton University Press.

Doherty, J. (2003). Ground Water Model Calibration Using Pilot Points and Regularization. *Groundwater*, 41(2):170–177.

Ebigbo, A., Niederau, J., Marquart, G., Dini, I., Thorwart, M., Rabbel, W., Pechnig, R., Bertani, R., and Clauser, C. (2016). Influence of depth, temperature, and structure of a crustal heat source on the geothermal reservoirs of Tuscany: numerical modelling and sensitivity study. *Geothermal Energy*, 4(1):5.

Eftang, J. L., Knezevic, D. J., and Patera, A. T. (2011). An hp certified reduced basis method for parametrized parabolic partial differential equations. *Mathematical and Computer Modelling of Dynamical Systems*, 17(4):395–422.

Eftang, J. L., Patera, A. T., and Rønquist, E. M. (2010). An "hp" Certified Reduced Basis Method for Parametrized Elliptic Partial Differential Equations. *SIAM Journal on Scientific Computing*, 32(6):3170–3200.

Eftang, J. L. and Stamm, B. (2012). Parameter multi-domain "hp" empirical interpolation. *International Journal for Numerical Methods in Engineering*, 90(4):412–428.

Eker, S., Rovenskaya, E., Obersteiner, M., and Langan, S. (2018). Practice and perspectives in the validation of resource management models. *Nature communications*, 9(1):5359.

Evensen, G. (1994). Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research: Oceans*, 99(C5):10143–10162.

Evensen, G. (2003). The Ensemble Kalman Filter: Theoretical formulation and practical implementation. *Ocean dynamics*, 53(4):343–367.

Fernández, M., Eguía, P., Granada, E., and Febrero, L. (2017). Sensitivity analysis of a vertical geothermal heat exchanger dynamic simulation: Calibration and error determination. *Geothermics*, 70:249–259.

Fletcher, R. and Reeves, C. M. (1964). Function minimization by conjugate gradients. *The computer journal*, 7(2):149–154.

Floris, F., Bush, M., Cuypers, M., Roggero, F., and Syversveen, A. R. (2001). Methods for quantifying the uncertainty of production forecasts: a comparative study. *Petroleum Geoscience*, 7(S):S87–S96.

Förster, A. (2001). Analysis of borehole temperature data in the Northeast German Basin: continuous logs versus bottom-hole temperatures. *Petroleum Geoscience*, 7(3):241–254.

Fowler, A. C. and Fowler, A. C. (1997). *Mathematical Models in the Applied Sciences*, volume 17. Cambridge University Press.

Frangos, M., Marzouk, Y., Willcox, K., and van Bloemen Waanders, B. (2010). Surrogate and reduced-order modeling: a comparison of approaches for large-scale statistical inverse problems [chapter 7].

Freymark, J., Bott, J., Cacace, M., Ziegler, M., and Scheck-Wenderoth, M. (2019). Influence of the Main Border Faults on the 3D Hydraulic Field of the Central Upper Rhine Graben. *Geofluids*, 2019.

Freymark, J., Sippel, J., Scheck-Wenderoth, M., Bär, K., Stiller, M., Fritsche, J.-G., and Kracht, M. (2017). The deep thermal field of the Upper Rhine Graben. *Tectonophysics*, 694:114–129.

Fuchs, S. and Balling, N. (2016). Improving the temperature predictions of subsurface thermal models by using high-quality input data. part 1: Uncertainty analysis of the thermal-conductivity parameterization. *Geothermics*, 64:42–54.

Galbally, D., Fidkowski, K., Willcox, K., and Ghattas, O. (2010). Non-linear model reduction for uncertainty quantification in large-scale inverse problems. *International journal for numerical methods in engineering*, 81(12):1581–1608.

Ganther, T. (2018). Validation of the FE and RB Implementation of the DwarfElephant Package through Diffusion Controlled Temperature and Pressure Benchmarks. Bacheolor's thesis, RWTH Aachen University, Aachen.

Geem, Z. W. (2012). *Optimization in Civil & Environmental Engineering*. Archives contemporaines.

GeORG-Projektteam (2013). Geopotenziale des tieferen Untergrundes im Oberrheingraben. *Fachlich- Technischer Abschlussbericht des INTERREG- Projekts GeORG, Teil 1, LGRB-Informationen*, 28:1–104.

Geuzaine, C. and Remacle, J.-F. (2009). Gmsh: A 3-D finite element mesh generator with built-in pre-and post-processing facilities. *International journal for numerical methods in engineering*, 79(11):1309–1331.

Ghasemi, M. and Gildin, E. (2016). Model order reduction in porous media flow simulation using quadratic bilinear formulation. *Computational Geosciences*, 20(3):723–735.

Gillijns, S., Mendoza, O. B., Chandrasekar, J., De Moor, B., Bernstein, D., and Ridley, A. (2006). What Is the Ensemble Kalman Filter and How Well Does it Work? In *2006 American Control Conference*, pages 6–pp. IEEE.

Goldberg, D. E. (1989). Genetic Algorithms in Search, Optimization and Machine Learning. Addsion-Wesley Longman. *Reading*.

Gosses, M., Nowak, W., and Wöhling, T. (2018). Explicit treatment for Dirichlet, Neumann and Cauchy boundary conditions in POD-based reduction of groundwater models. *Advances in water resources*, 115:160–171.

Grepl, M. (2005). *Reduced-basis Approximation and A Posteriori Error Estimation for Parabolic Partial Differential Equations*. PhD thesis, Massachusetts Institute of Technology.

Grepl, M. A. (2012). Model order reduction of parametrized nonlinear reaction–diffusion systems. *Computers & chemical engineering*, 43:33–44.

Grepl, M. A. and Kärcher, M. (2011). Reduced basis a posteriori error bounds for parametrized linear-quadratic elliptic optimal control problems. *Comptes rendus mathematique*, 349(15-16):873–877.

Grepl, M. A. and Patera, A. T. (2005). A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 39(1):157–181.

Günther, T., Rücker, C., and Spitzer, K. (2006). Three-dimensional modelling and inversion of DC resistivity data incorporating topography – II. Inversion. *Geophysical Journal International*, 166(2):506–517.

Haasdonk, B. (2017). Reduced Basis Methods for Parametrized PDEs–A Tutorial Introduction for Stationary and Instationary Problems. *Model reduction and approximation: theory and algorithms*, 15:65.

Hain, S., Ohlberger, M., Radic, M., and Urban, K. (2018). A hierarchical a posteriori error estimator for the Reduced Basis Method. *Advances in Computational Mathematics*, pages 1–24.

Hammerschmidt, M., Herrmann, S., Pomplun, J., Zschiedrich, L., Burger, S., and Schmidt, F. (2015). Reduced basis method for Maxwell's equations with resonance phenomena. In *Optical Systems Design 2015: Computational Optics*, volume 9630, page 96300R. International Society for Optics and Photonics.

Hecht, F., Pironneau, O., Le Hyaric, A., and Ohtsuka, K. (2007). FreeFem++. *Numerical Mathematics and Scientific Computation. Laboratoire JL Lions, Université Pierre et Marie Curie, http://www.freefem.org/ff+*, 3.

Hendricks Franssen, H. and Kinzelbach, W. (2008). Real-time groundwater flow modeling with the Ensemble Kalman Filter: Joint estimation of states and parameters and the filter inbreeding problem. *Water Resources Research*, 44(9).

Herman, J. and Usher, W. (2017). SALib: An open-source Python library for Sensitivity Analysis. *J. Open Source Softw*, 2(9):97.

Hernandez, V., Roman, J. E., and Vidal, V. (2005). SLEPc: A scalable and flexible toolkit for the solution of eigenvalue problems. *ACM Trans. Math. Software*, 31(3):351–362.

Herrmann, F. J., Friedlander, M. P., and Yilmaz, O. (2012). Fighting the Curse of Dimensionality: Compressive sensing in exploration seismology. *IEEE Signal Processing Magazine*, 29(3):88–100.

Hess, M. W. and Benner, P. (2013a). Fast evaluation of Time–Harmonic Maxwell's Equations Using the Reduced Basis Method. *IEEE Transactions on Microwave Theory and Techniques*, 61(6):2265–2274.

Hess, M. W. and Benner, P. (2013b). Reduced basis generation for maxwell's equations by rigorous error estimation. In *Proceedings of 19th International Conference on the Computation of Electromagnetic Fields (COMPUMAG 2013), Paper PD2-10 (2 pp.)*.

Hess, M. W., Grundel, S., and Benner, P. (2015). Estimating the Inf-Sup Constant in Reduced Basis Methods for Time-Harmonic Maxwell's Equations. *IEEE Transactions on Microwave Theory and Techniques*, 63(11):3549–3557.

Hess, W. M. and Benner, P. (2014). A reduced basis method for microwave semiconductor devices with geometric variations. *COMPEL: The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, 33(4):1071–1081.

Hesthaven, J. S., Rozza, G., Stamm, B., et al. (2016). *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*. SpringerBriefs in Mathematics, Springer.

Himpe, C. and Ohlberger, M. (2015). Data-driven combined state and parameter reduction for inverse problems. *Advances in Computational Mathematics*, 41(5):1343–1364.

Huang, P. (2018). Investigating different formulations for hydrothermal convection in geothermal systems. Master's thesis, RWTH Aachen University, Aachen.

Huang, X.-Y., Xiao, Q., Barker, D. M., Zhang, X., Michalakes, J., Huang, W., Henderson, T., Bray, J., Chen, Y., Ma, Z., et al. (2009). Four-dimensional variational data assimilation for wrf: Formulation and preliminary results. *Monthly Weather Review*, 137(1):299–314.

Huynh, D., Nguyen, N., Rozza, G., and Patera, A. Documentation for rbMIT Software: I. Reduced basis (RB) for Dummies. Massachusetts Institute of Technology, 2007–2010. `http://augustine.mit.edu/methodology/methodology_rbMIT_System.htm`.

Iglesias, M. and Stuart, A. M. (2014). Inverse Problems and Uncertainty Quantification. *SIAM News*, pages 2–3.

Illies, J. (1972). The Rhine graben rift system-plate tectonics and transform faulting. *Geophysical surveys*, 1(1):27–60.

Jacquey, A., Cacace, M., Blöcher, G., Milsch, H., and Deon, F. From localized to homogeneous deformation of porous rocks–insights from laboratory experiments and numerical modelling. In *GeoProc 2017 Book of Abstracts: 6th International Conference on Coupled THMC Processes in Geosystems*, page 50.

Jaite, X. C. (2019). Evaluating the Effect of Uncertainty in Geological Modeling on Groundwater Flow Simulations. Master's thesis, RWTH Aachen University, Aachen. (in preperation).

Jones, E., Oliphant, T., and Peterson, P. (2014). SciPy: Open source scientific tools for Python.

Jülich Supercomputing Centre (2016). JURECA: General-purpose supercomputer at Jülich Supercomputing Centre. *Journal of large-scale research facilities*, 2(A62).

Jung, N. (2012). *Error Estimation for Parametric Model Order Reduction and Its Application*. VDI Verlag.

Kärcher, M., Boyaval, S., Grepl, M. A., and Veroy, K. (2018). Reduced basis approximation and a posteriori error bounds for 4D-Var data assimilation. *Optimization and Engineering*, 19(3):663–695.

Kärcher, M. and Grepl, M. A. (2014). A certified reduced basis method for parametrized elliptic optimal control problems. *ESAIM: Control, Optimisation and Calculus of Variations*, 20(2):416–441.

Kärcher, M., Tokoutsi, Z., Grepl, M. A., and Veroy, K. (2017). Certified reduced basis methods for parametrized elliptic optimal control problems with distributed controls. *Journal of Scientific Computing*, pages 1–32.

Kastner, O., Sippel, J., Zimmermann, G., and Huenges, E. (2015). Assessment of Geothermal Heat Provision from Deep Sedimentary Aquifers in Berlin/Germany: A Case Study. *Assessment*, 19:25.

Keller, J., Hendricks Franssen, H.-J., and Marquart, G. (2018). Comparing Seven Variants of the Ensemble Kalman Filter: How Many Synthetic Experiments Are Needed? *Water Resources Research*, 54(9):6299–6318.

Keller, J., Hendricks Franssen, H.-J., and Nowak, W. (2019). Pilot Point Ensemble Kalman Filter for geostatistical inversion and data assimilation. *Water Resources Research*. (submitted).

Khaitan, S. K. and Gupta, A. (2012). *High Performance Computing in Power and Energy Systems*. Springer.

Khalajzadeh, V., Heidarinejad, G., and Srebric, J. (2011). Parameters optimization of a vertical ground heat exchanger based on response surface methodology. *Energy and Buildings*, 43(6):1288–1294.

Khuri, A. I. and Mukhopadhyay, S. (2010). Response surface methodology. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(2):128–149.

Kirk, B. S., Peterson, J. W., Stogner, R. H., and Carey, G. F. (2006). libMesh: a C++ library for parallel adaptive mesh refinement/coarsening simulations. *Engineering with Computers*, 22(3-4):237–254.

Knezevic, D. J. and Peterson, J. W. (2011). A high-performance parallel implementation of the certified reduced basis method. *Computer Methods in Applied Mechanics and Engineering*, 200(13):1455–1466.

Konrad, F., Savvatis, A., Degen, D., Einsiedl, F., Wellmann, F., and Zoßeder, K. (2020). IProductivity enhancement of geothermal wells through fault zones: efficient numerical evaluation of a parameter space for the Upper Jurassic aquifer of the North Alpine Foreland Basin. *Geothermics*. (in preparation).

Konrad, F., Savvatis, A., Wellmann, F., and Zoßeder, K. (2019). Hydraulic behavior of fault zones in pump tests of geothermal wells - a parametric analysis using numerical simulations for the upper jurassic aquifer of the north alpine foreland basin. *Geothermal Energy*.

Larrañaga, P. and Lozano, J. A. (2001). *Estimation of distribution algorithms: A new tool for evolutionary computation*, volume 2. Springer Science & Business Media.

LaVenue, A. M. and Pickens, J. F. (1992). Application of a Coupled Adjoint Sensitivity and Kriging Approach to Calibrate a Groundwater Flow Model. *Water Resources Research*, 28(6):1543–1569.

LaVenue, A. M., RamaRao, B. S., De Marsily, G., and Marietta, M. G. (1995). Pilot point methodology for automated calibration of an ensemble of conditionally simulated transmissivity fields: 2. application. *Water Resources Research*, 31(3):495–516.

Le Dimet, F.-X. and Talagrand, O. (1986). Variational algorithms for analysis and assimilation of meteorological observations: theoretical aspects. *Tellus A: Dynamic Meteorology and Oceanography*, 38(2):97–110.

Lehmann, H., Wang, K., and Clauser, C. (1998). Parameter identification and uncertainty analysis for heat transfer at the KTB drill site using a 2-D inverse method. *Tectonophysics*, 291(1-4):179–194.

Leucci, G. and Greco, F. (2012). 3D ERT survey to reconstruct archaeological features in the subsoil of the Spirito Santo church ruins at the site of Occhiolà (Sicily, Italy). *Archaeology*, 1(1):1–6.

Li, L., Zhou, H., Gómez-Hernández, J. J., and Franssen, H.-J. H. (2012). Jointly mapping hydraulic conductivity and porosity by assimilating concentration data via ensemble Kalman filter. *Journal of hydrology*, 428:152–169.

Liao, B., Xu, N., Wang, Q., and Li, X. (2019). Automated Calibration of Geological Models. Student project, RWTH Aachen University, Aachen.

Liao, Q. and Willcox, K. (2015). A Domain Decomposition Approach for Uncertainty Analysis. *SIAM Journal on Scientific Computing*, 37(1):A103–A133.

Lieberman, C., Willcox, K., and Ghattas, O. (2010). Parameter and State Model Reduction for Large-Scale Statistical Inverse Problems. *SIAM Journal on Scientific Computing*, 32(5):2523–2542.

Limberger, J., van Wees, J.-D., Tesauro, M., Smit, J., Bonté, D., Békési, E., Pluymaekers, M., Struijk, M., Vrijlandt, M., Beekman, F., et al. (2018). Refining the thermal structure of the European lithosphere by inversion of subsurface temperature data. *Global and planetary change*, 171:18–47.

Linde, N., Ginsbourger, D., Irving, J., Nobile, F., and Doucet, A. (2017). On uncertainty quantification in hydrogeology and hydrogeophysics. *Advances in water resources*, 110:166–181.

Lorenc, A. (1981). A Global Three-Dimensional Multivariate Statistical Interpolation Scheme. *Monthly Weather Review*, 109(4):701–721.

Lorenc, A., Ballard, S., Bell, R., Ingleby, N., Andrews, P., Barker, D., Bray, J., Clayton, A., Dalby, T., Li, D., et al. (2000). The Met. Office global three-dimensional variational data assimilation scheme. *Quarterly Journal of the Royal Meteorological Society*, 126(570):2991–3012.

Lorenc, A. C. (1986). Analysis methods for numerical weather prediction. *Quarterly Journal of the Royal Meteorological Society*, 112(474):1177–1194.

Lowrie, W. (2007). *Fundamentals of Geophysics*. Cambridge university press.

Manzano, G., Galve, F., Zambrini, R., and Parrondo, J. M. (2016). Entropy production and thermodynamic power of the squeezed thermal reservoir. *Physical Review E*, 93(5):052120.

Manzoni, A., Pagani, S., and Lassila, T. (2016). Accurate Solution of Bayesian Inverse Uncertainty Quantification Problems Combining Reduced Basis Methods and Reduction Error Models. *SIAM/ASA Journal on Uncertainty Quantification*, 4(1):380–412.

Meixner, J., Schill, E., Grimmer, J. C., Gaucher, E., Kohl, T., and Klingler, P. (2016). Structural control of geothermal reservoirs in extensional tectonic settings: An example from the Upper Rhine Graben. *Journal of Structural Geology*, 82:1–15.

Miao, T., Lu, W., Lin, J., Guo, J., and Liu, T. (2019). Modeling and uncertainty analysis of seawater intrusion in coastal aquifers using a surrogate model: a case study in Longkou, China. *Arabian Journal of Geosciences*, 12(1):1.

Michot, D., Benderitter, Y., Dorigny, A., Nicoullaud, B., King, D., and Tabbagh, A. (2003). Spatial and temporal monitoring of soil water content with an irrigated corn crop cover using surface electrical resistivity tomography. *Water Resources Research*, 39(5).

Midttømme, K., Roaldset, E., and Aagaard, P. (1998). Thermal conductivity claystones and mudstones of selected from England. *Clay Minerals*, 33(1):131–145.

Milk, R., Rave, S., and Schindler, F. (2016). pyMOR–Generic Algorithms and Interfaces for Model Order Reduction. *SIAM Journal on Scientific Computing*, 38(5):S194–S216.

Mo, S., Shi, X., Lu, D., Ye, M., and Wu, J. (2019). An adaptive Kriging surrogate method for efficient uncertainty quantification with an application to geological carbon sequestration modeling. *Computers & Geosciences*.

Mottaghy, D., Pechnig, R., and Vogt, C. (2011). The geothermal project Den Haag: 3D numerical models for temperature prediction and reservoir simulation. *Geothermics*, 40(3):199–210.

Myers, R. H., Montgomery, D. C., and Anderson-Cook, C. M. (2016). *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*. John Wiley & Sons.

Navarro, M., Le Maître, O. P., Hoteit, I., George, D. L., Mandli, K. T., and Knio, O. M. (2018). Surrogate-based parameter inference in debris flow model. *Computational Geosciences*, 22(6):1447–1463.

Navon, I. M. (2009). Data Assimilation for Numerical Weather Prediction: A Review. In *Data assimilation for atmospheric, oceanic and hydrologic applications*, pages 21–65. Springer.

Nield, D. A. and Bejan, A. (2017). Heat Transfer Through a Porous Medium. In *Convection in Porous Media*, pages 37–55. Springer.

Noack, V., Scheck-Wenderoth, M., and Cacace, M. (2012). Sensitivity of 3D thermal models to the choice of boundary conditions and thermal properties: a case study for the area of Brandenburg (NE German Basin). *Environmental Earth Sciences*, 67(6):1695–1711.

Noack, V., Scheck-Wenderoth, M., Cacace, M., and Schneider, M. (2013). Influence of fluid flow on the regional thermal field: results from 3D numerical modelling for the area of Brandenburg (North German Basin). *Environmental earth sciences*, 70(8):3523–3544.

Pagani, S., Manzoni, A., and Quarteroni, A. (2017). Efficient State/Parameter Estimation in Nonlinear Unsteady PDEs by a Reduced Basis Ensemble Kalman Filter. *SIAM/ASA Journal on Uncertainty Quantification*, 5(1):890–921.

Panzeri, M., Riva, M., Guadagnini, A., and Neuman, S. P. (2014). Comparison of Ensemble Kalman Filter groundwater-data assimilation methods based on stochastic moment equations and Monte Carlo simulation. *Advances in Water Resources*, 66:8–18.

Patil, A., Huard, D., and Fonnesbeck, C. J. (2010). PyMC: Bayesian Stochastic Modelling in Python. *Journal of statistical software*, 35(4):1.

Pauwels, H., Fouillac, C., and Fouillac, A.-M. (1993). Chemistry and isotopes of deep geothermal saline fluids in the Upper Rhine Graben: Origin of compounds and water-rock interactions. *Geochimica et Cosmochimica Acta*, 57(12):2737–2749.

Peherstorfer, B., Willcox, K., and Gunzburger, M. (2018). Survey of Multifidelity Methods in Uncertainty Propagation, Inference, and Optimization. *Siam Review*, 60(3):550–591.

Permann, C. J., Gaston, D. R., Andrš, D., Carlsen, R. W., Kong, F., Lindsay, A. D., Miller, J. M., Peterson, J. W., Slaughter, A. E., Stogner, R. H., and Martineau, R. C. (2020). MOOSE: Enabling massively parallel multiphysics simulation. *SoftwareX*, 11:100430.

Perrone, A., Lapenna, V., and Piscitelli, S. (2014). Electrical resistivity tomography technique for landslide investigation: A review. *Earth-Science Reviews*, 135:65–82.

Poulet, T. and Veveakis, M. (2016). A viscoplastic approach for pore collapse in saturated soft rocks using REDBACK: An open-source parallel simulator for Rock mEchanics with Dissipative feedBACKs. *Computers and Geotechnics*, 74:211–221.

Poulet, T., Veveakis, M., Paesold, M., and Regenauer-Lieb, K. (2014). REDBACK: An open-source highly scalable simulation tool for rock mechanics with dissipative feedbacks. In *AGU Fall Meeting Abstracts*.

Pribnow, D. and Clauser, C. (2000). Heat and fluid flow at the soultz hot dry rock system in the rhine graben. In *World Geothermal Congress, Kyushu-Tohoku, Japan*, pages 3835–3840.

Pribnow, D. and Schellschmidt, R. (2000). Thermal Tracking of Upper Crustal Fluid Flow in the Rhine Graben. *Geophysical Research Letters*, 27(13):1957–1960.

Prud'homme, C., Chabannes, V., Doyeux, V., Ismail, M., Samake, A., and Pena, G. (2012). Feel++: A computational framework for Galerkin Methods and Advanced Numerical Methods. In *ESAIM: Proceedings*, volume 38, pages 429–455. EDP Sciences.

Prud'homme, C., Rovas, D. V., Veroy, K., Machiels, L., Maday, Y., Patera, A. T., and Turinici, G. (2002). Reliable Real-Time Solution of Parametrized Partial Differential Equations: Reduced-Basis Output Bound Methods. *Journal of Fluids Engineering*, 124(1):70–80.

Quarteroni, A., Manzoni, A., and Negri, F. (2015). *Reduced Basis Methods for Partial Differential Equations: An Introduction*. UNITEXT. Springer International Publishing.

Quinao, J. J. and Zarrouk, S. J. (2014). Applications of experimental design and response surface method in probabilistic geothermal resource assessment–preliminary results. In *Proceedings, 39th Workshop on geothermal reservoir engineering. Stanford: Stanford University*.

RamaRao, B. S., LaVenue, A. M., De Marsily, G., and Marietta, M. G. (1995). Pilot point methodology for automated calibration of an ensemble of conditionally simulated transmissivity fields: 1. Theory and computational experiments. *Water Resources Research*, 31(3):475–493.

Rawlins, F., Ballard, S., Bovis, K., Clayton, A., Li, D., Inverarity, G., Lorenc, A., and Payne, T. (2007). The Met Office global four-dimensional variational data assimilation scheme. *Quarterly Journal of the Royal Meteorological Society*, 133(623):347–362.

Ray, S., Mukherjee, J., and Mandal, S. (2015). Modelling nitrogen and carbon cycles in Hooghly estuary along with adjacent mangrove ecosystem. In *Developments in Environmental Modelling*, volume 27, pages 289–320. Elsevier.

Rizzo, C. B., de Barros, F. P., Perotto, S., Oldani, L., and Guadagnini, A. (2017). Adaptive POD model reduction for solute transport in heterogeneous porous media. *Computational Geosciences*, pages 1–12.

Rodell, M., Chen, J., Kato, H., Famiglietti, J. S., Nigro, J., and Wilson, C. R. (2007). Estimating groundwater storage changes in the Mississippi River basin (USA) using GRACE. *Hydrogeology Journal*, 15(1):159–166.

Rosenbrock, H. (1960). An automatic method for finding the greatest or least value of a function. *The Computer Journal*, 3(3):175–184.

Rousset, M. A., Huang, C. K., Klie, H., and Durlofsky, L. J. (2014). Reduced-order modeling for thermal recovery processes. *Computational Geosciences*, 18(3-4):401–415.

Rovas, D., Machiels, L., and Maday, Y. (2006). Reduced-basis output bound methods for parabolic problems. *IMA journal of numerical analysis*, 26(3):423–445.

Rovas, D. V. (2003). *Reduced-Basis Output Bound Methods for Parametrized Partial Differential Equations*. PhD thesis, Massachusetts Institute of Technology.

Rozza, G., Huynh, D. B. P., and Patera, A. T. (2007). Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. *Archives of Computational Methods in Engineering*, 15(3):1.

Rücker, C., Günther, T., and Spitzer, K. (2006). Three-dimensional modelling and inversion of dc resistivity data incorporating topography – I. Modelling. *Geophysical Journal International*, 166(2):495–505.

Saltelli, A. (2002). Making best use of model evaluations to compute sensitivity indices. *Computer physics communications*, 145(2):280–297.

Saltelli, A. (2019). A short comment on statistical versus mathematical modelling. *Nature communications*, 10(1):3870.

Saltelli, A., Annoni, P., Azzini, I., Campolongo, F., Ratto, M., and Tarantola, S. (2010). Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index. *Computer Physics Communications*, 181(2):259–270.

Saltelli, A., Tarantola, S., Campolongo, F., and Ratto, M. (2004). Sensitivity analysis in practice: a guide to assessing scientific models. *Chichester, England*.

Salvatier, J., Wiecki, T. V., and Fonnesbeck, C. (2016). Probabilistic programming in Python using PyMC3. *PeerJ Computer Science*, 2:e55.

Sasaki, Y. (1970). *Some basic formalisms in numerical variational analysis*. Citeseer.

Shen, P. and Beck, A. (1992). Paleoclimate change and heat flow density inferred from temperature data in the Superior Province of the Canadian Shield. *Global and planetary change*, 6(2-4):143–165.

Siavashi, M. and Jamali, M. (2016). Heat transfer and entropy generation analysis of turbulent flow of TiO2-water nanofluid inside annuli with different radius ratios using two-phase mixture model. *Applied Thermal Engineering*, 100:1149–1160.

Siripunvaraporn, W. (2012). Three-Dimensional Magnetotelluric Inversion: An Introductory Guide for Developers and Users. *Surveys in geophysics*, 33(1):5–27.

Sobol, I. M. (2001). Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Mathematics and computers in simulation*, 55(1-3):271–280.

Song, X., Zhang, J., Zhan, C., Xuan, Y., Ye, M., and Xu, C. (2015). Global sensitivity analysis in hydrological modeling: Review of concepts, methods, theoretical framework, and applications. *Journal of hydrology*, 523:739–757.

Stober, I. and Bucher, K. (2015). Hydraulic and hydrochemical properties of deep sedimentary reservoirs of the Upper Rhine Graben, Europe. *Geofluids*, 15(3):464–482.

Tang, Y., Reed, P., Van Werkhoven, K., and Wagener, T. (2007). Advancing the identification and evaluation of distributed rainfall-runoff models using global sensitivity analysis. *Water Resources Research*, 43(6).

The CGAL Project (2019). *CGAL User and Reference Manual*. CGAL Editorial Board, 4.14 edition.

Tiefe Geothermie, P. (2007). Nutzungen der geothermischen Energie aus dem tiefen Untergrund (Tiefe Geothermie)–Arbeitshilfe für Geologische Dienste.

Toselli, A. and Widlund, O. (2006). *Domain Decomposition Methods – Algorithms and Theory*, volume 34. Springer Science & Business Media.

United Nations (2012). The Millennium Development Goals Report 2012.

University of Stuttgart, University of Münster, University of Ulm, and Sandia Lawrence Livermore National Research Laboratory. `http://www.ians.uni-stuttgart.de/MoRePaS/software/rbmatlab/1.13.10/doc/index.html`.

Vaidya, N. (2020). *Biophysics model development describing 3D energy-tissue interaction for thermal ablation treatments in cancer care*. PhD thesis, RWTH Aachen University. (in preperation).

van Griensven, A. v., Meixner, T., Grunwald, S., Bishop, T., Diluzio, M., and Srinivasan, R. (2006). A global sensitivity analysis tool for the parameters of multi-variable catchment models. *Journal of hydrology*, 324(1-4):10–23.

van Leeuwen, P. J. (2010). Nonlinear data assimilation in geosciences: an extremely efficient particle filter. *Quarterly Journal of the Royal Meteorological Society*, 136(653):1991–1999.

Vasseur, G., Bernard, P., Van de Meulebrouck, J., Kast, Y., and Jolivet, J. (1983). Holocene paleotemperatures deduced from geothermal measurements. *Palaeogeography, Palaeoclimatology, Palaeoecology*, 43(3-4):237–259.

Veroy, K. and Patera, A. (2005). Certified real-time solution of the parametrized steady incompressible Navier–Stokes equations: rigorous reduced-basis a posteriori error bounds. *International Journal for Numerical Methods in Fluids*, 47(8-9):773–788.

Veroy, K., Prud'homme, C., Rovas, D. V., and Patera, A. T. (2003). A Posteriori Error Bounds for Reduced-Basis Approximation of Parametrized Noncoercive and Nonlinear Elliptic Partial Differential Equations. In *Proceedings of the 16th AIAA computational fluid dynamics conference*, volume 3847, pages 23–26. Orlando, FL.

Vidal, J. and Genter, A. (2018). Overview of naturally permeable fractured reservoirs in the central and southern Upper Rhine Graben: Insights from geothermal wells. *Geothermics*, 74:57–73.

Vidal, J., Genter, A., Schmittbuhl, J., Whitechurch, H., Baujard, C., and Damalais, E. (2015). Evolution of concepts for the geothermal projects in the Upper Rhine Graben. In *European Geothermal Workshop, Strasbourg, France*.

Vogt, C., Iwanowski-Strahser, K., Marquart, G., Arnold, J., Mottaghy, D., Pechnig, R., Gnjezda, D., and Clauser, C. (2013). Modeling contribution to risk assessment of thermal production power for geothermal reservoirs. *Renewable energy*, 53:230–241.

Vogt, C., Mottaghy, D., Wolf, A., Rath, V., Pechnig, R., and Clauser, C. (2010). Reducing temperature uncertainties by stochastic geothermal reservoir modelling. *Geophysical Journal International*, 181(1):321–333.

Volkwein, S. (2011). Model reduction using proper orthogonal decomposition. *Lecture Notes, Institute of Mathematics and Scientific Computing, University of Graz. see http://www. uni-graz. at/imawww/volkwein/POD. pdf*.

Volkwein, S. (2013). Proper orthogonal decomposition: Theory and reduced-order modelling. *Lecture Notes, University of Konstanz*, 4(4):1–29.

von Bülow, R. (2020). *Integrated Hydrogeophysical and Structural Investigation of the Subsurface*. PhD thesis, RWTH Aachen University. (in preperation).

von Bülow, R., Degen, D., Klitsch, N., Veroy, K., and Wellmann, F. (2020). Global Sensitivity Analysis to Investigate the Application Areas of ERT Configurations. (in preperation).

Vrugt, J. A., Diks, C. G., Gupta, H. V., Bouten, W., and Verstraten, J. M. (2005). Improved treatment of uncertainty in hydrologic modeling: Combining the strengths of global optimization and data assimilation. *Water resources research*, 41(1).

Wagner, R. and Clauser, C. (2005). Evaluating thermal response tests using parameter estimation for thermal conductivity and thermal capacity. *Journal of Geophysics and Engineering*, 2(4):349–356.

Wainwright, H. M., Finsterle, S., Jung, Y., Zhou, Q., and Birkholzer, J. T. (2014). Making sense of global sensitivity analyses. *Computers & Geosciences*, 65:84–94.

Wellmann, F. and Caumon, G. (2018). 3-D Structural geological models: Concepts, methods, and uncertainties. *Advances in Geophysics*, 59:1–121.

Wellmann, J. F. and Reid, L. B. (2014). Basin-scale geothermal model calibration: Experience from the Perth Basin, Australia. *Energy Procedia*, 59:382–389.

Zabinsky, Z. B. (2013). *Stochastic Adaptive Search for Global Optimization*, volume 72. Springer Science & Business Media.

Zaitchik, B. F., Rodell, M., and Reichle, R. H. (2008). Assimilation of GRACE Terrestrial Water Storage Data into a Land Surface Model: Results for the Mississippi River Basin. *Journal of Hydrometeorology*, 9(3):535–548.

Zhan, C.-S., Song, X.-M., Xia, J., and Tong, C. (2013). An efficient integrated approach for global sensitivity analysis of hydrological model parameters. *Environmental Modelling & Software*, 41:39–52.

Zhou, H., Gomez-Hernandez, J. J., Franssen, H.-J. H., and Li, L. (2011). An approach to handling non-Gaussianity of parameters and state variables in ensemble Kalman filtering. *Advances in Water Resources*, 34(7):844–864.

Zhou, Q. Y., Shimada, J., and Sato, A. (2001). Three-dimensional spatial and temporal monitoring of soil water content using electrical resistivity tomography. *Water Resources Research*, 37(2):273–285.

Zlotnik, S., Díez, P., Modesto, D., and Huerta, A. (2015). Proper generalized decomposition of a geometrically parametrized heat problem with geophysical applications. *International Journal for Numerical Methods in Engineering*, 103(10):737–758.