ulm university universität
uulm

Universität Ulm

Fakultät für Mathematik und Wirtschaftswissenschaften

Institut für Numerische Mathematik

Reduced Basis Methods for parabolic PDEs with Parameter Functions in High Dimensions and Applications in Finance

Dissertation zur Erlangung des Doktorgrades

Dr. rer. nat.

der Fakultät für Mathematik und Wirtschaftswissenschaften

der Universität Ulm

vorgelegt von

Antonia Christine Mayerhofer

aus Göttingen

Ulm, im Januar 2016

Dekan: Prof. Dr. Werner Smolny
Gutachter:
Prof. Dr. Karsten Urban
Prof. Dr. Robert Stelzer
Tag der Abgabe: 19. Januar 2016
Tag der Promotion: 13. Mai 2016

Abstract

The present thesis deals with variants of the space-time reduced basis method for parametrised parabolic partial differential equations. The reduced basis method is a well-known projection based model reduction technique for parameter dependent problems. We consider parabolic partial differential equations in space-time variational formulation. The formulation includes integration over space and time and a model reduction is achieved in both, space and time dimension.

One objective of this thesis is the development of a reduced basis method to handle functions as parameters. A parameter in the reduced basis method is usually a vector of real numbers that describes the underlying model properties. The more general concept of a parameter space that is a subspace of a Hilbert space requires new reduced basis generation processes. In particular, the initial value of a parabolic partial differential equation is considered as a function parameter. A Two-Step Greedy Method is presented where two reduced bases are constructed, one for the approximation of the initial value and one for the evolution of the solution. The decomposition into two steps allows for a better approximation error control. A-priori as well as a-posteriori error bounds are provided.

As the space-time reduced basis method deals with time as additional dimension, the reduced basis generation process is computationally expensive. We apply the H-Tucker low rank tensor format in the reduced basis offline phase to reduce the computational costs. The high dimensional linear system is decomposed in its tensor product components and can efficiently be solved applying the low rank tensor method. The resulting solution of the high dimensional problem implies an additional approximation error that has to be considered in the basis generation process. Error analysis is provided to combine both model reduction methods.

For derivative pricing in finance, one needs to calculate the conditional expectation of the discounted payoff under a risk neutral measure. To this end one can equivalently solve the associated parabolic partial differential equation in diffusion based models. If we want to apply the reduced basis method as model reduction scheme, we usually have to construct a new reduced basis for each payoff function. To avoid this effort, the reduced basis method for parameter functions can be applied. Two standard option pricing models are considered, the Black-Scholes model and the Heston model. Both models provide a closed form solution such that the numerical results can easily be verified.

Zusammenfassung

In der vorliegenden Arbeit werden zwei Varianten der Space-Time Reduzierten Basis Methode für parametrische parabolische partielle Differentialgleichungen vorgestellt. Die Reduzierte Basis Methode ist eine verbreitete projektionsbasierte Modelreduktionsmethode für parameterabhängige Probleme. Parabolische partielle Differentialgleichungen werden in sogenannter space-time variationeller Formulierung betrachtet, welche sowohl die Integration über Raum als auch über Zeit beinhaltet, wodurch auch die Modelreduktion sowohl in Raum- als auch in Zeitdimension greift.
Zunächst wird der Parameterbegriff der Reduzierten Basis Methode auf Funktionen als Parameter erweitert. Als Parameter versteht man ursprünglich einen reelen Vektor, dessen Einträge die zugrundeliegenden Modeleigenschaften definieren. Die Erweiterung auf Funktionen (aus einem Hilbertraum) als Parameter erfordert neue Basisgenerierungsstrategien. Falls die Anfangswerte von parabolischen Differentialgleichungen als Funktionsparameter aufgefasst werden, können in einer zweistufigen Greedy Methode zwei reduzierte Basen erstellt werden, wobei die erste den Anfangswert und die zweite die Entwicklung der Lösung über die Zeit approximiert. Durch die Aufteilung in zwei Schritte kann der Approximationsfehler mit Hilfe der entwickelten a-priori und a-posteriori Fehlerschätzer besonders gut kontrolliert werden.
Die Zeit wird in der Space-Time Reduzierten Basis Methode als zusätzliche Dimension erfasst, weswegen die Basisgenerierung rechnerisch aufwendig ist. Diesem Problem widmet sich der zweite Teil der Arbeit. Hier wird das H-Tucker Low Rank Tensor Format in der Offline-Phase verwendet, um den numerischen Aufwand zu reduzieren. Das ursprüngliche hochdimensionale System wird hierbei in seine Tensorproduktkomponenten zerlegt und kann dann über das Low Rank Tensor Format effizient gelöst werden. Allerdings beinhaltet die so erhaltene Lösung einen zusätzlichen Approximationsfehler, was im Basisgenerierungsprozess beachtet werden muss.
Anwendung finden beide Methoden in der Bepreisung von Finanzderivaten. Anstatt den bedingten Erwartungswert des diskontierten Payoffs unter einem risikoneutralen Maß zu bestimmen, kann alternativ eine assoziierte parabolische partielle Differentialgleichung gelöst werden. Bei direkter Anwendung der Reduzierten Basis Methode müsste hierbei für jede neue Payofffunktion eine eigene reduzierte Basis konstruiert werden. Dies kann durch die Verwendung von Parameterfunktionen vermieden werden. Zwei klassische Optionspreismodelle – das Black-Scholes und das Heston Model – werden betrachtet, wobei beide Modelle eine Verifizierung der numerischen Ergebnisse erlauben.

Contents

1. Introduction

Fitting the real world in a mathematical discription has always some limitation. And the more complex a model gets the more challenging is the evaluation. This thesis is concerned with the model order reduction of parametrised parabolic partial differential equations. A reduced basis method is developed that allows for infinite dimensional function spaces as parameter spaces. Additionally the reduced basis method is combined with a low rank tensor format for efficient offline computations and the treatment of higher dimensional state spaces.

1.1. The Reduced Basis Method

Many real world applications are modeled by parametrised partial differential equations. Those models have often to be solved repeatedly for new parameter sets in a limited amount of time. The solution with standard numerical methods like the finite element method of finite volume leads to high dimensional linear systems of dimension $\mathcal{N} \gg 0$. Model reduction techniques are required if the arising numerical effort exceeds the given computational resources in terms of hardware or computational time. The class of projection based model reduction techniques lowers the original model state space dimension by projection onto a lower dimensional approximation space. The model reduction in the reduced basis method is achieved by approximation of the state space w.r.t. the occurring model parameters. For a sample set of parameters $\mu_1, \dots, \mu_N$ their corresponding solutions $u(\mu_i)$, for $i = 1, \dots, N$, are computed in the so-called *offline phase*. Here, the high dimensional linear system has to be solved by standard numerical methods. The reduced basis space is spanned by the precomputed solutions. In the *online phase*, the reduced basis space replaces the original solution space and the solutions are computed solving a corresponding N dimensional linear system. Under the assumption of smooth dependence of the system operators on the parameter μ, one expects $0 < N \ll \mathcal{N}$.

The reduced basis method has been applied to various partial differential equations and was extended for several frameworks. Our focus is on parabolic partial differential equations that are given in a space-time variational formulation. The space-time variational formulation is given by the application of test functions and integration over space and time. The corresponding solutions are elements of appropriate Bochner spaces. The model reduction is in space dimensions and time

dimension. This stands in contrast to reduced basis methods that are based on the standard weak formulation for parabolic partial differential equations and an underlying time stepping scheme.

1.2. The Objective

Parameter spaces in the reduced basis method are commonly subspaces of the real valued vector space $\mathbb{R}^d$. The expression 'high-dimensional' referred so far to large values of $d \gg 0$. The extension to parameter spaces that are subspaces of function spaces has not yet been considered. Functions as parameters allow for very flexible problem formulations. So far, for every possible function, a new reduced basis space had to be constructed. If there are finitely many functions that are a-priori known this is an expensive but a feasible strategy. As soon as the class of functions gets less precise, a more general solution is required. A standard parameter sampling procedure in the offline phase is performed on a discrete subspace of the original parameter space – the training set – that reflects the main properties of the parameter space. New basis sampling strategies are necessary for functions as parameters.

Time as additional dimension in the space-time reduced basis method increases the computation effort in the offline phase. Imposing a discretisation that is equivalent to a standard time stepping scheme allows for cheap computations of the detailed solutions. However, those discretisations do not in general lead to a well-posed discrete problem formulation. To overcome these problems and at the same time to allow for higher space dimensions the reduced basis method is combined with a low-rank tensor approximation scheme. The later implies a very efficient storage scheme for the operator equations and the ansatz and test functions. The ansatz space with state space $\Omega \subset \mathbb{R}^{\mathcal{N}_1 \times \ldots \times \mathcal{N}_d}$ is decomposed into its tensor product components. The corresponding coefficients of the solution of the partial differential equation build a tensor in $\mathbb{R}^{\mathcal{N}_1 \times \ldots \times \mathcal{N}_d}$. The matrix representation of the discrete operator has to be in a so-called CP decomposition. Then, every matrix-matrix or matrix-vector operation required in the offline phase of the reduced basis method can be performed without establishing the full system. To guarantee numerical feasibility an approximation of the detailed solutions is necessary. The reduced basis is therefore spanned by snapshots that only solve the discrete problem formulation up to a given tolerance. This approximation has to be considered in the basis sampling procedure and has an influence on the reduced basis approximation quality. Thus, the a-priori error control is essential.

1.3. Application in Option Pricing

We apply the developed methods in derivative pricing in finance. Stochastic processes are used to describe the dynamics of the underlying assets. Under suitable assumptions the expected value of a financial derivative is equivalent to the solution of a partial differential equation. Derivative pricing models in general have a huge amount of parameters. These parameters change for different underlyings and market situations. The application of the reduced basis method offers the possibility for e.g. a fast calibration or pricing process.

In finance, price processes in standard pricing models are often modeled as Itô diffusions, i.e., stochastic processes that satisfy a standard stochastic differential equation driven by a Brownian motion. We focus on parabolic partial differential equations that occur in finance when considering the expected value of an Itô diffusion. Those models can easily be extended to jump-diffusion models or to higher dimensions for pricing basket options, i.e. options with multiple underlyings. The restriction to pricing models that provide a closed form solution is not mandatory. The reduced basis method is fully based on numerical method that are applicable to a wide class of diffusion models. The associated parabolic partial differential equations have diffusion, convection and reaction terms with right-hand side equal to zero. The payoff of the option defines the final condition. The payoff that defines the option type is subject to change and thus a parameter function in the reduced basis method.

1.4. Thesis Outline

The thesis is divided into four parts.

The first part recalls the space-time reduced basis method. We provide the underlying functional analytic setting and state the space-time variational formulation. Well-posedness of the problem formulation is discussed. Possible finite element discretisations are presented. A general introduction to the reduced basis method is given. Different reduced basis simulation techniques for a stable space-time reduced basis approximation are presented. The underlying implementation is shortly introduced. We conclude the first part with an application of the standard space-time reduced basis method on a standard parabolic partial differential equation.

Parameter functions are the subject of the second part. A reduced basis method is introduced where the initial value of a parabolic partial differential equation is the parameter (function). The underlying assumption is a given good approximation of the function parameter domain in the first place. Wavelet bases offer the possibility to achieve good approximation results for less accurately described parameter domains. The reduced basis method for parameter functions is separated into two steps. The first results in a reduced basis approximation of the initial

value where the reduced basis for the evolution of the solution is established in the second. An a-posteriori online efficiently evaluable error estimator is provided as well as an a-priori error bound for the online approximation. Two variants of the greedy method are presented: the first fixes the reduced basis space offline; the second allows to choose a customized reduced basis space online that is optimal for the new parameter function. This flexibility comes with an increased storage effort. The numerical experiments at the end of the section show the applicability of the method and the quality of the model reduction.

The third part is concerned with the feasibility of the possibly computationally expensive offline phase of the space-time reduced basis method and a step towards higher space dimensions. We present the standard low rank tensor methods that provide a setting to store very high dimensional problems efficiently and perform computations with an, in comparison very small computational effort. We provide the necessary error analysis for the usage of the H-Tucker format in the offline phase of the space-time reduced basis method. A reduced basis for a 2D parabolic partial differential equation is computed in the numerical example at the end of the section.

The fourth part provides the application of the afore developed methods on standard derivative pricing models. For the functional analytical setting we introduce weighted Sobolev spaces. The standard numerical schemes for option pricing are presented. The payoff function is considered as parameter function. Therefore a specific treatment of the in finance a-priori unbounded space domain is required. As standard example, the reduced basis method for parameter functions is applied to the Black-Scholes model. Further, we introduce the Heston model, a stochastic volatility model. We again apply the developed reduced basis methods and discuss the difficulties arising in a real world application. Since both models provide a closed form solution our method can easily be validated.

A summary and a short outlook are given at the end.

2. Space-Time Reduced Basis Method

2.1. Introduction

The Reduced Basis Method (RBM) is a model order reduction method for parametrised partial differential equations (PPDEs). In this context, we are interested in the model parameters of the PDE. These parameters specify model properties e.g. the conductivity of a thermal block or the current stock market situation. The reduced basis method is a useful tool for solving a PDE for lots of different parameters very fast (in *real time*), very often (*multi-query* context) or on a small device. Possible scenarios are e.g. computations on a board computer of a car or an optimisation process for model calibration in finance.

The present thesis is in particular concerned with the space-time RBM and its use for parabolic PDEs in a space-time variational formulation. In contrast to the standard time-stepping approaches that are shortly described in Appendix B it involves a reduction in the discrete space and time dimension. A first certified space-time RBM for parabolic PDEs with homogenous initial condition was proposed by K. Urban and A. T. Patera in [UP14]. An a-posteriori error bound was introduced and its long time integral behaviour was investigated. The method is an alternative to other approaches where the error bounds may grow exponentially in time when the (nonlinear) operator is non-coervice, cf. [NRP09, KNP11]. Applications to nonlinear problems followed in [YPU14] for the Burgers equation. Homogeneous initial conditions were used and the error bound was combined with the existing Brezzi-Rappaz-Raviart (BRR) theory, that provides a framework for non-linear problems, and hp-splitting techniques. Further, they used an interpolation strategy for the construction of the reduced solution instead of a projection based method. M. Yano extended this approach and applied it to the Boussinesq equation in [Yan14]. He removed the interpolation-based approximation and used a hp-Petrov-Galerkin-projection-based approximation instead. A further application of the space-time RBM was for time-periodic problems, presented by K. Steih and K. Urban, [SU12]. In this setting the initial condition of the parabolic problem changes for periodicity. Further, the space-time approach was adapted by S. Glas and K. Urban to variational inequalities in [GU14]. We applied the space-time approach to the Heston model and presented a stable test space construction for the Petrov-Galerkin projection in combination with a treatment for parameter functions in [MU14], cf. Section 3.2.

2.2. Preliminaries

2.2.1. Space-Time Formulation

The space-time reduced basis method we are going to use to solve parametrised parabolic PDEs relies on a variational formulation in appropriate function spaces. For the introduction of the corresponding ansatz and test spaces, we give a short functional analytical overview of Bochner(-Lebesgue) spaces. In order and notation we mainly follow [Sho97, Ch. III]. We introduce the general evolution equation and its better known weak formulation first. We continue with the variational problem formulation and state existence and uniqueness results of the solution, cf. Ch. Schwab and R. Stevenson [SS09].

In order to better distinguish the vector valued functions taking their values in Banach spaces of standard Lebesgue integrable functions, we denote their norm by two lines "$\|\cdot\|$" whereas the standard norms are denoted by only one line "$|\cdot|$". To avoid confusions, we indicate the respective space additionally in the index.

2.2.1.1. Bochner Spaces and Weak Formulation

We extend the concept of Lebesgue measurable functions to Banach space valued functions. For a generalised concept of measurable and integrable step functions we generalise the concept of integrability. We restrict our presentation to functions defined on the real interval $I = (0, T)$ for $0 < T < \infty$.

Definition 2.2.1. (Bochner integrability.) [Sho97, Thm. 1.2] Let $(I, \mathcal{F}, \mu)$ be a measurable space and B be a Banach space equipped with norm $|\cdot|_B$. Let $f : I \to B$ be measurable in the sense that there is a sequence of measurable step functions such that $f_n \to f$ in B a.s. for $s \in I$. Then, f is integrable if and only if $s \mapsto |f(s)|_B$ is (Lebesgue) integrable.

Definition 2.2.2. [Sho97, p. 103] Let $1 \leq p < \infty$. Let B be a Banach space with norm $|\cdot|_B$ and dual B'. We denote by $L_p(I; B)$ resp. $L_p(0, T; B)$ the space of (equivalence classes of) functions $f : I \to B$ such that $|f(\cdot)|_B \in L_p(I; \mathbb{R})$, i.e.

$$\left(\int_I |f(t)|_B^p dt \right)^{1/p} < \infty.$$

We often refer to $L_p(I; B)$ as Bochner or Bochner-Lebesgue space. We do not detail the generalisation for $p = \infty$ as it is straightforward.

Proposition 2.2.3. [DL92, Prop. 1] For any Banach space B, the space $L_p(I;B)$ is a Banach space with the respective norm $\|f\|_{L_p(I;B)} = (\int_I |f(s)|_B^p ds)^{1/p}$ for $1 \leq p < \infty$.

Theorem 2.2.4. *[Sho97, Thm 1.5] Let $1 < p < \infty$ and $\frac{1}{p} + \frac{1}{p'} = 1$. Let B be reflexive, i.e. B can be isomorphically identified with its bidual space. Then we can identify the dual space $L_p(I;B)'$ with $L_{p'}(I;B')$.*

Proof. Explicit construction of the isomorphism. □

After the generalisation of integrability of vector valued functions, we pass on to vector-valued distributions. In the following we restrict to Hilbert spaces. For a more general introduction we refer to [DL92, Ch. XVIII §1.1].

Definition 2.2.5. [WZ76, Def. 3.3.1, 3.3.2, Satz 3.3.3] Let $\langle \cdot, \cdot \rangle_{V' \times V}$ denote the duality pairing of V and V'. By $\mathcal{C}^\infty(I;H)' := \mathcal{L}(\mathcal{C}^\infty(I);H)$ we denote the space of all linear functionals mapping from $\mathcal{C}^\infty(I)$ onto H.

- Let H be a Hilbert space. We call the linear map $u : \mathcal{C}^\infty(I) \to H$ a distribution with values in H, i.e., $u \in \mathcal{C}^\infty(I;H)'$, if for all compact $K \subset I$ there exist $p, C \geq 0$ such that $|\langle u, \phi \rangle|_H \leq C \sup_{x \in K} \sum_{s \leq p} |\phi^{(s)}(x)|$ for all $\phi \in \mathcal{C}^\infty(I)$, $\operatorname{supp}(\phi) \subset K$.
- The weak distributional derivative $\dot{u} \in \mathcal{C}^\infty(\Omega;H)'$ is defined by $\langle \dot{u}, \phi \rangle := (-1)\langle u, \phi^{(1)} \rangle$.

Proposition 2.2.6. [WZ76, Satz 3.3.5] Let V, H be Hilbert spaces, $V \subset H$ continuously embedded. Then,

$$\mathcal{C}^\infty(I;V)' \subset \mathcal{C}^\infty(I;H)'.$$

Remark 2.2.7. [WZ76, p. 153] For $f \in L_2(I;V)$ and V Hilbert space, $f \in \mathcal{C}^\infty(I;V)'$ and $\dot{f} \in \mathcal{C}^\infty(I;V')'$.

The last step is to introduce the so-called Gelfand triple, cf. [WZ76, p.127]: Let H be a Hilbert space and V be a Banach space. Let V be dense and continuously embedded in H and H be identified with its dual H' ($H \cong H'$). The embedding implies a dense embedding of the associated dual spaces and using $H \cong H'$ we obtain the Gelfand triple $V \hookrightarrow H \hookrightarrow V'$. The dual space V' is again a Banach space and in particular this allows to identify $(\cdot, \cdot)_H$ with $\langle \cdot, \cdot \rangle_{V' \times V}$ using the continuous extension of $(\cdot, \cdot)_H$ on $V' \times V$.

Denote by $|\cdot|_V$ and $|\cdot|_H$ the induced norms on V and H respectively. For $1 = \frac{1}{p} + \frac{1}{p'}$, consider the Banach space

$$\mathbb{X}_p = \{u \in L_p(0,T;V) : \dot{u} \in L_{p'}(0,T;V')\} \tag{2.2.1}$$

with the norm

$$\|u\|_{\mathbb{X}_p}^2 = \|u\|_{L_p(I;V)}^2 + \|\dot{u}\|_{L_{p'}(I;V')}^2. \tag{2.2.2}$$

We will use the notations $\dot{u}$, $\frac{\partial u}{\partial t}$ or $\frac{du}{dt}$ interchangeably.

Proposition 2.2.8. [Sho97, Prop. 1.2, Cor. 1.1]

- The Banach space $\mathbb{X}_p$ is embedded in $\mathcal{C}([0,T];H)$, i.e., there is a constant C for which $\|u\|_{\mathcal{C}([0,T],H)} \leq C\|u\|_{\mathbb{X}_p}, u \in \mathbb{X}_p$.
- For $u \in \mathbb{X}$ the function $|u(\cdot)|_H^2$ is absolutely continuous on $[0,T]$ and $\frac{d}{dt}|u(t)|_H^2 = 2\dot{u}(t)(u(t))$ a.e. $t \in [0,T]$.

In the following we are mostly interested in the case $p = 2$ and V being a Hilbert space. With Theorem 2.2.4 the dual of $L_2(0,T;V)$ is given by $L_2(0,T;V')$. In particular $\mathbb{X} := \mathbb{X}_2$ is a Hilbert space, cf. [WZ76, Prop. 3.3.7.], [DL92, Prop. 6].

Let $u_0 \in H$ and $g \in L_2(I;V')$ be given. Consider a bilinear form $a : V \times V \to \mathbb{R}$ such that for each pair $u, v \in V$ the function $a(u,v)$ is continuous (resp. bounded): there is a $C_a > 0$ such that

$$|a(u,v)| \leq C_a|u|_V|v|_V \quad \forall\ u,v \in V. \tag{2.2.3}$$

Let $\mathcal{A} \in \mathcal{L}(V,V')$ be the induced linear operator given due to Riesz-Fréchet, cf. Proposition A.1.3, by

$$\langle \mathcal{A}u, v\rangle_{V'\times V} = a(u,v) \quad \forall\ u,v \in V.$$

Proposition 2.2.9. [Sho97, Prop. 2.1] The following two problem formulations are equivalent. Find u where $u(t) \in V$:

$$\begin{aligned} \dot{u}(t) + \mathcal{A}u(t) &= g(t) \quad \text{in } V' \text{ a.e. } t \in I, \\ u(0) &= u_0 \quad \text{in } H. \end{aligned} \tag{2.2.4a}$$

$$\begin{aligned} (\dot{u}(t), v)_H + \langle \mathcal{A}u(t), v\rangle_{V'\times V} &= \langle g(t), v\rangle_{V'\times V} \quad \forall\ v \in V, \text{ a.e. } t \in I, \\ u(0) &= u_0 \quad \text{in } H. \end{aligned} \tag{2.2.4b}$$

Remark 2.2.10. For u as element in $\mathbb{X}$, the initial condition is meaningful by Proposition 2.2.8, i.e., $\lim_{t\to 0}\langle u(t), v\rangle_{V'\times V} = (u(0), v)_H$. Further by not evaluating at a time point t we can understand the equations in $\mathbb{X}'$. The variational formulation introduced in the next section follows.

2.2.1.2. Variational Formulation

For the variational formulation we require an appropriate test space. Following [SS09] we define

$$\mathbb{Y} := L_2(I;V) \times H \tag{2.2.5}$$

equipped with the norm

$$\|v\|_{\mathbb{Y}}^2 = \|v_1\|_{L_2(I,V)}^2 + |v_2|_H^2 \tag{2.2.6}$$

for every $v = (v_1, v_2)$ in $\mathbb{Y}$. For the linear operator $\mathcal{A}(t) \in \mathcal{L}(V, V')$, $I = (0, T)$ with $0 < T < \infty$ and $g \in L_2(I; V')$, consider again the associated bilinear form $a : V \times V \to \mathbb{R}$ with $\langle \mathcal{A}u, v\rangle_{V'\times V} = a(u, v)$ for all $u, v \in V$. Define for $u \in \mathbb{X}$ and $v = (v_1, v_2) \in \mathbb{Y}$:

$$b(u, v) := \int_I \langle \dot{u}(t), v_1(t)\rangle_{V'\times V}\, dt + \int_I a(u(t), v_1(t))dt + (u(0), v_2)_H, \tag{2.2.7a}$$

$$f(v) := \int_I \langle g(t), v_1(t)\rangle_{V'\times V}\, dt + (u_0, v_2)_H. \tag{2.2.7b}$$

Proposition 2.2.11. [SS09, (5.4)] The variational formulation of the equivalent problem formulations in Proposition 2.2.9 is given by

$$\text{find } u \in \mathbb{X} : \quad b(u, v) = f(v) \quad \forall v \in \mathbb{Y}. \tag{2.2.8}$$

Proposition 2.2.12. The problem formulations (2.2.8) and (2.2.4b) are equivalent.

Proof. Suppose that $u \in \mathbb{X}$ solves (2.2.4b). Obviously u is a solution of (2.2.8). If u is a solution of (2.2.8), we know that

$$\int_I \langle \dot{u}(t), v_1(t)\rangle_{V'\times V}\, dt + \int_I a(u(t), v_1(t))dt + (u(0), v_2)_H = \int_I \langle g(t), v_1(t)\rangle_{V'\times V}\, dt + (u_0, v_2)_H$$

for arbitrary $v_1 \in L_2(I; V)$ and $v_2 \in H$. In particular

$$(u(0), w)_H = (u_0, w)_H$$

for all $w \in V \subset H$, i.e. $u(0) = u_0$ a.e. in the domain of H. Further

$$\int_I \langle \dot{u}(t) + \mathcal{A}u(t) - g(t), \tilde{w}(t)\rangle_{V'\times V}\, dt = 0.$$

for arbitrary $\tilde{w} \in L_2(I; V)$. Thus $\langle \dot{u}(t) + \mathcal{A}u(t) - g(t), \tilde{w}(t)\rangle_{V'\times V} = 0$, a.e. in $[0, T]$, i.e. (2.2.4b) holds for all $(\tilde{w}(t) =)v \in V$. □

Remark 2.2.13. Instead of Equation (2.2.8) we can consider the variational formulation introduced in e.g. [Sho97, Prop. 2.1 (c)]. Given $v \in H^1(0, T; H)$, $v(T) \equiv 0$,

$$-\int_I (u(t), \dot{v}(t))_H dt + \int_I \mathcal{A}u(t)(v(t))dt = \int_I g(t)(v(t))dt + (u_0, v(0))_H. \tag{2.2.9}$$

The initial condition enters naturally the equation by integration by parts over time of the left-

hand side of the integrated weak formulation. Equation (2.2.8) has advantages for its later use in the reduced basis methods for parameter functions, cf. Section 3.2.

2.2.1.3. Existence and Uniqueness of the Solution

We have several equivalent problem formulation, thus several possibilities to show the well-posedeness of the problem and existence and uniqueness of a solution. As we will further use the results, we show existence and uniqueness of the solution in space-time variational formulation. We therefore use the following result of J. Nečas [Neč64]:

Proposition 2.2.14. cf. [Bab71, Thm. 2.1], [Bra07, III. §3, Satz 3.6] Let X and Y be Hilbert spaces and $b : \mathsf{X} \times \mathsf{Y} \to \mathbb{R}$ a bilinear form with associated linear operator $\mathcal{B} : \mathsf{X} \to \mathsf{Y}'$,

$$\langle \mathcal{B}u, v\rangle := b(u,v) \quad \forall\ v \in \mathsf{Y}.$$

The linear operator $\mathcal{B}$ is an isomorphism if and only if for the associated bilinear form b there exist $C_b \geq 0$ and $\beta > 0$ such that

$$\sup_{u\in\mathsf{X}} \sup_{v\in\mathsf{Y}} \frac{|b(u,v)|}{\|u\|_{\mathsf{X}}\|v\|_{\mathsf{Y}}} \leq C_b \qquad \text{(continuity)}, \tag{2.2.10a}$$

$$\inf_{u\in\mathsf{X}} \sup_{v\in\mathsf{Y}} \frac{b(u,v)}{\|u\|_{\mathsf{X}}\|v\|_{\mathsf{Y}}} \geq \beta > 0 \qquad \text{(inf-sup condition)} \tag{2.2.10b}$$

and

$$\text{for every } 0 \neq v \in \mathsf{Y} \text{ exists } u \in \mathsf{X} \ :\ b(u,v) \neq 0 \quad \text{(surjectivity)}. \tag{2.2.10c}$$

□

It remains to prove the three properties for the bilinear form b defined by Equation (2.2.7a) using the following properties of the associated bilinear form $a(\cdot,\cdot)$. We call a bilinear form $a(\cdot,\cdot)$ bounded and say that it satisfies the Gårding inequality if there exist $C_a > 0$, $\alpha_a > 0$ and $\lambda_a \in \mathbb{R}$ such that for all $\phi, \psi \in V$ the following holds

$$|a(\phi,\psi)| \leq C_a |\phi|_V |\psi|_V, \qquad \text{(continuity)} \tag{2.2.11a}$$

$$a(\phi,\phi) + \lambda_a |\phi|_H^2 \geq \alpha_a |\phi|_V^2 \qquad \text{(Gårding)}. \tag{2.2.11b}$$

The following theorem states well-posedeness of (2.2.8).

Theorem 2.2.15. *[SS09, Thm. 5.1] For the variational formulation* (2.2.8) *let* $a(\cdot,\cdot)$ *satisfy* (2.2.11a) *and* (2.2.11b)*. Then, there exist* C_b *and* β *such that the bilinear form* $b(\cdot,\cdot)$ *defined in* (2.2.7a) *is continuous and satisfies the inf-sup condition. In addition,* $b(\cdot,\cdot)$ *is surjective and there exists a unique solution* u *for* (2.2.8)*.*

Proof. Cf. [SS09, App. A]. The proof is an application of Proposition 2.2.14. An upper bound for the continuity constant C_b is explicitly constructed as well as surjectivity can be shown. For the inf-sup constant, the idea is to define $z_w := (A^*)^{-1}\dot{w}$ for $0 \neq w \in \mathbb{X}$. Here $A^* : V \to V'$ denotes the adjoint, i.e. $\langle A^*\phi, \psi\rangle_{V'\times V} := a(\psi, \phi)$ Using $v_w = (z_w + w, w(0))$ as test function, an explicit lower bound for β can be stated. □

We provide the mentioned lower bound of β. Therefore, we introduce the following quantities:

$$M_e := \sup_{0\neq w\in\mathbb{X}} \frac{|w(0)|_H}{\|w\|_{\mathbb{X}}} \qquad \text{and} \qquad \varrho := \sup_{0\neq\phi\in V} \frac{|\phi|_H}{|\phi|_V}.$$

Remark 2.2.16. It is known that M_e is uniformly bounded and only depends on the final time point T when T tends to zero ([DL92, Ch. XVIII, Proof of Thm. 1], [SS09]).

Corollary 2.2.17. *[UP14, Prop. 2.2, Cor. 2.7], [SS09, App. A] Define*

$$\beta^{LB}_{coer}(\alpha, \lambda, C) := \frac{\min(1, C^{-2})(\alpha - \lambda\varrho^2)}{\sqrt{2\max(1, \alpha^{-2}) + M_e^2}}$$

and

$$\beta^{LB}_{time}(T, \alpha, \lambda, C) := \frac{e^{-2\lambda T}}{\sqrt{\max(2, 1 + 2\lambda^2\varrho^4)}}\beta^{LB}_{coer}(\alpha, 0, C + \lambda\varrho^2).$$

The lower bound β^{LB} for the inf-sup-constant β in Theorem 2.2.15 is given by

$$\beta^{LB}(T, \alpha_a, \lambda_a, C_a) = \max(\beta^{LB}_{coer}(\alpha_a, \lambda_a, C_a), \beta^{LB}_{time}(T, \alpha_a, \lambda_a, C_a)).$$

Remark 2.2.18. The inf-sup lower bound $\beta^{\mathrm{LB}}_{\mathrm{coer}}$ is only meaningful, i.e. positive, if $\alpha_a - \lambda_a\varrho^2 > 0$, which is equivalent to the bilinear form a being coercive, i.e. (2.2.11b) for $\lambda = 0$ is satisfied. If a satisfies only the Gårding inequality, $\beta^{\mathrm{LB}}_{\mathrm{coer}}$ may be negative. Then, we consider w.l.o.g. the transformation $\hat{u} = ue^{-\lambda_a t}$ and the corresponding variational formulation, such that the transformed bilinear form $\hat{a}$ is coercive. Back-transformation yields $\beta^{\mathrm{LB}}_{\mathrm{time}}$ as lower bound of the inf-sup constant of the original bilinear form.

We presented the inf-sup lower bounds regarding the norm of $\mathbb{X}$ introduced in (2.2.2). Analogously to [UP14] a different norm can be used. Then, the lower bound slightly changes.

Corollary 2.2.19. *Cf. [UP14, Proposition 2.2] Theorem 2.2.15 holds for $\|\cdot\|_{\mathbb{X},T}$ given by*

$$\|u\|^2_{\mathbb{X},T} := \|u\|^2_{L_2(I,V)} + \|\dot{u}\|^2_{L_2(I,V')} + |u(T)|^2_H \tag{2.2.12}$$

for $u \in \mathbb{X}$. Let $\alpha_a - \lambda_a \varrho^2 > 0$. We obtain the following lower bound of the inf-sup constant

$$\beta_{coer}^{LB,\mathbb{X},T}(\alpha_a, \lambda_a, C_a) = \frac{\min(\min(1, C_a^{-2})(\alpha_a - \lambda_a \varrho^2), 1)}{\sqrt{2\max(1, \alpha_a^{-2}) + M_e^2}}.$$

Proof. The proof follows again [SS09, Appendix A]. □

Remark 2.2.20. (a) The norm given in Corollary 2.2.19 is well-posed since $\mathbb{X} \hookrightarrow \mathcal{C}(I; H)$ (cf. Proposition 2.2.8). By adding $|u(T)|_H^2$, the focus is set on the final time-point, which is the most relevant in e.g. option pricing. Despite that, using $\|\cdot\|_{\mathbb{X},T}$ improves the inf-sup constant in special cases, cf. Corollary 2.2.22.

(b) Homogeneous initial conditions can be included in the ansatz space $\mathbb{X}$ and the test space reduces to $L_2(I; V)$. The above estimate changes slightly since $M_e = 0$, [UP14, Prop. 2.2, Cor. 2.7] [MU14, Rem. 3.2].

(c) The lower bound can further be improved by replacing the coercivity constant α_a in the denominator by $0 < \beta_a^* \leq \inf_{\phi \in V} \sup_{\psi \in V} \frac{a(\phi,\psi)}{|\phi|_V |\psi|_V}$, cf. [UP14, Prop. 2.2].

Example 2.2.21. [UP14, Prop. 2.2, Rem. 2.4] Consider $\mathcal{A} \equiv -\Delta$, $V = H_0^1(\Omega)$, $H = L_2(\Omega)$. Define $|\phi|_V^2 = a(\phi, \phi)$ such that $C_a = 1$, $\lambda_a = 0$, $\alpha_a = 1$. We obtain $\beta \geq \frac{1}{\sqrt{2+M_e^2}}$.

This lower bound can be improved to $\beta \geq 1$.

Corollary 2.2.22. *cf. [UP14, Cor. 2.5] For a symmetric coercive bilinear form $a : V \times V \to \mathbb{R}$ and $|\cdot|_V^2 := a(\cdot, \cdot)$ the inf-sup constant is given by $\beta_{coer}^{LB,\mathbb{X},T} = 1$.*

Proof. Let $w \in \mathbb{X}$. For $v_w = (v_1, v_2) = ((A^*)^{-1}\dot{w} + w, w(0))$ consider

$$\begin{aligned}
\|v_w\|_{\mathbb{Y}}^2 &= \|v_1\|_{L_2(I;V)}^2 + |v_2|_H^2 \\
&= \|(A^*)^{-1}\dot{w} + w\|_{L_2(I;V)}^2 + |w(0)|_H^2 \\
&= \|\dot{w}\|_{L_2(I;V')}^2 + \|w\|_{L_2(I;V)}^2 + 2\int_I ((A^*)^{-1}\dot{w}, w)_V dt + |w(0)|_H^2.
\end{aligned}$$

Using $2\int_I ((A^*)^{-1}\dot{w}, w)_V dt = |w(T)|_H^2 - |w(0)|_H^2$ we obtain

$$\|v\|_{\mathbb{Y}}^2 = \|w\|_{\mathbb{X}}^2 \Leftrightarrow \|v\|_{\mathbb{Y}} = \|w\|_{\mathbb{X}}.$$

The bilinear form $b(\cdot,\cdot)$ satisfies

$$\begin{aligned} b(w, v_w) &= \int_I (\dot{w}(t), v_w(t))_{V' \times V} + a(w(t), v_w(t))dt + |w(0)|_H \\ &= \int_I a((A^*)^{-1}\dot{w}(t) + w(t), v_w(t))dt + |w(0)|_H \\ &= \|(A^*)^{-1}\dot{w} + w\|_{L_2(I;V)} + |w(0)|_H^2 \\ &= \|w\|_{\mathbb{X}}^2 = \|w\|_{\mathbb{X}}\|v\|_{\mathbb{Y}} \end{aligned}$$

and $\beta_{\text{coer}}^{\text{LB},\mathbb{X},T} = 1$. □

Certainly, well-posedness can be shown directly for the weak-formulation (2.2.4b) as in e.g. [LM72, Ch. 4 Thm. 4.1]. As our variational formulation is an equivalent problem formulation, we would immediately obtain the well-posedeness of the problem as well as existence and uniqueness of the solution. However, the just discussed inf-sup constant is an important quantity for the a-posteriori error estimation in the space-time reduced basis method. Furthermore the approach is analogous for proving well-posedness of the discrete problem formulation in the next section.

2.2.2. Discretisation of the Space-Time Variational Formulation

In this section, we discuss finite element approximations for solving the parabolic PDE numerically in space-time variational formulation. In a space-time discretisation scheme, we discretise in space and time at once. Most numerical approaches do not use the functional analytical background provided in terms of Bochner spaces. In contrary, M. Griebel and D. Oeltz developed a sparse grid space-time framework in [GO07]. They still rely on the weak formulation but with an additional integration over time, keeping the initial condition separate. The solution is given as an element of a tensor space that is isomorphic to the introduced Bochner space $\mathbb{X}$. Hierarchical bases are used and some convergence results are stated. Further, Ch. Schwab and R. Stevenson introduced a space-time adaptive wavelet method in [SS09]. Here, the underlying variational formulation is the one we introduced in the previous section. Deriving stable space-time Petrov-Galerkin finite element discretisations was the subject of the thesis of R. Andreev, [And12]. Different discretisation schemes are presented along with discussions on well-posedness and stability. Useful is the observation that for particular chosen finite element time discretisations solving the resulting high dimensional linear system is equivalent to solving a corresponding time stepping scheme as those schemes have the advantage of providing a greater computational feasibility. Unfortunately, as was shown in [And12], they are not unconditionally stable in an inf-sup sense and well-posedness of the discrete problem formulation may fail, cf. Section 2.2.2.1. We present the discretisation that leads to the Crank-Nicolson time stepping scheme in detail in Section 2.2.2.2. There, ansatz and test space have the same dimension. For $\mathbb{X}^{\mathcal{N}} \subset \mathbb{X}$, $\mathbb{Y}^{\mathcal{N}} \subset \mathbb{Y}$

and $\dim(\mathbb{X}^{\mathcal{N}}) = \mathcal{N} = \dim(\mathbb{Y}^{\mathcal{N}})$ with $\mathcal{N} < \infty$, the discrete problem formulation of (2.2.8) is given by

$$\text{find } u^{\mathcal{N}} \in \mathbb{X}^{\mathcal{N}} : \quad b(u^{\mathcal{N}}, v^{\mathcal{N}}) = f(v^{\mathcal{N}}) \quad \forall\, v^{\mathcal{N}} \in \mathbb{Y}^{\mathcal{N}}. \tag{2.2.13}$$

In general, we have to solve a Petrov-Galerkin problem, i.e., $\mathbb{X}^{\mathcal{N}} \neq \mathbb{Y}^{\mathcal{N}}$. In [And12] was shown that there are pairs of unconditionally stable discretisation spaces $\mathbb{X}^{\mathcal{N}} \subset \mathbb{X}$ and $\mathbb{Y}^{\mathcal{M}} \subset \mathbb{Y}$ such that $\dim(\mathbb{X}^{\mathcal{N}}) = \mathcal{N} \leq \mathcal{M} = \dim(\mathbb{Y}^{\mathcal{M}})$. We give a short introduction in Section 2.2.2.3.

2.2.2.1. Well-posedness and Stability

The discrete space-time variational formulation leads to a quasi-optimality result for the approximation that can be used in the error estimation of the reduced basis method. We state the theoretical result in the upcoming Proposition. It is an adaption of Nečas result introduced in Proposition 2.2.14 on finite dimensional approximation spaces. For finite elements it was first applied in [Bab71, BA72].

Proposition 2.2.23. cf. [Bra07, Hilfssatz 3.7] Let X, Y be real Hilbert spaces. Let Proposition 2.2.14 hold for a bilinear form $b : \mathsf{X} \times \mathsf{Y} \to \mathbb{R}$. Consider non-trivial closed subspaces $\{0\} \neq \mathsf{X}^{\mathcal{N}} \subseteq \mathsf{X}$ and $\{0\} \neq \mathsf{Y}^{\mathcal{N}} \subseteq \mathsf{Y}$ such that b satisfies (2.2.10) (a) - (c) for $\mathsf{X}^{\mathcal{N}}$ instead of X and $\mathsf{Y}^{\mathcal{N}}$ instead of Y. Let $C_b^{\mathcal{N}}$ be the discrete continuity constant and $\beta^{\mathcal{N}}$ the discrete inf-sup constant. Then, there exists a unique $u^{\mathcal{N}} \in \mathsf{X}^{\mathcal{N}}$ such that

$$b(u^{\mathcal{N}}, v^{\mathcal{N}}) = f(v^{\mathcal{N}}) \quad \forall\, v^{\mathcal{N}} \in \mathsf{Y}^{\mathcal{N}}$$

and $\|u^{\mathcal{N}}\|_{\mathsf{X}} \leq \frac{C_b^{\mathcal{N}}}{\beta^{\mathcal{N}}} \|u\|_{\mathsf{X}}$ for $u \in \mathsf{X}$ being the corresponding solution. Furthermore, a quasi-optimality estimate holds, i.e.,

$$\|u - u^{\mathcal{N}}\|_{\mathsf{X}} \leq (1 + \frac{1}{\beta^{\mathcal{N}}} C_b^{\mathcal{N}}) \inf_{w^{\mathcal{N}} \in \mathsf{X}^{\mathcal{N}}} \|u - w^{\mathcal{N}}\|_{\mathsf{X}}.$$

□

Remark 2.2.24. Since $\mathsf{X}^{\mathcal{N}} \subset \mathsf{X}$, the equality $\|u\|_{\mathsf{X}^{\mathcal{N}}} = \|u\|_{\mathsf{X}}$ holds true for all $u \in \mathsf{X}^{\mathcal{N}}$.

Continuity on the discrete spaces directly follows from continuity on the Hilbert spaces X and Y, as long as we consider conforming discretisations $\mathsf{X}^{\mathcal{N}} \subset \mathsf{X}$ and $\mathsf{Y}^{\mathcal{N}} \subset \mathsf{Y}$:

$$\sup_{u \in \mathsf{X}^{\mathcal{N}}} \sup_{v \in \mathsf{Y}^{\mathcal{N}}} \frac{|b(u,v)|}{\|u\|_{\mathsf{X}} \|v\|_{\mathsf{Y}}} \leq \sup_{u \in \mathsf{X}} \sup_{v \in \mathsf{Y}} \frac{|b(u,v)|}{\|u\|_{\mathsf{X}} \|v\|_{\mathsf{Y}}} \leq C_b.$$

Assuming $\dim(\mathsf{X}^{\mathcal{N}}) = \dim(\mathsf{Y}^{\mathcal{N}}) < \infty$, injectivity of the linear operator in $\mathcal{L}(\mathsf{X}^{\mathcal{N}}, (\mathsf{Y}^{\mathcal{N}})')$ implies already bijectivity. Therefore, the assumptions of Proposition 2.2.23 are satisfied if and only if

the discrete inf-sup condition

$$\inf_{u\in\mathsf{X}^{\mathcal{N}}}\sup_{v\in\mathsf{Y}^{\mathcal{N}}}\frac{|b(u,v)|}{\|u\|_{\mathsf{X}}\|v\|_{\mathsf{Y}}}\geq\beta^{\mathcal{N}}>0 \tag{2.2.14}$$

holds. The inf-sup condition therefore implies the well-posedness of the discretised problem.

Remark 2.2.25. In the Hilbert space setting, the quasi-optimality condition can be improved, [XZ03, Kat60],

$$\|u-u^{\mathcal{N}}\|_{\mathsf{X}}\leq\frac{1}{\beta^{\mathcal{N}}}C_b^{\mathcal{N}}\inf_{w^{\mathcal{N}}\in\mathsf{X}^{\mathcal{N}}}\|u-w^{\mathcal{N}}\|_{\mathsf{X}}\leq(1+\frac{1}{\beta^{\mathcal{N}}}C_b^{\mathcal{N}})\inf_{w^{\mathcal{N}}\in\mathsf{X}^{\mathcal{N}}}\|u-w^{\mathcal{N}}\|_{\mathsf{X}}.$$

For the computation of the inf-sup constant we can equivalently solve a generalised eigenvalue problem. We shortly recall the main steps and refer to [UP14, p. 13] and references therein for details. Let the assumptions of Proposition 2.2.23 be fulfilled. To compute the inf-sup constant, we first apply the Riesz Representation Theorem and define the supremizer $S_u\in\mathsf{Y}$ for given $u\in\mathsf{X}$ by

$$(S_u,v)_{\mathsf{Y}}=b(u,v)\quad\forall v\in\mathsf{Y}.$$

Lemma 2.2.26. *For the setting given in Proposition 2.2.23,* $S_u=\arg\sup_{v\in\mathsf{Y}^{\mathcal{N}}}\frac{b(u,v)}{\|v\|_{\mathsf{Y}}}$.

Proof. Recall that S_u is defined as the Riesz representant of b in Y, i.e. $\|b(u,\cdot)\|_{\mathsf{Y}'}=\|S_u\|_{\mathsf{Y}}$. Then,

$$\sup_{v\in\mathsf{Y}^{\mathcal{N}}}\frac{b(u,v)}{\|v\|_{\mathsf{Y}}}=\|S_u\|_{\mathsf{Y}}=\frac{(S_u,S_u)_{\mathsf{Y}}}{\|S_u\|_{\mathsf{Y}}}=\frac{b(u,S_u)}{\|S_u\|_{\mathsf{Y}}}.$$

□

Obviously, $\beta^{\mathcal{N}}\leq\inf_{u\in\mathsf{X}^{\mathcal{N}}}\frac{\|S_u\|_{\mathsf{Y}}}{\|u\|_{\mathsf{X}}}$.

Let $\mathsf{X}^{\mathcal{N}}=\operatorname{span}\{\phi_1,\dots,\phi_{\mathcal{N}}\}$ and $\mathsf{Y}^{\mathcal{N}}=\operatorname{span}\{\psi_1,\dots,\psi_{\mathcal{N}}\}$. For any $u\in\mathsf{X}^{\mathcal{N}}$,

$$\frac{\|S_u\|_{\mathsf{Y}}}{\|u\|_{\mathsf{X}}}=\frac{\mathbf{u}^T\mathbf{B}^T\mathbf{Y}^{-1}\mathbf{B}\mathbf{u}}{\mathbf{u}^T\mathbf{X}\mathbf{u}}$$

where $\mathbf{u}:=(u_i)_{i=1,\dots,\mathcal{N}}$ is the coefficient vector of $u=\sum_{i=1}^{\mathcal{N}}u_i\phi_i$, $\mathbf{B}:=(b(\phi_j,\psi_i))_{i,j=1,\dots,\mathcal{N}}$ is the corresponding cross-Gramian and $\mathbf{Y}$ and $\mathbf{X}$ are the Gramians w.r.t. the given bases. To compute $\beta^{\mathcal{N}}=\inf_{u\in\mathsf{X}^{\mathcal{N}}}\frac{\|S_u\|_{\mathsf{Y}}}{\|u\|_{\mathsf{X}}}$ the generalised eigenvalue problem

$$\mathbf{B}^T\mathbf{Y}^{-1}\mathbf{B}\mathbf{u}=\lambda\mathbf{X}\mathbf{u}$$

can be solved correspondingly. The inf-sup constant $\beta^{\mathcal{N}}$ is given by the square root of the smallest eigenvalue.

2.2.2.2. Crank Nicolson Time Stepping Scheme

Throughout this thesis, the finite element method is used to compute the 'truth' (exact solution of the PDE) that is required for the reduced basis method. Hence, fine discretisations are needed as the model reduction depends on the exacteness of the PDE solutions. To meet these requirements, the FEM solution should be as computationally feasible as possible. Therefore, we introduce a space-time discretisation that is equivalent to solving a Crank Nicolson time stepping scheme. The advantage is that we solve a linear equation system in every time step that only depends on the discrete space dimension. In particular, it is not required to establish the full space-time system. The discretisation does not imply the inf-sup stability in general but a condition at the step-size of the time discretisation has to be satisfied.

Since $L_2(I;V)$ is isomorphic to $L_2(I) \otimes V$ and $H^1(I;V')$ to $H^1(I) \otimes V'$, [Nou14, Sec. 1.5], we get for the trial and test spaces $\mathbb{X}$ and $\mathbb{Y}$

$$\mathbb{X} = L_2(I;V) \cap H^1(I;V') = H^1(I) \otimes V$$

and

$$\mathbb{Y} = L_2(I;V) \times H = (L_2(I) \otimes V) \times H.$$

For the function space V, we consider the finite subspace $V^{\mathcal{J}}$, $\dim(V^{\mathcal{J}}) = \mathcal{J}$, spanned by a nodal basis $\boldsymbol{\phi} := \{\phi_1, \dots, \phi_{\mathcal{J}}\}$ with respect to the triangulation $\mathcal{T}^{\mathcal{J}}_{\text{space}}$ of the underlying space Ω.

For the finite-dimensional temporal subspaces $E^{\mathcal{K}+1} \subset H^1(I)$ and $F^{\mathcal{K}} \subset L_2(I)$, we will consider the triangulation of the interval I given by $\mathcal{T}^{\mathcal{K}}_{\text{time}} := \{t^k = k\Delta t : 0 \leq k \leq \mathcal{K}, \Delta t = \frac{T}{\mathcal{K}}\}$.

Define $E^{\mathcal{K}+1}$ on the trial side by the piecewise linear space $\text{span}\{\boldsymbol{\sigma}\} = \text{span}\{\sigma^0, \dots, \sigma^{\mathcal{K}}\}$ with respect to the triangulation $\mathcal{T}^{\mathcal{K}}_{\text{time}}$. For every $\mathcal{K} > k > 0$, we choose σ^k as the hat function with nodes t^{k-1}, t^k and t^{k+1} and the remaining ones are defined by $\sigma^0 = \frac{t^1 - t}{\Delta t}\chi_{[0,t^1]}$ and $\sigma^{\mathcal{K}} = \frac{t - t^{\mathcal{K}-1}}{\Delta t}\chi_{[t^{\mathcal{K}-1}, t^{\mathcal{K}}]}$, Figure 2.2.1. Furthermore, $F^{\mathcal{K}}$ is chosen as the finite subspace spanned by $\boldsymbol{\tau} = \{\tau^1, \dots, \tau^{\mathcal{K}}\}$ with respect to $\mathcal{T}^{\mathcal{K}}_{\text{time}}$, where τ^k is the characteristic function χ_{I^k} on $I^k := (t^{k-1}, t^k]$. With this preparation at hand, the discrete approximation subspaces of $\mathbb{X}$ and $\mathbb{Y}$ are defined as

$$\mathbb{X}^{\mathcal{N}} := E^{\mathcal{K}+1} \otimes V^{\mathcal{J}} \tag{2.2.15a}$$

and

$$\mathbb{Y}^{\mathcal{N}} := F^{\mathcal{K}} \otimes V^{\mathcal{J}} \times V^{\mathcal{J}} \tag{2.2.15b}$$

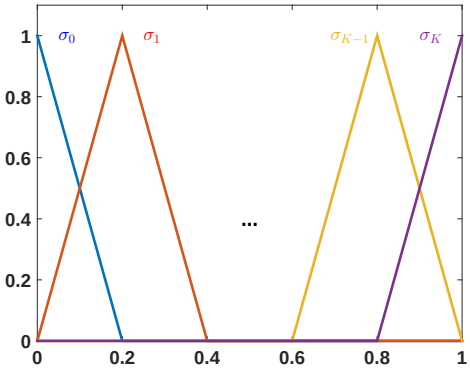


Figure 2.2.1.: Basis functions of $E^{\mathcal{K}+1}$ for $I = (0, 1)$.

where $\dim(E^{\mathcal{K}+1}) = \mathcal{K}+1$, $\dim(F^{\mathcal{K}}) = \mathcal{K}$ and $\dim(V^{\mathcal{J}}) = \mathcal{J}$, hence $\mathcal{N} := \dim(\mathbb{X}^{\mathcal{N}}) = \mathcal{J}(\mathcal{K}+1) = \mathcal{J}\mathcal{K} + \mathcal{J} = \dim(\mathbb{Y}^{\mathcal{N}})$.

Remark 2.2.27. Ansatz and test spaces are thus given by

$$\begin{aligned}\mathbb{Y}^{\mathcal{N}} &= \operatorname{span}\{\ (\tau^\ell \otimes \phi_j, 0), (0, \psi_m) \mid \ell = 1, \dots, \mathcal{K}; j, m = 1, \dots \mathcal{J}\ \} \\ &=: \operatorname{span}\{\ y_i \mid i = 1, \dots, \mathcal{K}\mathcal{J} + \mathcal{J}\ \},\end{aligned}$$

where $(\ell, j) \mapsto i$ for $i = 1, \dots, \mathcal{K}\mathcal{J}$ as well as $m \mapsto i$ for $i = \mathcal{K}\mathcal{J} + 1, \dots, \mathcal{K}\mathcal{J} + \mathcal{J}$ and

$$\begin{aligned}\mathbb{X}^{\mathcal{N}} &= \operatorname{span}\{\ \sigma^\ell \otimes \phi_j \mid \ell = 0, \dots, \mathcal{K}; j = 1, \dots, \mathcal{J}\ \} \\ &=: \operatorname{span}\{\ x_i \mid i = 1, \dots, (\mathcal{K}+1)\mathcal{J}\ \}\end{aligned}$$

with a unique map $(\ell, j) \mapsto i$.

With these notations at hand, the discrete variational formulation (2.2.13) is equivalent to the (large but sparse) quadratic linear system (LS)

$$\mathbf{B}\mathbf{u} = \mathbf{f},$$

where $\mathbf{B} = [b(x_i, y_j)]_{i=1,\dots,(\mathcal{K}+1)\mathcal{J}, j=1,\dots,(\mathcal{K}+1)\mathcal{J}}$ and $\mathbf{f} = (f(y_j))_{j=1,\dots,(\mathcal{K}+1)\mathcal{J}}$. The coefficient vector of the finite element solution $u^{\mathcal{N}} \in \mathbb{X}^{\mathcal{N}}$ is denoted by $\mathbf{u}$. Even though the matrix is sparse, it is very large. To avoid the very expensive tensor evaluation, the time discretisation is evaluated directly and we show that the Crank Nicolson time-stepping scheme is an equivalent problem formulation.

Time stepping For given functions $w = \sum_{k=0}^{\mathcal{K}} \sum_{i=1}^{\mathcal{J}} w_i^k (\sigma^k \otimes \phi_i) \in \mathbb{X}^{\mathcal{N}}$ and $v = (z, h) = (\sum_{k=1}^{\mathcal{K}} \sum_{j=1}^{\mathcal{J}} z_j^k (\tau^k \otimes \phi_j), \sum_{m=1}^{\mathcal{J}} h_m \psi_m) \in \mathbb{Y}^{\mathcal{N}}$, we obtain, cf. [UP14, And12, MU14],

$$\begin{aligned} b(w,v) &= \int_I \langle \dot{w}(t), z(t)\rangle_{V'\times V} + a(w(t), z(t))dt + (w(0), h)_H \\ &= \sum_{k=0}^{\mathcal{K}} \sum_{l=1}^{\mathcal{K}} \sum_{i,j=1}^{\mathcal{J}} w_i^k z_j^l \int_I \langle \dot{\sigma}^k(t)\phi_i, \tau^l(t)\phi_j\rangle_{V'\times V} + a(\sigma^k(t)\phi_i, \tau^l(t)\phi_j)dt \\ &\quad + \sum_{k=0}^{\mathcal{K}} \sum_{i,j=1}^{\mathcal{J}} w_i^k h_j (\sigma^k(0)\phi_i, \phi_j)_H \\ &= \sum_{k=0}^{\mathcal{K}} \sum_{l=1}^{\mathcal{K}} \sum_{i,j=1}^{\mathcal{J}} w_i^k z_j^l (\dot{\sigma^k}, \tau^l)_{L_2(I)} (\phi_i, \phi_j)_H + (\sigma^k, \tau^l)_{L_2(I)} a(\phi_i, \phi_j) \\ &\quad + \sum_{i,j=1}^{\mathcal{J}} w_i^0 h_j (\phi_i, \phi_j)_H. \end{aligned}$$

For $k \geq 0$ and $\ell \geq 1$ we have $(\dot{\sigma^k}, \tau^\ell)_{L_2(I)} = \delta_{k,\ell} - \delta_{k+1,\ell}$ and $(\sigma^k, \tau^\ell)_{L_2(I)} = \frac{\Delta t}{2}(\delta_{k,\ell} + \delta_{k+1,\ell})$, in particular $(\dot{\sigma^0}, \tau^\ell)_{L_2(I)} = -\delta_{1,\ell}$ and $(\sigma^0, \tau^\ell)_{L_2(I)} = \frac{\Delta t}{2}\delta_{1,\ell}$. Set

$$\mathbf{M}_{\text{space}}^{\mathcal{J}} := ((\phi_i, \phi_j)_H)_{j,i=1,\dots,\mathcal{J}} \quad \text{and} \quad \mathbf{A}_{\text{space}}^{\mathcal{J}} := (a(\phi_i, \phi_j))_{j,i=1,\dots,\mathcal{J}}.$$

Remark 2.2.28. The initial condition u_0 is left in a general setting as an element of the Hilbert space H. The numerical treatment of the initial value will be discussed in Section 3.2. So, we do not specify the discretisation of the initial value on the right-hand side in detail but assume that u_0 has a finite expansion $\sum_{\ell=1}^{\mathcal{L}} u_0^\ell \delta^\ell$, $0 < \mathcal{L} < \infty$.

Let $v = (\tau^\ell \otimes \phi_j, 0)$ for some ℓ (≥ 1) and $j \in \{1, \dots, \mathcal{J}\}$. We obtain

$$\begin{aligned} b(w, (\tau^\ell \otimes \phi_j, 0)) &= \sum_{k=0}^{\mathcal{K}} \sum_{i=1}^{\mathcal{J}} \left[w_i^k (\dot{\sigma^k}, \tau^\ell)_{L_2(I)} (\phi_i, \phi_j)_H + (\sigma^k, \tau^\ell)_{L_2(I)} a(\phi_i, \phi_j) \right] \\ &= \left[\mathbf{M}_{\text{space}}^{\mathcal{J}} (\mathbf{w}^\ell - \mathbf{w}^{\ell-1}) + \mathbf{A}_{\text{space}}^{\mathcal{J}} \frac{\Delta t}{2} (\mathbf{w}^\ell + \mathbf{w}^{\ell-1}) \right]_j \end{aligned}$$

with the coefficient vectors $\mathbf{w}^\ell = (w_i^\ell)_{i=1,\dots,\mathcal{J}}$ for all $\ell = 0, \dots, \mathcal{K}$. On the right-hand side, we use a trapezoidal approximation

$$\begin{aligned} f((\tau^\ell \otimes \phi_j, 0)) &= \int_I \langle g(t), \tau^\ell \otimes \phi_j(t, .)\rangle_{V'\times V} dt = \int_{I^\ell} \langle g(t), \tau^\ell(t)\phi_j\rangle_{V'\times V} dt \\ &\approx \frac{\Delta t}{2} \langle g(t^{\ell-1}) + g(t^\ell), \phi_j\rangle_{V'\times V}. \end{aligned} \tag{2.2.16}$$

The remaining basis functions of the test space are time-independent: For any $m = 1, \dots, \mathcal{J}$, we

have

$$b(w,(0,\phi_m)) = \sum_{i,j=1}^{\mathcal{J}} w_i^0(\phi_i,\phi_m)_H = [\mathbf{M}_{\text{space}}^{\mathcal{J}}\mathbf{w}^0]_m$$

and

$$f((0,\phi_m)) = (u_0,\phi_m)_H = \sum_{\ell=1}^{\mathcal{L}} u_0^\ell(\delta^\ell,\phi_m)_H =: [\mathbf{u_0}]_m.$$

The assumption of Remark 2.2.28 on the finite representation of the initial condition is needed to evaluate the scalar product in H numerically. As a consequence, we get the following Crank-Nicolson scheme

$$\mathbf{M}_{\text{space}}^{\mathcal{J}}\mathbf{w}^0 = \mathbf{u_0}, \tag{2.2.17a}$$

$$(2\mathbf{M}_{\text{space}}^{\mathcal{J}} + \Delta t\mathbf{A}_{\text{space}}^{\mathcal{J}})\mathbf{w}^\ell = (2\mathbf{M}_{\text{space}}^{\mathcal{J}} - \Delta t\mathbf{A}_{\text{space}}^{\mathcal{J}})\mathbf{w}^{\ell-1} + \Delta t\langle g(t^{\ell-1}) + g(t^\ell),\phi_j\rangle_{V'\times V}, \quad \ell = 1,\dots,\mathcal{K}. \tag{2.2.17b}$$

Introducing implicit and explicit operators, $\mathbf{L}_{\text{im}}$ and $\mathbf{L}_{\text{ex}}$,

$$\begin{aligned}\mathbf{L}_{\text{im}} &= 2\mathbf{M}_{\text{space}}^{\mathcal{J}} + \Delta t\mathbf{A}_{\text{space}}^{\mathcal{J}},\\ \mathbf{L}_{\text{ex}} &= 2\mathbf{M}_{\text{space}}^{\mathcal{J}} - \Delta t\mathbf{A}_{\text{space}}^{\mathcal{J}},\\ \mathbf{b}^{\ell-1} &= \Delta t\langle g(t^{\ell-1}) + g(t^\ell),\phi_j\rangle_{V'\times V}\end{aligned}$$

and in every time step the following system of equations is to be solved

$$\mathbf{L}_{\text{im}}\mathbf{w}^\ell = \mathbf{L}_{\text{ex}}\mathbf{w}^{\ell-1} + \mathbf{b}^{\ell-1}.$$

Stability For symmetric bilinear forms $a : V \times V \to \mathbb{R}$, R. Andreev showed in [And12, Sec. 5.2.3 A] that the inf-sup constant for the chosen discretisation w.r.t. the presented space-time norms $\|\cdot\|_{\mathbb{X}}$ and $\|\cdot\|_{\mathbb{Y}}$ is bounded from below (i.e. $\inf_{\mathcal{N}}\beta^{\mathcal{N}} > 0$) by

$$K_{V'\times V}(V^{\mathcal{J}},V^{\mathcal{J}}) := \inf_{0\neq u\in V^{\mathcal{J}}}\sup_{0\neq v\in V^{\mathcal{J}}}\frac{\langle u,v\rangle_{V'\times V}}{|u|_V|v|_V} > 0, \tag{2.2.18}$$

if the CFL[1] condition

$$\text{CFL}_{\mathcal{N}} := \max\Delta\mathcal{T}_{\text{time}}^{\mathcal{K}}\sup_{0\neq\psi\in V^{\mathcal{J}}}\frac{|\psi|_V}{|\psi|_{V'}} > 0$$

is satisfied for $\mathcal{N} > 0$. Here, $\Delta\mathcal{T}_{\text{time}}^{\mathcal{K}} := \{t^k - t^{k-1} : 0 \le k \le \mathcal{K}\}$. For the triangulation $\mathcal{T}_{\text{time}}^{\mathcal{K}}$ applies $\max\Delta\mathcal{T}_{\text{time}}^{\mathcal{K}} = \Delta t = \frac{T}{\mathcal{K}}$. The results were first published in [And13] and detailed again in [And14] for the special case of the heat equation.

[1] Denotes in general conditions for numerical stability, called after R. Courant, K. Friedrichs and H. Lewy.

Proposition 2.2.29. [And14, Prop. 2] Let $V^{\mathcal{J}} \subset V$, $E^{\mathcal{K}+1} \subset H^1(I)$ and $F^{\mathcal{K}} \subset L_2(I)$ be defined as above. Then, there exists a constant $\beta^0 > 0$, independent of $V^{\mathcal{J}}$, $E^{\mathcal{K}+1}$ and $F^{\mathcal{K}}$, such that the discrete inf-sup condition (2.2.14) holds for $\mathbb{X}^{\mathcal{N}}$ and $\mathbb{Y}^{\mathcal{N}}$ given in (2.2.15) with

$$\beta^{\mathcal{N}} > \beta^0 K_{V' \times V}(V^{\mathcal{J}}, V^{\mathcal{J}}) \min\{1, \mathrm{CFL}_{\mathcal{N}}^{-1}\}.$$

□

Guaranteeing (inf-sup) stability of the discretised problem is equivalent to requirements for the discretisation in time and the discretisation in space separately. In time, we have to guarantee $\mathrm{CFL}_{\mathcal{N}}^{-1} > 0$, i.e. the discretisation has to be chosen 'fine enough'. The CFL condition can not be improved in general for the space-time discretisation that is equivalent to the Crank Nicolson time-stepping scheme (cf. [And12, Ex. 5.2.12]). In space, we have to guarantee $K_{V' \times V}(V^{\mathcal{J}}, V^{\mathcal{J}}) > 0$.

The next proposition shows the connection between $K_{V' \times V}$ and the H-orthogonal projection from V to its closed subspace $V^{\mathcal{J}}$.

Proposition 2.2.30. [And12, Prop. 4.4.8] Let $V \hookrightarrow H \cong H' \hookrightarrow V'$ be a Gelfand triple of Hilbert spaces. Let Q be the H-orthogonal projector onto a closed subspace $U \subset V$. Then, for any $\kappa > 0$ the following are equivalent

(a) $K_{V' \times V}(U, U) \geq \kappa$

(b) $|Qv|_V \leq \kappa^{-1} |v|_V$ for all $v \in V$.

We give an example of a stable finite element discretisation of $H_0^1(D)$, $D \subset \mathbb{R}^d$. This is the d-dimensional generalisation of the discretisation used in [And12, Ch. 8].

Example 2.2.31. Let $D = (-1, 1)^d \subset \mathbb{R}^d$ and define $V := H_0^1(D)$, $H := L_2(D)$. Consider the tensor product spaces isomorphic to V and H, i.e. $H_0^1(D) \cong H_0^1(D^{(1)}) \otimes \ldots \otimes H_0^1(D^{(d)})$ and $H := L_2(D) \cong L_2(D^{(1)}) \otimes \ldots \otimes L_2(D^{(d)})$, $D^{(i)} = (-1, 1)$ for all $i = 1, \ldots, d$, cf. [Hac12a, Ch. 4]. To ease the notation, we consider on all tensor subspaces an equidistant grid $\mathcal{T}^\ell = \{x_k = -1 + k\Delta t : 0 \leq k \leq 2^{\ell+1}, \Delta t = \frac{2}{2^{\ell+1}}\}$.

On each space $H_0^1(D^{(i)})$, let $\boldsymbol{\phi}_i$ be the finite element basis analogously defined to the one for $E^{\mathcal{K}+1}$, i.e. consider piecewise linear hat functions on $\mathcal{T}^\ell$, setting $\sigma^0 \equiv \sigma^{2^{\ell+1}} \equiv 0$ to ensure $\boldsymbol{\phi}_i \subset H_0^1(D^{(i)})$, $|\boldsymbol{\phi}_i| = 2^{\ell+1} - 1$. The tensorised basis $\boldsymbol{\phi} = \boldsymbol{\phi}_1 \otimes \ldots \otimes \boldsymbol{\phi}_d$ is a tensor product finite element basis of $V^{\mathcal{J}}$, $\mathcal{J} := (2^{\ell+1} - 1)^d$, [BS94, (3.5.2)].

Claim. There exists $\kappa > 0$ with $K_{V' \times V}(V^{\mathcal{J}}, V^{\mathcal{J}}) \geq \kappa$.

Proof. Let $d = 1$, [And12, Sec. 8.2]. Let $Q_\ell : L_2(D) \to V_\ell$ and $R_\ell : H_0^1(D) \to V_\ell$ denote the $L_2(D)$-orthogonal and $H_0^1(D)$-orthogonal surjective projectors. For $C_{\mathrm{dir}}, C_{\mathrm{inv}} > 0$, the direct estimate, [BS94, (4.4.20) Thm.],

$$|\xi - R_\ell \xi|_{L_2(D)} \leq C_{\mathrm{dir}} 2^{-\ell} |\xi|_{H_0^1(D)} \quad \forall\, \xi \in H_0^1(D) \quad \forall\, \ell \in \mathbb{N}_0$$

and the inverse estimate, [BS94, (4.5.11) Thm.],

$$|\xi_\ell|_{H_0^1(D)} \leq C_{\mathrm{inv}} 2^{\ell} |\xi_\ell|_{L_2(D)} \quad \forall \xi \in V_\ell \quad \forall \ell \in \mathbb{N}_0$$

imply stability of the $L_2(D)$-orthogonal projector Q_ℓ in $H_0^1(D)$ uniformly in $\ell \in \mathbb{N}_0$: There exists $C_{\mathrm{stab}} > 0$ such that $|Q_\ell \xi|_{H_0^1(D)} \leq C_{\mathrm{stab}} |\xi|_{H_0^1(D)}$ for all $\ell \in \mathbb{N}_0$, $\xi \in H_0^1(D)$, [And12, Lemma 8.2.1].

For the general case $d > 1$ we use the tensor product representation of $v \in H_0^1(D)$ given by $v_1 \otimes \ldots \otimes v_d$. Since, [AT14, p. 8],

$$|v_1 \otimes \ldots \otimes v_d|^2_{H_0^1(D)} = \sum_{\mu=1}^{d} \Big(|v_\mu|^2_{H_0^1(D^{(\mu)})} \prod_{\mu \neq \mu' = 1}^{d} |v_{\mu'}|^2_{L_2(D^{(\mu')})} \Big), \tag{2.2.19}$$

we get for $C_{\mathrm{stab}} > 0$ as above

$$\begin{aligned} C^2_{\mathrm{stab}} |v|^2_{H_0^1(D)} = C^2_{\mathrm{stab}} |v_1 \otimes \ldots \otimes v_d|^2_{H_0^1(D)} &= \sum_{\mu=1}^{d} (C^2_{\mathrm{stab}} |v_\mu|^2_{H_0^1(D^{(\mu)})} \prod_{\mu \neq \mu' = 1}^{d} |v_{\mu'}|^2_{L_2(D^{(\mu')})}) \\ &\geq \sum_{\mu=1}^{d} (|Q_\ell v_\mu|^2_{H_0^1(D^{(\mu)})} \prod_{\mu \neq \mu' = 1}^{d} |Q_\ell v_{\mu'}|^2_{L_2(D^{(\mu')})}) = |Q_\ell v|^2_{H_0^1(D)}. \end{aligned}$$

Taking the square root on both sides leads to

$$|Q_\ell v|_{H_0^1(D)} \leq C_{\mathrm{stab}} |v|_{H_0^1(D)}.$$

With the application of Proposition 2.2.30 the lower bound $\kappa := C^{-1}_{\mathrm{stab}} > 0$ follows. □

A further possibility is to introduce a natural norm on the discretised spaces, [UP14, And12]. The norm is natural in the sense that it avoids the (in general not defined) point evaluation of the considered Lebesgue function (equivalence class representant). For $w \in \mathbb{X}$ set $\bar{w}^k := (\Delta t)^{-1} \int_{I^k} w(t) dt \in V$ and $\bar{w} := \sum_{k=1}^{\mathcal{K}} \tau_k \otimes \bar{w}^k \in L_2(I; V)$, then

$$\|w\|^2_{\mathbb{X},\mathrm{bar}} := \|\bar{w}\|^2_{L_2(I;V)} + \|\dot{w}\|^2_{L_2(I;V')} + |w(T)|^2_H, \tag{2.2.20}$$

$$\beta^{\mathcal{N},\mathrm{bar}} := \inf_{w^{\mathcal{N}} \in \mathbb{X}^{\mathcal{N}}} \sup_{v^{\mathcal{N}} \in \mathbb{Y}^{\mathcal{N}}} \frac{b(w^{\mathcal{N}}, v^{\mathcal{N}})}{\|w^{\mathcal{N}}\|_{\mathbb{X},\mathrm{bar}} \|v^{\mathcal{N}}\|_{\mathbb{Y}}},$$

$$\gamma^{\mathcal{N},\mathrm{bar}} := \sup_{w^{\mathcal{N}}\in\mathbb{X}^{\mathcal{N}}} \sup_{v^{\mathcal{N}}\in\mathbb{Y}^{\mathcal{N}}} \frac{b(w^{\mathcal{N}}, v^{\mathcal{N}})}{\|w^{\mathcal{N}}\|_{\mathbb{X},\mathrm{bar}}\|v^{\mathcal{N}}\|_{\mathbb{Y}}}.$$

For $\|\cdot\|_{\mathbb{X},\mathrm{bar}}$ as norm on $\mathbb{X}^{\mathcal{N}}$, the following proposition shows stability for $b(\cdot,\cdot)$. The proposition is a straightforward extension of [UP14, Prop. 2.9].

Proposition 2.2.32. Let $a(.,.)$ be symmetric, bounded and coercive and set $|\phi|_V^2 := a(\phi,\phi)$, $\phi \in V$ then $\beta^{\mathcal{N},\mathrm{bar}} = C_b^{\mathcal{N},\mathrm{bar}} = 1$.

Proof. Extension of [UP14, Prop. 2.9] for non-homogeneous initial condition. □

2.2.2.3. Minimal Residual Approach

So far, the chosen time discretisation in trial and test space results in the equivalence to the Crank Nicolson scheme. Unfortunately we get the dependency on the CFL condition. More general approaches allow for unconditionally stable pairs of discrete trial and test spaces. These generalised time discretisations were subject of [And12]. It was shown that, as long as some embedding results for the discretised spaces are valid, we obtain inf-sup stability for the discretised pair of trial and test spaces, [And12, Sec. 5.2.3 B]. The dimension of the discrete trial and test spaces are not equal any more and least square methods have to be applied, [And12, Ch. 4]. Thus, instead of the usual discrete variational formulation (2.2.13), the discrete (functional) residual minimisation problem will be solved, i.e.

$$\text{find } u^{\mathcal{N}} \in \mathbb{X}^{\mathcal{N}}: \quad u^{\mathcal{N}} = \arg\min_{w^{\mathcal{N}}\in\mathbb{X}^{\mathcal{N}}} \sup_{v^{\mathcal{M}}\in\mathbb{Y}^{\mathcal{M}}\setminus\{0\}} \frac{|b(w^{\mathcal{N}}, v^{\mathcal{M}}) - f(v^{\mathcal{M}})|}{|v^{\mathcal{M}}|_{\mathbb{Y}}}. \tag{2.2.21}$$

Well-posedness of the minimal residual approach is provided in the following theorem.

Theorem 2.2.33. *[And12, Thm. 4.1.9] Let* X, Y *be real Hilbert spaces,* $b : \mathsf{X}\times\mathsf{Y} \to \mathbb{R}$ *a bilinear form and* $\mathsf{X}^{\mathcal{N}} \times \mathsf{Y}^{\mathcal{M}} \subseteq \mathsf{X}\times\mathsf{Y}$ *be a non-trivial pair of subspaces*[2] *that satisfies the discrete inf-sup condition for* b*,*

$$\inf_{u^{\mathcal{N}}\in\mathsf{X}^{\mathcal{N}}\setminus\{0\}} \sup_{v^{\mathcal{M}}\in\mathsf{Y}^{\mathcal{M}}\setminus\{0\}} \frac{|b(u^{\mathcal{N}}, v^{\mathcal{M}})|}{\|u^{\mathcal{N}}\|_{\mathsf{X}}\|v^{\mathcal{M}}\|_{\mathsf{Y}}} =: \beta^{(\mathcal{N},\mathcal{M})} > 0. \tag{2.2.22}$$

Then, for any $u \in \mathsf{X}$ *there exists a unique* $u^{\mathcal{N}} \in \mathsf{X}^{\mathcal{N}}$ *which satisfies*

$$u^{\mathcal{N}} = \arg\inf_{w^{\mathcal{N}}\in\mathsf{X}^{\mathcal{N}}} \Big(\sup_{v^{\mathcal{M}}\in\mathsf{Y}^{\mathcal{M}}\setminus\{0\}} \frac{|b(w^{\mathcal{N}} - u, v^{\mathcal{M}})|}{\|v^{\mathcal{M}}\|_{\mathsf{Y}}} \Big) \tag{2.2.23}$$

for u *solving the non-discretised PDE. Moreover, there holds the quasi-optimality estimate*

$$\|u - u^{\mathcal{N}}\|_{\mathsf{X}} \le \frac{C_b}{\beta^{(\mathcal{N},\mathcal{M})}} \inf_{w^{\mathcal{N}}\in\mathsf{X}^{\mathcal{N}}} \|u - w^{\mathcal{N}}\|_{\mathsf{X}}. \tag{2.2.24}$$

[2] $\mathsf{X}^{\mathcal{N}} \neq \{0\}$, $\mathsf{Y}^{\mathcal{M}} \neq \{0\}$

The constant C_b denotes the (discrete) continuity constant of the bilinear form b.

Proof. For a proof we refer to the proof of [And12, Thm. 4.1.9]. □

2.2.3. Introduction to the Reduced Basis Method (RBM)

Up to now, we considered parabolic PDEs or, more general, evolutionary problems without focus on the parameters that probably define the (bi)linear forms. If such model parameters vary often, an efficient model reduction scheme should be applied to avoid computationally expensive recomputations of the same problem repeatedly. Especially if parameters change in real time problems, an application of a model reduction method is not optional. The basic idea of the reduced basis method appeared in the late 70s and early 80s. For a historical overview on the early work we refer to [Rhe93]. In first approaches, parameter dependent systems with only one parameter were considered. Steps have been taken towards a-posteriori error estimates and the approach was extended to multi-parameter problems, cf. [FR83, Rhe93, BR95]. The reduced basis method as used in this thesis was introduced in a series of papers in the beginning of the 20th century, cf. [MPR01, PRV$^+$02, VPRP03] and [PR07] and references therein. Since then, the reduced basis method was adapted to very different applications and a wide field of partial differential equations.

2.2.3.1. The Reduced Basis Method

We present the standard RB procedure in the upcoming section. Parts of this chapter were published in [MU14, Section 2.2]. We recall the main properties of the RBM. For details, we refer to the survey article [Haa14] and the books [PR07, QMN15, HRS15].

Let X and Y be (infinite-dimensional) Hilbert spaces. Let $\mathcal{D} \subset \mathbb{R}^p$, $p > 0$, be the parameter space[3]. Let $b : \mathsf{X} \times \mathsf{Y} \times \mathcal{D} \to \mathbb{R}$ be a parametric bilinear form on $\mathsf{X} \times \mathsf{Y}$ and $f : \mathsf{Y} \times \mathcal{D} \to \mathbb{R}$ a parametric linear functional, $f(\cdot;\mu) \in \mathsf{Y}'$ for $\mu \in \mathcal{D}$. For a given parameter $\mu \in \mathcal{D}$, we are interested in solving the problem:

$$\text{Find } u(\mu) \in \mathsf{X}: \quad b(u(\mu), v; \mu) = f(v;\mu) \quad \forall v \in \mathsf{Y}. \tag{2.2.25}$$

Well-posedness of (2.2.25) is assumed for all $\mu \in \mathcal{D}$ throughout this section.

[3] A more general setting is considered in Chapter 3.2.

Example 2.2.34. Let $\mathcal{D} \subset \mathbb{R}^d$ be the parameter space. For the special case of parabolic μPDEs, we define the ansatz space $\mathsf{X} := \mathbb{X}$, cf. Equation (2.2.1) and the test space $\mathsf{Y} := \mathbb{Y}$ as defined in Equations (2.2.5) and define for $w \in \mathbb{X}$, $v = (z, h) \in \mathbb{Y}$ and $\mu \in \mathcal{D}$

$$b(w, v; \mu) := \int_I \langle \dot{w}(t), z(t) \rangle_{V' \times V} \, dt + \int_I a(w(t), z(t); \mu) dt + (w(0), h)_H$$
$$f(v; \mu) := \int_I \langle g(t; \mu), z(t) \rangle_{V' \times V} \, dt + (u_0, h)_H.$$

This is the space-time variational formulation given in Equation (2.2.8), with the explicit parameter dependence detailed in $a(\cdot, \cdot; \cdot)$ and on the right-hand side.

The next step is the assumption that a finite discretisation $\mathsf{X}^{\mathcal{N}} \subset \mathsf{X}$, $\mathsf{Y}^{\mathcal{N}} \subset \mathsf{Y}$ available, in the sense that the discrete problem formulation

$$\text{find } u^{\mathcal{N}}(\mu) \in \mathsf{X}^{\mathcal{N}} : \quad b(u^{\mathcal{N}}(\mu), v^{\mathcal{N}}; \mu) = f(v^{\mathcal{N}}; \mu) \quad \forall v^{\mathcal{N}} \in \mathsf{Y}^{\mathcal{N}}$$

is still well-posed.

The important observation for the RBM is that we are not interested in the full space X but only in the subspace $\mathsf{M} := \{u(\mu) \in \mathsf{X} : \mu \in \mathcal{D}\}$. Thus, we are going to replace the approximation space $\mathsf{X}^{\mathcal{N}}$ by a much smaller approximation space $\mathsf{X}_N \subset \mathsf{X}^{\mathcal{N}}$ for M. For the construction of X_N we use $\mathsf{X}^{\mathcal{N}}$: in the *offline* phase we compute *high dimensional* or often called *detailed* or *truth* solutions $u^{\mathcal{N}}(\mu) \in \mathsf{X}^{\mathcal{N}} \subset \mathsf{X}$ with standard numerical methods, like the finite element method. A standard assumption is that the detailed solution is a good approximation of the true problem solution in X. We construct an RB as a set of detailed solutions $u^i := u^{\mathcal{N}}(\mu^i)$, so-called *snapshots*, for a given parameter sample set $\mu^1, \dots, \mu^N \in \mathcal{D}$ that span a subspace X_N of the detailed approximation space $\mathsf{X}^{\mathcal{N}}$.

The model reduction takes place in the *online* phase. We consider the ansatz space $\mathsf{X}_N := \operatorname{span}\{u^i : i = 1, \dots, N\}$ and a suitable (reduced) test space $\mathsf{Y}_N \subset \mathsf{Y}^{\mathcal{N}}$ (for simplicity both of dimension N). The reduced problem formulation reads

$$\text{find } u_N(\mu) \in \mathsf{X}_N : \quad b(u_N(\mu), v_N; \mu) = f(v_N; \mu) \quad \forall v_N \in \mathsf{Y}_N. \tag{2.2.26}$$

If $N \ll \mathcal{N}$, the reduced model is feasible online even though the stiffness matrix of the system on the left-hand side is in general densely populated. To guarantee existence and uniqueness of the RB solution, the reduced problem formulation again has to be well-posed. For a visualisation, we refer to Figure 2.2.2.

Remark 2.2.35. By construction of X_N, the approximation error of the detailed solution is a lower

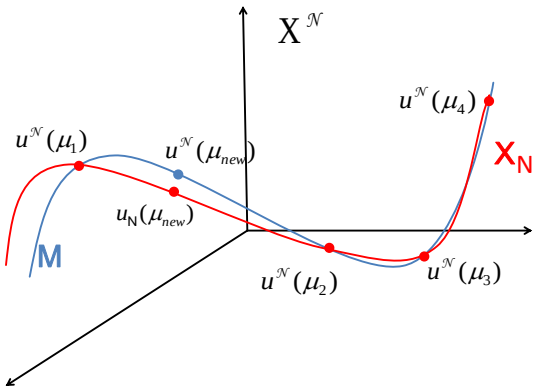


Figure 2.2.2.: The reduced basis approximation.

bound of the RB approximation error:

$$\inf_{u^{\mathcal{N}} \in \mathsf{X}^{\mathcal{N}}} \|u - u^{\mathcal{N}}\|_{\mathsf{X}} \leq \inf_{u_N \in \mathsf{X}_N} \|u - u_N\|_{\mathsf{X}}.$$

Thus, the approximation quality of the RB solution can not be better than the one of the discrete detailed solution. In addition, the RB approximation error is bounded by $\|u - u_N\| \leq \|u - u^{\mathcal{N}}\| + \|u^{\mathcal{N}} - u_N\|$. If we assume $u^{\mathcal{N}}(\mu)$ to be a sufficiently good approximation of $u(\mu)$, the focus lies on the error between $u^{\mathcal{N}}(\mu)$ and $u_N(\mu)$.

Remark 2.2.36. In the space-time setting of Example 2.2.34, well-posedness is related to an inf-sup condition, Section 2.2.1.3, that has to be satisfied for the entire parameter space, i.e.

$$\inf_{\mu \in \mathcal{D}} \inf_{w \in \mathbb{X}} \sup_{v \in \mathbb{Y}} \frac{b(w, v; \mu)}{\|w\|_{\mathbb{X}} \|v\|_{\mathbb{Y}}} \geq \beta > 0. \tag{2.2.27}$$

A corresponding discrete inf-sup condition has again to hold true for the detailed discretisation and again independent of the parameter space:

$$\inf_{\mu \in \mathcal{D}} \inf_{w^{\mathcal{N}} \in \mathbb{X}^{\mathcal{N}}} \sup_{v^{\mathcal{N}} \in \mathbb{Y}^{\mathcal{N}}} \frac{b(w^{\mathcal{N}}, v^{\mathcal{N}}; \mu)}{\|w^{\mathcal{N}}\|_{\mathbb{X}} \|v^{\mathcal{N}}\|_{\mathbb{Y}}} \geq \beta^{\mathcal{N}} > 0 \quad \forall \mathcal{N} \in \mathbb{N}. \tag{2.2.28}$$

The problem formulation in the reduced basis spaces has thus also to satisfy stability in a uniform inf-sup-sense, i.e.,

$$\inf_{\mu \in \mathcal{D}} \inf_{w_N \in \mathbb{X}_N} \sup_{v_N \in \mathbb{Y}_N} \frac{b(w_N, v_N; \mu)}{\|w_N\|_{\mathbb{X}} \|v_N\|_{\mathbb{Y}}} \geq \beta_N > 0 \quad \forall N \in \mathbb{N}. \tag{2.2.29}$$

Explicit constructions of a test space for the online procedure of parabolic PDEs in space-time variational formulation are described in Section 2.3.

The ultimate efficiency aim is to realise an online complexity that is independent of $\mathcal{N}$. In

particular, all quantities that depend on $\mathcal{N}$ and that are necessary for solving the system of linear equations in (2.2.26) have to be precomputed in the offline phase. The key for that realisation is the assumption that bilinear form and right-hand side functional are separable w.r.t. the parameter, i.e.,

$$b(w,v;\mu) = \sum_{q=1}^{Q_b} \theta_q^b(\mu)\, b_q(w,v), \qquad f(v;\mu) = \sum_{q=1}^{Q_f} \theta_q^f(\mu) f_q(v), \tag{2.2.30}$$

with functions θ_q^b, $\theta_q^f : \mathcal{D} \to \mathbb{R}$ and parameter-independent forms $b_q : \mathsf{X} \times \mathsf{Y} \to \mathbb{R}$ and $f_q : \mathsf{Y} \to \mathbb{R}$. In the RB literature this is often called *affine decomposition* w.r.t. the parameter. For models that do not naturally provide this decomposition, there are strategies like the Empirical Interpolation Method (EIM) that approximate this structure for the use in the reduced basis method, cf. [BMNP04].

Let $\{v^j : 1 \le j \le N\}$ be a basis of the space Y_N. The reduced problem formulation (2.2.26) reads

$$\text{find } \mathbf{u}_N := (u_{N,i})_{i=1,\dots,N} \in \mathbb{R}^N : \sum_{i=1}^{N} u_{N,i}\, b(u^i, v^j; \mu) = f(v^j;\mu) \quad \forall j = 1,\dots,N. \tag{2.2.31}$$

Furthermore, let $u^i = \sum_{n=1}^{\mathcal{N}} \alpha_n^i x_n$ be the representation of the snapshots in a basis $\{x_n : n = 1,\dots,\mathcal{N}\}$ of $\mathsf{X}^{\mathcal{N}}$ and correspondingly $v^j = \sum_{n=1}^{\mathcal{N}} \beta_n^j y_n$ in a basis $\{y_n : n = 1,\dots,\mathcal{N}\}$ of $\mathsf{Y}^{\mathcal{N}}$. If (2.2.30) holds, the computation of the stiffness matrix and the right-hand side of the reduced linear system can be split in an offline/online fashion as follows $(i,j, = 1,\dots,N)$

$$\begin{aligned} b(u^i, v^j;\mu) &= \sum_{n,n'=1}^{\mathcal{N}} \alpha_n^i \beta_{n'}^j\, b(x_n, y_{n'};\mu) = \sum_{n,n'=1}^{\mathcal{N}} \alpha_n^i \beta_{n'}^j \sum_{q=1}^{Q_b} \theta_q^b(\mu)\, b_q(x_n, y_{n'}) \\ &= \sum_{q=1}^{Q_b} \theta_q^b(\mu) \sum_{n,n'=1}^{\mathcal{N}} \alpha_n^i \beta_{n'}^j\, b_q(x_n, y_{n'}) =: \sum_{q=1}^{Q_b} \theta_q^b(\mu) [\mathbf{B}_q]_{i,j}, \end{aligned}$$

$$f(v^j;\mu) = \sum_{n=1}^{\mathcal{N}} \beta_n^j f_q(y_n;\mu) = \sum_{q=1}^{Q_f} \theta_q^f(\mu) \sum_{n=1}^{\mathcal{N}} \beta_n^j f_q(y_n) =: \sum_{q=1}^{Q_f} \theta_q^f(\mu) (\mathbf{f}_q)_j.$$

The terms $[\mathbf{B}_q]_{i,j}$ and $(\mathbf{f}_q)_j$ are μ-independent and can thus be computed offline with complexity $\mathcal{O}(N^2 \mathcal{N} Q_b)$ resp. $\mathcal{O}(N \mathcal{N} Q_f)$. Online, for a new parameter $\mu \in \mathcal{D}$, the stiffness matrix of the RB system is obtained by computing:

$$[\mathbf{B}_N(\mu)]_{i,j} = b(u^i, v^j;\mu) = \sum_{q=1}^{Q_b} \theta_q^b(\mu) [\mathbf{B}_q]_{i,j}, \qquad i,j = 1,\dots N, \tag{2.2.32}$$

which requires an online-complexity of $\mathcal{O}(Q_b N^2)$ that is independent of $\mathcal{N}$ (resp. $\mathcal{O}(Q_f N)$ for the right-hand side).

Example 2.2.37. For $\mathsf{X} = \mathsf{Y} = H_0^1(0, 2\pi)$ we consider the heat equation

$$-\mu\Delta u = \sin(x) \quad x \in \Omega = (0, 2\pi), \mu \in [0.5, 1.5].$$

We assume homogeneous Dirichlet boundary conditions. The weak formulation reads: Find $u \in H_0^1(0, 2\pi)$ such that

$$\mu \int_{(0,2\pi)} \nabla u(x) \nabla v(x) dx = \int_{(0,2\pi)} \sin(x) v(x) dx \quad \forall\, v \in H_0^1(0, 2\pi).$$

The problem formulation is well-posed with coercivity constant $\inf_{u \in H_0^1(0,2\pi)} \frac{b(u,u;\mu)}{|u|^2_{H_0^1}} = \alpha(\mu) = \mu$ since $b(u, u; \mu) = \mu \int_{(0,2\pi)} \nabla u(x) \nabla u(x) dx = \mu |u|_{H_0^1}$. We skip the high dimensional discretisation as we can directly compute the solution for the parameter $\mu = 1$. We set the basis function $u^1 := u(1) = \sin(x)$ and define the one dimensional RB approximation spaces by $\mathsf{X}_1 := \operatorname{span}\{u^1\} =: \mathsf{Y}_1$. Then,

$$\mathbf{B}_1 := b(u^1, u^1) = \int_{(0,2\pi)} \cos^2(x) dx = \pi, \quad \mathbf{f}_1 := f(u^1) = \int_{(0,2\pi)} \sin^2(x) dx = \pi.$$

For the new parameter $\mu = 0.5$ we solve

$$0.5 \mathbf{B}_1 \mathbf{u}_1 = \mathbf{f}_1 \Leftrightarrow \mathbf{u}_1 = \frac{1}{0.5} = 2.$$

Hence, the solution is given by $u_1(0.5) = 2u^1 = 2\sin(x)$. Verifying, that $-0.5\Delta 2\sin(x) = -\nabla\cos(x) = \sin(x)$, we have computed the exact solution for the new parameter $\mu = 0.5$ by solving a $1D$ linear equation system. Thus, the high dimensional problem is reduced to a one dimensional one.

2.2.3.2. The Greedy Algorithm

The snapshots defining the reduced space X_N are determined by the (offline-)selection of parameter samples $S_N := \{\mu^i : i = 1, \dots, N\}$ and those samples are often computed by maximising a computable error estimate $\Delta_N(\mu)$ w.r.t. the parameter μ.

This can be done e.g. by nonlinear optimisation or a greedy method w.r.t. a finite so called *training set* $\mathcal{D}^{\text{train}} \subset \mathcal{D}$. For the training set, we define $\mathsf{M}(\mathcal{D}^{\text{train}}) := \{u(\mu) \in \mathsf{X}^{\mathcal{N}} : \mu \in \mathcal{D}^{\text{train}}\}$.

The basic greedy procedure is presented in Algorithm 2.2.1. In the i-th greedy step we consider the current RB approximation space X_{i-1} and identify the error maximising element

$u^i = \arg\sup_{u\in\mathsf{M}(\mathcal{D}^{\text{train}})} \inf_{w\in\mathsf{X}_{i-1}} \|u-w\|_{\mathsf{X}}$. The RB basis is extended by u^i. The RB approximation space is finally given as $\mathsf{X}_N := \operatorname{span}(\Xi_{\mathsf{X}}^N)$. Since the RB solution u^i is given by the (Petrov-) Galerkin projection of $u \in \mathsf{M}(\mathcal{D}^{\text{train}})$ onto X_i, we compute the current RB approximation u_i. Since

$$\|u^{\mathcal{N}} - u_N\|_{\mathsf{X}} \leq C \inf_{w\in\mathsf{X}_N} \|u^{\mathcal{N}} - w\|_{\mathsf{X}}$$

we replace the computation of the infimum by the error calculation $\|u^{\mathcal{N}} - u_i\|_{\mathsf{X}}$ for $u^{\mathcal{N}} \in \mathsf{M}(\mathcal{D}^{\text{train}})$.

To evaluate the supremum, we have to compute all solutions $u^{\mathcal{N}} \in \mathsf{M}(\mathcal{D}^{\text{train}})$ for the given training set $\mathcal{D}^{\text{train}}$. Those computations are computationally expensive and the procedure is time-consuming for large training sets. The computational effort can be avoided if an a-posteriori

Algorithm 2.2.1 Greedy Algorithm

Input: $\mathcal{D}^{\text{train}} \subset \mathcal{D}$ training set, tol > 0 tolerance, $\Xi_{\mathsf{X}}^0 = \emptyset$, i $= 1$.
Output: Ξ_{X}^N RB basis.

1: **while** $\sup_{u\in\mathsf{M}(\mathcal{D}^{\text{train}})} \inf_{w\in\mathsf{X}_{i-1}} \|u-w\|_{\mathsf{X}} >$ tol **do**
2: $\quad u^i := \arg\sup_{u\in\mathsf{M}(\mathcal{D}^{\text{train}})} \inf_{w\in\mathsf{X}_{i-1}} \|u-w\|_{\mathsf{X}}$.
3: $\quad \Xi_{\mathsf{X}}^i = \Xi_{\mathsf{X}}^{i-1} \cup \{u^i\}$.
4: $\quad \mathsf{X}_i := \operatorname{span}\{\Xi_{\mathsf{X}}^i\}$, $i \leftarrow i+1$.
5: **end while**
6: N = i.

error estimator $\Delta_N(\mu)$ is available that approximates the error $\|u^{\mathcal{N}} - u_N\|$. We refer to [Haa14, Rem. 2.39] for different possible error indicator that can be used in the greedy procedure.

The standard error estimator for an inf-sup stable problem is given in the next Proposition.

Proposition 2.2.38. Let $\mathsf{X}_N \subset \mathsf{X}^{\mathcal{N}}$ and $u_N(\mu) \in \mathsf{X}_N$ be the solution of (2.2.26). Defining the *residual* by $r_N(v;\mu) := f(v;\mu) - b(u_N(\mu), v;\mu) = b(u^{\mathcal{N}}(\mu) - u_N(\mu), v;\mu)$ for all $v \in \mathsf{Y}^{\mathcal{N}}$, we obtain the following error estimate

$$\|u^{\mathcal{N}}(\mu) - u_N(\mu)\|_{\mathsf{X}} \leq \frac{1}{\beta^{\mathcal{N}}} \|r_N(\cdot;\mu)\|_{(\mathsf{Y}^{\mathcal{N}})'} =: \Delta_N(\mu), \tag{2.2.33}$$

where $\beta^{\mathcal{N}}$ denotes the inf-sup constant given in Equation (2.2.28) of $b(\cdot,\cdot;\mu)$ on $\mathsf{X}^{\mathcal{N}} \times \mathsf{Y}^{\mathcal{N}}$.

Proof. It immediately follows from (2.2.28) and the definition of the dual norm that

$$\beta^{\mathcal{N}} \|u^{\mathcal{N}}(\mu) - u_N(\mu)\|_{\mathsf{X}} \leq \sup_{v\in\mathsf{Y}^{\mathcal{N}}} \frac{b(u^{\mathcal{N}}(\mu) - u_N(\mu), v;\mu)}{\|v\|_{\mathsf{Y}}} = \|r_N(\cdot;\mu)\|_{(\mathsf{Y}^{\mathcal{N}})'}.$$

□

The error estimator $\Delta_N(\mu)$ given in Proposition 2.2.38 is offline-online efficient since one can derive an affine decomposition w.r.t. the parameter as in (2.2.30) also for the residual $\|r_N(\cdot;\mu)\|_{(\mathsf{Y}^{\mathcal{N}})'}$, cf. Section 2.3.2.

Remark 2.2.39. The error estimator for the space-time reduced basis method requires the (discrete) inf-sup constant. Its computation can be realised by using the successive constraint method (SCM), which was introduced by D. Huyn, G. Rozza, S. Sen and A. Patera, [HRSP07], as an offline-online efficient method for estimating the otherwise computationally costly inf-sup or – for coercive problems – the coercivity constant. The main idea is to precompute $\beta(\mu_i)$ for μ_i in a sample set $\{\mu_1, \dots, \mu_K\} = \mathcal{D}^{\mathrm{SCM}} \subset \mathcal{D}$ and to use these values to construct a lower bound $\beta_{\mathrm{LB}}(\mu)$ for the inf-sup constant β for any other parameter. This is realised by a greedy approach offline and solving a linear optimisation problem online.

The idea behind the greedy approach is to approximate the set $\mathsf{M}(\mathcal{D}) = \{u^{\mathcal{N}}(\mu) \in \mathsf{X}^{\mathcal{N}} : \mu \in \mathcal{D}\} \subset \mathsf{X}^{\mathcal{N}}$ up to a given target tolerance by the RB approximation space X_N with $\dim(\mathsf{X}_N) = N$ and $N \ll \mathcal{N}$. Optimal subspaces $\mathsf{X}_N \subset \mathsf{X}^{\mathcal{N}}$ for the approximation can be identified considering the Kolmogorov N-width

$$d_N(\mathsf{M})_{\mathsf{X}} = \inf_{\substack{\mathsf{X}_N \subset \mathsf{X}^{\mathcal{N}} \\ \dim \mathsf{X}_N = N}} \sup_{u \in \mathsf{M}(\mathcal{D})} \inf_{w \in \mathsf{X}_N} \|u - w\|_{\mathsf{X}}. \tag{2.2.34}$$

Convergence results for the greedy procedure are achieved by applying the convergence rates of the Kolmogorov N-width, cf. [DPW14, BCD$^+$11].

If the error $\|u^{\mathcal{N}} - u_N\|_{\mathsf{X}}$ is replaced by an error estimator, we have to ensure that the greedy procedure succeeds in finding (close to) optimal subspaces X_N.

Proposition 2.2.40. [DPW14, Def. 1.1, Rem. 1.2] Let $\mu \in \mathcal{D}$ and let $\Delta_N(\mu)$ be a tight surrogate in the sense that there exist constants $C, c > 0$ such that

$$c\ \Delta_N(\mu) \leq \|u^{\mathcal{N}}(\mu) - u_N(\mu)\|_{\mathsf{X}} \leq C\Delta_N(\mu). \tag{2.2.35}$$

Then,

$$\max \inf_{w \in \mathsf{X}_N} \|u^{\mathcal{N}} - w\|_{\mathsf{X}} \leq \kappa \|u^{\mathcal{N}}(\mu) - u_N(\mu)\|_{\mathsf{X}},$$

with $\kappa := \inf\{\frac{C}{c} : C, c \text{ satisfy } (2.2.35)\ \forall \mu \in \mathcal{D}, \forall N \in \mathbb{N}\}$.

The approximation space X_N improves if the chosen error estimate $\Delta_N(\mu)$ has an associated κ close to one.

2.3. Reduced Basis Simulation for Petrov-Galerkin Problems

The space-time RBM is based on the space-time variational formulation introduced in Equation (2.2.8). This is in contrast to time-stepping RB methods for parabolic PDEs presented in Appendix B. We introduce three different possibilities to compute the RB solution within the space-time RBM:

First, we explore the minimal residual approach used in [Yan14]. This was presented in detail in [MPR02] for non-coercive problems, including a theoretical analysis of the associated inf-sup constant.

The second possibility is the explicit construction of an inf-sup stable reduced test space. This is a direct application of the theory developed for the Navier-Stokes problem, by G. Rozza and K. Veroy, [RV07], that was adapted by A.-L. Gerner and K. Veroy in [GV12] to general saddle point problems.

As a third possibility, the associated normal equation is to be solved.

The following proposition shows that all three approaches are equivalent on finite dimensional approximation spaces and lead to the same RB solution.

Proposition 2.3.1. Let $\mathsf{X}^{\mathcal{N}} := \operatorname{span}\{x_1, \dots, x_{\mathcal{N}}\}$ and $\mathsf{Y}^{\mathcal{N}} := \operatorname{span}\{y_1, \dots, y_{\mathcal{N}}\}$ be the high dimensional discretised system and let $\operatorname{span}\{u^1, \dots, u^N\} =: \mathsf{X}_N \subset \mathsf{X}^{\mathcal{N}}$ be a reduced basis space. For a given parameter μ, let $\mathsf{Y}_N(\mu) \subset \mathsf{Y}^{\mathcal{N}}$ be a stable RB test space, i.e. let the inf-sup stability be guaranteed for X_N and Y_N. Define

$$\mathbf{Y}_{k,j} = (y_k, y_j)_{\mathsf{Y}}, \quad \mathbf{B}_{j,i} = b(u^i, y_j), \quad \mathbf{f}_k = f(y_k) \qquad \forall\, i = 1, \dots, N, \quad k, j = 1, \dots, \mathcal{N}.$$

Then for $u_N = \sum_{i=1}^{N} \alpha_i u^i \in \mathsf{X}_N$ the following are equivalent:

(1) $b(u_N, v; \mu) = f(v; \mu) \quad \forall\, v \in \mathsf{Y}^N(\mu)$,

(2) $u_N = \arg\inf_{u \in \mathsf{X}_N} \|b(u, \cdot; \mu) - f(\cdot; \mu)\|_{(\mathsf{Y}^{\mathcal{N}})'}$,

(3) for $\mathbf{u} = (\alpha_i)_{i=1,\dots,N}$: $\quad \mathbf{B}^T \mathbf{Y}^{-1} \mathbf{B} \mathbf{u} = \mathbf{B}^T \mathbf{Y}^{-1} \mathbf{f}$.

Proof. This is an adaptation of [And12, Prop. 4.2.5]. We replace the high dimensional trial space by the RB trial space. □

Throughout this section, we use the notations introduced in Section 2.2.3.1.

Note that the RB simulation is not only part of the online phase of the RBM but can already be used offline in the parameter sampling procedure 2.2.3.2. We detail its appearance in a greedy procedure in Algorithm 2.3.1 using the function

$$\texttt{RB-SIM} : \mathcal{D}^{\text{train}} \times \mathbb{N} \to \mathsf{X}^{\mathcal{N}}, \quad (\mu, N) \mapsto u_N(\mu) \tag{2.3.1}$$

and defining $\Delta_N^{\texttt{RB-SIM}}(\mu) = \frac{1}{\beta_{\text{LB}}} \| b(\texttt{RB-SIM}(\mu, N), \cdot; \mu) - f(\cdot; \mu) \|_{(\mathsf{Y}^{\mathcal{N}})'}$.

Algorithm 2.3.1 Greedy Algorithm

Input: $\mathcal{D}^{\text{train}} \subset \mathcal{D}$ training set, tol > 0 RB target tolerance.
Output: Ξ_{X}^N RB basis.

1: Choose $\mu \in \mathcal{D}^{\text{train}}$, compute $u^1 := u^{\mathcal{N}}(\mu)$, $\Xi_{\mathsf{X}}^1 = \{u^1\}$, $\ell = 1$.
2: **while** $\max_{\mu \in \mathcal{D}^{\text{train}}}(\Delta_\ell^{\texttt{RB-SIM}}(\mu)) >$ tol **do**
3: $\quad \mu = \arg\max_{\mu \in \mathcal{D}^{\text{train}}}(\Delta_\ell^{\texttt{RB-SIM}}(\mu))$.
4: $\quad$ Compute $u^{\ell+1} := u^{\mathcal{N}}(\mu)$, $\Xi_{\mathsf{X}}^{\ell+1} := \Xi_{\mathsf{X}}^{\ell} \cup \{u^{\ell+1}\}$, $\ell \leftarrow \ell + 1$.
5: **end while**
6: $N := \ell$.

2.3.1. Offline-Online Decomposition in the Space-Time RBM

We recall Example 2.2.34 where we defined for $w \in \mathbb{X}$, $v = (z, h) \in \mathbb{Y}$ and $\mu \in \mathcal{D}$

$$\begin{aligned} b(w, v; \mu) &:= \int_I \langle \dot{w}(t), z(t) \rangle_{V' \times V} \, dt + \int_I a(w(t), z(t); \mu) dt + (w(0), h)_H, \\ f(v; \mu) &:= \int_I \langle g(t; \mu), z(t) \rangle_{V' \times V} \, dt + (u_0, h)_H. \end{aligned}$$

Here and in the following we assume the affine decomposition w.r.t. the parameter given by

$$\begin{aligned} b(w, v; \mu) &= \int_I \langle \dot{w}(t), z(t) \rangle_{V' \times V} \, dt + \sum_{q=1}^{Q_a} \int_I \theta_q^a(\mu) a_q(w(t), z(t)) dt + (w(0), h)_H \\ &= \sum_{q=1}^{Q_b} \theta_q^b(\mu) b_q(w, v) \end{aligned}$$

with $Q_b := Q_a + 2$, $b_1(w, v) := \int_I \langle \dot{w}(t), z(t) \rangle_{V' \times V} \, dt$, $b_{Q_b}(w, v) := (w(0), h)_H$ and $\theta_1^b(\mu) = \theta_{Q_b}^b(\mu) = 1$ as well as $b_i(w, v) := \int_I a_{i-1}(w, z) dt$, $\theta_i^b \equiv \theta_{i-1}^a$ for $i = 2, \ldots, Q_a + 1$. For the

right-hand side we assume a decomposition of the right-hand side g and receive

$$\begin{aligned} f(v;\mu) &= \int_I \left\langle \sum_{q=1}^{Q_g} \theta_q^g(\mu) g_q(t), z(t) \right\rangle_{V' \times V} dt + (u_0, h)_H \\ &= \sum_{q=1}^{Q_f} \theta_q^f(\mu) f_q(v) \end{aligned}$$

for $\theta_q^f \equiv \theta_q^g$, $f_q(v) := \int_I \langle g_q(t), z(t) \rangle_{V' \times V} \, dt$ for $q = 1, \dots, Q_f$ and $f_{Q_f}(v) := (u_0, h)_H$, $\theta_{Q_f}^f(\mu) := 1$, $Q_f := Q_g + 1$.

2.3.2. Minimal Residual Projection

By Proposition 2.3.1 the reduced solution u_N in (2.2.26) is equivalently given by Proposition 2.3.1 (2). The dual norm of the residual $r(v; u, \mu) = b(u, v; \mu) - f(v; \mu)$, $v \in \mathsf{Y}^{\mathcal{N}}$, $u \in \mathsf{X}^{\mathcal{N}}$ is computable using its Riesz representant $\hat{r}(u; \mu) \in \mathsf{Y}^{\mathcal{N}}$ given by $(\hat{r}(u; \mu), v)_{\mathsf{Y}} = r(v; u, \mu)$ for all $v \in \mathsf{Y}^{\mathcal{N}}$. The RB solution is given by

$$u_N = \arg \inf_{u \in \mathsf{X}_N} \|r(\cdot; u, \mu)\|_{(\mathsf{Y}^{\mathcal{N}})'} = \arg \inf_{u \in \mathsf{X}_N} \sup_{v \in \mathsf{Y}^{\mathcal{N}}} \frac{r(v; u, \mu)}{\|v\|_{\mathsf{Y}}} = \arg \inf_{u \in \mathsf{X}_N} \|\hat{r}(u; \mu)\|_{\mathsf{Y}}.$$

By definition, $\|\hat{r}(u; \mu)\|_{\mathsf{Y}}$ is offline-online decomposable in the following way. Let $\mathsf{X}^{\mathcal{N}} := \operatorname{span}\{ x_i : i = 1, \dots, \mathcal{N} \}$, $\mathsf{Y}^{\mathcal{N}} := \operatorname{span}\{ y_i : i = 1, \dots, \mathcal{N} \}$ and $\mathsf{X}_N := \operatorname{span}\{u^1, \dots, u^N\}$. Every basis function $u^i \in \mathsf{X}_N$ is an element of $\mathsf{X}^{\mathcal{N}}$, thus $u^i = \sum_{j=1}^{\mathcal{N}} \alpha_j^i x_j$. Let $\boldsymbol{\alpha}^i := (\alpha_j^i)_{j=1,\dots,N}$ be the associated coefficient vector. Offline, we precompute the matrix/vector representations

$$\begin{aligned} &[\mathbf{BB}_{q,q'}^N]_{i,j} = (\boldsymbol{\alpha}_i)^T (\mathbf{B}_q)^T \mathbf{Y}^{-1} \mathbf{B}_{q'} \boldsymbol{\alpha}_j, \quad (\mathbf{BF}_{q,\ell}^N)_i = (\boldsymbol{\alpha}^i)^T (\mathbf{B}_q)^T \mathbf{Y}^{-1} \mathbf{f}_\ell = (\mathbf{FB}_{\ell,q}^N)^T \\ &\text{and} \quad \mathbf{FF}_{\ell,\ell'}^N = (\mathbf{f}_\ell)^T \mathbf{Y}^{-1} \mathbf{f}_{\ell'}. \end{aligned}$$

for all $q, q' = 1, \dots, Q_b$, $i, j = 1, \dots, N$ and $\ell, \ell' = 1, \dots, Q_f$ where

$$\begin{aligned} [\mathbf{Y}]_{i,j} = (y_i, y_j)_{\mathsf{Y}}, \quad [\mathbf{B}_q]_{j,i} = b_q(x_i, y_j), \text{ and } (\mathbf{f}_\ell)_i = f_\ell(y_i) \\ \forall q = 1, \dots, Q_b, \quad \ell = 1, \dots, Q_f, \quad i, j = 1, \dots, \mathcal{N}. \end{aligned}$$

Let $u_N(\mu) = \sum_{i=1}^N \alpha_i(\mu) u^i \in \mathsf{X}_N$ and define the coefficient vector by $\mathbf{u} := (\alpha_i(\mu))_{i=1,\dots,N}$. Since $\|\hat{r}(u;\mu)\|_{\mathsf{Y}}^2 = (\hat{r}(u;\mu), \hat{r}(u;\mu))_{\mathsf{Y}} = \langle r(\cdot, u;\mu), \hat{r}(u;\mu)\rangle_{\mathsf{Y}'\times\mathsf{Y}}$ we compute the residual by

$$\begin{aligned}(\mathbf{Y}^{-1}(\mathbf{Bu}-\mathbf{f}))^T\mathbf{Y}(\mathbf{Y}^{-1}(\mathbf{Bu}-\mathbf{f})) = &\sum_{q,q'=1}^{Q_b} \theta_q^b(\mu)\theta_{q'}^b(\mu)\mathbf{u}^T\mathbf{B}\mathbf{B}_{q,q'}\mathbf{u} \\ &- \sum_{q=1}^{Q_b}\sum_{\ell=1}^{Q_f}\theta_q^b(\mu)\theta_\ell^f(\mu)(\mathbf{u}^T\mathbf{B}\mathbf{F}_{q,\ell} + \mathbf{F}\mathbf{B}_{\ell,q}\mathbf{u}) \\ &+ \sum_{\ell,\ell'=1}^{Q_f}\theta_\ell^f(\mu)\theta_{\ell'}^f(\mu)\mathbf{F}\mathbf{F}_{\ell,\ell'}.\end{aligned}$$

The online computations are independent of the high dimension $\mathcal{N}$, i.e. of complexity $\mathcal{O}(N^2 Q_b)$. In Equation (2.3.1) the unconstrained nonlinear optimisation problem $\texttt{RB-SIM}(\mu,\ell) = \texttt{MinRes}(\mu,\ell) = \arg\min_{u\in\mathsf{X}_\ell}\|\hat{r}(u;\mu)\|_{\mathsf{Y}}$ is to be solved by e.g. a line-search algorithm. The value of the residual is given immediately and can be used for the error estimator.

The inf-sup stability of the RB system is deduced from the inf-sup stability of the discrete high-dimensional system. For given parameter $\mu \in \mathcal{D}$,

$$\beta_N(\mu) = \inf_{u\in\mathsf{X}_N}\sup_{y\in\mathsf{Y}^{\mathcal{N}}}\frac{b(u,v;\mu)}{\|u\|_{\mathsf{X}}\|v\|_{\mathsf{Y}}} \geq \inf_{u\in\mathsf{X}^{\mathcal{N}}}\sup_{y\in\mathsf{Y}^{\mathcal{N}}}\frac{b(u,v;\mu)}{\|u\|_{\mathsf{X}}\|v\|_{\mathsf{Y}}} \geq \beta^{\mathcal{N}}(\mu) > 0.$$

2.3.3. Petrov-Galerkin Projection

The construction of a stabilising reduced test space is necessary for the Petrov-Galerkin projection. One strategy is to enrich the test space within the greedy procedure by the supremizers of the current trial space. This method has been applied to general saddle point problems in [GV12] and to transport-dominated problems in [DPW14]. The later presented a double-greedy scheme to enrich the test space in a greedy procedure until a required stabilisation is reached. We present the construction for a parameter dependent test space.

For every parameter μ, the RB test space Y_N has to guarantee inf-sup stability of the RB system, i.e.,

$$\inf_{u\in\mathsf{X}_N}\sup_{v\in\mathsf{Y}_N}\frac{b(u,v;\mu)}{\|u\|_{\mathsf{X}}\|v\|_{\mathsf{Y}}} > 0.$$

We follow the notations of the previous Section 2.3.2 and define for $u^i \in \mathsf{X}^{\mathcal{N}}$ and a given parameter $\mu \in \mathcal{D}$ (in particular we do not focus on the associated parameter μ_i!) the supremizing operator by $S_{u^i}(\mu) = \arg\sup_{v\in\mathsf{Y}^{\mathcal{N}}}\frac{b(u^i,v;\mu)}{\|v\|_{\mathsf{Y}}}$. As the supremizer is the Riesz representant given by

$(S_{u^i}(\mu), v)_{\mathsf{Y}} = b(u, v; \mu)$ for all $v \in \mathsf{Y}^{\mathcal{N}}$, the associated coefficient vector in $\mathsf{Y}^{\mathcal{N}}$ is given by

$$\mathbf{S}_{u^i}(\mu) := \mathbf{Y}^{-1} \sum_{q=1}^{Q_b} \theta_q^b(\mu) \mathbf{B}_q \boldsymbol{\alpha}_i = \sum_{q=1}^{Q_b} \theta_q^b(\mu) \mathbf{Y}^{-1} \mathbf{B}_q \boldsymbol{\alpha}_i =: \sum_{q=1}^{Q_b} \theta_q^b(\mu) \mathbf{S}_q^i.$$

For all RB basis elements u^i, $i = 1, \dots, N$, we compute the supremizing components S_q^i for $q = 1, \dots, Q_b$ given by $(S_q^i, v)_{\mathsf{Y}} = b_q(u, v)\ \forall\, v \in \mathsf{Y}^{\mathcal{N}}$. The (offline) test space is given by $\mathsf{Y}_{\text{offline}}^{NQ_b} = \operatorname{span}\{S_1^1, \dots, S_{Q_b}^1, \dots, S_1^N, \dots, S_{Q_b}^N\}$.

Online, for a new parameter $\mu \in \mathcal{D}$ an associated optimal test space $\mathsf{Y}_N(\mu) = \{S^1(\mu), \dots, S^N(\mu)\}$ is considered, where $S^i(\mu) := \sum_{q=1}^{Q_b} \theta_q^b(\mu) S_q^i$. The RB-problem formulation is as introduced in (2.2.26) given $\mathsf{Y}_N(\mu)$ as test space. For every parameter $\mu \in \mathcal{D}$ inf-sup stability is guaranteed by the test space construction:

$$\begin{aligned}
\beta^{\mathcal{N}}(\mu) &= \inf_{u \in \mathsf{X}^{\mathcal{N}}} \sup_{v \in \mathsf{Y}^{\mathcal{N}}} \frac{b(u, v; \mu)}{\|u\|_{\mathsf{X}} \|v\|_{\mathsf{Y}}} \leq \inf_{u \in \mathsf{X}_N} \sup_{v \in \mathsf{Y}^{\mathcal{N}}} \frac{b(u, v; \mu)}{\|u\|_{\mathsf{X}} \|v\|_{\mathsf{Y}}} \\
&= \sup_{v \in \mathsf{Y}^{\mathcal{N}}} \frac{\sum_{i=1}^N \alpha_i(\mu) b(u^i, v; \mu)}{\|\sum_{i=1}^N \alpha_i(\mu) u^i\|_{\mathsf{X}} \|v\|_{\mathsf{Y}}} = \frac{\sum_{i=1}^N \alpha_i(\mu) b(u^i, S_{u^i}(\mu); \mu)}{\|\sum_{i=1}^N \alpha_i(\mu) u^i\|_{\mathsf{X}} \|S_{u^i}(\mu)\|_{\mathsf{Y}}} \\
&\leq \sup_{v \in \mathsf{Y}_N(\mu)} \frac{b(\sum_{i=1}^N \alpha_i(\mu) u^i, v; \mu)}{\|\sum_{i=1}^N \alpha_i(\mu) u^i\|_{\mathsf{X}} \|v\|_{\mathsf{Y}}}.
\end{aligned}$$

The greedy algorithm 2.3.1 is extended by the test space construction, cf. line 7 in Algorithm 2.3.2.

Algorithm 2.3.2 Greedy Algorithm with Test Space Construction

Input: $\mathcal{D}^{\text{train}} \subset \mathcal{D}$ training set, $\text{tol} > 0$ RB target tolerance.
Output: Ξ_{X}^N RB ansatz space basis, $\Xi_{\mathsf{Y}}^{NQ_b}$ RB test space basis.

1: Choose $\mu \in \mathcal{D}^{\text{train}}$, compute $u^1 := u^{\mathcal{N}}(\mu)$, $\Xi_X^1 = \{u^1\}$, $\ell = 1$.
2: Compute $S_1^1, \dots, S_{Q_b}^1$, $\Xi_Y^{Q_b} = \{S_1^1, \dots, S_{Q_b}^1\}$
3: **while** $\max_{\mu \in \mathcal{D}^{\text{train}}} (\Delta_\ell^{\texttt{RB-SIM}}(\mu)) > \text{tol}$ **do**
4: $\quad \mu' = \arg\max_{\mu \in \mathcal{D}^{\text{train}}} (\Delta_\ell^{\texttt{RB-SIM}}(\mu))$
5: $\quad$ Compute $u^{\ell+1} := u^{\mathcal{N}}(\mu')$, $\Xi_X^{\ell+1} := \Xi_X^\ell \cup \{u^{\ell+1}\}$, $\ell \leftarrow \ell + 1$
6: $\quad$ Compute $S_1^\ell, \dots, S_{Q_b}^\ell$, $\Xi_Y^{\ell Q_b} \leftarrow \Xi_Y^{(\ell-1)Q_b} \cup \{S_1^\ell, \dots, S_{Q_b}^\ell\}$
7: **end while**
8: $N := \ell$.

The RB simulation is given by $\texttt{RB-SIM}(\mu, \ell) := \texttt{PGP}(\mu, \ell) = \sum_{i=1}^N (\mathbf{B}_N(\mu)^{-1} \mathbf{f}_N(\mu))_i u^i$. Offline, we additionally have to precompute $b_q(u^i, S_j^{q'})$ for all $q, q' = 1, \dots, Q_b$ and $i, j = 1, \dots, N$ to achieve the desired independence of $\mathcal{N}$. The computational effort is given by $\mathcal{O}(Q_b^2 N^2)$, independent of $\mathcal{N}$, and Q_b-times the one of a standard Galerkin projection where $\mathsf{X}_N = \mathsf{Y}_N$.

Remark 2.3.2. There are several simplifications and strategies that avoid the resulting μ dependence of the test space, as e.g. in [RV07, GV12]. For these strategies the inf-sup stability for the RB spaces may not be provable any more but was shown in numerical experiments only. A suboptimal test space in the sense that inf-sup stability is not theoretically guaranteed any more results in a non-bijective operator $\mathbf{B}_N$ and may cause instabilities in the greedy procedure.

As already observed in [RV07, Sec. 7], large RB space dimensions N lead to algebraic instabilities concerning the condition of B_N. The algebraic stability can be improved by not taking the snapshots directly as basis functions but orthogonalising first. Orthogonalisation applied to the trail space is straight forward, using the well-known Gram-Schmidt algorithm. Since the test space is parameter dependent, an orthogonalisation of the test space can only be performed online. We refer to [RV07, Sec. 7] for details.

2.3.4. Normal Equation Projection

The normal equation stated in Proposition 2.3.1 (3) can also be used for the RB simulation. This possibility was mentioned in [DPW14] but not used, because of their usage of a μ-dependent test space norm and the therewith associated computational effort. The normal equation does not need an explicit construction of a reduced test space neither does it require a numerical optimisation procedure. In comparison to the minimal residual approach, we compute only two of the already presented operators in Section 2.3.2 and solve the problem

$$\text{find } u_N = \sum_{i=1}^{N} (\mathbf{u})_i u^i \in \mathsf{X}_N : \sum_{q,q'=1}^{Q_b} \theta_q^b(\mu)\theta_{q'}^b(\mu)\mathbf{B}\mathbf{B}_{q,q'}\mathbf{u} = \sum_{q=1}^{Q_b}\sum_{\ell=1}^{Q_f} \theta_q^b(\mu)\theta_\ell^f(\mu)\mathbf{B}\mathbf{F}_{q,\ell}.$$

Online, the computational effort $\mathcal{O}(Q_b^2 N^2)$ is independent of the high dimension $\mathcal{N}$. The inf-sup stability of the RB system is again implied by the inf-sup stability of the truth discretisation,

$$\beta_N(\mu) = \inf_{u\in\mathsf{X}_N}\sup_{v\in\mathsf{Y}^\mathcal{N}} \frac{b(u,v;\mu)}{\|u\|_\mathsf{X}\|v\|_\mathsf{Y}} \geq \inf_{u\in\mathsf{X}^\mathcal{N}}\sup_{v\in\mathsf{Y}^\mathcal{N}} \frac{b(u,v;\mu)}{\|u\|_\mathsf{X}\|v\|_\mathsf{Y}} = \beta^\mathcal{N}(\mu) > 0.$$

The RB simulation is given by $\texttt{RB-SiM}(\mu,\ell) = \texttt{GNE}(\mu,\ell) = \sum_{i=1}^{N} (\mathbf{B}\mathbf{B}^{-1}(\mu)\mathbf{B}\mathbf{F}(\mu))_i u^i$.

In the numerical experiments throughout this thesis, we use the normal equation approach to compute the RB solution. The approach does neither require an additional optimisation algorithm nor an additional storage of an RB test space.

```
model = test_model;
model_data = gen_model_data(model);
detailed_data = gen_detailed_data(model,model_data);
reduced_data = gen_reduced_data(model,detailed_data);
rb_sim_data = rb_simulation(model,reduced_data);
uN = rb_reconstruction(model,reduced_data);
```

Figure 2.4.1.: RBMatlab command chain for the reduced basis approximation.

```
function detailed_data = gen_detailed_data(model,model_data)
% Reduced basis generation
% Input: model and mesh specifications
% Output: Reduced basis
detailed_data.RB{1} = init_data(model,model_data);
detailed_data = rb_extension(model,detailed_data);
```

Figure 2.4.2.: Reduced basis generation.

2.4. Implementation

2.4.1. Software

The implementation is realised in MATLAB [MAT14]. All numerical experiments are embedded in the working environment of `RBMatlab` [RBM13], a software project of the University of Stuttgart and the University of Münster, Germany, that provides a MATLAB library for the reduced basis method. We present the standard command chain in Figure 2.4.1. The partial differential equation with all its properties is specified in `test_model`. In particular, the mesh size and parameter domain specifications are given. All model specifications are stored in the MATLAB structure array `model`. The second step is the mesh generation and storage in `model_data`. The reduced basis offline phase is performed by calling `gen_detailed_data`, cf. Figure 2.4.2. First, an initial basis element is computed. Second, the actual reduced basis generation is performed in `rb_extension`. In the greedy procedure the functions `gen_reduced_data` and `rb_simulation` are called. The reduced basis is stored in `detailed_data.RB`. The actual precomputations are then realised by calling `gen_reduced_data`. The structure `reduced_data` only contains $\mathcal{N}$-independent quantities that are used in the reduced basis simulation `rb_simulation` where the actual online computations are performed. The coefficient vector of the reduced basis solution and the (estimated) error of the approximation are stored in `rb_sim_data`, cf. Figure 2.4.3.

As the space-time framework is not available in the toolbox, just a few original functions of `RBMatlab` remain in the program. However, the implementation follows its original structure.

```
rb_sim_data =
  uN: 'reduced basis coefficients'
  Delta: 'estimated reduced basis approximation error'
```

Figure 2.4.3.: MATLAB structure array `rb_sim_data`.

For the later use in the context of low rank tensor methods, the underlying finite element solver needs to return the discrete model in a very specific form and has been implemented in addition. For the same reason, the implementation provides an interface with the Hierarchical Tucker Toolbox [Hie13]. Both, the FEM implementation as well as the interface resulted in connection with F. Grimmers Master thesis [Gri15]. The H-Tucker toolbox was developed by Ch. Tobler and D. Kressner in 2012, cf. the manual [KT12].

2.4.2. Numerical Realisation of the space-time RBM

For discrete ansatz and test spaces $\mathbb{X}^{\mathcal{N}} = E^{\mathcal{K}+1} \otimes V^{\mathcal{J}}$ and $\mathbb{Y}^{\mathcal{N}} = F^{\mathcal{K}} \otimes V^{\mathcal{J}} \times V^{\mathcal{J}}$, cf. Equation (2.2.15), the problem formulation is given by

$$\text{For } \mu \in \mathcal{D} \subset \mathbb{R}^d \text{ find } u \in \mathbb{X}^{\mathcal{N}} : \; b(u, v; \mu) = f(v; \mu) \quad \forall\, v \in \mathbb{Y}^{\mathcal{N}}.$$

The bilinear form $b(\cdot, \cdot; \cdot)$ as well as the linear functional $f(\cdot; \cdot)$ are defined in Example 2.2.34.

The matrix representation of the left-hand side is given by

$$\mathbf{B}(\mu) = \begin{pmatrix} \mathbf{C}_{\text{time}} \otimes \mathbf{M}_{\text{space}} + \mathbf{M}_{\text{time}} \otimes \mathbf{A}_{\text{space}}(\mu) \\ (1, 0, \dots, 0) \otimes \mathbf{M}_{\text{space}} \end{pmatrix}.$$

For $i = 0, \dots, \mathcal{K}$ and $j = 1, \dots, \mathcal{K}$ the temporal matrices have the components

$$[\mathbf{C}_{\text{time}}]_{ji} = \int_I \dot{\sigma}_i(t)\tau_j(t)dt, \quad [\mathbf{M}_{\text{time}}]_{ji} = \int_I \sigma_i(t)\tau_j(t)dt \text{ and } (1, 0, \dots, 0)^T \in \mathbb{R}^{\mathcal{K}+1}.$$

For $i, j = 1, \dots, \mathcal{J}$ the spatial matrices are given by

$$[\mathbf{M}_{\text{space}}]_{ji} = \int_\Omega \phi_i(x)\phi_j(x)dx, \quad [\mathbf{A}_{\text{space}}]_{ji} = \int_\Omega a(\phi_i(x), \phi_j(x))dx.$$

The right-hand side is given by

$$\mathbf{f}(\mu) = \begin{pmatrix} \text{vec}(\mathbf{g}_1(\mu)) \\ \mathbf{M}_{\text{space}}\mathbf{u}_0 \end{pmatrix}$$

for $\mathbf{g}_1(\mu) = (\int_I \langle g(t; \mu), \tau^\ell(t)\phi_j \rangle_{V' \times V} dt)_{j=1,\dots,\mathcal{J}, \ell=1,\dots,\mathcal{K}}$ and a coefficient vector $\mathbf{u}_0$ corresponding to a chosen expansion of the initial value in $H^{\mathcal{L}}$, $\mathcal{L} \in \mathbb{N}$. For $\mathbf{I}_{\text{time}} = [(\tau^i, \tau^j)_{L_2(I)}]_{i,j=1,\dots,\mathcal{K}}$ the

discrete operator of the test space norm

$$\|v\|_{\mathbb{Y}}^2 = \|v_1\|_{L_2(I,V)}^2 + |v_2|_H^2$$

for every $v = (v_1, v_2)$ in $\mathbb{Y}^{\mathcal{N}}$ is given by

$$\mathbf{Y} = \begin{pmatrix} \mathbf{I}_{\text{time}} \otimes \mathbf{V}_{\text{space}} & 0 \\ 0 & \mathbf{M}_{\text{space}} \end{pmatrix}$$

and the induced operator of the ansatz space norm $\|\cdot\|_{\mathbb{X}}$ defined by

$$\|u\|_{\mathbb{X}}^2 = \|u\|_{L_2(I;V)}^2 + \|\dot{u}\|_{L_2(I;V')}^2 \quad \text{for } u \in \mathbb{X}^{\mathcal{N}}$$

is given by, cf. [UP14, Sec. 3.3],

$$\mathbf{X} = \mathbf{M}_{\text{time}} \otimes \mathbf{V}_{\text{space}} + \mathbf{V}_{\text{time}} \otimes (\mathbf{M}_{\text{space}} \mathbf{V}_{\text{space}}^{-1} \mathbf{M}_{\text{space}})$$

for $\mathbf{V}_{\text{time}} = [(\dot{\sigma}^{k-1}, \dot{\sigma}^{\ell-1})_{L_2(I)}]_{k,\ell=1,\dots,\mathcal{K}+1}$.

The computational effort is reduced by using $(\mathbf{T} \otimes \mathbf{X})\text{vec}(w) = \text{vec}(\mathbf{X}\mathbf{w}\mathbf{T}^T)$ for $\mathbf{T} \in \mathbb{R}^{m\times m}$, $\mathbf{X} \in \mathbb{R}^{n\times n}$ and $\mathbf{w} \in \mathbb{R}^{n\times m}$, $n, m \in \mathbb{N}$. Here, the vec-operator $\text{vec} : \mathbb{R}^{n_1\times\dots\times n_d} \to \mathbb{R}^{n_1\cdots n_d}$ maps a tensor to a (column) vector in *reverse* lexicographical order, cf. Section 4.2.1. The structure of $\mathbf{Y}$ allows for a separated computation of the inverse:

$$\mathbf{Y}^{-1} = \begin{pmatrix} \mathbf{I}_{\text{time}}^{-1} \otimes \mathbf{V}_{\text{space}}^{-1} & 0 \\ 0 & \mathbf{M}_{\text{space}}^{-1} \end{pmatrix}.$$

Remark 2.4.1. Including time as an additional dimension in the variational formulation has advantages regarding the error estimates and the computational complexity of the online phase. However, problems may arise in the numerical verification of the inf-sup stability and the computational feasibility of the offline computations. The actual snapshot computation can be reduced to the same computational effort as the one of a standard time stepping scheme applying a corresponding discretisation, cf. Section 2.2.2.2. For a symmetric bilinear form $a(\cdot,\cdot)$ in the variational formulation the inf-sup stability can be guaranteed for fine time discretisation but a numerical verification is required for non-symmetric $a(\cdot,\cdot)$, again cf. Section 2.2.2.2. Applying a low rank tensor format in the offline phase of the RBM further improves the computational feasibility as the explicit establishment of the full matrices does not become necessary. For details we refer to Chapter 4.

2.5. Numerical Example

This numerical example provides a first insight into the space-time reduced basis method. We consider a parabolic PDE with diffusion, convection and reaction terms.

2.5.1. Diffusion-Convection-Reaction Equation

Let $V := H_0^1(\Omega) \hookrightarrow L_2(\Omega) =: H$. Let $I = (0, 0.3)$ be the time interval. Let $\mathcal{D} := [0.5, 1.5] \times [0, 1] \times [0, 1] \subset \mathbb{R}^3$ be the parameter domain. For fix parameter $\mu = (\alpha, \beta, \gamma)^T \in \mathcal{D}$ find u that solves

$$\begin{aligned} \dot{u} - \nabla \cdot \alpha \nabla u + \beta \nabla u + \gamma u = g \quad & \text{on } \Omega \times I = (0,1) \times (0, 0.3), \\ u(\omega, t) = 0 \quad & \forall\ \omega \in \partial\Omega, t \in (0, 0.3) \\ u(\omega, 0) = u_0(\omega) = \sin(2\pi\omega) \quad & \forall\ \omega \in \Omega. \end{aligned}$$

The right-hand side is computed exactly for $\mu = (1, 0.5, 0.5) \in \mathcal{D}$ for the solution $u(\omega, t) = \sin(2\pi\omega)\cos(4\pi t)$ with

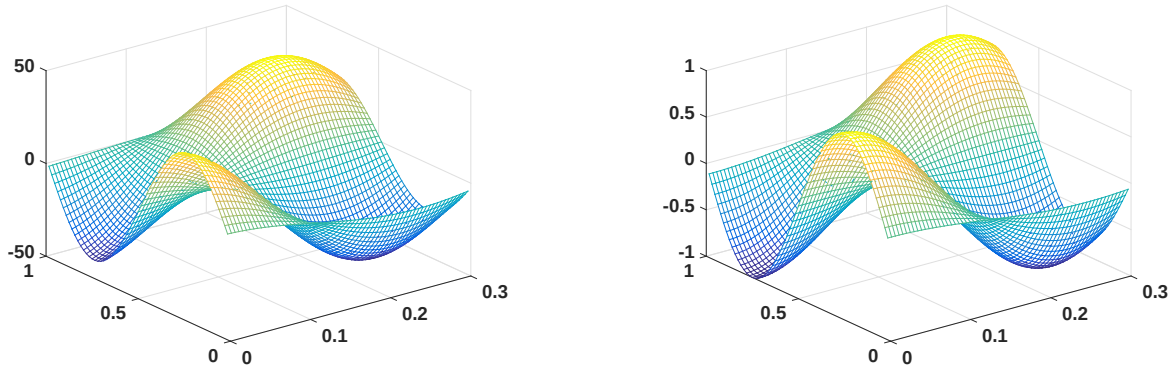
$$g(\omega, t) = \sin(2\pi\omega)\Big((4\pi^2 + 0.5)\cos(4\pi t) - 4\pi \sin(4\pi t)\Big) + \pi \cos(2\pi\omega)\cos(4\pi t),$$

cf. Figure 2.5.1. For $v = (v_1, v_2) \in \mathbb{Y} = L_2(I; H_0^1(\Omega)) \times L_2(\Omega)$ the space-time variational formulation reads find $u \in \mathbb{X} = L_2(I; H_0^1(\Omega)) \cap H^1(I; H^{-1}(\Omega))$ such that

$$\begin{aligned} \int_I \int_\Omega \dot{u}(\omega, t) v_1(\omega, t) d\omega + \int_\Omega \alpha \nabla u(\omega, t) \nabla v_1(\omega, t) + \beta \nabla u(\omega, t) v_1(\omega, t) + \gamma u(\omega, t) v_1(\omega, t) d\omega dt \\ + \int_\Omega u(\omega, 0) v_2(\omega) d\omega = \int_I \int_\Omega g(\omega, t) v_1(\omega, t) d\omega dt + \int_\Omega u_0(\omega) v_2(\omega) d\omega. \end{aligned}$$

2.5.2. Well-posedness

The problem is well-posed in the Hilbert space setting as long as $a(u, v; \mu) := \int_\Omega \alpha \nabla u(\omega, t) \nabla v_1(\omega, t) + \beta \nabla u(\omega, t) v_1(\omega, t) + \gamma u(\omega, t) v_1(\omega, t) d\omega$ satisfies a Gårding inequality and is bounded, cf. Section 2.2.1.3. This is satisfied if $\mathcal{A}(\mu)$ given by $\langle \mathcal{A}(\mu) u, v \rangle_{V' \times V} = a(u, v; \mu)$ is an elliptic operator, [Eva99, Ch. 6 Thm. 2]. Since all coefficients are $L_\infty(\Omega)$ and $\alpha \in [0.5, 1.5]$ is positive, the parabolic PDE is well-posed.



(a) (b)

Figure 2.5.1.: The right-hand side of the PDE is given in (a). The exact solution for $\mu = (1, 0.5, 0.5)$ is given in (b).

2.5.3. Discretisation

We follow Section 2.2.2.2 for the discretisation. We introduce a linear finite element approximation in space and denote the set of basis functions by $\{\phi_1, \dots, \phi_{\mathcal{J}}\}$. For $L_1 \in \mathbb{N}$ the number of intervals is given by 2^{L_1}. The number of basis functions is given by $\mathcal{J} := 2^{L_1} - 1$. The temporal domain contains 2^{L_2} intervals for fixed $L_2 \in \mathbb{N}$ and $\mathcal{K} := 2^{L_2} + 1$ basis functions are used. With this discretisation at hand, the discrete operator $\mathbf{B}$ of the left-hand side is given by

$$\mathbf{B} = \begin{pmatrix} \mathbf{C}_{\text{time}} \otimes \mathbf{M}_{\text{space}} + \alpha \mathbf{M}_{\text{time}} \otimes \mathbf{A}_{\text{space}} + \beta \mathbf{M}_{\text{time}} \otimes \mathbf{B}_{\text{space}} + \gamma \mathbf{M}_{\text{time}} \otimes \mathbf{C}_{\text{space}} \\ (1, 0, \dots, 0) \otimes \mathbf{M}_{\text{space}} \end{pmatrix}.$$

The temporal matrices are specified in Section 2.4.2. For $i, j = 1, \dots, \mathcal{J}$ the spatial matrices are given by

$$[\mathbf{M}_{\text{space}}]_{ji} = \int_\Omega \phi_i(x)\phi_j(x)dx, \qquad [\mathbf{A}_{\text{space}}]_{ji} = \int_\Omega \nabla\phi_i(x)\nabla\phi_j(x)dx,$$
$$[\mathbf{B}_{\text{space}}]_{ji} = \int_\Omega \nabla\phi_i(x)\phi_j(x)dx, \qquad [\mathbf{C}_{\text{space}}]_{ji} = \int_\Omega \phi_i(x)\phi_j(x)dx.$$

The affine decomposition w.r.t. the parameter has four components:

$$\mathbf{B} = \begin{pmatrix} \mathbf{C}_{\text{time}} \otimes \mathbf{M}_{\text{space}} \\ (1, 0, \dots, 0) \otimes \mathbf{M}_{\text{space}} \end{pmatrix} + \alpha \begin{pmatrix} \mathbf{M}_{\text{time}} \otimes \mathbf{A}_{\text{space}} \\ 0 \end{pmatrix} + \beta \begin{pmatrix} \mathbf{M}_{\text{time}} \otimes \mathbf{B}_{\text{space}} \\ 0 \end{pmatrix} + \gamma \begin{pmatrix} \mathbf{M}_{\text{time}} \otimes \mathbf{C}_{\text{space}} \\ 0 \end{pmatrix}.$$

The first is independent of the actual parameters, i.e. $\theta_b^1(\mu) = 1$ for arbitrary $\mu = (\alpha, \beta, \gamma)^T \in \mathcal{D}$.

The right-hand side is parameter independent and only has one component and a μ-independent coefficient. In Figure 2.5.2, the error between the true solution and the finite element solution is detailed for the parameter $\mu = (1, 0.5, 0.5)$. The finite element solution is projected onto the grid given by $L_1 = L_2 = 10$. The analytic solution is evaluated on the same grid. We achieve an exponential convergence w.r.t. the space discretisation for fine time discretisations.

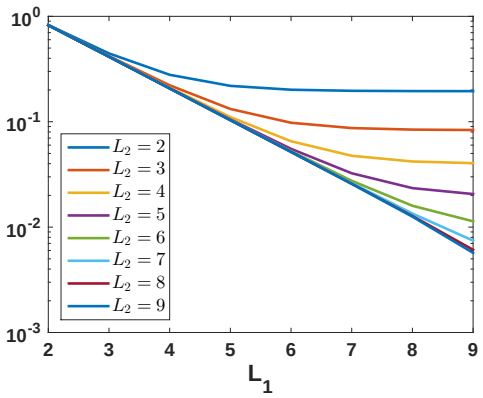
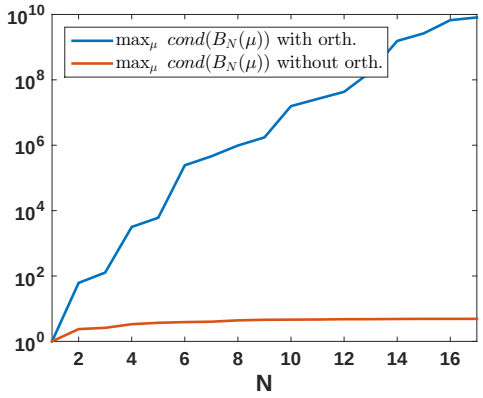


Figure 2.5.2.: Finite element approximation error $\|u(1, 0.5, 0.5) - u^{\mathcal{N}}(1, 0.5, 0.5)\|_{\mathbb{X},\mathrm{bar}}$ measured by projection on the grid given by $L_1 = L_2 = 10$.

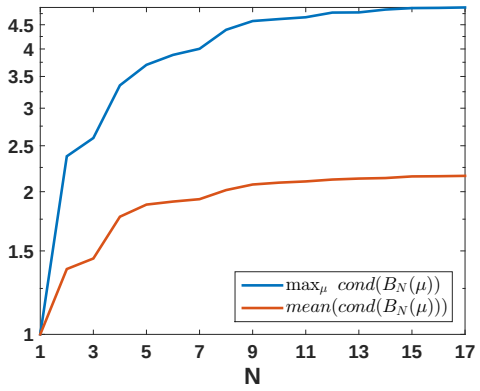
2.5.4. Offline and Online Computations

We set $L_1 = L_2 = 6$ for a finite element approximation error below 10^{-1}. The training set $\mathcal{D}^{\mathrm{train}}$ is given by 17 equidistantly distributed points in every parameter direction. Running the offline phase, we monitor the condition number of the system matrix $\mathbf{B}_N(\mu)$, cf. Figure 2.5.3. In Figure 2.5.3 (a) we compare the maximal condition number over all μ in dependence of N once with using the snapshots directly and once with taking orthonormalised snapshots in the reduced basis. The condition number is bounded using an orthonormalised reduced basis, cf. Figure 2.5.3 (b). The condition number behaves like expected and already observed in [RV07, Sec. 7] for the Stokes equation. The greedy decay is shown in Figure 2.5.5 (a). For $N = 16$ we reach an RB-tolerance of 10^{-3} using an orthonormalised reduced basis. We use an estimated lower bound for the inf-sup constant, $\beta_{\mathrm{LB}} = 0.2$, cf. Figure 2.5.4. The error estimator presented in Proposition 2.2.38 is capable to catch the error behaviour, cf. Figure 2.5.5 (a). Performing an SCM would improve the results and would provide an online feasible strategy for computing the required inf-sup constants.

In Figure 2.5.5 (b) the RB approximation error for the subset $[0.5, 1.5]\times\{0\}\times\{0\}$ of the parameter domain is shown and compared to the error estimator. Both, the error and the estimator are bounded by the chosen approximation tolerance `tol` $= 0.001$, as expected. Further, we can see



(a) Maximal condition using orthonormalised RB vs. snapshots directly.



(b) Maximal condition vs. mean for orthonormalised RB.

Figure 2.5.3.: Condition of the RB system matrix $B_N(\mu)$.

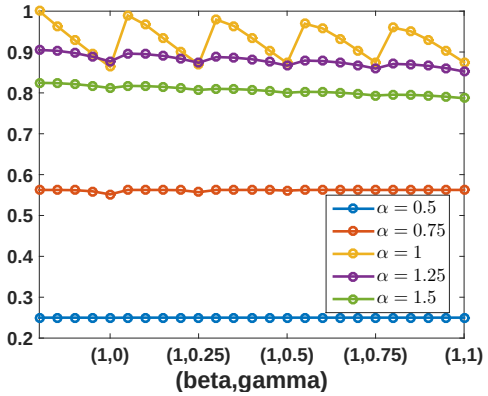
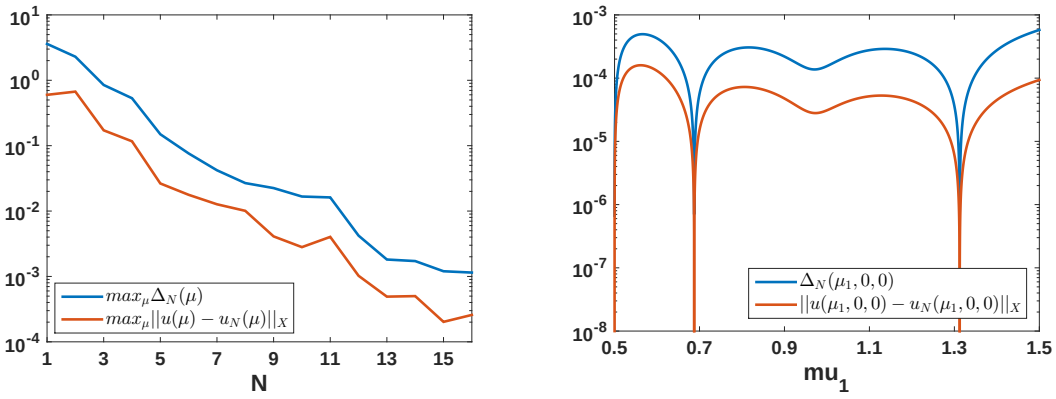


Figure 2.5.4.: Inf-sup values for a test set of the parameter domain.

that there are three RB basis elements that correspond to the solution of a parameter given of the form $(\mu_1, 0, 0)$.

We achieve a reduction of the high dimension $\mathcal{N} = 2^6 2^6 = 4096$ to low dimension $N = 16$.



(a) RB approximation error decay in the greedy procedure.

(b) RB approximation error in comparison to the error estimator for the full parameter domain $\mathcal{D}$.

Figure 2.5.5.: Offline and online RB approximation errors.

3. Parameter Functions

Parts presented in this chapter are published in [MU14]. The notation was adapted to the present thesis and the framework has been extended.

3.1. Introduction

Even if the reduced basis method is a model reduction method specifically designed for parameter dependent PDEs, a high dimensional parameter space stays a challenging task. When speaking about high dimensional parameter spaces we have to distinguish between standard parameter spaces $\mathcal{D} \subset \mathbb{R}^d$ for large $0 \ll d$ and (infinite dimensional) function spaces as parameter spaces, i.e., $\mathcal{D} \subset H$ for H being a Hilbert space.

A parameter space $\mathcal{D} \subset \mathbb{R}^d$ may called high dimensional for some arbitrary large $0 \ll d \in \mathbb{N}$. Note that, depending on the application $\mathcal{D} \subset \mathbb{R}^5$ might already be high dimensional. The question here is how to construct a reduced basis in the first place: The Greedy procedure which is usually used for the construction relies on a training set that has to describe the parameter space very well. The offline procedure may become infeasible for very large sampling sets. The works of J. Hesthaven, B. Stamm, S. Zhang, D. Devaud , A. Manzoni and G. Rozza, [HSZ14, HZ11, DMR13], focus on these high dimensionalities. Facing large parameter spaces, one can use parameter domain decomposition techniques, sensitivity analysis approaches or adequate Greedy methods.

This section is concerned with parameter spaces that are not or not entirely a subspace of $\mathbb{R}^d$ anymore but a subspace of an infinite-dimensional Hilbert space H. The before described high dimensionality may still occur, but in addition we face a whole new type of parameter space - (a subspace of) an infinite dimensional function space. The idea is to be able to construct a reduced basis that not only allows to solve the problem for new standard parameters to a given accuracy but also to allow the variation of the initial condition of the parabolic PDE as a function parameter. State of the art is to construct a new reduced basis if the model changes in a way that a new function is needed at some point. E.g. in derivative pricing, this provides us with the flexibility to construct one reduced basis that can be used for very different and possibly customized payoff functions.

3.2. Reduced Basis Method for Parameter Functions in the Initial Condition

Recall the space-time framework introduced in Chapter 2. The initial condition of the parabolic PDE is the parameter function $\mu_0 \in \mathcal{D}_0 \subset H$ for a Hilbert space H. The parameter space includes a standard parameter $\mu_1 \in \mathcal{D}_1 \subset \mathbb{R}^d$, $d \in \mathbb{N}$, in addition to the parameter functions and is given by $\mathcal{D} = \mathcal{D}_0 \times \mathcal{D}_1$. The parameter dependence of the (bi)linear forms of the space-time variational formulation is given by, cf. Example 2.2.34,

$$\begin{aligned} b(u, v; \mu_1) &:= \int_I \langle \dot{u}(t), z(t) \rangle_{V' \times V} \, dt + \int_I a(u(t), z(t); \mu_1) dt + (u(0), h)_H \\ &=: b_1(u, z; \mu_1) + (u(0), h)_H \end{aligned} \tag{3.2.1a}$$

and

$$f(v; \mu) := \int_I \langle g(t; \mu_1), z(t) \rangle_{V' \times V} \, dt + (\mu_0, h)_H =: g_1(z; \mu_1) + (\mu_0, h)_H \tag{3.2.1b}$$

for $\mu = (\mu_0, \mu_1) \in \mathcal{D}$ and $u \in \mathbb{X}$, $(z, h) \in \mathbb{Y}$. The problem formulation is of the form (2.2.25), i.e.

$$\text{find } u(\mu) \in \mathbb{X} \text{ such that } b(u(\mu), v; \mu_1) = f(v; \mu) \quad \forall \, v \in \mathbb{Y}. \tag{3.2.2}$$

The bilinear form $b(\cdot, \cdot; \mu_1)$ only depends on μ_1, whereas the right-hand side $f(\cdot; \mu)$ depends on the full parameter $\mu = (\mu_0, \mu_1)$. For the numerical treatment, we consider non-trivial $\mathcal{N}$-dimensional discrete subspaces $\mathbb{X}^{\mathcal{N}} \subset \mathbb{X}$ and $\mathbb{Y}^{\mathcal{N}} \subset \mathbb{Y}$ as proposed in Section 2.2.2.2 and we assume well-posedness of the discrete problem forumlation. For the new greedy procedure as well as the error estimator we separate the space-time variational formulation into two subproblems: The first gives an approximation of the solution at time $t = 0$ using the initial value only and the second determines the evolution of the solution u. We specify the parameter dependence of the solution by introducing a second argument, e.g. $u^{\mathcal{N}}(\mu)(0) = u^{\mathcal{N}}(0; \mu)$. In the notation of Section 2.2.2.2 define $E_0^1 := \langle \sigma^0 \rangle$ and $E_1^{\mathcal{K}} := \langle \sigma^1, \dots, \sigma^{\mathcal{K}} \rangle$. The ansatz space, given in Equation (2.2.1) is isomorphic to

$$\mathbb{X}^{\mathcal{N}} = E_0^1 \otimes V^{\mathcal{J}} \oplus E_1^{\mathcal{K}} \otimes V^{\mathcal{J}} =: \mathsf{Q}^{\mathcal{J}} \oplus \mathsf{W}^{\mathcal{I}}$$

for $\mathcal{I} := \mathcal{K}\mathcal{J}$ and as before, cf. Equation (2.2.5)

$$\mathbb{Y}^{\mathcal{N}} = F^{\mathcal{K}} \otimes V^{\mathcal{J}} \times V^{\mathcal{J}} =: \mathsf{Z}^{\mathcal{I}} \times V^{\mathcal{J}}. \tag{3.2.3}$$

By definition, $\mathsf{Q}^{\mathcal{J}} \hookrightarrow \mathbb{X}$ and $\mathsf{W}^{\mathcal{I}} \hookrightarrow \mathbb{X}$ are canonically embedded. Further

$$\mathsf{W}^{\mathcal{I}} \cong \{w \in \mathbb{X} : w(0) = 0\} = \operatorname{span}\{\sigma^k \otimes \phi_i : \, k = 1, \dots, \mathcal{K}, \, i = 1, \dots, \mathcal{J}\}.$$

We divide the problem for $\mu = (\mu_0, \mu_1) \in \mathcal{D}$ as follows:

$$\text{(a) Find } q(\mu_0) \in \mathsf{Q}^{\mathcal{J}}: \qquad (q(0;\mu_0), h)_H = (\mu_0, h)_H \qquad \forall h \in V^{\mathcal{J}}, \tag{3.2.4a}$$

$$\text{(b) Find } w(\mu) \in \mathsf{W}^{\mathcal{I}}: \qquad b_1(w, z; \mu_1) = f_1(z; \mu_1, q(\mu_0)) \qquad \forall z \in \mathsf{Z}^{\mathcal{I}}, \tag{3.2.4b}$$

with the modified bilinear forms defined in (3.2.1) and the right-hand side $f_1(z; \mu_1, q) := g_1(z; \mu_1) - b_1(q, z; \mu_1)$.

Remark 3.2.1. We do not specify the embeddings $\mathsf{Q}^{\mathcal{J}} \hookrightarrow \mathbb{X}$ and $\mathsf{W}^{\mathcal{I}} \hookrightarrow \mathbb{X}$ in the notation, i.e. we denote the elements in $\mathsf{Q}^{\mathcal{J}}$ resp. $\mathsf{W}^{\mathcal{I}}$ by the same letter as their images in $\mathbb{X}$.

Proposition 3.2.2. The discrete problem formulations (3.2.4) and (3.2.2) are equivalent.

Proof. If q and w are the solutions of (3.2.4) we define $u := q + w \in \mathbb{X}$ satisfying (3.2.2). For a solution of (3.2.2) we separate the solution into two parts, $u^{\mathcal{N}} = \sigma^0 \otimes u^{\mathcal{N}}(0) + \sum_{k=1}^{\mathcal{K}} \sum_{\ell=1}^{\mathcal{J}} \alpha_k^\ell \sigma^k \otimes \phi_\ell =: q_0 + w^{\mathcal{I}}$. Then, q_0 satisfies Equation (3.2.4a) and $w^{\mathcal{I}} \in \mathsf{W}^{\mathcal{I}}$ satisfies Equation (3.2.4b). □

For $\mu := (\mu_0, \mu_1) \in \mathcal{D}$ the residual reads

$$\begin{aligned} r_N(v;\mu) &= f(v;\mu) - b(u_N(\mu), v; \mu) \\ &= f(v;\mu) - b(u_N(\mu), v; \mu) \\ &= g_1(z;\mu_1) + (\mu_0, h)_H - b_1(u_N(\mu), z; \mu_1) + (u_N(\mu)(0), h)_H \\ &= g_1(z;\mu_1) - b_1(u_N(\mu), z; \mu_1) + (\mu_0 - (u_N(\mu))(0), h)_H \\ &=: r_{N,1}(z;\mu) + r_{N,0}(h;\mu), \end{aligned}$$

for any $v = (z, h) \in \mathbb{Y}^{\mathcal{N}}$. Recall, that we need to construct a reduced basis that ensures a small residual for the full parameter space $\mathcal{D}$. In order to do so, we need an efficient online computation of the error estimator $\Delta_N(\mu) = \frac{\|r_N(\cdot;\mu)\|_{(\mathbb{Y}^{\mathcal{N}})'}}{\beta^{\mathcal{N}}}$ introduced in Proposition 2.2.38 realised by the decomposition of the residual $r_N(v;\mu)$ into its affine components. This is no problem for $r_{N,1}(z;\mu)$ due to the separation properties of g_1 and b_1, cf. Section 2.3.1. The second part $r_{N,0}(h;\mu)$ needs special treatment in the online procedure. By choosing appropriate approximation spaces, the separation allows to control the error in the initial value directly using the same analysis as for the general approximation error of the function μ_0.

The following estimate is immediate:

$$\begin{aligned} \|r_N(\cdot;\mu)\|_{\mathbb{Y}'} &= \sup_{v \in \mathbb{Y}} \frac{r_N(v;\mu)}{\|v\|_{\mathbb{Y}}} = \sup_{(z,h) \in \mathbb{Y}^{\mathcal{N}}} \frac{r_{N,1}(z;\mu) + r_{N,0}(h;\mu)}{(\|z\|_{\mathsf{Z}}^2 + \|h\|_H^2)^{1/2}} \\ &= \|r_{N,1}(\cdot;\mu)\|_{\mathsf{Z}'} + \sup_{h \in \mathcal{H}} \frac{r_{N,0}(h;\mu)}{\|h\|_H} \leq \|r_{N,1}(\cdot;\mu)\|_{\mathsf{Z}'} + \|\mu_0 - u_N(0;\mu)\|_H \\ &=: R_{N,1}(\mu) + R_{N,0}(\mu_0). \end{aligned}$$

At a first glance it seems that $r_{N,1}$ (and $R_{N,1}$) only depends on μ_1. However, the RB solution $u_N(\mu)$ involves both μ_0 and μ_1 so that both parameters enter. As already mentioned earlier, the approximation of the initial value (and hence $R_{N,0}$), depends only on μ_0. For $\Delta_N^1(\mu) := \frac{1}{\beta^{\mathcal{N}}} R_{N,1}(\mu)$ and $\Delta_N^0(\mu) := \frac{1}{\beta^{\mathcal{N}}} R_{N,0}(\mu_0)$, the separation of the error estimator follows,

$$\Delta_N(\mu) = \Delta_N^1(\mu) + \Delta_N^0(\mu_0). \tag{3.2.5}$$

3.2.1. A Two-Step Greedy Method

In this first approach, cf. [MU14], we choose in a given approximation space $\mathcal{D}_0^{\mathcal{L}}$ of $\mathcal{D}_0$ the most relevant functions for the parameter space. We fix the approximation of the function μ_0 offline in a first step and do not take the operator of the PDE into account. In a second step, we perform a Greedy procedure to obtain a reduced basis for the evolution of the PDE solution. Online, we stay with the a-priori chosen sample set for the approximation of the initial value. The separated system is used in the reduced basis construction.

Given the parameter space $\mathcal{D} = \mathcal{D}_0 \times \mathcal{D}_1 \subset H \times \mathbb{R}^d$, we assume a finite dimensional approximation space of the function parameter space $\mathcal{D}_0$, i.e., μ_0 is approximately given by $\tilde{\mu}_0 \in \mathcal{D}_0^{\mathcal{L}} = \operatorname{span}\{\delta^1, \dots, \delta^{\mathcal{L}}\}$. Note, that the set $\{\delta^1, \dots, \delta^{\mathcal{L}}\}$ does not have to be a basis in the classical sense but can be a collection of functions that describe the parameter space as good as possible. Using the approximation space $\mathcal{D}_0^{\mathcal{L}}$, we get the usual separation of the form $(\tilde{\mu}_0, h)_H = \sum_{\ell=1}^{\mathcal{L}} \beta_\ell(\tilde{\mu}_0)\,(\delta_\ell, h)_H$, for the right-hand side of the initial value problem. The second part $R_{N,0}$ of the residual is offline-online decomposable using the finite description $\mathcal{D}_0^{\mathcal{L}}$. However, we will see that in the online computations we may have a linear dependence on $\mathcal{L}$, when the function μ_0 is applied in the right-hand side.

Remark 3.2.3. (a) A first idea by introducing a finite dimensional approximation space is to transfer the parameter function space $\mathcal{D}_0^{\mathcal{L}}$ to a standard parameter space $\mathbb{R}^{\mathcal{L}}$ by mapping $\mu_0 \mapsto (\beta_1(\mu_0), \dots, \beta_{\mathcal{L}}(\mu_0))$. This results in an $\mathcal{L}$-dimensional parameter space containing the coefficients β_ℓ as parameters. However, this may become computationally infeasible: in general the coefficients cannot be bounded and if they can we would need to work with an $\mathcal{L}$-dimensional hypercube of potentially very large side lengths. Consequently, determining parameter samples for snapshots e.g. by a greedy method might be extremely costly, not only for large values of $\mathcal{L}$. In particular, the transformation neglects the fact that $\mu_0 \in \operatorname{span}\{\delta_1, \dots, \delta_{\mathcal{L}}\}$.

(b) For a full error control we require some knowledge of the approximation error $\|\mu_0 - \tilde{\mu}_0\|_H$ affecting $\Delta_N^0(\mu)$.

The separation of the error estimate in Equation (3.2.5) suggests to compute parameter samples (and snapshots) in two steps, namely first to determine N_0 samples μ_0^i for the initial value by maximising $R_{N,0}(\mu_0)$ w.r.t. $\mu_0 \in \mathcal{D}_0$ and second to consider the evolution and compute samples μ^j by maximising $R_{N,1}(\mu^j)$ using the before-computed snapshots for the initial value contained in $\mathsf{Q}^{\mathcal{J}}$. This corresponds to separated computations.

Let us now describe the two parts in detail. We remark that even though we describe a greedy method, one could also use a different method to determine appropriate parameter samples e.g. by using nonlinear optimisation w.r.t. the error estimate, [BTWG08, UVZ14]. The separation into two steps is independent of the particular maximisation strategy.

3.2.1.1. Offline Phase - Initial Value Approximation

The first step is to generate a reduced basis for the initial value, i.e., we need the solution at $t = 0$, which only depends on the parameter function $\mu_0 \in \mathcal{D}_0$, as we have seen in (3.2.4). For a given tolerance $\mathrm{tol}_0 > 0$, we are looking for a sample set $S_0^{N_0} := \{\mu_0^1, \dots, \mu_0^{N_0}\}$ and corresponding solutions $q^i := q(\mu_0^i)$. We denote the corresponding elements in $\mathsf{Q}^{\mathcal{J}}$ by $u_0^i := \sigma^0 \otimes q^i \in \mathsf{Q}^{\mathcal{J}}$, $i = 1, \dots, N_0$ and the embedding $u_0^i + 0 \in \mathbb{X}^{\mathcal{N}}$ with the same letter u_0^i (i.e., $u_0^i := \sigma^0 \otimes q^i + w^i$, $w^i \in \mathsf{W}^{\mathcal{I}}$, $w^i \equiv 0$).

For a parameter $\mu_0 \in \mathcal{D}_0$, the corresponding solution $q(\mu_0) \in V^{\mathcal{J}}$ is determined by

$$(q(\mu_0), \phi_j)_H = (\mu_0, \phi_j)_H, \qquad 1 \le j \le \mathcal{J}, \tag{3.2.6}$$

where $\{\phi_1, \dots, \phi_{\mathcal{J}}\}$ is the chosen (FEM) basis for $V^{\mathcal{J}}$. Given a sample set $S_0^{N_0}$ and a corresponding RB space $V_{N_0} = \{q^1, \dots, q^{N_0}\}$ (where we should have $N_0 \ll \mathcal{J}$) a corresponding RB initial value approximation $u_{N_0}^{\mathrm{init}}(\mu_0^*)$ for some new parameter $\mu_0^* \in \mathcal{D}_0$ is determined by solving the system of linear equations corresponding to

$$(q_{N_0}(\mu_0^*), q^i)_H = (\mu_0^*, q^i)_H, \quad \forall\ 1 \le i \le N_0 \quad \text{and} \quad q_{N_0}(\mu_0^*) = \sum_{i=1}^{N_0} \alpha_i(\mu_0^*) q^i. \tag{3.2.7}$$

The inner products $(\mu_0^*, q^i)_H$ need to be computed online efficient (i.e., with complexity independent of $\mathcal{J}$). Then, the error contribution reads

$$R_{N,0}(\mu_0) = \|\mu_0 - q_{N_0}(\mu_0)\|_H.$$

Working with a finite approximation $\mathcal{D}_0^{\mathcal{L}} = \mathrm{span}\{\delta^1, \dots, \delta^{\mathcal{L}}\}$ of $\mathcal{D}_0$ the sample set $\{\mu_0^1, \dots, \mu_0^{N_0}\}$ can be determined in different ways:

For a Proper Orthogonal Decomposition (POD) compute the Gramian $\mathbf{M}_H^{\mathcal{L}} := \left((\delta^\ell, \delta^{\ell'})_H\right)_{\ell,\ell'=1,\dots,\mathcal{L}}$ and choose $\mu_0^1, \dots, \mu_0^{N_0}$ as the (orthogonalised) eigenfunctions corresponding to the N_0 largest eigenvalues of $\mathbf{M}_H^{\mathcal{L}}$. If $\mathcal{D}_0^{\mathcal{L}}$ is chosen well, this approach results in the best H-orthogonal choice. The obvious drawback is the strong dependence on the choice of the δ^i, $i = 1, \dots, \mathcal{L}$.

For a greedy procedure, one chooses a finite training set $\mathcal{D}_0^{\text{train}} \subset \mathcal{D}_0$ and determines parameter samples by maximising $R_{N,0}(\mu_0)$ over $\mu_0 \in \mathcal{D}_0^{\text{train}}$. We obtain the greedy scheme in Algorithm 3.2.1. Of course $\mathcal{D}_0^{\text{train}} \subset \mathcal{D}_0^{\mathcal{L}}$ is a reasonable choice.

Algorithm 3.2.1 Initial Value Greedy

Input: Training set of initial values $D_0^{\text{train}} \subset \mathcal{D}_0$, tolerance $\text{tol}_0 > 0$
Output: RB basis Ξ_{N_0}, RB space V_{N_0}, sample set $S_0^{N_0}$

1: Choose $\mu_0^1 \in D_0^{\text{train}}$, $S_0^1 := \{\mu_0^1\}$, compute $q(\mu_0^1)$ as in (3.2.6), normalise $\Xi_1 := \{q(\mu_0^1)\}$.
2: **for** $j = 1, \dots, N_0^{\max}$ **do**
3: $\quad \mu_0^{j+1} = \arg\max_{\mu_0 \in \mathcal{D}_0^{\text{train}}} R_{j,0}(\mu_0)$
4: $\quad$ **if** $R_{j,0}(\mu_0^{j+1}) < \text{tol}_0$ **then** $N_0 := j$, $V_{N_0} := \text{span}(\Xi_{N_0})$; **Stop end if**
5: $\quad$ Compute $q(\mu_0^{j+1}) \in V^{\mathcal{J}}$ as in (3.2.6).
6: $\quad S_0^{j+1} := S_0^j \cup \{\mu_0^{j+1}\}$, $\Xi_{j+1} := \Xi_j \cup \{q(\mu_0^{j+1})\}$, orthonormalise Ξ_{j+1}.
7: **end for**

It remains to discuss the efficient computation of the error term $R_{N_0,0}(\mu_0)$ for a given parameter $\mu_0 \in \mathcal{D}_0$. Note, that we obtain a set of orthonormal functions Ξ_{N_0} as an output of Algorithm 3.2.1. Hence, the RB approximation $q_{N_0}(\mu_0)$ coincides with the H-orthogonal projection of μ_0 to V_{N_0}. This means that $R_{N_0,0}(\mu_0)$ is the error of the best approximation of μ_0 in V_{N_0}. There are different possibilities to compute this error:

- If μ_0 is given as closed formula, then an efficient quadrature may be used.
- If μ_0 has a finite expression in terms of a stable basis $\{\delta^1, \dots, \delta^{\mathcal{L}}\}$, one may use an efficient quadrature to precompute $(\delta_i, q^j)_H$ or transform $q_{N_0}(\mu_0)$ into that basis and use the coefficients of the difference.
- One could compute an orthonomal basis for the complement $V^{\mathcal{J}} \ominus V_{N_0}$ and approximate $R_{N_0,0}(\mu_0)$ by computing coefficients of μ_0 w.r.t. that complement basis (e.g. in terms of wavelets).

3.2.1.2. Offline Phase - Evolution Greedy

The next step is to find a basis for the part of the solution u in $\mathsf{W}^{\mathcal{I}}$, given the already determined reduced space V_{N_0}. For a parameter $\mu = (\mu_0, \mu_1) \in \mathcal{D}$ and an approximation $q_{N_0}(\mu_0)$, the

evolution part $w \in \mathsf{W}^{\mathcal{I}}$ is computed as

$$b_1(w(\mu), z; \mu_1) = g_1(z; \mu_1) - b_1(\sigma^0 \otimes q_{N_0}(\mu_0), z; \mu_1) \quad \forall\ z \in \mathsf{Z}^{\mathcal{I}}.$$

A reduced basis approximation corresponding to $\mu = (\mu_0, \mu_1) \in \mathcal{D}$ requires to first determine

$$q_{N_0}(\mu_0) = \sum_{i=1}^{N_0} \alpha_i(\mu_0)\, q^i \tag{3.2.8a}$$

using (3.2.7). Then, given parameter samples $S_1^{N_1} = \{\mu^1, \dots, \mu^{N_1}\} \subset \mathcal{D} = \mathcal{D}_0 \times \mathcal{D}_1$ (to be determined e.g. by a second greedy described below) and corresponding snapshots $w^i := w(\mu^i) \in \mathsf{W}^{\mathcal{I}}$, $i = 1, \dots, N_1$, a reduced basis approximation $w_{N_1}(\mu) \in \mathsf{W}_{N_1} = \operatorname{span}\{w^i : 1 \le i \le N_1\}$ is determined by

$$b_1(w_{N_1}(\mu), z_{N_1}; \mu_1) = g_1(z_{N_1}; \mu_1) - b_1(\sigma^0 \otimes q_{N_0}(\mu_0), z_{N_1}; \mu_1), \tag{3.2.8b}$$

for all $z_{N_1} \in \mathsf{Z}_{N_1}$, where Z_{N_1} is a stable reduced space corresponding to W_{N_1} w.r.t. the inner product b_1 in the sense that

$$\inf_{w_{N_1} \in \mathsf{W}_{N_1}} \sup_{z_{N_1} \in \mathsf{Z}_{N_1}} \frac{b_1(w_{N_1}, z_{N_1}; \mu_1)}{\|w_{N_1}\|_{\mathsf{W}}\, \|z_{N_1}\|_{\mathsf{Z}}} \ge \beta_1(\mu_1) > 0 \tag{3.2.9}$$

independent of $N_1 \to \infty$, see Section 3.2.1.5 below. Here, β_1 is the inf-sup constant of the bilinear form b_1 w.r.t. $\mathsf{W}_{N_1} \times \mathsf{Z}_{N_1}$.

It is readily seen that the right-hand side of (3.2.8b) admits a separation w.r.t. the parameter. In fact, recalling the affine decomposition w.r.t. the parameter, we have

$$\begin{aligned} f_1(z_{N_1}; \mu_1, \sigma^0 \otimes q_{N_0}(\mu_0)) &= g_1(z_{N_1}; \mu_1) - b_1(\sigma^0 \otimes q_{N_0}(\mu_0), z_{N_1}; \mu_1) \\ &= \sum_{q=1}^{Q_g} \theta_q^g(\mu_1)\, g_q(z_{N_1}) + \sum_{q=1}^{Q_b} \theta_q^b(\mu_1)\, b_q(\sigma^0 \otimes q_{N_0}(\mu_0), z_{N_1}) \\ &= \sum_{q=1}^{Q_g} \theta_q^g(\mu_1)\, g_q(z_{N_1}) + \sum_{q=1}^{Q_b} \sum_{n=1}^{N_0} \theta_q^b(\mu_1)\, \alpha_{0,n}(\mu_0)\, b_q(\sigma^0 \otimes q^n, z_{N_1}) \\ &=: \sum_{q=1}^{Q_g + N_0\, Q_b} \theta_q^{f_1}(\mu)\, f_q(z_{N_1}; \sigma^0 \otimes q^n) \end{aligned} \tag{3.2.10}$$

with obvious definitions of the involved terms. Hence, we obtain an efficient offline-online splitting both for the computation of the reduced basis approximation $w_{N_1}(\mu)$ and of the residual $r_{N,1}(\mu; z) = g_1(\mu_1; z) - b_1(\mu_1; u_N(\mu), z)$, where we set $u_N(\mu) := u_{N_0}^{\mathrm{init}}(\mu_0) + w_{N_1}(\mu) =$

$\sigma^0 \otimes q_{N_0}(\mu_0) + w_{N_1}(\mu)$, which means that

$$\begin{aligned} r_{N_1,1}(z;\mu) &= g_1(z;\mu_1) - b_1(u_N(\mu), z;\mu_1) \\ &= g_1(z;\mu_1) - b_1(\sigma^0 \otimes q_{N_0}(\mu_0), z;\mu_1) - b_1(w_{N_1}(\mu), z;\mu_1), \end{aligned}$$

which coincides with the residual of (3.2.8b). The error estimator is given by (cf. (3.2.5)):

$$\Delta^1_{N_1}(\mu) := \frac{\|r_{N_1,1}(\mu)\|_{\mathsf{Z}'}}{\beta_{\mathrm{LB}}} = \frac{R_{N_1,1}(\mu)}{\beta_{\mathrm{LB}}},$$

where β_{LB} is a lower bound of the inf-sup constant of the bilinear form b. We obtain a – more or less – standard greedy scheme described in Algorithm 3.2.2.

Algorithm 3.2.2 Evolution Greedy

Input: training set $\mathcal{D}^{\mathrm{train}} \subset \mathcal{D}$, tolerance $\mathrm{tol}_1 > 0$
Output: RB basis Ξ_{N_1}, RB space W_{N_1}, sample set $S_1^{N_1}$

1: Choose $\mu^1 \in \mathcal{D}^{\mathrm{train}}$, $\mu^1 := (\mu_0^1, \mu_1^1)$, $S_1^1 := \{\mu^1\}$
2: Compute RB approximation $q_{N_0}(\mu_0^1) \in V_{N_0}$ as in (3.2.8a)
3: Compute $w(\mu^1) \in \mathsf{W}^{\mathcal{I}}$ as in (3.2.8b), $\Xi_1 = \{w(\mu^1)\}$
4: **for** $j = 1, \ldots, N_1^{\max}$ **do**
5: $\quad \mu^{j+1} = \arg\max_{\mu \in \mathcal{D}^{\mathrm{train}}} \Delta_j^1(\mu)$
6: $\quad$ **if** $\Delta_j^1(\mu^{j+1}) < \mathrm{tol}_1$ **then** $N_1 := j$, $\mathsf{W}_{N_1} := \mathrm{span}(\Xi_{N_1})$; **Stop end if**
7: $\quad$ Compute the RB approximation $q_{N_0}(\mu_0^{j+1}) \in S_0^{N_0}$ as in (3.2.8a)
8: $\quad$ Compute $w(\mu^{j+1}) \in \mathsf{W}^{\mathcal{I}}$ as in (3.2.8b)
9: $\quad S_1^{j+1} := S_1^j \cup \{\mu^{j+1}\}$, $\Xi_{j+1} := \Xi_j \cup \{w(\mu^{j+1})\}$
10: **end for**

The question arises how to choose the training set $\mathcal{D}^{\mathrm{train}}$ in Algorithm 3.2.2, in particular the training samples for the initial value parameter. To simplify an a-posteriori online error control it is convenient to take the sample set $S_0^{N_0}$ as the initial value training set in the evolution greedy. Some experiments on the interaction of sample set $S_0^{N_0}$ and $\mathcal{D}_0^{\mathrm{train}}$ are to be found in [MU14, Sec. 5.3].

3.2.1.3. Online Phase

The reduced basis $\{q^1, \ldots, q^{N_0}\} \subset V^{\mathcal{J}}$ for the initial value and the reduced basis $\{w^1, \ldots, w^{N_1}\} \subset \mathsf{W}^{\mathcal{I}}$ for the evolution of the solution are now available. For any new parameter $(\mu_0, \mu_1) \in \mathcal{D}$ we first have to solve

$$(\sum_{i=1}^{N_0} \bar{\alpha}_i (q^i, q^j)_H)_{j=1,\ldots,N_0} = ((\mu_0, q^j)_H)_{j=1,\ldots,N_0}.$$

This is equivalent to solve the system of linear equations

$$((q^i, q^j)_H)_{j,i=1,\dots,N_0}\bar{\boldsymbol{\alpha}} = ((\mu_0, q^j)_H)_{j=1,\dots,N_0}, \quad (\bar{\boldsymbol{\alpha}})_i = \bar{\alpha}_i.$$

The right hand side leads (as described before) to either

(1) a quadrature rule to evaluate the scalar product in H or

(2) if $\mu_0 = \sum_{\ell=1}^{\mathcal{L}} \alpha_\ell \delta^\ell$ we can detail

$$(\mu_0, q^j)_H = (\sum_{\ell=1}^{\mathcal{L}} \alpha_\ell \delta^\ell, q^j)_H = (((\delta^\ell, q^j)_H)_{\ell=1,\dots,\mathcal{L}})^T \boldsymbol{\alpha}$$

and pre-compute $(\delta^i, q^j)_H$ or

(3) the approximation error of μ_0 on $S_0^{N_0}$ is available and we just pre-compute (μ_0^i, q^j) for all $i = 1, \dots, N_0$ and directly add the approximation error in the estimator.

Remark 3.2.4. In case (2), the online effort depends on the high dimension $\mathcal{L}$ as the number of multiplications on the right-hand side. Further, if the expansion coefficients of μ_0 in $\mathcal{D}^{\mathcal{L}}$ resp. $S_0^{N_0}$ are available we can precompute $\left(\sum_{j=1}^{N_0} \frac{(\delta^\ell, q^j)_H}{(q^i, q^j)_H}\right)_{i=1,\dots,N_0}$ for all $\ell = 1, \dots, \mathcal{L}$ in (2) resp. $\left(\sum_{j=1}^{N_0} \frac{(\mu_0^\ell, q^j)_H}{(q^i, q^j)_H}\right)_{i=1,\dots,N_0}$ for all $\ell = 1, \dots, N_0$ in (3).

The evolutionary part can be computed as in (3.2.8) independent of the high-dimension $\mathcal{N}$ since b_1 and f_1 are offline-online decomposable.

3.2.1.4. Online RB Approximation Error

The RB solution computed online may not respect the chosen greedy tolerances. The reason is that the training set of the parameter function space contains only single functions but linear combination of these functions are considered online.

By applying the RB to a new parameter $\mu = (\mu_0, \mu_1)$, we have already seen in Equation (3.2.5) that the error bound can be separated into two parts. The first part contains $\|\mu_0 - u^N(\mu_0, \mu_1)(0)\|_H$ which is equivalent to the error between μ_0 and its projection onto $\text{span}(S_0^{N_0})$:

$$\|\mu_0 - \tilde{\mu}_0\|_H.$$

The error needs to be computed online efficient, i.e. independent of $\mathcal{N}$. We therefore need either an efficient error estimator for the approximation or in the case of a finite description $\mathcal{D}_0^{\mathcal{L}}$ of $\mathcal{D}_0$ we can use the basis $\delta^1, \dots, \delta^{\mathcal{L}}$ and receive the offline-online decomposability but with an online

dependence on $\mathcal{L}$. In any case, the approximation error depends on the approximated function and may highly differ in the applications.

The second part of the error estimator is offline-online decomposable and efficiently computable. We deduce an upper bound of the estimator for the evolutionary part in Proposition 3.2.5 below by assuming that the sample set $S_0^{N_0}$ for the initial value equals the training set of the evolution greedy.

Proposition 3.2.5. Let $V_{N_0} = \operatorname{span}\{q^1, \dots, q^{N_0}\}$, $W_{N_1} = \operatorname{span}\{w^1, \dots, w^{N_1}\}$ and $P : \mathcal{D}_0 \to S_0^{N_0}$ the H-projection of $\mathcal{D}_0$ onto $S_0^{N_0}$. For μ_0 define $\tilde{\mu}_0 := P(\mu_0)$, $\tilde{\mu}_0 = \sum_{\ell=1}^{N_0} d_\ell(\mu_0)\mu_0^\ell$. For given parameter $(\mu_0, \mu_1) \in \mathcal{D}$ let $u_N(\mu_0, \mu_1) = u_{N_0}^{\text{init}} + w_{N_1}$ be the RB approximation. The initial value is given by $u_{N_0}^{\text{init}}(\mu_0) = \sum_{\ell=1}^{N_0} d_\ell(\mu_0) q^\ell$ and the evolutionary part by $w_{N_1} = \sum_{\ell=1}^{N_1} c_\ell(\mu_0, \mu_1) w^\ell$. Then,

$$\|g_1(\cdot) - b_1(u_N(\mu_0, \mu_1), \cdot; \mu_1)\|_{\mathbb{Y}'} \leq |1 - \sum_{\ell=1}^{N_0} d_\ell(\mu_0)| \ \|g_1(\cdot)\|_{\mathbb{Y}'} + |\sum_{\ell=1}^{N_0} d_\ell(\mu_0)| \text{tol}_1,$$

for tol_1 defined as in Algorithm 3.2.2.

Proof.

$$\begin{aligned}
&\|g_1(\cdot) - b_1(u_N(\mu_0, \mu_1), \cdot; \mu_1)\|_{\mathbb{Y}'} \\
&= \|g_1(\cdot) - b_1((u_{N_0}^{\text{init}} + w_{N_1})(\mu_0, \mu_1), \cdot; \mu_1)\|_{\mathbb{Y}'} \\
&= \|g_1(\cdot) - \sum_{k=1}^{N_0} d_k(\mu_0) b_1(\sigma^0 \otimes q^k, \cdot; \mu_1) - \sum_{\ell=1}^{N_1} c_\ell(\mu_0, \mu_1) b_1(w^\ell, \cdot; \mu_1)\|_{\mathbb{Y}'} \\
&\leq \|g_1(\cdot) - \sum_{k=1}^{N_0} d_k(\mu_0) \Big(b_1(\sigma^0 \otimes q^k, \cdot; \mu_1) + \sum_{\ell=1}^{N_1} \tilde{c}_\ell^k(\mu_0^k, \mu_1) b_1(w^\ell, \cdot; \mu_1) \Big) \|_{\mathbb{Y}'} \\
&= \|g_1(\cdot) + \sum_{k=1}^{N_0} d_k(\mu_0) \Big(g_1(\cdot) - b_1(\sigma^0 \otimes q^k, \cdot; \mu_1) - \sum_{\ell=1}^{N_1} \tilde{c}_\ell^k(\mu_0^k, \mu_1) b_1(w^\ell, \cdot; \mu_1) \Big) - \sum_{k=1}^{N_0} d_k(\mu_0) g_1(\cdot) \|_{\mathbb{Y}'} \\
&\leq |1 - \sum_{k=1}^{N_0} d_k(\mu_0)| \ \|g_1(\cdot)\|_{\mathbb{Y}'} + |\sum_{k=1}^{N_0} d_k(\mu_0)| \text{tol}_1.
\end{aligned}$$

For the third inequality the parameters $c_\ell(\mu_0, \mu_1)$ are replaced by $\sum_{k=1}^{N_0} d_k(\mu_0) \tilde{c}_\ell^k(\mu_0^k, \mu_1)$ for $\ell = 1, \dots, N_1$. The coefficient $\tilde{c}_\ell^k(\mu_0^k, \mu_1)$ is given by solving (3.2.8) for the initial value $\mu_0^k \in S_0^{N_0}$, for $k = 1, \dots, N_0$ and $\ell = 1, \dots, N_1$. □

The upper bound can be evaluated before performing the actual RB approximation. It connects the online RB approximation error and the previously used greedy tolerance. Further, we can explore that the chosen parameter function has an impact on the RB approximation in terms of

its expansion coefficients and that the approximation might improve if the right-hand side of the PDE equals to zero.

3.2.1.5. Stable RB Test Spaces for the Separated System

As for the space-time RBM in general, it remains to construct a stable test space Z_{N_1} in the sense of (3.2.9). As before, this space is constructed using appropriate supremizers in an efficient offline-online manner.

Let $\{w^1, \dots, w^{N_1}\}$ be the basis of W_{N_1} and fix $\mu_1 \in \mathcal{D}_1$. Then, the supremizer $S_{w^n}(\mu_1) \in \mathsf{Z}^{\mathcal{I}}$, $1 \le n \le N_1$, is defined by the relation

$$S_{w^n}(\mu_1) := \arg\sup_{z \in \mathsf{Z}^{\mathcal{I}}} \frac{b_1(\mu_1; w^n, z)}{\|z\|_{\mathsf{Z}}}, \quad 1 \le n \le N_1.$$

In order to compute this quantity, recall that $\{\zeta_i := \tau^k \otimes \phi_j : 1 \le k \le \mathcal{K}, 1 \le j \le \mathcal{J}, i = (k,j)\}$ is the basis of $\mathsf{Z}^{\mathcal{I}}$ and note that

$$\begin{aligned}\|z\|_{\mathsf{Z}}^2 = \|z\|_{L_2(I;V)}^2 &= \sum_{k,k'=1}^{\mathcal{K}} \sum_{j,j'=1}^{\mathcal{J}} z_j^k z_{j'}^{k'} \, (\tau^k, \tau^{k'})_{L_2(I)} \, (\phi_j, \phi_{j'})_V \\ &= \boldsymbol{z}^T (\mathbf{I}_{\text{time}}^{\mathcal{K}} \otimes \mathbf{M}_{\text{space}}^{\mathcal{J}}) \boldsymbol{z} =: \boldsymbol{z}^T \mathbf{Z}^{\mathcal{I}} \boldsymbol{z}\end{aligned}$$

with the Gramian matrices $\mathbf{M}_{\text{space}}^{\mathcal{J}} = ((\phi_i, \phi_j)_V)_{i,j=1,\dots,\mathcal{J}}$ for $V^{\mathcal{J}}$ (w.r.t. the V-inner product) and $\mathbf{I}_{\text{time}}^{\mathcal{K}} = ((\tau^k, \tau^{k'})_{L_2(I)})_{k,k'=1,\dots,\mathcal{K}} = \Delta t \, \mathbb{1} \in \mathbb{R}^{\mathcal{K} \times \mathcal{K}}$. Next, let the expansion of w^n in terms of the full basis $\{\varpi_i := \sigma^k \otimes \phi_j : i = (k,j), 1 \le k \le \mathcal{K}, 1 \le j \le \mathcal{J}\}$ of $\mathsf{W}^{\mathcal{I}}$ be denoted by

$$w^n = \sum_{i=1}^{\mathcal{I}} \omega_i^n \varpi_i, \qquad \boldsymbol{\omega}^n := (\omega_i^n)_{i=1,\dots,\mathcal{I}}.$$

Then, setting $z_i := z_j^k$, $i = (k,j)$, we get

$$b_1(\mu_1; w^n, z) = \sum_{i,i'=1}^{\mathcal{I}} \omega_i^n z_{i'} b_1(\mu_1; \varpi_i, \zeta_{i'}) = (\boldsymbol{z})^T \mathbf{B}^{\mathcal{I}}(\mu_1) \boldsymbol{\omega}^n.$$

The vector $\boldsymbol{s}_n(\mu_1)$ containing the expansion coefficients of $S_{w^n}(\mu_1)$ is given by

$$\boldsymbol{s}_n(\mu_1) = (\mathbf{Z}^{\mathcal{I}})^{-1} \mathbf{B}^{\mathcal{I}}(\mu_1) \boldsymbol{w}^n.$$

In view of the separation of $b_1(\mu_1; \cdot, \cdot)$ w.r.t. the parameter μ_1 we have the decomposition $\mathbf{B}^{\mathcal{I}}(\mu_1) = \sum_{q=1}^{Q_b} \theta_q^b(\mu_1) \, \mathbf{B}_q^{\mathcal{I}}$ with parameter-independent matrices $\mathbf{B}_q^{\mathcal{I}}$. Hence, we obtain the representation $\boldsymbol{s}_n(\mu_1) = \sum_{q=1}^{Q_b} \theta_q^b(\mu_1) (\mathbf{Z})^{-1} \mathbf{B}_q^{\mathcal{I}} \boldsymbol{w}^n$ and the terms $\boldsymbol{z}_q^n := (\mathbf{Z}^{\mathcal{I}})^{-1} \mathbf{B}_q^{\mathcal{I}} \boldsymbol{w}^n$ can be computed

offline (as they are parameter-independent). Since the μ_1-dependent supremizers can be build by linear combinations (with μ_1-dependent coefficients) of the functions $z_q^n \in \mathsf{Z}^{\mathcal{I}}$ corresponding to the coefficient vectors $\boldsymbol{z}_q^n$, $1 \leq n \leq N_1$, $1 \leq q \leq Q_b$, we choose for every $\mu_1 \in \mathcal{D}_1$

$$\mathsf{Z}_{N_1}(\mu_1) := \operatorname{span}\{S_{w^1}(\mu_1), \ldots, S_{w^{N_1}}(\mu_1)\}$$

as reduced test space, where $S_{w^n}(\mu_1) = \sum_{q=1}^{Q_b} \theta_q^b(\mu_1)\, z_q^n$. Of course, instead of explicitly constructing the stabilising test space we can reformulate the (sub-)problem into the generalised Gauss normal equation and solve it accordingly.

3.2.2. Best N-term Approximation Approach

So far we have chosen offline the (RB) approximation space $\mathcal{D}_0^{\mathcal{L}}$ of the initial value. Unfortunately, the initial value part of the error will get dominant in situations where the functions in the parameter space cannot be approximated sufficiently well in the finite dimensional space $\mathcal{D}_0^{\mathcal{L}}$. In the next section we generalise the presented procedure for the offline phase such that for every parameter function μ_0 its (quasi-)best N-term approximation can be chosen online. For this, we consider wavelet Riesz bases for $\mathcal{D}_0 \subset H$. Every parameter function $\mu_0 \in \mathcal{D}_0$ then has an in general infinite wavelet expansion. The advantage is that the absolute values of the wavelet coefficients $|d_{j,k}|$ of μ_0 may have an exponential decay regarding their level j. Using a truncated wavelet basis as training set in the Evolution Greedy, we can hope for a small RB approximation error for sparse representations of μ_0 that coincides with small RB initial value bases online. Further, the norm equivalences associated with the Riesz bases ensure that the approximation error in the initial value is online computable with very reasonable effort. However, the method has increased memory requirements in comparison to the Two-Step Greedy method with a fixed approximation space for the initial value.

3.2.2.1. Short Introduction to Wavelets

We follow [Dah03] and [Urb09] for this very short overview of the wavelet features that are the most important in this context.

Definition 3.2.6. [Urb09, Def. 5.3] A countable collection of elements $\mathbf{h} := \{\, h_\lambda \,:\, \lambda \in \mathcal{I} \,\}$, $\mathcal{I} \subset \mathbb{Z}$, of a separable Hilbert space H is called a Riesz basis of H if each element $v \in H$ has an expansion $v = \sum_{\lambda\in\mathcal{I}} v_\lambda h_\lambda$ in terms of $\mathbf{h}$, and there exist constants $0 < c_{\mathbf{h}} \leq C_{\mathbf{h}} < \infty$ such that

$$c_{\mathbf{h}} \sum_{\lambda\in\mathcal{I}} |v_\lambda|^2 \leq \left\| \sum_{\lambda\in\mathcal{I}} v_\lambda h_\lambda \right\|_H^2 \leq C_{\mathbf{h}} \sum_{\lambda\in\mathcal{I}} |v_\lambda|^2.$$

For every element the sequence of coefficients is uniquely determined. A Riesz basis generalises the concept of an orthogonal basis.

For a construction of a wavelet basis, we introduce the concept of a Multiresolution Analysis, i.e., nested spaces equipped with certain properties. For any function $g : \mathbb{R} \to \mathbb{R}$, we define

$$g_{[j,k]}(x) := 2^{j/2} g(2^j x - k), \quad x \in \mathbb{R},$$

for $j \in \mathbb{N}_0$ and $k \in \mathbb{Z}$.

Definition 3.2.7. [Urb09, Def. 2.3] A sequence $(S_j)_{j \geq j_0}$ of spaces $S_j \subset L_2(\mathbb{R})$ is called Multiresolution Analysis (MRA), if for $j \geq j_0$

(a) the spaces are nested, i.e. $S_j \subset S_{j+1}$;

(b) their union is dense in $L_2(\mathbb{R})$, i.e. $\overline{\cup_{j \geq j_0} S_j} = L_2(\mathbb{R})$;

(c) their intersection is trivial, i.e. $\cap_{j \geq j_0} S_j = \{0\}$;

(d) there exists a function ϕ such that each set $\Phi_j := \{\phi_{[j,k]} : k \in \mathbb{Z}\}$ is a uniformly stable basis for S_j;

(e) the spaces arise by scaling: $f \in S_j \Leftrightarrow f(2\cdot) \in S_{j+1}$, $j \geq j_0$ (dilation);

(f) the spaces are shift-invariant, i.e., $f \in S_0 \Leftrightarrow f(\cdot - k) \in S_0$, $k \in \mathbb{Z}$.

The index j is referred to as the level. Every element in a high level space S_J can be decomposed into its part in S_{J-1} and into the part that lives in the complement of S_{J-1} in S_J. The basis elements of the complement spaces $W_{J-1} = S_J \ominus S_{J-1}$ are called wavelets on the corresponding level $J-1$. By iteration beginning on some high level J down to the coarsest level j_0 the space S_J decomposes to $S_{j_0} \bigoplus_{j=j_0}^{J-1} W_j$, its multiscale representation [Urb09, (5.3)]. This decomposition is only stable if the complement spaces W_j are uniformly stable in the sense that the angle between S_j and W_j is uniformly bounded. The orthogonal complement of S_j in S_{j+1} would be the optimal choice under this criteria but is at the same time restrictive for a wavelet construction: the only compactly supported scaling functions are those of the family of Daubechies scaling functions, [Urb09, Sec. 5.3].

For a general construction of wavelets we refer to [Urb09, Sec. 5.2 and 5.3].

Since the union of all spaces S_j given in Definition 3.2.7 is dense in $L_2(\mathbb{R})$, cf. Definition 3.2.7 (b), the set of all wavelets $\{\psi_{[j,k]} : j \geq j_0, k \in \mathbb{Z}\}$ forms an (orthonormal) basis of $L_2(\mathbb{R})$.

Definition 3.2.8. [Urb09, p. 140] The multiscale representation of a function $f_J \in S_J$ is its decomposition into the part in S_{j_0} and all the details in the orthogonal complements: $f_J = c_0 \phi_{[j_0,0]} + \sum_{j=j_0}^{J-1} \sum_{k \in \mathcal{J}_j \subset \mathbb{Z}} d_{j,k} \psi_{[j,k]}$.

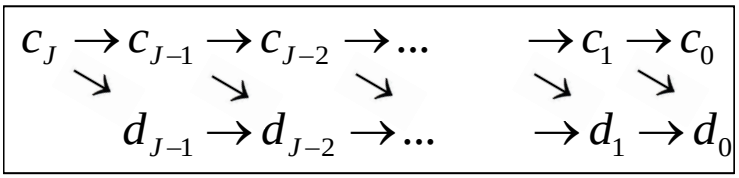


Figure 3.2.1.: Fast wavelet transformation: decomposition of coefficient vector $\mathbf{c}_J$ into its detail parts and one scale part.

Remark 3.2.9. Without loss of generality we assume in the following that the lowest level is given by $j_0 = 0$.

Every element $f \in S_J$ has now two equivalent representations:

(1) $f = \sum_{k\in\mathcal{J}_J} c_{J,k}\phi_{[J,k]}$

(2) $f = \sum_{j=-1}^{J-1}\sum_{k\in\mathcal{J}_j} d_{j,k}\psi_{[j,k]}$.

Here, $\mathcal{J}_{-1} := \{0\}$ and $\psi_{[-1,0]} := \phi_{[0,0]}$. Given expansion (1), the expansion in terms of wavelets can be deduced using the fast wavelet transformation (FWT): Let $\mathbf{c}_j = (c_{j,k})_{k\in\mathcal{J}_j}$ be the coefficient vector on level j. We define the down-sampling operator [Urb09, (5.17)]

$$\downarrow\colon \ell(\mathbb{Z}) \to \ell(\mathbb{Z}), \quad (\downarrow \mathbf{c})_k := c_{2k},\ k \in \mathbb{Z}.$$

Application of the down-sampling operator on the coefficient vector $\mathbf{c}_j$ of level j we obtain the associated coefficient vectors $\mathbf{c}_{j-1}$ and $\mathbf{d}_{j-1}$ of level $j-1$ such that

$$f = \sum_{k\in\mathcal{J}_j} c_{j,k}\phi_{[j,k]} = \sum_{k\in\mathcal{J}_{j-1}} d_{j-1,k}\psi_{[j-1,k]} + \sum_{k\in\mathcal{J}_{j-1}} c_{j-1,k}\phi_{[j-1,k]}.$$

The mapping $T_{j-1,j} : c_j \mapsto (c_{j-1}, d_{j-1})$ is called the wavelet transform, cf. [Urb09, p. 143]. The procedure is visualised in Figure 3.2.1. A reconstruction can be achieved by using the so-called inverse wavelet transform. For details we refer to [Urb09, 5.4.1].

Example 3.2.10. (Haar Basis.) [Urb09, Sec. 5.12] The Haar basis is an orthonormal wavelet basis for $L_2(0,1)$. The construction uses the characteristic function on $[0,1)$, $\phi^{\text{Haar}} := \chi_{[0,1)}$. The Haar function system corresponding to ϕ^{Haar} is the simplest example of an MRA. The so called *mother* wavelet is given by

$$\psi(x) = \begin{cases} 1 & 0 \le x < 0.5 \\ -1 & 0.5 \le x < 1. \end{cases}$$

The set $\boldsymbol{\psi}_j^{\text{Haar}} := \{\psi_{[j,k]} : k \in \mathcal{J}_j \subset \mathbb{Z}\}$ is an orthonormal basis for W_j. The wavelet basis functions for the first three levels $j = 0, 1, 2$ are shown in Figure 3.2.2.

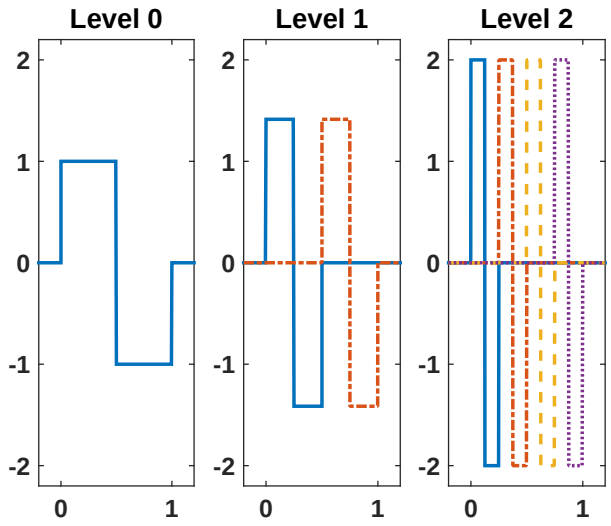


Figure 3.2.2.: The Haar basis elements for level 0, 1, and 2.

For the application in the space-time RBM for parameter functions we are interested in wavelets having compact support that behaves like $2^{-|j|}$ that build Riesz bases. The norm of a function given in its wavelet expansion is equivalently given by the $\ell_2(\mathcal{J})$-norm of its wavelet coefficients. Here, $\ell_2(\mathcal{J})$ denotes the space of all sequences $\mathbf{c} = (c_\lambda)_{\lambda\in\mathcal{J}} \subset H$ which satisfy $|\mathbf{c}|^2_{\ell_2(\mathcal{J})} := \sum_{\lambda\in\mathcal{J}} |c_\lambda|^2 < \infty$. The Jackson inequality below shows that the decay of the wavelet coefficients is related to the approximation level and the regularity of the considered function:

$$\inf_{v_j\in S_j} |v - v_j|_{L_2(\Omega)} \lesssim 2^{-js}|v|_{H^s(\Omega)}, \quad v \in H^s(\Omega),\ s \le \gamma,$$

where γ denotes the highest possible order of approximation, [Urb09, (5.30)]. The $\lesssim$ stands for $\le$ up to a multiplicative constant.

Let v_J be the wavelet expansion of v up to level J. Then,

$$|v - v_J|^2_{L_2(\Omega)} \lesssim C \sum_{j>J} \sum_{k\in\mathcal{J}_j} |d_{j,k}|^2.$$

Consequently, when the parameter functions are approximated by a truncated wavelet expansion, the approximation error can be computed very efficiently online in terms of the additional wavelet expansion coefficients.

3.2.2.2. Offline Phase - Initial Value Approximation with Wavelets

We assume a given wavelet Riesz basis of our parameter function space H and an underlying finite element basis $\{\phi_1, \dots, \phi_{\mathcal{J}}\}$ of $V^{\mathcal{J}}$ defined on a triangulation of step size $\frac{1}{2^\ell}$ in every spatial direction. By construction we expect that the coefficients on higher levels play a minor role in the function representation. Thus, we focus on all wavelet basis functions up to level ℓ. We impose them as initial value of the PDE and precompute the associated (initial value) snapshots $q^{j,k} \in V^{\mathcal{J}}$ that satisfy:

$$(q^{j,k}, \phi_i)_H = (\psi_{[j,k]}, \phi_i)_H \quad \forall\, i = 1, \dots, \mathcal{J}, \forall\, j = -1, \dots, \ell, k = 0, \dots, 2^j - 1.$$

In contrary to the previously described Two-Step Greedy method we will not take all the $2^{\ell+1}$ snapshots into account in the online phase. Nevertheless, we store all $2^{\ell+1}$ initial values in an initial value library

$$\mathbb{L}_{\text{init}} = \{q^{j,k} \ : \ j = -1, \dots, \ell, k = 0, \dots, 2^j - 1\}.$$

3.2.2.3. Offline Phase - Evolution Greedy for every wavelet basis function

For the evolutionary part, we perform an Evolution Greedy with training set $\{\psi_{[j,k]}\} \times \mathcal{D}_1^{\text{train}}$ for every wavelet up to the prefixed level ℓ. In contrast to Section 3.2.1.2, we do not compute only one reduced basis of $W^{\mathcal{I}}$ for one very large training set in $\mathcal{D}$. At the end we have $2^{\ell+1}$ RB spaces $W_{N_1^{j,k}} = \text{span}\{w_{j,k}^1, \dots, w_{j,k}^{N_1^{j,k}}\}$ of dimension $N_1^{j,k}$ that we collect in an Evolution Library

$$\mathbb{L}_{\text{evol}} = \{\{w_{j,k}^1, \dots, w_{j,k}^{N_1^{j,k}}\} \ : \ j = -1, \dots, \ell, k = 0, \dots, 2^j - 1\}.$$

We precompute all components that are required online for the quasi basis $\mathbb{L}_{\text{evol}}$.

3.2.2.4. Online Phase with Online Orthonormalisation

The online phase is adapted for the given parameter function μ_0. We consider its wavelet expansion and focus on the N wavelets with the largest coefficients (up to truncation of the level according to the finite element discretisation). This corresponds to a (quasi-)best N-term approximation:

$$H_N = \inf_{\substack{U \subset H, \\ \dim(U) = N}} \left(\inf_{\tilde{\mu}_0 \in U} \|\mu_0 - \tilde{\mu}_0\|_{L_2(\Omega)} \right).$$

Let j_i and k_i for $i = 1, \dots, N$ be the indices of the N wavelets. The initial value part can now directly be computed by

$$u_N^{\text{init}} = \sum_{i=1}^{N} d_{j_i,k_i} \sigma^0 \otimes q^{j_i,k_i}.$$

The evolutionary part is computed by taking the set $\{w_{j_i,k_i}^{\ell} : i = 1, \dots, N, \ell = 1, \dots, N_1^{j_i k_i}\}$ as the new approximation space.

We now choose the corresponding parts of the precomputed quantities and get an $N_1(N) \times N_1(N)$ linear equation system where $N_1(N) = \sum_{i=1}^{N} N_1^{j_i,k_i}$ depends again on the number of terms used for the approximation.

In a standard reduced basis offline phase, the greedy procedure itself ensures that linear dependent snapshots do not occure: After the parameter that maximises the greedy error is identified, the corresponding snapshot will be added to the reduced basis. Thus, all PDE solutions that are linear multiples of that snapshot are taken into account in the next step and their greedy error equals zero.

In the described RB setting we choose the RB basis books online out of our library $\mathbb{L}_{\text{evol}}$. It is possible that the new basis composition contains linear dependent snapshots as they have been sampled in different greedy procedures. We address the problem by performing an online orthonormalisation so that we can identify and delete linear dependent snapshots.

Orthonormalisation of the RB basis is equivalent to precondition the online system matrix $\mathbf{B}_{N_1(N)}(\mu)$ with a preconditioner $\mathbf{S}$ that is given by $\mathbf{S}^T[(w^i, w^j)_{\mathbb{X}}]_{i,j=1,\dots,N_1(N)}\mathbf{S} = \mathbb{1}$.

Proposition 3.2.11. The preconditioner that is equivalent to a basis orthonormalisation is explicitly given by $\mathbf{S} := \mathbf{U}^{-1}\mathbf{D}^{-1/2}$. The matrices $\mathbf{D}$ and $\mathbf{U}$ are given by a Singular Value Decomposition (SVD) of the Gramian $[(w^i, w^j)_{\mathbb{X}}]_{i,j=1,\dots,N_1(N)} = U^T D U$.

Proof.

$$\begin{aligned}(\mathbf{U}^{-1}\mathbf{D}^{-1/2})^T[(w^i, w^j)_{\mathbb{X}}]_{i,j=1,\dots,N_1(N)}\mathbf{U}^{-1}\mathbf{D}^{-1/2} &= (\mathbf{U}^{-1}\mathbf{D}^{-1/2})^T\mathbf{U}^T\mathbf{D}\mathbf{U}\mathbf{U}^{-1}\mathbf{D}^{-1/2} \\ &= \mathbf{D}^{-T/2}\mathbf{U}^{-T}\mathbf{U}^T\mathbf{D}\mathbf{U}\mathbf{U}^{-1}\mathbf{D}^{-1/2} = \mathbb{1}.\end{aligned}$$

□

As the preconditioner $\mathbf{S}$ depends on the chosen best N-term approximation again ($\mathbf{S} = \mathbf{S}(N) =: \mathbf{S}_N$), we are only able to precompute the Gramian matrix for the full library. Online, we get the Gramian $[(w^i, w^j)_{\mathbb{X}}]_{i,j=1,\dots,N_1(N)}$ by just choosing the corresponding precomputed parts. However, we have to perform the singular value decomposition online and apply the constructed preconditioner to the operator $B_{N_1(N)}(\mu)$. Even though all computations are performed in terms

of the RB-basis dimension $N_1(N)$ this increases the computational effort online. The new online phase is detailed in Algorithm 3.2.3.

Algorithm 3.2.3 Online Phase with Online Orthonormalisation

Input: $N_1 := N_1(N)$, $\mathbb{L}_{\text{init}}$, $\mathbb{L}_{\text{evol}}$, new parameter $\mu \in \mathcal{D}$.
Output: RB solution $u_{N_1}(\mu)$, estimator $\Delta_{N_1}(\mu)$

1: Compute $\mathbf{B}_{N_1}(\mu)$, $\mathbf{F}_{N_1}(\mu)$
2: Perform SVD $\rightarrow \mathbf{S}_{N_1}$
3: Orthogonalisation: $\tilde{\mathbf{B}}_{N_1}(\mu) = \mathbf{S}_{N_1}^T \mathbf{B}_{N_1}(\mu)\mathbf{S}_{N_1}$, $\tilde{\mathbf{F}}_{N_1}(\mu) = \mathbf{S}_{N_1}^T \mathbf{F}_{N_1}(\mu)$
4: **if** $\det(\tilde{\mathbf{B}}_{N_1}(\mu)) = 0$ **then** Delete zero rows/columns **end if**
5: Solve $\tilde{u}_{N_1}(\mu) = (\tilde{\mathbf{B}}_{N_1}(\mu))^{-1}\tilde{\mathbf{F}}_{N_1}(\mu)$
6: Get $u_{N_1}(\mu) = \mathbf{S}_{N_1}\tilde{u}_{N_1}(\mu)$
7: Compute $\Delta_{N_1}(\mu)$

3.2.2.5. Online RB Approximation Error

As for the fixed approximation spaces we can use the separation of the standard error estimator to receive an offline-online efficient error bound. The approximation error in the initial value is just the sum over all remaining wavelet coefficients associated to the functions that are not included in the approximation space. As we can now choose the best N-term approximation for every function, the approximation error of the initial value can be significantly smaller in comparison to the one in Section 3.2.1. The evolutionary part can – as before – be bounded in terms of the right-hand side and the afore used (Evolution) Greedy tolerance, cf. Proposition 3.2.5. Again, the right-hand side has an influence on the quality of the approximation.

3.2.2.6. Extensions

As soon as we solve a problem with source term equal to zero, it is possible that the PDE solution is given in terms of the RB expansion. If the right-hand side is different to zero the RB approximation quality is restricted by the fact that we only precompute snapshots solving the inhomogeneous PDE. Performing the above described basis generation of the evolutionary part for the homogeneous PDE and extending the library $\mathbb{L}_{\text{evol}}$ by these bases may improve the performance of the online phase. Even though this might be a solution for some applications, the offline and online computational effort and the storage requirements double.

3.3. Generalisations

We point out a few things to be kept in mind when generalising the use of parameter functions in the reduced basis method. We consider an example to get an idea on how a parameter function in the bilinear form of the left-hand side influences the approximation.

Example 3.3.1. We are interested in the solution u of

$$\begin{aligned} -\mu_0 \Delta u &= g \quad \text{on } \Omega = (0,1) \\ u &= 0 \quad \text{on } \partial\Omega, \end{aligned}$$

for some positive definite function $\mu_0 \in \mathcal{D} \subset \mathcal{C}^\infty(\Omega)$. The weak formulation is straightforward and reads: Find $u \in V = H_0^1(\Omega)$ such that

$$a(u, v; \mu) := \int_\Omega \mu_0(x) \nabla u(x) \nabla v(x) dx = \int_\Omega g(x) v(x) dx =: f(v) \quad \forall v \in V.$$

We assume that the computation of $u^{\mathcal{N}}(\mu_0)$ is possible, i.e. that μ_0 is given e.g. in closed form. The standard RB error estimator for coercivity constant $\alpha > 0$ is given by

$$|u^{\mathcal{N}}(\mu_0) - u_N(\mu_0)|_V \leq \frac{1}{\alpha} |f(\cdot) - a(u_N(\mu_0), \cdot; \mu_0)|_{V'}.$$

For the parameter function μ_0 we denote by $\tilde{\mu}_0$ the approximation of μ_0 in terms of given functions $\delta^1, \ldots, \delta^{\mathcal{L}}$, e.g. wavelets. The residual is given by

$$\begin{aligned} r(v; \mu_0) &= f(v) - \int_\Omega \tilde{\mu}_0(x) \nabla u_N(x) \nabla v(x) dx \\ &= f(v) - \sum_{i=1}^{\mathcal{L}} \alpha_i(\mu_0) \int_\Omega \delta^i(x) \nabla u_N(x) \nabla v(x) dx. \end{aligned}$$

A standard greedy procedure can be performed for finitely many functions $\delta^1, \ldots, \delta^{\mathcal{L}}$, analogous to the presented Evolution Greedy procedure. The offline-online decomposition of the error estimator is straightforward using the standard affine decomposition. The influence of the function approximation error received by projecting μ_0 onto $\operatorname{span}\{\delta^1, \ldots, \delta^{\mathcal{L}}\}$ is not explicitly given.

So, for a parameter space $\mathcal{D}_0 \times \mathcal{D}_1 \subset F \times \mathbb{R}^d$, with function space F, given ansatz space X and test space Y and a well-posed problem formulation

$$\text{find } u \in \mathsf{X}: \quad b(u, v; \mu) = f(v) \quad \forall\, v \in \mathsf{Y}$$

we have at least to assume an affine decomposition w.r.t. the parameter in the sense that

$$b(u, v; \mu) = \sum_{q=1}^{Q_b} \theta_q(\mu_1) b_q(u, v) + b_{Q_b+1}(u, v; \mu_0).$$

For the RB simulation, we require the bilinear form to be linear in μ_0. In general, the treatment of parameter functions either in the bilinear form $b(\cdot, \cdot)$ or in the right-hand side depends highly on the actual problem formulation.

3.4. Numerical Examples

The numerical experiments test the different methods for parameter (functions) in the initial condition. The influence of the approximation error of the parameter function in the RB error (estimator) is investigated. We consider a 1D diffusion problem and keep the influences of the standard parameters as simple as possible. The numerical solution follows Section 2.2.2.2. We will precompute all necessary inf-sup constants such that a possible approximation of the constants with e.g. an SCM, cf. Remark 2.2.39 does not influence the results. In addition we evaluate the approximation error in the discrete natural norm presented in Equation (2.2.20), i.e., for $w \in \mathbb{X}^{\mathcal{N}} \subset \mathbb{X}$

$$\|w\|^2_{\mathbb{X},\mathrm{bar}} := \|\bar{w}\|^2_{L_2(I;V)} + \|\dot{w}\|^2_{L_2(I;V')} + \|w(T)\|^2_H,$$

for $\bar{w}^k := (\Delta t)^{-1} \int_{I^k} w(t)\, dt \in V$ and $\bar{w} := \sum_{k=1}^{\mathcal{K}} \tau_k \otimes \bar{w}^k \in L_2(I; V)$. In Proposition 2.2.32 was shown that the stability can be improved in the case of a symmetric bilinear form $a(\cdot, \cdot)$.

3.4.1. Model Problem

For the Hilbert spaces $V := H^1_0(0, 1)$ and $H := L_2(0, 1)$ we consider

$$\begin{aligned} \dot{u}(x, t) - \mu_1 \Delta u(x, t) &= g(x) \quad \text{for } (x, t) \in [0, 1] \times [0, 0.3] \\ u(x, 0) &= \mu_0. \end{aligned}$$

The parameter space is given by $\mathcal{D} = \mathcal{D}_0 \times \mathcal{D}_1$ where

$$\mathcal{D}_0 = L_2(0, 1), \quad \mathcal{D}_1 = [0.5, 1.5].$$

We take the Haar wavelets as basis of $L_2(\Omega)$ that we presented already in Example 3.2.10. The error in the initial value is measured in terms of the coefficients of the wavelets not taken into

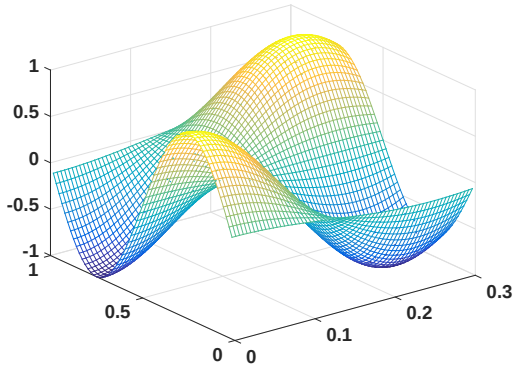


Figure 3.4.1.: The right-hand side g_{sin}.

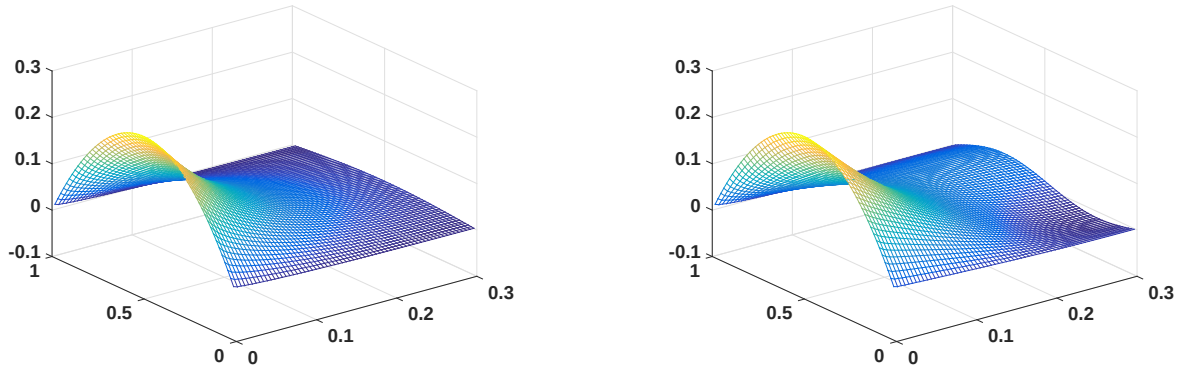
account in the approximation. In a numerical framework, we compute the sum over all wavelet coefficients up to truncation on some fixed high level. We impose the space-time discretisation that is equivalent to the Crank-Nicolson time stepping scheme, cf. Section 2.2.2.2. We perform the discretisation with step size $\Delta x = \Delta t = \frac{1}{2^6}$.

We have not yet specified the right-hand side. The a-priori bound deduced in Proposition 3.2.5 shows that our evolution approximation can be significantly better when the PDE has a right-hand side equal to zero. We therefore start with a comparison of two settings. The first is $g_{\text{zero}}(x) = 0$ and the second is a time-depending full right-hand side $g_{\text{sin}}(x,t) = \sin(2\pi x)\cos(4\pi t)$, cf. Figure 3.4.1. The greedy tolerance of the Evolution Greedy stays the same in all experiments, i.e. $\text{tol}_1 = 0.001$. The training set of the standard parameter space $\mathcal{D}_1$ is given by $\mathcal{D}_1^{\text{train}} = \{0.5 + k\frac{1}{17} \ : \ k = 0, \dots, 17\}$.

For online tests we test with two different parameters. The first is given by $\mu_{\text{smooth}} = (\mu_0, \mu_1) = (x(1-x), 1) \notin \mathcal{D}^{\text{train}}$. The function μ_0 is smooth and a sparse wavelet representation approximates the function well, cf. Figure 3.4.5 (a). The (FEM) solutions of the PDE for the respective right-hand sides are shown in Figure 3.4.2. The parameter function of the second parameter $\mu_{\text{L2}} = (\mu_0, \mu_1) = (|x - 0.5|^{1/2}, 1) \notin \mathcal{D}^{\text{train}}$ is in $L_2(\Omega)$ only, cf. Figure 3.4.3. The space-time variational formulation allows explicitly for an initial condition with less regularity than the PDE solution. The finite element solutions are displayed in Figure 3.4.4. Because of the lower regularity the wavelet expansion coefficients have a slow decay, cf. Figure 3.4.5 (b).

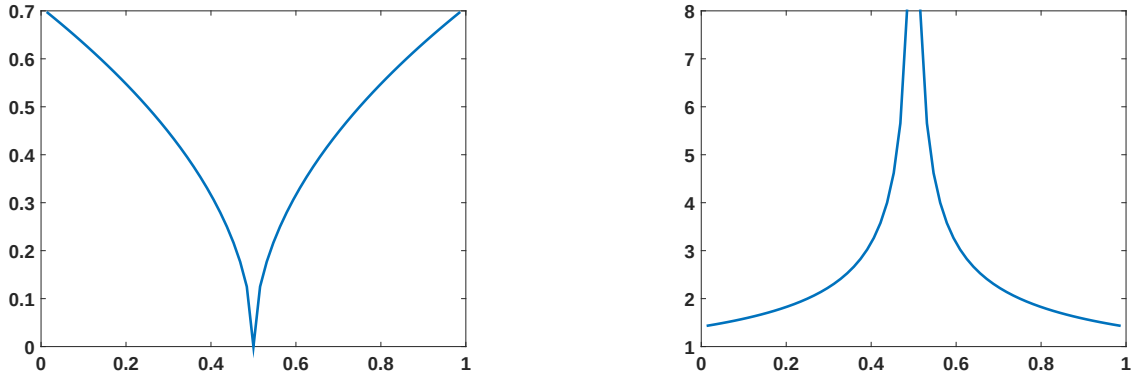
3.4.2. Fix Initial Value Approximation Online

We start with the Two-Step Greedy method. As we are using a wavelet basis we already know by its construction that - at least for smooth functions - lower level wavelets have a higher impact in the approximation procedure. This is not reflected by the basis functions themselves as we



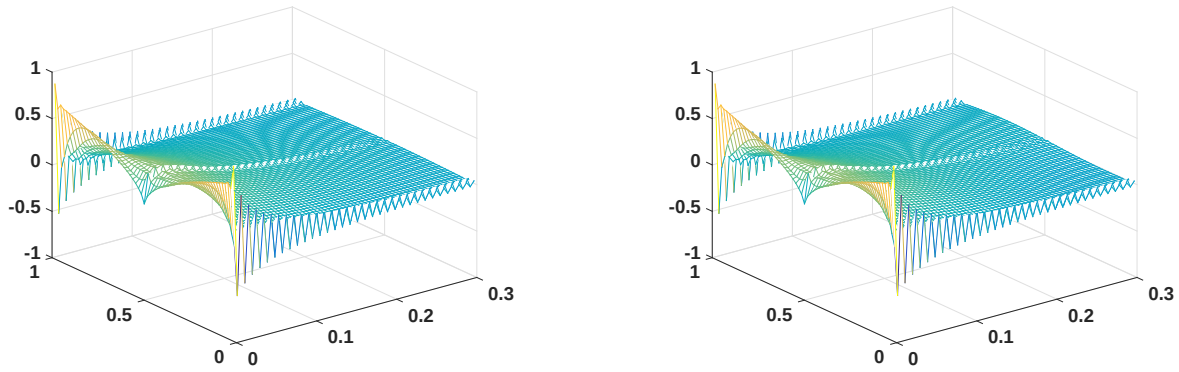
(a) $g = g_{\text{zero}}$ (b) $g = g_{\text{sin}}$

Figure 3.4.2.: Detailed solution for μ_{smooth}.



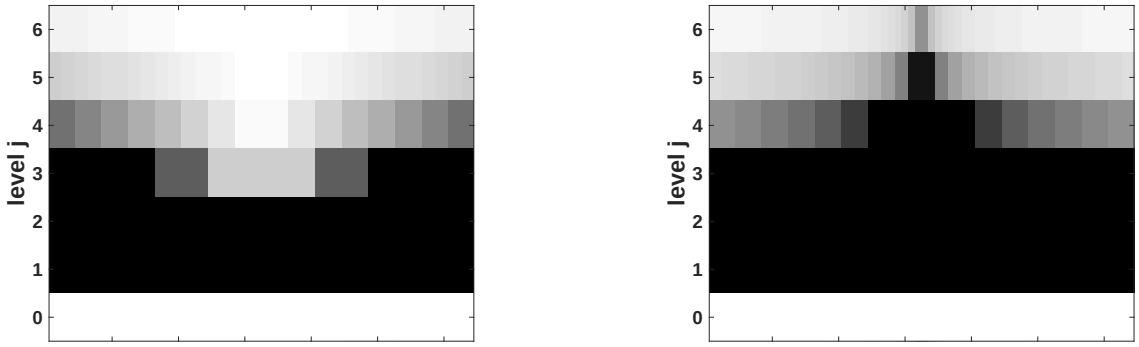
(a) (b)

Figure 3.4.3.: The parameter function of μ_{L2} and its weak derivative.



(a) $g = g_{\text{zero}}$ (b) $g = g_{\text{sin}}$

Figure 3.4.4.: Detailed solution for μ_{L2}.



(a) μ_{smooth} (b) μ_{L2}

Figure 3.4.5.: Wavelet expansion coefficients decay. Dark colors correspond to high absolut coefficient values.

can see by performing the Initial Value Greedy introduced in Algorithm 3.2.1. The training set is given by $\mathcal{D}_0^{\text{train}} := \{\psi_{[j,k]} : j = -1, \dots, \ell, k = 0, \dots, 2^j - 1\}$ and the tolerance $\text{tol}_0 = 0.001$. As expected, performing an initial value greedy for wavelets does not lead to the desired result. The procedure detects the wavelets on high levels as they have (by construction) a very small support and only reflect very local detail information. The greedy error shown in Figure 3.4.6 (a) stagnates until the snapshots for the higher levels are collected in the basis as one can see in Figure 3.4.6 (b). We can summarise that

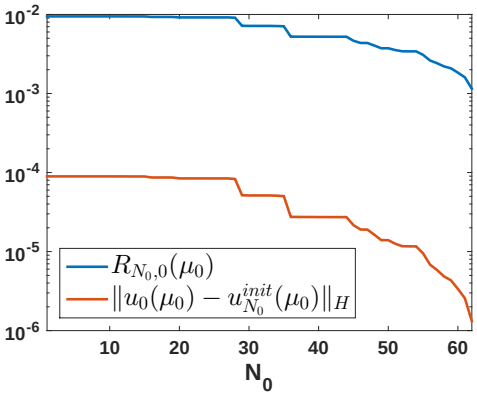
- the additional information of the expected size of the expansion coefficients is not used

and

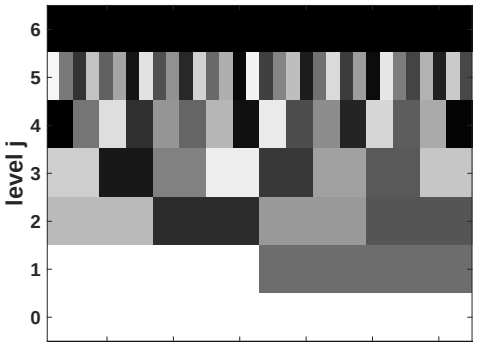
- the small support of the high level wavelets is responsible for locally very different initial values that correspond to very different PDE solutions.

Even including a scaling of the high level wavelets e.g. with 2^{-j} on level j does not change the order in which the wavelets are chosen.

In fact, the first step can be reduced to fixing an N_0 that is convenient for the application as there exist additional information on the basis construction itself. The first N_0 wavelets are now taken for the approximation in the initial value and the reduced basis $\{q^1, \dots, q^{N_0}\}$ is stored for $N_0 \in \{2^2, 2^3, 2^4, 2^5, 2^6\}$. Starting with $g \equiv g_{\text{zero}}$ we perform the Evolution Greedy five times: for $i = 2, \dots, 6$ we set $N_0 = 2^i$ and consider the training set $\mathcal{D}^{\text{train}} = \{\psi_{[j,k]} : j = 0, \dots, i; k = 0, \dots, 2^j - 1\} \times \mathcal{D}_1^{\text{train}}$. The greedy error decay for all $i = 2, \dots, 6$ behaves very similar for g_{zero} and g_{sin}. We show the decay for $g = g_{\text{zero}}$ and $i = 2, \dots, 5$ in Figure 3.4.7 (a), (c), (e), (g). There are five parameters in the training set



(a) Error decay for all wavelets up to level 6 in the training set.

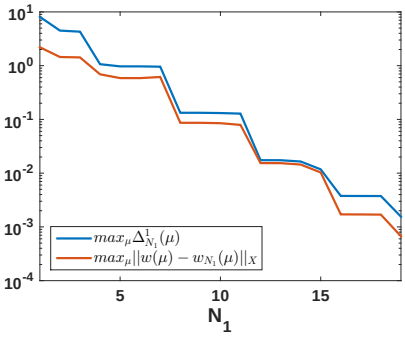


(b) Chosen sample set: darker colored wavelets are chosen first.

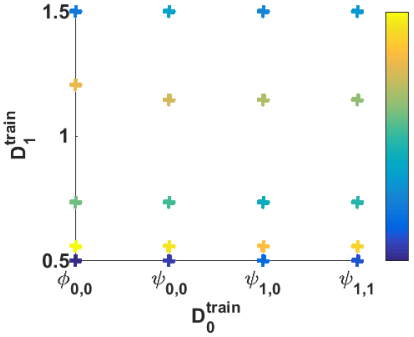
Figure 3.4.6.: Initial Value Greedy

$\mathcal{D}_1^{\text{train}}$ that have the most impact on the RB approximation of the evolution of the solution. As the wavelet basis functions are very different, the Greedy ensures that for all functions the respective five snapshots are in the RB space, cf. Figure 3.4.7 (b), (d), (f), (h). Online, for the new parameter $\mu_{\text{smooth}} = (x(1-x), 1)$ the error of the RB solution to the Crank-Nicolson detailed solution (Figure 3.4.8 (a)) depends on the considered wavelets. In fact, for $g = g_{\text{zero}}$, an approximation up to level four suffices for the PDE solution as one can see by the break down going over to the point were 2^5 wavelets are included in the training set and for the initial value approximation, Figure 3.4.8 (a). At this point, the error estimator $\Delta_N(\mu)$ overestimates the actual error because the value of the the wavelet approximation error part $\Delta_{N_0}^0 = \frac{1}{\beta^N}(\sum |d_\ell|^2)^{1/2}$ remains high in comparison to $\Delta_{N_1}^1$, cf. Figure 3.4.8 (b). The initial value approximation error is constantly added to the system even though the PDE system smooths the detail information from higher wavelet levels. A non-zero right-hand side has an impact on the approximation quality. For $g = g_{\text{sin}}$, it is not sufficient to consider only wavelets up to level four, cf. Figure 3.4.9 (a). The RB approximation error stays on the same level (> 0.01) for the reduced basis approximations corresponding to $i = 2, 3, 4$. The reduced basis – constructed for single wavelet functions as initial values – is not able to approximate the solution for the new initial value given in terms of a wavelet expansion, cf. Figure 3.4.9 (b). The influence of the right-hand side on the solution is obviously high.

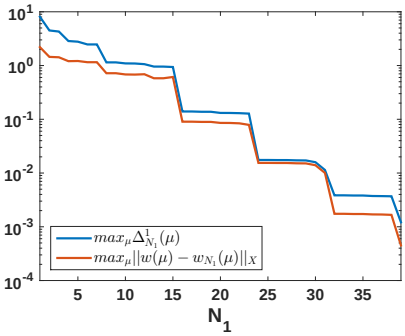
The parameter function of the parameter $\mu_{\text{L2}} = (|x - 0.5|^{1/2}, 1)$ is less smooth. Its wavelet expansion coefficients have a higher absolute value around the salient point 0.5. The low-level wavelets that we take into account first do not entirely reflect this behaviour, cf. Figure 3.4.10 (b). The PDE solution is influenced by the initial value but can still be approximated in case $g = g_{\text{zero}}$, Figure 3.4.10 (a). Here, the initial value approximation error is the dominating



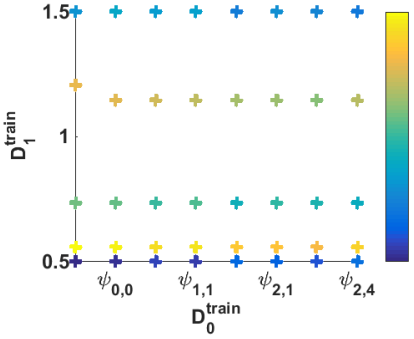
(a) Error and estimator $\Delta_{N_1,1}$ for $g = g_{\text{zero}}$ and $i = 2$.



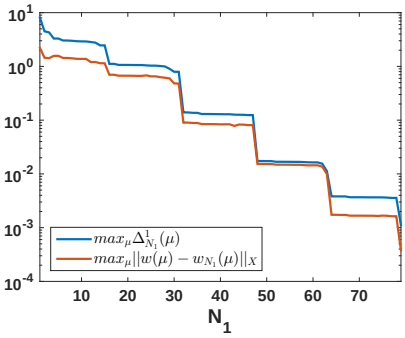
(b) Sample set for $g = g_{\text{zero}}$ and $i = 2$.



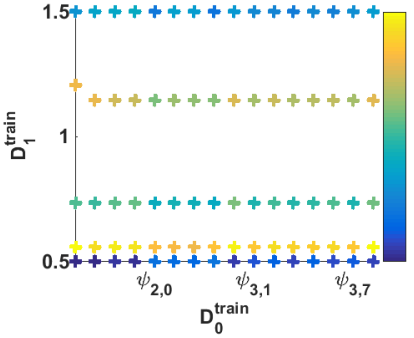
(c) Error and estimator $\Delta_{N_1,1}$ for $g = g_{\text{zero}}$ and $i = 3$.



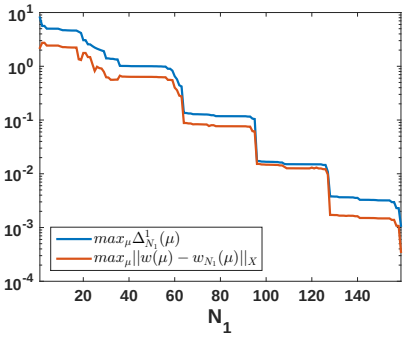
(d) Sample set for $g = g_{\text{zero}}$ and $i = 3$.



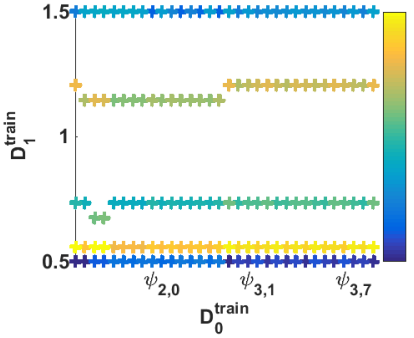
(e) Error and estimator $\Delta_{N_1,1}$ for $g = g_{\text{zero}}$ and $i = 4$.



(f) Sample set for $g = g_{\text{zero}}$ and $i = 4$.

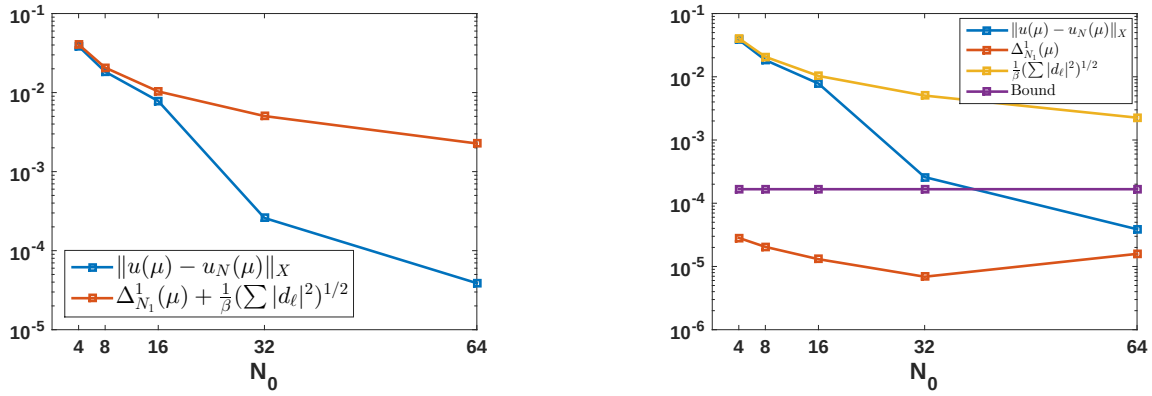


(g) Error and estimator $\Delta_{N_1,1}$ for $g = g_{\text{zero}}$ and $i = 5$.



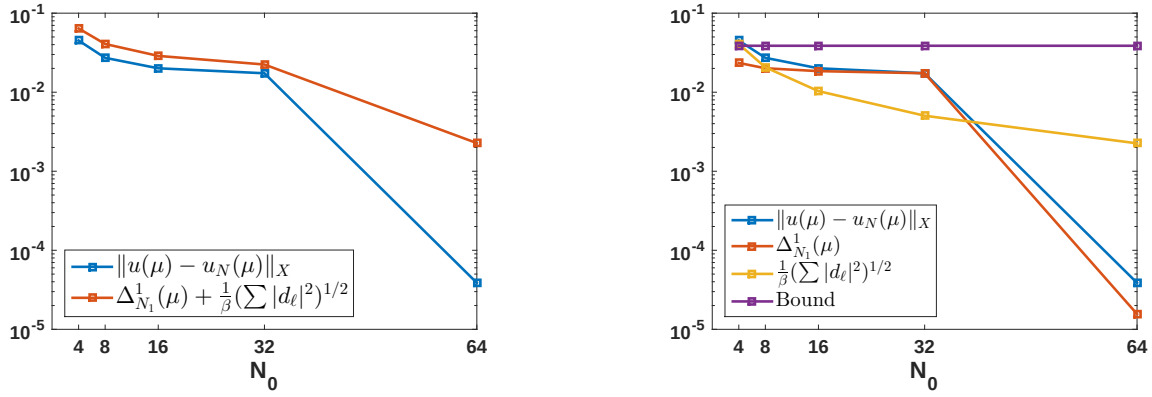
(h) Sample set for $g = g_{\text{zero}}$ and $i = 5$.

Figure 3.4.7.: Evolution Greedy performance for different $\mathcal{D}^{\text{train}}$ and $g = g_{\text{zero}}$.



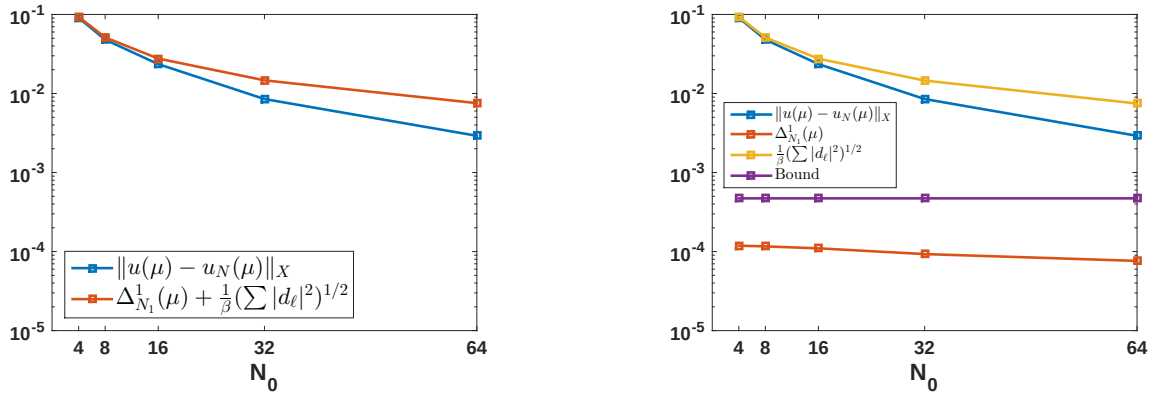
(a) (b)

Figure 3.4.8.: Error and estimator for the online procedure of μ_{smooth} and right-hand side g_{zero}.



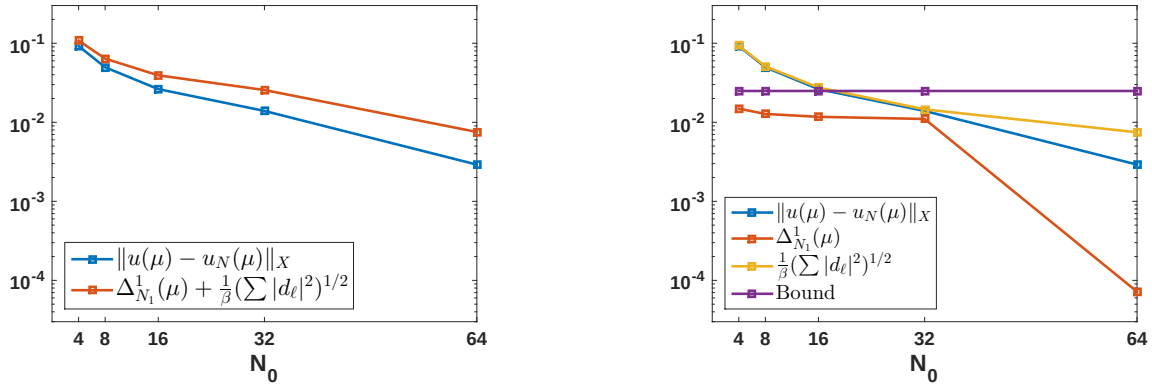
(a) (b)

Figure 3.4.9.: Error and estimator for the online procedure of μ_{smooth} and right-hand side g_{sin}.



(a) (b)

Figure 3.4.10.: Error and estimator for the online procedure of μ_{L2} and right-hand side g_{zero}.



(a) (b)

Figure 3.4.11.: Error and estimator for the online procedure of μ_{L2} and right-hand side g_{sin}.

part. For the non-zero right-hand side g_{sin} the RB approximation actually gets better as for the smooth initial value, cf. Figure 3.4.11. The RB basis seems to catch the solution behaviour much better for a non-smooth parameter. Consequently, the linear combination of snapshots corresponding to single wavelet functions reflect the solution behaviour much better if it includes very local changes due to the initial value. The dimensions of the two different reduced bases (for $g = g_{\mathrm{sin}}$ and $g = g_{\mathrm{zero}}$) for $i = 2, \ldots, 6$ are stated in Table 3.4.1. The greedy procedure stops with the same amount of basis functions for both right-hand sides. The high dimensional discretisation is of dimension $\mathcal{N} = 2^6 2^6 = 4096$. Performing the Crank Nicolson time stepping scheme we solve $2^6 = 64$ times a 64-dimensional linear equation system. Using all wavelets up to level four leads to an approximation that is about one order of magnitude higher than the original greedy tolerance solving one 32 dimensional and one 160 dimensional linear equation system. We reduce the discretised system of dimension $\mathcal{N} = 4096$ to an RB system of dimension

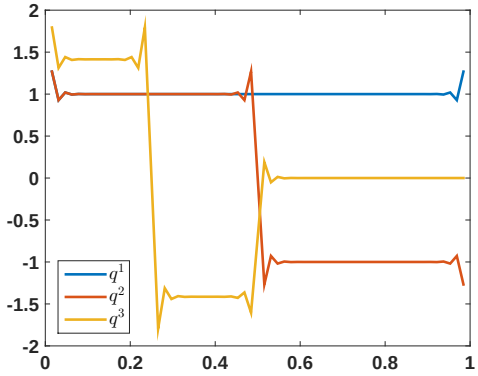


Figure 3.4.12.: The first three elements of $\mathbb{L}_{\text{init}}$.

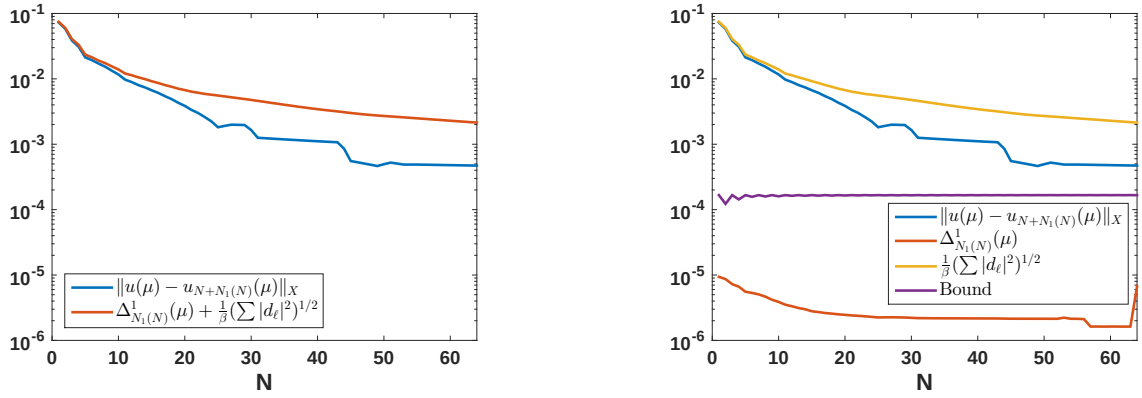
$N = 192$. Further, an involved right-hand side negatively affects the RB approximation quality.

i =	2	3	4	5	6
g_{zero}	4 + 20	8 + 40	16 + 80	32 + 160	64 + 282
g_{sin}	4 + 20	8 + 40	16 + 80	32 + 160	64 + 284

Table 3.4.1.: Reduced basis space dimension: $N = N_0 + N_1$.

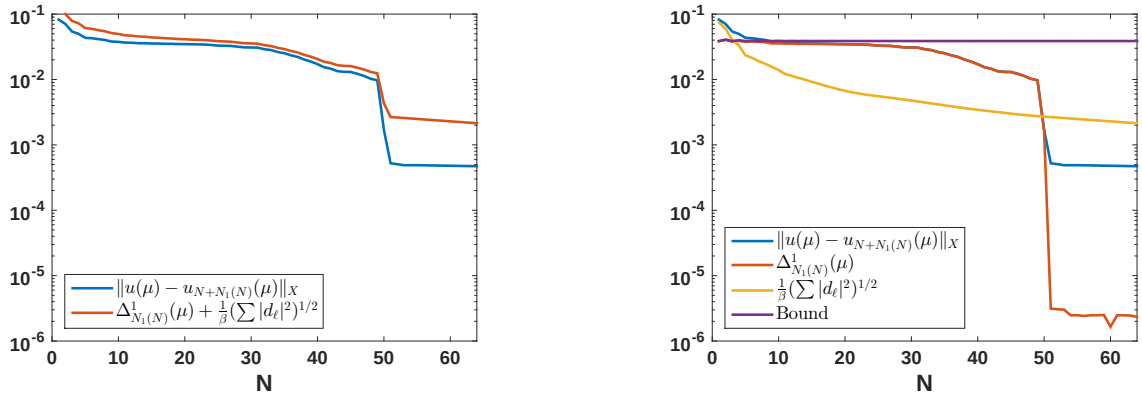
3.4.3. Best N-Term Approximation Online

The second approach is to use the best N-term approximation of the parameter function for the approximation in the initial condition. The snapshot library of the initial value contains 2^6 elements, whereby the first three are visualised in Figure 3.4.12. The Evolution Greedy is now performed 2^6 times with training sets $\mathcal{D}^{\text{train}} = \{\psi_{[j,k]}\} \times \mathcal{D}_1^{\text{train}}$ for $j = -1, \ldots, \ell$, $k = 0, \ldots, 2^j - 1$. For both right-hand sides g_{zero} and g_{sin} the Evolution Library contains 2^6 reduced bases of dimension four to five. The reduced basis approximation for the parameter with smooth initial value μ_{smooth} is significantly better for the problem formulation with right-hand side equal to zero, cf. Figure 3.4.13 and Figure 3.4.14. While the error and the estimator for g_{zero} and μ_{smooth} decrease, both stagnate for g_{sin} until 50 wavelets and their corresponding reduced basis (books) are taken into account. In both cases, the estimator behaves like the error but overestimates as soon as the approximation error of the initial condition is the dominating part. As before, the wavelets having a support that is smaller than the one of the underlying finite elements play a minor role for the PDE solution process. The RB approximation results of g_{zero} and g_{sin} do not essentially differ for the less smooth parameter function of μ_{L2}, cf. Figure 3.4.15 and Figure 3.4.16. Again, when g_{sin} is considered, the approximation improves significantly for values $N \geq 32$. Here, for



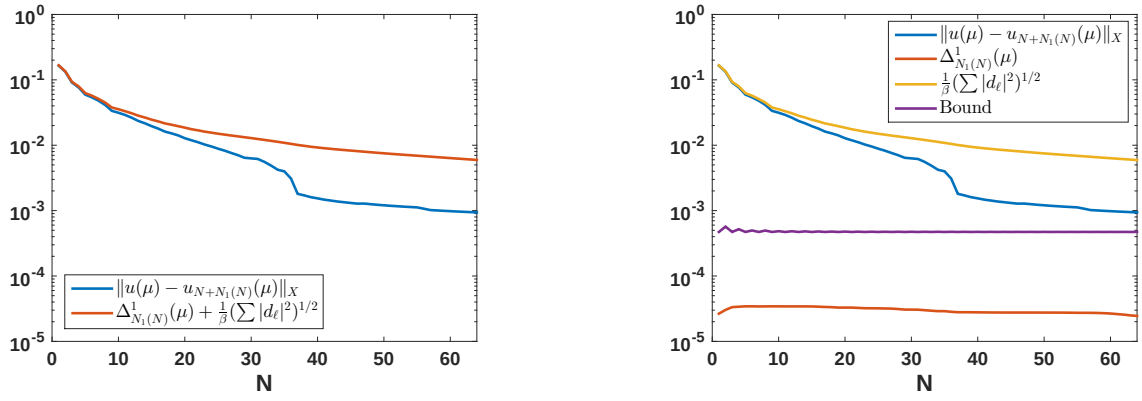
(a) (b)

Figure 3.4.13.: Error and estimator for the online procedure of $\mu = \mu_{\text{smooth}}$ and right-hand side g_{zero}.



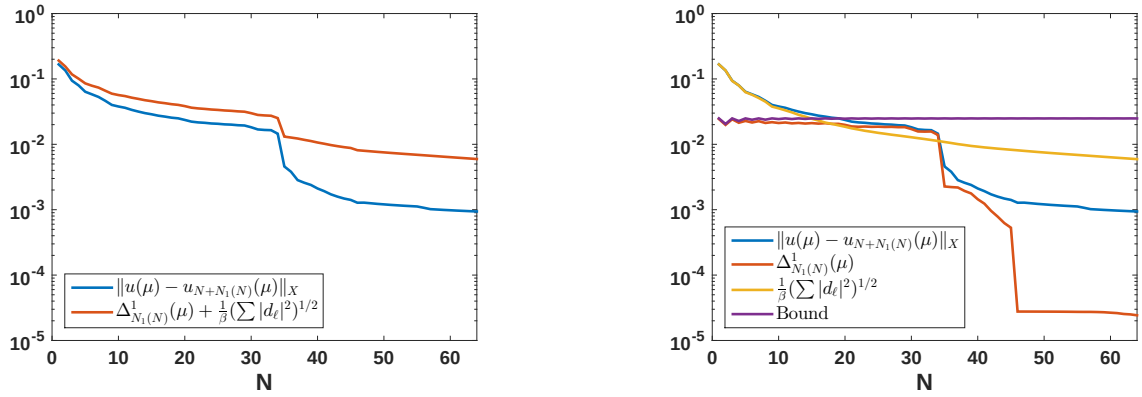
(a) (b)

Figure 3.4.14.: Error and estimator for the online procedure of $\mu = \mu_{\text{smooth}}$ and right-hand side g_{sin}.



(a) (b)

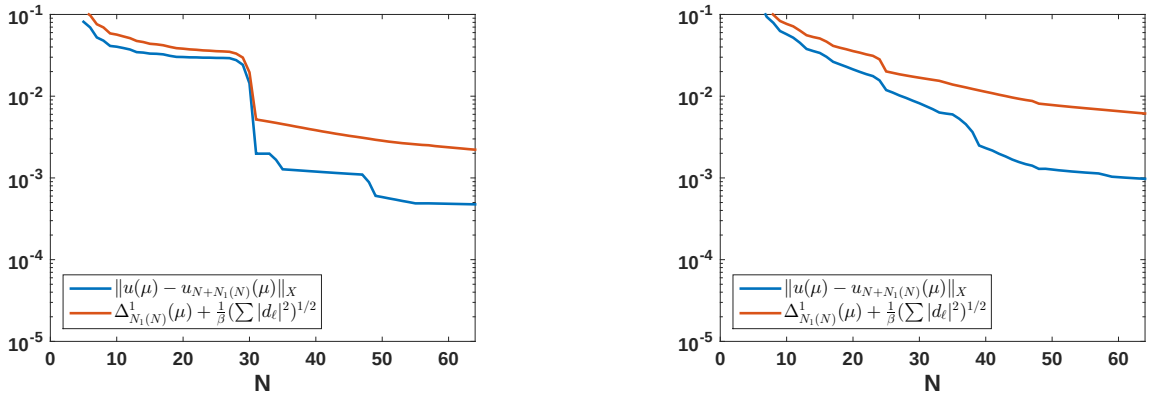
Figure 3.4.15.: Error and estimator for the online procedure of $\mu = \mu_{L2}$ and right-hand side g_{zero}.



(a) (b)

Figure 3.4.16.: Error and estimator for the online procedure of $\mu = \mu_{\text{L2}}$ and right-hand side g_{sin}.

the right-hand side equal to zero, the same but less pronounced improvement for $N \geq 32$ can be observed as the reduced bases books associated to higher level wavelets are more important for a good RB evolution approximation of the non-homogeneous problem. They are considered earlier for a non-smooth parameter function and the RB approximation gets better for smaller N. In contrast, the evolution approximation of the homogeneous problem is better for the smooth parameter function. Apparently, the wavelets that are necessary for a good approximation of the initial value and the reduced basis books that are relevant for a good approximation may not coincide. The given right-hand side $g = g_{\text{sin}}$ interacts with the input in form of the parameter function. In the last experiment we take additionally the extreme values of the parameter function into account by ensuring that the first wavelets in the approximation are the one on level 6 that cover the areas of the extrema $\max(\mu_0)$, $\min(\mu_0)$ and the zero with their support. After including these wavelets in the RB space, we continue adding the wavelets according to the best



(a) $g_{\sin}, \mu_{\text{smooth}}$ (b) $g_{\sin}, \mu_{\text{L2}}$

Figure 3.4.17.: Error and estimator for the online procedure taking extreme values and the best $N-5$-term approximation of the initial value into account.

N-term approximation as before. In particular, we include $\psi_{[6,1]}, \psi_{[6,30]}, \psi_{[6,31]}, \psi_{[6,63]}$ and $\psi_{[6,64]}$. As expected, the RB approximation of the solution for g_{zero} does not change as the effects of the wavelets on level six are smoothed out. The RB approximation for $g_{\sin}$ and μ_{smooth} improves significantly, cf. Figure 3.4.17 (a). The RB approximation quality has been satisfying without the extreme values for $g_{\sin}$ and μ_{L2}. It improves a little around the break $N = 30$, Figure 3.4.17 (b). The dimension of the reduced basis depends on the parameter function online. We have seen – performing the Initial Value Greedy – that a single wavelet as initial value results in a very specific PDE solution. Thus, the dimensions of the respective reduced basis books just differ by at most one or two. The number of basis functions grows linearly with N and does not depend on the scenario, as shown in Figure 3.4.18. Of course, depending on the scenario we receive different approximation qualities for the same RB size. For a better overview as well as a comparison to the previously presented method, we detail the dimensions for $N \in \{2^2, 2^3, 2^4, 2^5, 2^6\}$ in Table 3.4.2.

N =	4	8	16	32	64
$g_{\text{zero}}, \mu_{\text{smooth}}$	4 + 20	8 + 40	16 + 80	32 + 158	64 + 305
$g_{\sin}, \mu_{\text{smooth}}$	4 + 20	8 + 40	16 + 80	32 + 158	64 + 305
$g_{\text{zero}}, \mu_{\text{L2}}$	4 + 20	8 + 40	16 + 80	32 + 160	64 + 314
$g_{\sin}, \mu_{\text{L2}}$	4 + 20	8 + 40	16 + 80	32 + 160	64 + 314
$g_{\sin}, \mu_{\text{smooth}}$ + extrema	4 + 16	8 + 36	16 + 76	32 + 154	64 + 305
$g_{\sin}, \mu_{\text{smooth}}$ + extrema	4 + 16	8 + 36	16 + 76	32 + 156	64 + 312

Table 3.4.2.: Reduced basis space dimension: $N + N_1$.

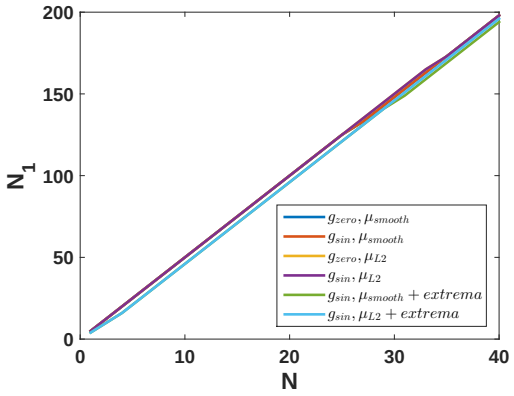


Figure 3.4.18.: Overview on RB dimensions in dependence on N.

3.4.4. Comparison and Outlook

Both procedures are decomposed into two steps. We are going to refer to the results of the first numerical experiments in Section 3.4.2 as the results for 'Fix-N' wavelets used for the parameter functions. In contrast, the second part uses the 'Best-N' wavelets in the initial value approximation. The amount of basis functions used in the approximation of the evolution stays the same: taking N offline chosen and fix wavelets or the best N-term approximation for the initial value leads to the same number N_1 of basis elements for the evolution, cf. Tables 3.4.1 and 3.4.2.

We are going to compare the error $\|u^{\mathcal{N}}(\mu) - u_N(\mu)\|_{\mathbb{X},\text{bar}}$ for $\mu = \mu_{\text{smooth}}$ and $\mu = \mu_{\text{L2}}$ that we receive in both methods, restricting to $N(= N_0) \in \{2^2, 2^3, 2^4, 2^5, 2^6\}$. For a zero right-hand side, the best N-term approximation improves the quality of the RB approximation for both test parameters at least up to $N = 16$, cf. Figure 3.4.19. For $N = 32$ there is a break for 'Fix-N' and the wavelets used on level 5 lead just to the reduced basis elements that catch perfectly the behaviour of the PDE solution for $\mu = \mu_{\text{smooth}}$. To cover such a behaviour, an adaptive reduced basis enrichment would be necessary online. The behaviour is more a happy coincidence than a rule. As we have seen, the 'Best-N' procedure is not working best for non-zero right-hand side w.r.t. the smooth parameter function. Given these results, it is not surprising that the 'Fix-N' procedure leads to better results, cf. Figure 3.4.20 (a). However, including the extremal values of the initial condition, the 'Best-N' results improve significantly for $N = 32$, Figure 3.4.20 (b). For μ_{L2}, both methods lead to the same approximation quality. Here, again the extreme values improve the approximation for $N = 32$. Note that the extreme values are taken first in the 'Best-N' procedure in Figure 3.4.20 (b), i.e. we only consider an $(N-5)$-term approximation for the initial value part instead of the best N-term approximation.

At the end it depends on the actual application for which procedure one decides. At least for

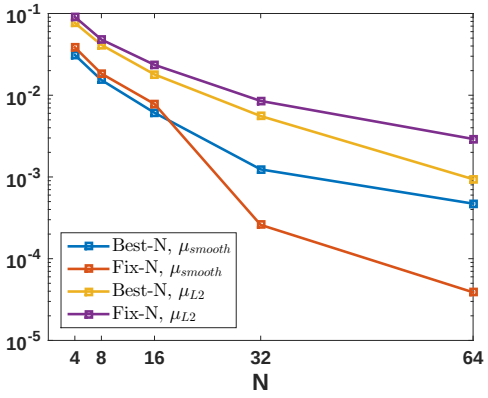
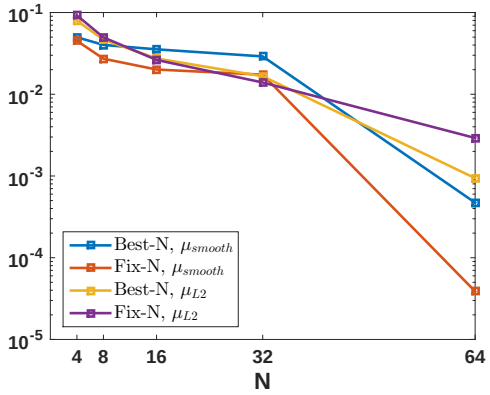
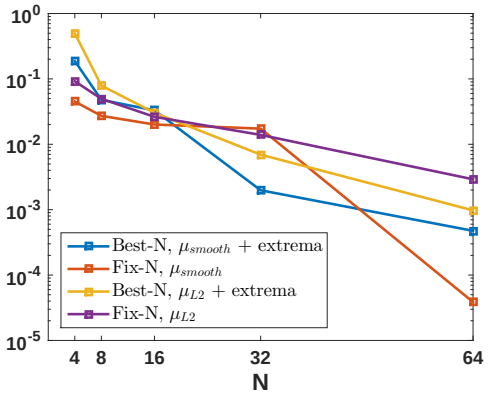


Figure 3.4.19.: Comparison of achieved online approximation errors for g_{zero} and $\mu \in \{\mu_{\text{smooth}}, \mu_{\text{L2}}\}$.



(a) No extremal values are taken into account.



(b) Including wavelets in the RB corresponding to the extremal values.

Figure 3.4.20.: Comparison of achieved online approximation errors for g_{zero} and $\mu \in \{\mu_{\text{smooth}}, \mu_{\text{L2}}\}$.

the use of a wavelet basis of the function parameter domain, the 'Best-N' procedure offers more flexibility regarding the size of the RB spaces. We do not enlarge level-wise but just as needed for the parameter function. Therefore, smaller bases can be used if the function parameter allows for. Further, for a right-hand side equal to zero, the 'Best-N' procedure allows for – in general – better approximation as the function approximation error is smaller. For a non-zero right-hand side we require the ad-hoc improvement of the basis by considering the wavelets corresponding to the extrem values. The disadvantage may be the required storage capacity to store the libraries needed. In addition the online approximation requires an additional computational effort for performing the singular value decomposition.

In our example we receive good results for $N_0 = 32$, i.e. we reduced the high-dimension $\mathcal{N} = 4096$ to about 190. The reduction in the initial value part might not be significant on the first glance for receiving satisfying results. However, we do not solve a linear equation system in that step but just perform the reconstruction using the corresponding wavelet coefficients. The reduction in the evolutionary part is – as expected – good: the reduction factor is about 12. The online computable error estimates in general follow the error behaviour.

4. H-Tucker Low Rank Tensor Format in the Reduced Basis Method

4.1. Introduction

Low rank tensor formats are storage formats that are based on a tensor product structure of the underlying space. The ansatz and test space in the space-time variational formulation provide this structure naturally separating space and time. The full tensor decomposition of the spatial domain is achieved by applying a tensorised basis in space too, cf. Example 2.2.31. For example, the solution of a time dependent 2D problem can be represented by a 3D tensor, with one time and two space dimensions. Low rank tensor formats efficiently use the tensor product structure to reduce the numerical effort in the solution process. The basic idea is to avoid the establishment of the full system and to perform all computations on the tensor components.

We give a short introduction to the tensor decomposition schemes most important for the present thesis following the introductions of T. G. Kolda and B. W. Bader [KB09], L. Grasedyck [Gra10] and of L. Grasedyck, D. Kressner and Ch. Tobler [GKT13]. For the introduction to the H-Tucker format that is the low rank format of choice we also follow the theses of A. Rupp [Rup14] and Ch. Tobler [Tob12]. We present the BPX-preconditioners for parabolic PDEs in space-time variational formulation, introduced by R. Andreev in [And12] and used in combination with the H-Tucker format in the paper of R. Andreev and Ch. Tobler [AT14] for solving parabolic problems. We investigate the application of the low-rank tensor format in the offline phase of the space-time reduced basis method. In general it reduces the numerical effort and allows to solve higher dimensional problems.

4.2. Preliminaries

4.2.1. CP Decomposition and Tucker Format

A d-order tensor is a multidimensional array $x \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ of size $n_1 \times \ldots \times n_d$. We denote the entries of x by $x_{i_1,\ldots,i_d}$, where $i_\mu \in \{1,\ldots,n_\mu\}$ for $1 \leq \mu \leq d$. For increasing d the number

of elements $N = \prod_{i=1}^{d} n_i$ increases exponentially. Low rank tensor formats are storage schemes that avoid this exponential storage effort.

We start by introducing the CANDECOMP/PARAFAC (CP) decomposition, cf. [KB09, Sec. 3]. The CP decomposition is based on factorising the entries of the tensor

$$x_{i_1,\dots,i_d} \approx \tilde{x}_{i_1}^{(1)} \dots \tilde{x}_{i_d}^{(d)}.$$

Collecting the factors of every entry in a corresponding vector $u^{(\mu)} = \begin{pmatrix} \tilde{x}_1^{(\mu)} & \cdots & \tilde{x}_{n_\mu}^{(\mu)} \end{pmatrix}^T \in \mathbb{R}^{n_\mu}$ we receive

$$\mathrm{vec}(x) \approx u^{(d)} \otimes \dots \otimes u^{(1)}.$$

Here, the vec-operator $\mathrm{vec} : \mathbb{R}^{n_1 \times \dots \times n_d} \to \mathbb{R}^{n_1 \cdots n_d}$ maps a tensor to a (column) vector in *reverse* lexicographical order. The order is chosen for convenience: MATLAB 's vec-operator uses this order. The tensor product denotes the standard Kronecker product. For a tensor x in CP format holds

$$\mathrm{vec}(x) = \sum_{i=1}^{r} u_i^{(d)} \otimes \dots \otimes u_i^{(1)}. \tag{4.2.1}$$

Definition 4.2.1. [KB09, Sec. 3.1] The tensor rank r is the minimum over all r such that x is exactly CP decomposable with r terms.

Example 4.2.2. The tensor $\begin{pmatrix} 3 & 6 & 9 \\ 4 & 8 & 12 \end{pmatrix} \in \mathbb{R}^{2\times 3}$ has the CP decomposition

$$\mathrm{vec}(x) = \begin{pmatrix} 3 & 4 & 6 & 8 & 9 & 12 \end{pmatrix}^T = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \otimes \begin{pmatrix} 3 \\ 4 \end{pmatrix}.$$

Its rank is equal to one.

The storage format is most efficient if the tensor rank is small as we have to store $(\sum_{i=1}^{d} n_i) \cdot r$ components. For detailed information, an historical background, computation techniques and applications we refer to [KB09, Sec. 3].

The Tucker decomposition introduced in [Tuc63], see also [Tuc66], follows the same principle, cf. [KB09, Sec. 4]. The tensor x is given by

$$\mathrm{vec}(x) = (u_d \otimes \dots \otimes u_1)\mathrm{vec}(c).$$

Here, $u_\mu \in \mathbb{R}^{n_\mu \times r_\mu}$ for all $1 \le \mu \le d$ are so called factor matrices. The core tensor is a multidimensional array $c \in \mathbb{R}^{r_1 \times \dots \times r_d}$. We call the vector $(r_1, \dots, r_d)$ the multilinear rank.

Definition 4.2.3. [Gra10, Def. 2.4] The Tucker rank is defined as the vector of (elementwise) minimal entries such that there exists (columnwise) orthonormal matrices u_i and a corresponding core tensor such that x is given in Tucker format.

The Tensor Train (TT) format and the H-Tucker format are further developments. We focus on the H-Tucker format as it was used for solving parabolic PDEs in space-time variational formulation before, [AT14]. Furthermore, the TT format can be considered as a special case of the H-Tucker format, [GH11, Sec. 3]. For a short introduction to the TT format we refer to [GKT13, Sec. 2.3]. For a comparison we also refer to [GH11].

We shortly explain the so-called tensor truncation that allows for using a low rank tensor format as a compression scheme. Even though we do not need the low rank tensor format as a model reduction method, the truncation to a tensor of bounded rank is required, e.g., in the solution process of a linear system in tensor format. Letters marked with an $\widehat{\cdot}$-symbol are missing in the corresponding representation, e.g. $(a_1, \dots, \widehat{a_k}, \dots, a_n)^T := (a_1, \dots, a_{k-1}, a_{k+1}, \dots, a_n)^T \in \mathbb{R}^{n-1}$. For $\mu \in \{1, \dots, d\}$ the μ-matricisation $x^{(\mu)}$ of x is an $n_\mu \times (n_1 \dots \widehat{n_\mu} \dots n_d)$-matrix given by the entries of x in rearranged order [GKT13, Sec. 2.2]

$$x^{(\mu)}_{i_\mu, \ell} = x_{i_1, \dots, i_d}, \quad \ell = 1 + \sum_{\nu<\mu} (i_\nu - 1) \prod_{\eta<\nu} n_\eta + \sum_{\nu>\mu} (i_\nu - 1) \prod_{\eta<\nu, \eta\neq\mu} n_\eta \tag{4.2.2}$$

In particular,

$$x^{(\mu)} = u_\mu c^{(\mu)} (u_d \otimes \dots \otimes \widehat{u_\mu} \otimes \dots \otimes u_1)^T, \quad c^{(\mu)} \in \mathbb{R}^{n_\mu \times n_1 \cdots \widehat{n_\mu} \cdots n_d}. \tag{4.2.3}$$

A more general matricisation is obtained by replacing μ by a set of indices $t \subset \{1, \dots, d\}$. The corresponding t-matricisation is the tensor $x^{(t)} \in \mathbb{R}^{n_{t_1} \cdots n_{t_{d-k}} \times n_{s_1} \cdots n_{s_k}}$ where $\{s_1, \dots, s_k\} = \{1, \dots, d\} \setminus t$. A higher order singular value decomposition (HOSVD) was introduced for compression, [DLDMV00]. Truncation of x to a tensor of a fixed tensor rank $\mathbf{m} := (m_1, \dots, m_d)$ is explained in the next definition.

Definition 4.2.4. [Gra10, Def. 2.5], [Rup14, Def. 5.2] Let $x \in \mathbb{R}^{\mathcal{I}}$, $\mathcal{I} \subset \mathbb{N}$ an index set. For $\mu = 1, \dots, d$ let

$$x^{(\mu)} = u_\mu \Sigma_\mu v_\mu^T, \quad u_\mu \in \mathbb{R}^{n_\mu \times n_\mu}$$

be the singular value decomposition of the μ-mode matricisation of the tensor x with diagonal matrix Σ_μ. The tensor x can be written in Tucker format

$$\mathrm{vec}(x) = (u_d \otimes \dots \otimes u_1) c$$

where

$$c = (u_d^T \otimes \dots \otimes u_1^T) \mathrm{vec}(x).$$

The truncation of x to a tensor of Tucker rank $\mathbf{m}$ is defined by

$$\begin{aligned}\mathrm{vec}(\mathcal{T}_{\mathbf{m}}(x)) &:= (\widetilde{u}_d^T \widetilde{u}_d \otimes \ldots \otimes \widetilde{u}_1^T \widetilde{u}_1)\mathrm{vec}(x) \\ &= (\widetilde{u}_d \otimes \ldots \otimes \widetilde{u}_1)c,\end{aligned}$$

where $\widetilde{u}_\mu$ is the matrix of the first m_μ columns of u_μ for all $1 \leq \mu \leq d$ and $c := (\widetilde{u}_d^T \otimes \ldots \otimes \widetilde{u}_1^T)\mathrm{vec}(x)$.

The approximation $\widetilde{x}$ is the quasi best approximation in the space of all tensors of multilinear rank $\mathbf{m}$: $T(\mathbf{m}) := \{x = (u_d \otimes \ldots \otimes u_1)\mathrm{vec}(c) \ : \ \mathrm{rank}(u_j) \leq m_j, j = 1, \ldots, d\}$ and

$$\|x - \widetilde{x}\|_2 \leq \sqrt{d} \min_{y \in T(\mathbf{m})} \|x - y\|_2.$$

4.2.2. H-Tucker Format

This section is based on the introductions given in [Rup14, Sec. 5] and [Gra10, Tob12]. The size of the core tensor in the Tucker decomposition becomes infeasibly large if a large Tensor rank $\mathbf{r} = (r_1, \ldots, r_d)$ is needed to approximate the tensor sufficiently well. The H-Tucker format is closely related to the Tucker format but further decomposes the core tensor c hierarchically. A hierarchical tree of tensors is the consequence. We consider an example for a better understanding.

Example 4.2.5. [Rup14, p. 74] Let $x \in \mathbb{R}^{n_1 \times \ldots \times n_4}$ be a 4-order tensor. Given a Tucker decomposition

$$\mathrm{vec}(x) = (u_4 \otimes \ldots \otimes u_1)\mathrm{vec}(c)$$

we decompose $c \in \mathbb{R}^{r_1 \times \ldots \times r_4}$ by

$$\mathrm{vec}(c) = (b_{\{3,4\}} \otimes b_{\{1,2\}})\mathrm{vec}(b_{\{1,2,3,4\}})$$

where $b_{\{1,2\}} \in \mathbb{R}^{r_1 r_2 \times r_{1,2}}$ and $b_{\{3,4\}} \in \mathbb{R}^{r_3 r_4 \times r_{3,4}}$, $b_{\{1,2,3,4\}} \in \mathbb{R}^{r_{1,2} \times r_{3,4}}$. Application of $(A \otimes B)(C \otimes D) = AC \otimes BD$ results in

$$\begin{aligned}\mathrm{vec}(x) &= (u_4 \otimes \ldots \otimes u_1)(b_{\{3,4\}} \otimes b_{\{1,2\}})\mathrm{vec}(b_{\{1,2,3,4\}}) \\ &= ((u_4 \otimes u_3)b_{\{3,4\}} \otimes (u_2 \otimes u_1)b_{\{1,2\}})\mathrm{vec}(b_{\{1,2,3,4\}}).\end{aligned}$$

The corresponding dimension tree is shown in Figure 4.2.1.

We now give the formal definition of a dimension tree.

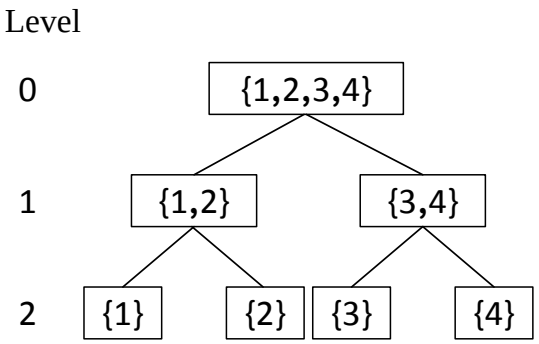


Figure 4.2.1.: Dimension tree structure of a 4-order tensor in H-Tucker format.

Definition 4.2.6. [Rup14, Def. 5.3], [Gra10, Def. 3.1] A dimension tree $\mathcal{T}_d$ is a tree with root $\{1,\dots,d\}$ such that each node $t \in \mathcal{T}_d$ is either a leaf and a singleton $t = \{\mu\}$, $\mu \in \{1,\dots,d\}$, or the union of two disjoint successors $t = t_\ell \cup t_r$. The level k of the tree is defined as the set of all nodes having a distance of k to the root, cf. Figure 4.2.1. We denote the set of nodes on level k of the tree by

$$\mathcal{T}_d^k = \{t \in \mathcal{T}_d \ : \ \text{level}(t) = k\}.$$

Remark 4.2.7. To be precise, $u_i := u_{\{i\}}$ in Example 4.2.5.

Definition 4.2.8. [Rup14, Def. 5.6] Let $\mathcal{T}_d$ be a dimension tree and $\{r_t\}_{t\in\mathcal{T}_d}$ be a family of non-negative integers. Let $(u_t)_{t\in\mathcal{T}_d}$ be matrices such that $u_t = (u_{t_r} \otimes u_{t_\ell})b_t$ with matrices $b_t \in \mathbb{R}^{r_{t_\ell} r_{t_r} \times r_t}$ for the inner nodes of the tree $\mathcal{I}(\mathcal{T}_d)$. The matrices u_t are called mode frames, the matrices b_t are called transfer tensors. Then, the collection $((b_t)_{t\in\mathcal{I}(\mathcal{T}_d)}, (u_t)_{t\in\mathcal{L}(\mathcal{T}_d)})$ is a hierarchical Tucker representation of the tensor $x = u_{\{1,\dots,d\}}$. The set $\mathcal{L}(\mathcal{T}_d)$ denotes the leaves of the dimension tree $\mathcal{T}_d$. If x is represented by a tree structure with corresponding mode frames and transfer tensors, we denote it as an element in $\mathcal{H}$.

If x is a d-order tensor and $((b_t)_{t\in\mathcal{I}(\mathcal{T}_d)}, (u_t)_{t\in\mathcal{L}(\mathcal{T}_d)})$ the corresponding H-Tucker representation, it can be written as, [Rup14, Lemma 5.3],

$$x = \bigotimes_{i=0}^{d-1} u_{\{d-i\}} \prod_{\ell=0}^{d-1} \bigotimes_{t\in\mathcal{T}_d^\ell} b_t. \tag{4.2.4}$$

The storage requirements are bounded by $(d-1)r^3 + r\sum_{\mu=1}^{d} n_\mu, \quad r = \max_{t\in\mathcal{T}_d} r_t$.

Remark 4.2.9. [Rup14, Lemma 5.2] For any tensor $x \in \mathbb{R}^{n_1\times\cdots\times n_d}$ a tree strucre can be obtained: For every t-matricisation $x^{(t)}$ with $t = t_\ell \cup t_r$ it holds $\text{span}(x^{(t)}) \subset \text{span}(x^{(t_r)} \otimes x^{(t_\ell)})$. Consequently, for any column u in $x^{(t)}$ there is a vector b such that $u = (x^{(t_r)} \otimes x^{(t_\ell)})b$. The

extension to u_t, u_{t_ℓ} and u_{t_r} as bases for $\operatorname{span}(x^{(t)})$, $\operatorname{span}(x^{(t_\ell)})$ and $\operatorname{span}(x^{(t_r)})$ respectively gives $u_t = (u_{t_r} \otimes u_{t_\ell})\operatorname{vec}(b_t)$.

To solve a system of linear equations the operator is required in H-Tucker format as well. We follow [Tob12, Sec. 2.3] and [AT14] for a short explanation regarding parabolic PDEs in space-time variational formulation. We continue using bold letters for coefficient vectors/tensors as well as vectors/matrices that represent (bi)linear forms in discrete (sub)spaces.

Example 4.2.10. Assuming a tensorised basis for the discrete approximation in space we deal with the discrete subspace $V^{\mathcal{J}} = V^{\mathcal{J}_1(1)} \otimes \cdots \otimes V^{\mathcal{J}_d(d)} \subset V$, cf. Example 2.2.31. A solution $u \in \mathbb{X}^{\mathcal{N}}$ can be represented by a tensor $\mathbf{u} \in \mathbb{R}^{\mathcal{J}_1 \times \ldots \times \mathcal{J}_d \times (\mathcal{K}+1)}$. More precise, let $\{\nu_1^{(k)}, \ldots, \nu_{\mathcal{J}_k}^{(k)}\}$ be the basis of $V^{\mathcal{J}_k,(k)}$, $1 \le k \le d$, and $\{e_0, \ldots, e_{\mathcal{K}}\}$ of the discrete temporal ansatz space $E^{\mathcal{K}+1}$. The tensor $\mathbf{u} \in \mathbb{R}^{\mathcal{J}_1 \times \ldots \times \mathcal{J}_d \times (\mathcal{K}+1)}$ is given by

$$\begin{aligned} u(t,x) &= \sum_{i_1=1}^{\mathcal{J}_1} \cdots \sum_{i_d=1}^{\mathcal{J}_d} \sum_{i_{d+1}=0}^{\mathcal{K}} \mathbf{u}_{i_1,\ldots,i_{d+1}} (e_{i_{d+1}} \otimes \nu_{i_d}^{(d)} \otimes \cdots \otimes \nu_{i_1}^{(1)})(x_1, \ldots, x_d, t) \\ &= \sum_{i_1=1}^{\mathcal{J}_1} \cdots \sum_{i_d=1}^{\mathcal{J}_d} \sum_{i_{d+1}=0}^{\mathcal{K}} \mathbf{u}_{i_1,\ldots,i_{d+1}} e_{i_{d+1}}(t) \nu_{i_d}^{(d)}(x_d) \cdots \nu_{i_1}^{(1)}(x_1). \end{aligned}$$

We can formally convert a matrix $\mathbf{A} \in \mathbb{R}^{m_1 \ldots m_d \times n_1 \ldots n_d}$ in H-Tucker format if it can be decomposed in the sum of tensor products of smaller matrices $A_j^{(\mu)} \in \mathbb{R}^{m_\mu \times n_\mu}$ for all $\mu = 1, \ldots, d$ and $j = 1, \ldots, R$:

$$\mathbf{A} = \sum_{j=1}^{R} A_j^{(d)} \otimes \cdots \otimes A_j^{(1)}.$$

Matrix operations as addition of two matrices or a matrix-matrix or matrix-vector multiplication can be performed directly in H-Tucker format without establishment of the full high dimensional system. The tensor representation of the matrix is given by the tensor $\bar{\mathbf{A}} \in \mathbb{R}^{n_1 m_1 \times \ldots \times n_d m_d}$ such that

$$\operatorname{vec}(\bar{\mathbf{A}}) = \sum_{j=1}^{R} \operatorname{vec}(A_j^{(d)}) \otimes \cdots \otimes \operatorname{vec}(A_j^{(1)}). \tag{4.2.5}$$

This corresponds to represent the matrix $\mathbf{A}$ by a vectorisation where each of the hierarchical blocks is converted columnwise with the vec-operator, cf. Figure 4.2.2. This is *not* the application of the vec-operator to the entire matrix. The right-hand side can be interpreted as a CP decomposition. In the following Lemma the conversion of the CP decomposition in H-Tucker format is described.

Lemma 4.2.11. *[Tob12, Sec. 3.8] Let $x \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ be in exact CP decomposition* (4.2.1)*. For a dimension tree $\mathcal{T}_d$ we define*

$$\forall\ t = \{\mu\} \in \mathcal{L}(\mathcal{T}_d): \quad [u_t]_{\cdot\, k} = u_k^{(\mu)}, \quad k = 1, \ldots, r, \quad u_t \in \mathbb{R}^{n_\mu \times r_\mu},\ r_\mu := r$$

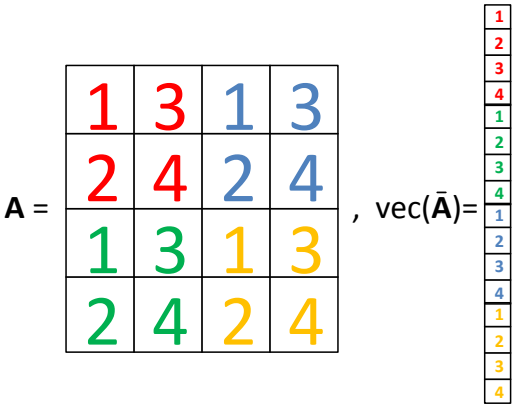


Figure 4.2.2.: A matrix $\mathbf{A} = A^{(2)} \otimes A^{(1)}$, $A^{(\mu)} \in \mathbb{R}^{2\times 2}$ for $\mu = 1, 2$ has four subblocks highlighted on the left-hand side. The corresponding vectorisation $\mathrm{vec}(\bar{\mathbf{A}})$ in tensor format is shown on the right-hand side.

and the transfer tensors are given by

$$\forall\, t \in \mathcal{I}(\mathcal{T}_d) \setminus \{1, \dots, d\}: \quad (b_t)_{i,j,k} := \begin{cases} 1 & i = k = j, \\ 0 & \textit{otherwise} \end{cases} \quad b_t \in \mathbb{R}^{r_t \times r_t \times r_t}, r_t := r.$$

The root transfer tensor is given by

$$(b_{\{1,\dots,d\}})_{i,j,1} := \begin{cases} 1 & i = j \\ 0 & \textit{otherwise} \end{cases}, \quad b_{\{1,\dots,d\}} \in \mathbb{R}^{r\times r\times 1}.$$

Moreover there is an isomorphism $\Phi : \mathcal{L}(\mathbb{R}^{n_1\times\cdots\times n_d}, \mathbb{R}^{m_1\times\cdots\times m_d}) \to \mathbb{R}^{n_1 m_1\times\ldots\times n_d m_d}$.

Example 4.2.12. Heat equation. Let $\Omega \subset \mathbb{R}$, $I = (0, T)$ for $0 < T < \infty$, $V = H_0^1(\Omega)$ and $H = L_2(\Omega)$. For $u_0 \in L_2(\Omega)$ the problem formulation denotes

$$\begin{aligned} \dot{u} - \Delta u &= 0 \\ u(x,t) &= 0 \quad \forall\, x \in \partial\Omega \\ u(x,0) &= u_0 \quad \forall\, x \in \Omega. \end{aligned}$$

The left-hand side of the space-time variational formulation is given by

$$b(u,v) = \int_I \int_\Omega \dot{u}(x,t) v_1(x,t) dx dt + \int_I \int_\Omega \nabla u(x,t) \nabla v_1(x,t) dx dt + \int_\Omega u(x,0) v_2(x) dx.$$

For the discretisation introduced in Section 2.2.2.2 the operator $\mathbf{B}$ is given by

$$\mathbf{B} = \mathbf{C}^{\text{time}} \otimes \mathbf{M}^{\text{space}} + \mathbf{A}^{\text{time}} \otimes \mathbf{L}^{\text{space}}.$$

where $[\mathbf{M}^{\text{space}}]_{ij} = \int_\Omega \phi_i(x)\phi_j(x)dx$, $[\mathbf{L}^{\text{space}}]_{ij} = \int_\Omega \dot{\phi}_i(x)\dot{\phi}_j(x)dx \quad \forall\, i,j = 1,\dots,\mathcal{J}$,

$$[\mathbf{C}^{\text{time}}]_{ij} = \begin{cases} \int_I \dot{\sigma}_{i-1}(t)\tau_j(t)dt & i = 1,\dots,\mathcal{K}+1, j = 1,\dots,\mathcal{K} \\ 1 & i = j = \mathcal{K}+1 \\ 0 & \text{else} \end{cases}$$

and $[\mathbf{A}^{\text{time}}]_{ij} = \begin{cases} \int_I \sigma_{i-1}(t)\tau_j(t)dt & i = 1,\dots,\mathcal{K}+1, j = 1,\dots,\mathcal{K} \\ 0 & \text{else} \end{cases}$.

The H-Tucker representation of $b(\cdot,\cdot)$ is given by

$$u^{(2)}_{\{2\}} = [\text{vec}(\mathbf{C}^{\text{time}}), \text{vec}(\mathbf{A}^{\text{time}})] \quad u^{(1)}_{\{1\}} = [\text{vec}(\mathbf{M}^{\text{space}}), \text{vec}(\mathbf{L}^{\text{space}})]$$

and $b_{\{1,2\}} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$.

For the matrix-vector product we first look at the result

$$\text{vec}(Ax) = \sum_{j=1}^{R} (A_j^{(d)} \otimes \cdots \otimes A_j^{(1)})\text{vec}(x), \quad A_j^{(\mu)} \in \mathbb{R}^{m_\mu \times n_\mu}.$$

Lemma 4.2.13. *The application of $\bar{\mathbf{A}} \in \mathcal{H}$ to a vector x in H-Tucker format results in a vector $Ax \in \mathcal{H}$. Let $((b_t^x)_{t\in\mathcal{I}(T_d)}, (u_t^x)_{t\in\mathcal{L}(T_d)})$ be the H-Tucker representation of x. The mode frames of Ax are defined by*

$$u_t^{Ax} = [A_1^{(t)}u_t^x, \dots, A_R^{(t)}u_t^x], \quad t \in \mathcal{L}(\mathcal{T}_d)$$

and its transfer tensors by

$$b_t^{Ax} = b_t^{\bar{\mathbf{A}}} \otimes b_t^x, \quad t \in \mathcal{I}(\mathcal{T}_d).$$

The transfer tensors $b_t^{\bar{\mathbf{A}}}$ of $\bar{\mathbf{A}}$ are given by applying Lemma 4.2.11 to the CP decomposition (4.2.5).

Remark 4.2.14. The upper indices 'x', 'Ax' and '$\bar{\mathbf{A}}$' are just labels and added for convenience.

The hierarchical rank grows to R. This is important in the usage within the reduced basis method.

Remark 4.2.15. We have not used the mode frames of the tensor representation $\bar{\mathbf{A}}$ of $\mathbf{A}$ explicitly in the matrix-vector product. However, with application of Lemma 4.2.11 the mode frames are defined by

$$[u_t^{\bar{\mathbf{A}}}]_{\cdot\,\mu} = \text{vec}(A_\mu^{(t)}), \quad A_\mu^{(t)} \in \mathbb{R}^{m_t \times n_t}, \mu = 1,\dots,R.$$

For operators in H-Tucker format the matrix-matrix product is an extension of the matrix-vector product introduced in Lemma 4.2.13 and we refer to [Rup14, Lemma 5.10] for details. We refer to Appendix C for the scalar multiplication in H-Tucker format as well as the addition of two tensors.

4.2.3. BPX-preconditioning for Parabolic PDEs

This section shortly introduces preconditioners for parabolic PDEs in discretised space-time variational formulation. A finite element discretisation with compactly supported basis results in sparse stiffness matrices. Unfortunately the condition of the matrix grows in dependence of the size of the matrices, [DK92]. R. Andreev introduced multilevel preconditioners for parabolic problems in space-time variational formulation. The preconditioners are based on the BPX[1]-preconditioners for elliptic problems, [BPX90]. They are optimal in the sense that the polynomial growth of the condition number with the size of the matrix reduces to a logarithmic growth, [DK92, BPX90]. The section mainly follows [And12, Tob12, AT14].

Proposition 4.2.16. [And12, Prop. 4.2.3] Suppose that all assumptions of Theorem 2.2.23 are valid. Let $\mathcal{M}$ and $\mathcal{N}$ be symmetric positive definite operators that satisfy the norm equivalences

$$d_{\mathcal{M}}\|\cdot\|_{\mathcal{M}} \leq \|\cdot\|_{\mathsf{X}} \leq D_{\mathcal{M}}\|\cdot\|_{\mathcal{M}} \text{ and } d_{\mathcal{N}}\|\cdot\|_{\mathcal{N}} \leq \|\cdot\|_{\mathsf{Y}} \leq D_{\mathcal{N}}\|\cdot\|_{\mathcal{N}}$$

for $0 < d_{\mathcal{M}} \leq D_{\mathcal{M}}$ and $0 < d_{\mathcal{N}} \leq D_{\mathcal{N}}$. Let $\mathcal{B}$ be the Riesz representation of the bilinear form $b(\cdot,\cdot)$, i.e. $\langle \mathcal{B}u, v\rangle = b(u,v)\ \forall u \in \mathbb{X}, v \in \mathbb{Y}$. The condition of $\mathcal{B}^* = \mathcal{M}^{-1/2}\mathcal{B}^T\mathcal{N}^{-1}\mathcal{B}\mathcal{M}^{1/2}$ is bounded:

$$\sqrt{k_2(\mathcal{B}^*)} \leq \frac{D_{\mathcal{N}}D_{\mathcal{M}}}{d_{\mathcal{N}}d_{\mathcal{M}}}\frac{C_b^{\mathcal{N}}}{\beta^{\mathcal{N}}}.$$

In the following, the explicit construction of such norm inducing operators are given.

4.2.3.1. Norm-inducing Operators

In the following we collect the requirements on the temporal and spatial domain for the construction of the BPX-preconditioners, [AT14, Sec. 2.3]. Recall first that the trial and test spaces, cf. Equation (2.2.1) and (2.2.5), can be splitted into tensor products,

$$\mathbb{X} = H^1(I) \otimes V, \quad \mathbb{Y} = L_2(I) \otimes V \times H.$$

The Hilbert space V is densly embedded in the Hilbert space H. We require a sequence of closed nested subspaces in the temporal and spatial domain, $\{0\} = E_0 \subset E_1 \subset \ldots \subset H^1(I)$ and

[1] Bramble-Pascial-Xu

$\{0\} = V_0 \subset V_1 \subset \ldots \subset V$, with $\cup_{k\in\mathbb{N}_0} E_k \subseteq L_2(I)$ and $\cup_{\ell\in\mathbb{N}_0} V_\ell \subseteq H$ dense. Further we assume given projections $P_k : L_2(I) \to E_k$ satisfying $E_k = E_{k-1} \oplus P_k(L_2(I))$ and $Q_\ell : H \to V_\ell$ with $V_\ell = V_{\ell-1} \oplus Q_\ell(H)$.

Remark 4.2.17. [And12, Sec. 6.2] In general the projection neither need to be surjectiv nor orthogonal for the respective inner products. Further, $\sum_{k\in\mathbb{N}} P_k = Id_{L_2(I)}$ and $\sum_{\ell\in\mathbb{N}} Q_\ell = Id_H$ considering pointwise convergence.

Assumption 4.2.18. *[And12, (6.2.1), (6.2.2), Ass. 6.2.1, Ass. 6.2.2] For the construction of norm equivalent BPX preconditioners we require the following norm equivalences:*

$$d_{L_2(I)}|f|^2_{L_2(I)} \leq \sum_{k\in\mathbb{N}} |P_k(f)|^2_{L_2(I)} \leq D_{L_2(I)}|f|^2_{L_2(I)} \quad \forall\ f \in L_2(I)$$

and

$$d_H|f|^2_H \leq \sum_{\ell\in\mathbb{N}} |Q_\ell(f)|^2_H \leq D_H|f|^2_H \quad \forall\ f \in H.$$

We further assume that there exist constants $d_{H^1(I)}, D_{H^1(I)}$ and a monotone sequence $(p_k)_{k\in\mathbb{N}} \subset (0,\infty)$, such that

$$d_{H^1(I)}|f|^2_{H^1(I)} \leq \sum_{k\in\mathbb{N}} p_k^2|P_k(f)|^2_{L_2(I)} \leq D_{H^1(I)}|f|^2_{H^1(I)} \quad \forall\ f \in H^1(I)$$

and constants d_V, D_V and again a monotone sequence $(q_\ell)_{\ell\in N} \subset (0,\infty)$, such that

$$d_V|f|^2_V \leq \sum_{\ell\in\mathbb{N}} q_\ell^2|Q_\ell(f)|^2_H \leq D_V|f|^2_V \quad \forall\ f \in V. \tag{4.2.6}$$

The later induces a multilevel norm equivalence on the dual space V' of V if Q_ℓ are H-orthogonal projectors on H [And12, Lemma 6.2.3]

$$D_V^{-1}|f|^2_{V'} \leq \sum_{\ell\in\mathbb{N}} q_\ell^{-2}|Q_\ell(f)|^2_H \leq d_V^{-1}|f|^2_{V'} \quad \forall\ f \in V.$$

We denote the norm-inducing operators on trial and test spaces by $\mathcal{M}$ and $\mathcal{N}$ respectively.

Proposition 4.2.19. [And12, Sec. 6.2.2] For any $w \in \mathbb{X}_\mathcal{M} := \cup_{k,\ell\in\mathbb{N}} E_k \otimes V_\ell$, $\widetilde{w} \in \mathbb{X}$ we define $\mathcal{M} : \mathbb{X}_\mathcal{M} \to \mathbb{X}'$ by

$$\langle \mathcal{M}w, \widetilde{w}\rangle_{\mathbb{X}'\times\mathbb{X}} = \sum_{k\in\mathbb{N}}\sum_{\ell\in\mathbb{N}} (q_\ell^2 + p_k^2 q_\ell^{-2})\langle (P_k \otimes Q_\ell)w, (P_k \otimes Q_\ell)\widetilde{w}\rangle_{L_2(I)\otimes H}.$$

The norm equivalence is given by

$$d_{L_2(I)}d_V\|w\|^2_{L_2(I)\otimes V} + d_{H^1(I)}d_{V'}\|w\|^2_{H^1(I)\otimes V'} \le \langle \mathcal{M}w, w\rangle_{\mathbb{X}'\times\mathbb{X}} \le D_{L_2(I)}D_V\|w\|^2_{L_2(I)\otimes V} + D_{H^1(I)}D_{V'}\|w\|^2_{H^1(I)\otimes V'}. \tag{4.2.7}$$

Proof. The norm equivalence is derived applying the norm equivalences stated in Assumption 4.2.18. □

The operator $\mathcal{M}$ can be extended to an s.p.d. operator $\mathcal{M} \in \mathrm{Iso}(\mathbb{X}, \mathbb{X}')$.

Proposition 4.2.20. [And12, Sec. 6.2.1] The norm-inducing operator $\mathcal{N} : \mathbb{Y}_{\mathcal{N}} := \cup_{k,\ell\in N} E_k \otimes V_\ell \times \cup_{\ell\in\mathbb{N}} V_\ell \to \mathbb{Y}'$ is decomposed into two parts, $\mathcal{N}w = (\mathcal{N}_1 w_1, \mathcal{N}_2 w_2)$ for $w = (w_1, w_2) \in \mathbb{Y}_{\mathcal{N}}$. For $w \in \mathbb{Y}_{\mathcal{N}}$, $\widetilde{w} \in \mathbb{Y}$ we define

$$\langle \mathcal{N}_1 w_1, \widetilde{w}_1\rangle_{\mathbb{Y}_1'\times\mathbb{Y}_1} := \sum_{k\in\mathbb{N}}\sum_{\ell\in\mathbb{N}} q_\ell^2 \langle (P_k\otimes Q_\ell)w_1, (P_k\otimes Q_\ell)\widetilde{w}_1\rangle_{L_2(I)\otimes H},$$

and

$$\langle \mathcal{N}_2 w_2, \widetilde{w}_2\rangle_H := \sum_{\ell\in\mathbb{N}} \langle Q_\ell w_2, Q_\ell \widetilde{w}_2\rangle_H.$$

Again the operator yields the norm equivalence to the respective test space norm $\|\cdot\|_{\mathbb{Y}}$ on $\mathbb{Y}_{\mathcal{N}}$.

The continuous extension $\mathcal{N} \in \mathrm{Iso}(\mathbb{Y}, \mathbb{Y}')$ is an s.p.d. operator [And12, 6.2.1 3.]. We verify in the next two examples that the necessary Assumptions 4.2.18 are satisfied in standard situations.

Example 4.2.21. For $V = H^1_0(D) \hookrightarrow L_2(D) = H$ we follow Example 2.2.31 for d open intervals $D^{(\mu)} \subset \mathbb{R}$, $\mu = 1, \dots, d$. Here, we do not restrict on the intervals $(-1, 1)$. Denote by $|D^{(\mu)}|$ the length of the interval. The nested subspaces $V^{(\mu)}_{\ell_\mu} \subset V$ are the standard conforming finite element spaces given by all continuous piecewise linear functions w.r.t. a uniform partition of $D^{(\mu)}$ in $2^{\ell_\mu+1}$ subintervals. The discretised Hilbert space is given by $V_{\boldsymbol{\ell}} = V^{(1)}_{\ell_1} \otimes \cdots \otimes V^{(d)}_{\ell_d} \subseteq V$, $\boldsymbol{\ell} = (\ell_1, \dots, \ell_d)$, $(\boldsymbol{\ell})_\mu = \ell_\mu$. Let $Q_{\boldsymbol{\ell}} : L_2(D) \to V_{\boldsymbol{\ell}} \cap (\sum_{\boldsymbol{\ell}\neq\boldsymbol{\ell}'\le\boldsymbol{\ell}} V_{\boldsymbol{\ell}'})^{\perp_H}$ be the $L_2(D)$-orthogonal surjective projection.

Recall Equation (2.2.19) ([AT14, p. 8])

$$|v_1\otimes\ldots\otimes v_d|^2_{H^1_0(D)} = \sum_{\mu=1}^d (|v_\mu|^2_{H^1_0(D^{(\mu)})} \prod_{\mu\neq\mu'=1}^d |v_{\mu'}|^2_{L_2(D^{(\mu')})}).$$

The norm equivalence stated in Equation (4.2.6) then holds for the monotone sequence

$$q_{\boldsymbol{\ell}} := \sqrt{\sum_{\mu=1}^d q^2_{\ell_\mu}} = \sqrt{\sum_{\mu=1}^d (\frac{2^{\ell_\mu+1}}{|D^{(\mu)}|})^2},$$

cf. [GO95, Thm. 3] and [AT14, (2.35)].

The temporal mesh of E_k is also given by the continuous piecewise linear functions on I with step size $\Delta t = \frac{|I|}{2^{k+1}}$. The $L_2(I)$-orthogonal surjective projection $P_k : H^1(I) \to E_k \cap (E_{k-1})^{\perp_{L_2(I)}}$ satsifies the required norm equivalences. The monotone sequence is given by $p_k := \frac{2^{k+1}}{|I|}$.

Example 4.2.22. Given the same setting as in the previous example but $V := H^1(D)$, $H := L_2(D)$ we detail

$$|v_1 \otimes \ldots \otimes v_d|^2_{H^1(D)} = \prod_{\mu'=1}^{d} |v_{\mu'}|^2_{L_2(D^{(\mu')})} + \sum_{\mu=1}^{d} (|v_\mu|^2_{H^1_0(D^{(\mu)})} \prod_{\mu \neq \mu'=1}^{d} |v_{\mu'}|^2_{L_2(D^{(\mu')})})$$

and the corresponding monotone sequence is defined as, [GO95, Thm. 3],

$$q_\ell := \sqrt{1 + \sum_{\mu=1}^{d} q^2_{\ell_\mu}} = \sqrt{1 + \sum_{\mu=1}^{d} (\frac{2^{\ell_\mu+1}}{|D^{(\mu)}|})^2}.$$

Again for the temporal related norm equivalence we receive $p_k := \frac{2^{k+1}}{|I|}$.

The construction of the preconditioners allows for an explicit representation of their inverse, [And12, Sec. 6.2.3]. Define for orthogonal projectors P_k and Q_ℓ where all multilevel equivalences hold as above [And12, Prop. 6.2.5]

$$\mathcal{M}_\pm = \sum_{k \in \mathbb{N}} \sum_{\ell \in \mathbb{N}} (q^2_\ell + p^2_k q^{-2}_\ell)^{\pm 1} (P_k \otimes Q_\ell). \tag{4.2.8}$$

It was shown in [And12, Prop 6.2.5] that $\mathcal{M} = \mathcal{M}_+$ and $\mathcal{M}^{-1} = \mathcal{M}_-$. Analogous we can define $\mathcal{N} = \mathcal{N}_+$ and $\mathcal{N}^{-1} = \mathcal{N}_-$ by

$$\mathcal{N}_\pm = (\sum_{k \in \mathbb{N}} \sum_{\ell \in \mathbb{N}} (q^2_\ell)^{\pm 1} (P_k \otimes Q_\ell), \sum_{\ell \in \mathbb{N}} Q_\ell). \tag{4.2.9}$$

4.2.3.2. Explicit Constructions in H-Tucker Format

Let $\{x_1, \ldots, x_{\mathcal{N}}\}$ and $\{y_1, \ldots, y_{\widetilde{\mathcal{N}}}\}$ be the (finite element) bases for ansatz and test space respectively. Define for $\mathcal{M}_\pm$ and $\mathcal{N}_\pm$ the operator

$$[\mathbf{M}_\pm]_{ij} := \langle \mathcal{M}_\pm x_i, x_j \rangle_{\mathbb{X}' \times \mathbb{X}} \text{ and } [\mathbf{N}_\pm]_{k\ell} := \langle \mathcal{N}_\pm y_k, y_\ell \rangle_{\mathbb{Y}' \times \mathbb{Y}}$$

for all $i, j = 1, \ldots, \mathcal{N}$, $k, \ell = 1, \ldots, \widetilde{\mathcal{N}}$ and the mass matrices

$$[\mathbf{M}_0]_{ij} := \langle x_i, x_j \rangle_{\mathbb{X}' \times \mathbb{X}} \text{ and } [\mathbf{N}_0]_{k\ell} := \langle y_k, y_\ell \rangle_{\mathbb{Y}' \times \mathbb{Y}}.$$

The matrix representation of $\mathcal{M}$ is $\mathbf{M}_+$ as is $\mathbf{N}_+$ for $\mathcal{N}$. The matrix representation of the inverse is given in the next proposition.

Proposition 4.2.23. [And12, Prop. 6.2.6], [AT14, Prop. 2.5] The matrix representation of $\mathcal{M}^{-1}$ is $\mathbf{M}_+^{-1} := \mathbf{M}_0^{-1}\mathbf{M}_-\mathbf{M}_0^{-1}$. For $\mathcal{N}^{-1}$ the matrix representation is given by $\mathbf{N}_+^{-1} := \mathbf{N}_0^{-1}\mathbf{N}_-\mathbf{N}_0^{-1}$.

Proof. It suffices to show $\mathbf{M}_0 = \mathbf{M}_-\mathbf{M}_0^{-1}\mathbf{M}_+$:

$$\begin{aligned}\bar{\mathbf{x}}^T\mathbf{M}_0\mathbf{x} = \langle \bar{x}, x\rangle_{\mathbb{X}'\times\mathbb{X}} &= \langle \mathcal{M}_+^{-1}\bar{x}, \mathcal{M}_+x\rangle_{\mathbb{X}'\times\mathbb{X}} = \langle \mathcal{M}_-\bar{x}, \mathcal{M}_+x\rangle_{\mathbb{X}'\times\mathbb{X}} \\ &= (\mathbf{M}_0^{-1}\mathbf{M}_-\bar{\mathbf{x}})^T\mathbf{M}_0(\mathbf{M}_0^{-1}\mathbf{M}_+\mathbf{x}) = \bar{\mathbf{x}}^T\mathbf{M}_-\mathbf{M}_0^{-1}\mathbf{M}_+\mathbf{x}\end{aligned}$$

for $\bar{\mathbf{x}}$ and $\mathbf{x}$ being the coefficient vectors of $\bar{x}$ and x in $\mathbb{Y}^{\mathcal{N}}$. The proof is analogous for $\mathbf{N}_+^{-1}$. □

For solving the parabolic PDE we require the tensor representation of the bilinear form $b(\cdot,\cdot)$ of the left-hand side denoted by $\mathbf{B} \in \mathcal{H}$ and of the right-hand side $\mathbf{f} \in \mathcal{H}$. One possibility is to follow [AT14], cf. [Tob12, Sec. 5], using a preconditioned low rank CGNR method for solving the linear equation system, i.e., the solution $u = \sum_{i=1}^{\mathcal{N}}(\mathbf{u})_i x_i \in \mathbb{X}$ is given by $\mathbf{M}^{-1}\mathbf{B}^T\mathbf{N}^{-1}\mathbf{B}\mathbf{u} = \mathbf{M}^{-1}\mathbf{B}^T\mathbf{N}^{-1}\mathbf{f}$. The problem formulation requires only the inverse of the preconditioners. For an explicit construction we detail all basis functions of $\mathbb{X}^{\mathcal{N}}$ in their tensor product representation

$$e_{i_{d+1}} \otimes v_{i_d}^{(d)} \otimes \cdots \otimes v_{i_1}^{(1)}$$

for $i_{d+1} = 0, \ldots, \mathcal{K}^E$, $i_\mu = 1, \ldots, \mathcal{J}_\mu$, $\mu = 1, \ldots, d$. We generalise to arbitrary test spaces, e.g. the minimal residual approach of [And12] where the temporal test space discretisation is a refinement of the temporal ansatz space discretisation. The basis of $\mathbb{Y}^{\widetilde{\mathcal{N}}}$ is given by

$$(f_{i_{d+1}} \otimes v_{i_d}^{(d)} \otimes \cdots \otimes v_{i_1}^{(1)}, 0) \text{ and } (0, \otimes v_{i_d}^{(d)} \otimes \cdots \otimes v_{i_0}^{(0)})$$

where $i_{d+1} = 0, \ldots, \mathcal{K}^F$, $i_\mu = 1, \ldots, \mathcal{J}_\mu$, $\mu = 1, \ldots, d$. Note that the number of basis functions in the spatial domain depends on the Hilbert space V and the underlying discretisation of the spatial domain Ω. We define the mass matrices by

$$[\mathbf{M}_e^{(d+1)}]_{ij} = \int_I e_i(t)e_j(t)dt, \quad [\mathbf{M}_f^{(d+1)}]_{ij} = \int_I f_i(t)f_j(t)dt, \quad [\mathbf{M}^{(\mu)}]_{i_\mu j_\mu} = \int_{\Omega^{(\mu)}} v_{i_\mu}^{(\mu)}(x)v_{j_\mu}^{(\mu)}(x)dx$$

where $\Omega = \Omega^{(1)} \times \cdots \times \Omega^{(d)}$ and $\mu = 1, \ldots, d$. The mass matrices $\mathbf{M}_0$ and $\mathbf{N}_0$ are given by

$$\begin{aligned}\mathbf{M}_0 &= \mathbf{M}_e^{(d+1)} \otimes \mathbf{M}^{(d)} \otimes \cdots \otimes \mathbf{M}^{(1)} \\ \mathbf{N}_0 &= \begin{pmatrix} \mathbf{M}_f^{(d+1)} & 0 \\ 0 & 1 \end{pmatrix} \otimes \mathbf{M}^{(d)} \otimes \cdots \otimes \mathbf{M}^{(1)}.\end{aligned}$$

For the inverse operator $\mathcal{M}^{-1}$ we also need $\mathbf{M}_+$ and $\mathbf{M}_-$. Analogously for $\mathcal{N}^{-1}$ we require $\mathbf{N}_+$ and $\mathbf{N}_-$. We give the explicit construction for $\mathcal{M}$ analogous to [AT14, Sec. 3.3]. Storing the PBX preconditioners in tensor format, the matrix-matrix product is performed in an efficient way. For this reason, an another interpretation of a matrix in tensor format, [Tob12, Eq. (3.23)], is required: Given a matrix $\mathbf{A}$ in the following format

$$\mathbf{A} = \sum_{j_d=1}^{R_d} \cdots \sum_{j_1=1}^{R_1} \mathbf{h}_{j_1 \dots j_d} (A_{j_d}^{(d)} \otimes \cdots \otimes A_{j_1}^{(1)}),$$

we can first store the coefficient tensor $\mathbf{h} \in \mathbb{R}^{R_1 \times \cdots \times R_d}$ in H-Tucker format. Instead of using the full coefficient tensor one may approximate the operator by truncation of $\mathbf{h} \in \mathcal{H}$. For the preconditioner $\mathcal{M}$ and $\mathcal{N}$ we refer to [AT14, Sec. 3.3] where this truncation process is described in detail. Let $u_t^{\mathbf{h}} \in \mathbb{R}^{R_t \times s_t}$ for all $t \in \mathcal{L}(\mathcal{T}_d)$ and $b_t^{\mathbf{h}} \in \mathbb{R}^{s_{t_\ell} s_{t_r} \times s_t}$ for all $\mathcal{I}(\mathcal{T}_d)$ be the mode frames and transfer tensors of the H-Tucker representation of $\mathbf{h}$. The tensor representation $\bar{\mathbf{A}} \in \mathcal{H}$ of $\mathbf{A}$ yields the same dimension tree $\mathcal{T}_d$. The mode frames are given by

$$u_t^{\bar{\mathbf{A}}} = (\mathrm{vec}(A_0^{(t)}), \dots, \mathrm{vec}(A_{R_t}^{(t)})) u_t^{\mathbf{h}} \in \mathbb{R}^{n_t m_t \times s_t} \quad \forall\, t \in \mathcal{L}(\mathcal{T}_d).$$

The transfer tensors are again the same as for the coefficient tensor, i.e., $b_t^{\bar{\mathbf{A}}} = b_t^{\mathbf{h}}$. The truncation of the coefficient tensor is equivalent to a quasi-optimal approximation of the operator in the Frobenius norm [Tob12, Sec. 3.8]. The approximation of the preconditioners has an effect on the preconditioning itself. The relation of the precondition effect to the size of the tensor ranks is explained in [AT14, Sec. 3.3] such that an effective truncation rank can be chosen.

The matrices $\mathbf{M}_+$ and $\mathbf{M}_-$ are now given by [AT14, (3.21)]

$$\mathbf{M}_\pm = \sum_{\ell_1=0}^{\mathcal{J}_1} \cdots \sum_{\ell_d=0}^{\mathcal{J}_d} \sum_{\ell_{d+1}=0}^{\mathcal{K}^E} (\mathbf{g}_{\ell_1 \dots \ell_{d+1}})^{\pm 1} (P_{\ell_{d+1}}^{(d+1)} \otimes \cdots \otimes P_{\ell_1}^{(1)})$$

where the tensor $\mathbf{g}$ is given by

$$\mathbf{g}_{\ell_1 \dots \ell_{d+1}} := \left(\sum_{\mu=1}^{d} q_{\ell_\mu}^2 \right) + p_{\ell_{d+1}}^2 \left(\sum_{\mu=1}^{d} q_{\ell_\mu}^2 \right)^{-1}.$$

The construction of the projection matrices $P_\ell^{(\mu)}$ for all $\mu = 1, \dots, d+1$, $\ell = 1, \dots, {R_\mu}^2$ follows with standard multigrid methods. The highest levels are deduced from the discretisation, i.e., $L_\mu := \mathcal{J}_\mu, \mu = 1, \dots, d$ and $L_{d+1} := \mathcal{K}^E$ resp. $\mathcal{K}^F$. We denote the mass matrices on level ℓ by

[2] $R_\mu = \mathcal{J}_\mu$ for $\mu = 1, \dots, d$ and $R_{d+1} = \mathcal{K}^E$

$M_\ell^{(\mu)}$, $\ell \in \{0, \dots, L_\mu\}$, $\mu = 1, \dots, d+1$. We define the projection operators by

$$P_\ell^{(\mu)} := \mathbf{M}^{(\mu)}(S^{(\mu)}_{L_\mu \nwarrow \ell}(\mathbf{M}_\ell^{(\mu)})^{-1} S^{(\mu)}_{L_\mu \swarrow \ell} - S^{(\mu)}_{L_\mu \nwarrow \ell-1}(\mathbf{M}_{\ell-1}^{(\mu)})^{-1} S^{(\mu)}_{L_\mu \swarrow \ell-1})\mathbf{M}^{(\mu)}$$
$$P_0^{(\mu)} := \mathbf{M}^{(\mu)}(S^{(\mu)}_{L_\mu \nwarrow 0}(\mathbf{M}_0^{(\mu)})^{-1} S^{(\mu)}_{L_\mu \swarrow 0})\mathbf{M}^{(\mu)}.$$

The prolongation from level ℓ_1 to level ℓ_2, $\ell_1 \leq \ell_2$ is given by $S_{\ell_2 \nwarrow \ell_1}$. Furthermore, $S_{\ell_1 \swarrow \ell_2} = S^T_{\ell_2 \nwarrow \ell_1}$ denotes the restriction.

Remark 4.2.24. [AT14, Sec. 3.3]

(a) The mass matrix $M_\ell^{(\mu)}$ on a lower level ℓ is given by $M_\ell^{(\mu)} = S_{\ell \swarrow L_\mu} \mathbf{M}^{(\mu)} S_{L_\mu \nwarrow \ell}$. Here, $\mathbf{M}^{(d+1)} := \mathbf{M}_e^{(d+1)}$.

(b) For all $\mu = 1, \dots, d+1$: $P_{\ell'}^{(\mu)}(\mathbf{M}^{(\mu)})^{-1} P_\ell^{(\mu)} = \delta_{\ell\ell'} P_\ell^{(\mu)} \quad \forall\, \ell, \ell' \in \{0, \dots, L_\mu\}$.

(b) $\sum_{\ell=1}^{R_\mu} P_\ell^{(\mu)} = M^{(\mu)}$

The operator $\mathcal{N}$ is given by

$$\mathbf{N} = \sum_{\ell_1=0}^{\mathcal{J}_1} \cdots \sum_{\ell_d=0}^{\mathcal{J}_d} \sum_{\ell_{d+1}=0}^{\mathcal{K}^F} \left(\sum_{\mu=1}^{d} q_{\ell_\mu}^2 \right) \begin{pmatrix} P_{\ell_{d+1}}^{(d+1)} & 0 \\ 0 & 0 \end{pmatrix} \otimes P_{\ell_d}^{(d)} \otimes \ldots \otimes P_{\ell_1}^{(1)}$$
$$+ \sum_{\ell_1=0}^{\mathcal{J}_1} \cdots \sum_{\ell_d=0}^{\mathcal{J}_d} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \otimes P_{\ell_d}^{(d)} \otimes \ldots \otimes P_{\ell_1}^{(1)}.$$

4.3. Using H-Tucker in the Space-Time RBM

One of the drawbacks of the space-time variational formulation are the very high dimensional linear systems after discretisation. Up to now we worked with a time discretisation such that the linear system is equivalent to a time stepping scheme. But even with the separation of the temporal and spatial domain for the solution process the method can be computationally expensive. We require e.g. the dual norm of the residual for computing the error estimate, that is not fully decomposable. Low rank tensor formats are mainly developed to have an efficient compression scheme of very high dimensional systems in terms of spatial dimensions. In finance this was used e.g. for pricing structured financial products [Rup14]. We do not focus on the low rank tensor format as a model reduction method in the first place. In fact we analyse a possible use in the offline phase of the space-time RBM. If all operations are performed in tensor format, it is highly efficient[3] and avoids the establishment of the full system. The numerical experiments in Section 4.4 show the possibilities and the limits using the H-Tucker format in the RB offline phase. The error analysis is crucial here. In the numerical experiments we use

[3] As long as the rank of the underlying operator does not exceed a certain level.

the Hierarchical Tucker Toolbox provided by D. Kressner and Ch. Tobler. A description can be found in [KT12] and [Tob12, Sec. 3]. The interface for using the H-Tucker toolbox in combination with the RBMatlab structure, where the numerical experiments are embedded in, was developed in conjunction with the master thesis of F. Grimmer [Gri15].

4.3.1. Parabolic Partial Differential Equation in H-Tucker Format

Following up the numerical realisation of the PDE operators in Section 2.4.2, we extend the operator separation to the spatial domain. All computations are performed in H-Tucker format. The full system does not have to be established when the operator matrices $\mathbf{B}$ and $\mathbf{f}$ are stored as tensors. This Section mainly follows [Tob12, Sec. 2.3] as well as [AT14].

For $\Omega \subset \mathbb{R}^d$ and a time interval $I \subset \mathbb{R}$, we are interested in a solution to

$$\frac{d}{dt}u(t,\omega) - \nabla \cdot (\alpha(t,\omega)\nabla u(t,\omega)) + \beta(t,\omega)\nabla u(t,\omega) + \gamma(t,\omega)u(t,\omega) = g(t,\omega) \quad (t,\omega) \in I \times \Omega$$
$$u(0,\omega) = u_0(\omega).$$

For the ease of the presentation we assume homogeneous Dirichlet boundary conditions

$$u(t,\omega) = 0 \quad (t,\omega) \in I \times \partial\Omega.$$

Suppose that the coefficients $\alpha : I \times \Omega \to \mathbb{R}^{d\times d}$, $\beta : I \times \Omega \to \mathbb{R}^d$ and $\gamma : I \times \Omega \to \mathbb{R}$ as well as the source term g and the initial value u_0 are separable in the temporal and all spatial directions:

$$\alpha_{ij}(t,\omega) = \alpha_{ij}^{(d+1)}(t)\prod_{\mu=1}^{d}\alpha_{ij}^{(\mu)}(\omega_\mu), \quad g(t,\omega) = g^{(d+1)}(t)\prod_{\mu=1}^{d} g^{(\mu)}(\omega_\mu),$$
$$\beta_i(t,\omega) = \beta_i^{(d+1)}(t)\prod_{\mu=1}^{d}\beta_i^{(\mu)}(\omega_\mu), \quad u_0(\omega) = \prod_{\mu=1}^{d} u_0^{(\mu)}(\omega_\mu),$$
$$\gamma(t,\omega) = \gamma^{(d+1)}(t)\prod_{\mu=1}^{d}\gamma^{(\mu)}(\omega_\mu).$$

Here, $\alpha = [\alpha_{ij}]_{i,j=1,\dots,d}$ and $\beta = (\beta_i)_{i=1,\dots,d}$ as well as $\omega = (\omega_1,\dots,\omega_d)^T \in \mathbb{R}^d$. Actually a finite sum of separable functions suffices. For the convenience of the reader we use one summand only. The tensor basis of the ansatz space $\mathbb{X}$ on $\Omega = \Omega^{(1)} \times \cdots \times \Omega^{(d)}$ is given by

$$\{e_{i_{d+1}} \otimes \phi_{i_d}^{(d)} \otimes \cdots \otimes \phi_{i_1}^{(1)} \,:\, i_{d+1} = 0,\dots,\mathcal{K}^E, i_\mu = 1,\dots,\mathcal{J}_\mu, \mu = 1,\dots,d\}$$

The basis for the test space $\mathbb{Y}$ is given by

$$\{(f_{j_{d+1}} \otimes \phi_{i_d}^{(d)} \otimes \cdots \otimes \phi_{i_1}^{(1)}, 0), (0, \phi_{i_d}^{(d)} \otimes \cdots \otimes \phi_{i_1}^{(1)}) \,:\, j_{d+1} = 0, \ldots, \mathcal{K}^F, i_\mu = 1, \ldots, \mathcal{J}_\mu, \mu = 1, \ldots, d\}.$$

Remark 4.3.1. We distinguish between $\mathcal{K}^E$ and $\mathcal{K}^F$ as we explicitly allow not only for the temporal discretisation introduced in Section 2.2.2.2 but for the minimal residual approach presented in Section 2.2.2.3 as well. There, the basis elements $f_{j_{d+1}}$ are linear hat functions on the refined discrete time interval and $\mathcal{K}^F = 2\mathcal{K}^E$ in that situation.

We define the mass matrices (for $\mu = 1, \ldots, d$)

$$[\mathbf{M}_e^{(d+1)}]_{ij} = \int_I e_i(t)e_j(t)dt, \quad [\mathbf{M}_f^{(d+1)}]_{ij} = \int_I f_i(t)f_j(t)dt \text{ and } [\mathbf{M}^{(\mu)}]_{ij} = \int_{\Omega^{(\mu)}} \phi_i^{(\mu)}(x)\phi_j^{(\mu)}(x)dx.$$

We add 'e' and 'f' to distinguish between temporal related matrices of ansatz and test space. We now state exemplarily the decomposition of the diffusive operator of the left-hand side in its tensor product components:

$$\begin{aligned}\mathbf{B}_{\text{diff}}(x) = &\sum_{\mu=1}^{d} \mathbf{M}_{\mu,\mu}^{(d+1)} \otimes \mathbf{M}_{\mu,\mu}^{(d)} \otimes \cdots \otimes \mathbf{M}_{\mu,\mu}^{(\mu+1)} \otimes \mathbf{A}^{(\mu)} \otimes \mathbf{M}_{\mu,\mu}^{(\mu-1)} \otimes \cdots \otimes \mathbf{M}_{\mu,\mu}^{(1)} \\ &+ \sum_{\substack{\mu,\mu'=1 \\ \mu\neq\mu'}}^{d} \mathbf{M}_{\mu,\mu'}^{(d+1)} \otimes \mathbf{M}_{\mu,\mu'}^{(d)} \otimes \cdots \otimes \mathbf{M}_{\mu,\mu'}^{(\mu+1)} \otimes \mathbf{V}_{\mu'}^{(\mu)} \otimes \mathbf{M}_{\mu,\mu'}^{(\mu-1)} \otimes \cdots \\ &\cdots \otimes \mathbf{M}_{\mu,\mu'}^{(\mu'+1)} \otimes \tilde{\mathbf{V}}_{\mu}^{(\mu')} \otimes \mathbf{M}_{\mu,\mu'}^{(\mu'-1)} \otimes \cdots \otimes \mathbf{M}_{\mu,\mu'}^{(1)}.\end{aligned}$$

Here, $\mathbf{M}_\mu^{(k)}$ are the mass matrices defined above but inducing now the respective operator coefficient for $k \leq d$: $[\mathbf{M}_{\mu,\mu'}^{(k)}]_{ij} = \int_{\Omega^{(\mu)}} \alpha_{\mu',\mu}^{(k)}(x)\phi_i^{(k)}(x)\phi_j^{(k)}(x)dx$. In addition, $[\widehat{\mathbf{M}}_{\mu,\mu'}^{(d+1)}]_{ij} = \int_I \alpha_{\mu,\mu'}^{(d+1)}(x)f_i(t)e_i(t)dt$, $\mathbf{M}_{\mu,\mu'}^{(d+1)} = \begin{pmatrix}\widehat{\mathbf{M}}^{(d+1)} \\ 0\end{pmatrix}$, $[\mathbf{A}^{(\mu)}]_{ij} = \int_{\Omega^{(\mu)}} \alpha_{\mu,\mu}^{(\mu)}\nabla\phi_i^{(\mu)}(x)\nabla\phi_j^{(\mu)}(x)dx$, $[\mathbf{V}_{\mu'}^{(\mu)}]_{ij} = \int_{\Omega^{(\mu)}} \alpha_{\mu',\mu}^{(\mu)}(x)\nabla\phi_i^{(\mu)}(x)\phi_j^{(\mu)}(x)dx$ and $[\tilde{\mathbf{V}}_{\mu'}^{(\mu)}]_{ij} = \int_{\Omega^{(\mu)}} \alpha_{\mu,\mu'}^{(\mu)}(x)\phi_i^{(\mu)}(x)\nabla\phi_j^{(\mu)}(x)dx$.

The right-hand side is given by

$$\mathbf{f} = \mathbf{f}^{(d+1)} \otimes \mathbf{f}^{(d)} \otimes \cdots \mathbf{f}^{(1)} + \mathbf{h}^{(d+1)} \otimes \mathbf{h}^{(d)} \otimes \cdots \otimes \mathbf{h}^{(1)}.$$

Where $(\mathbf{f}^{(\mu)})_i = \int_{\Omega^{(\mu)}} g^{(\mu)}(x)\phi_i(x)dx$, $(\mathbf{h}^{(\mu)})_i = \int_{\Omega^{(\mu)}} u_0^{(\mu)}(x)\phi_i(x)dx$, $\mathbf{f}^{(d+1)} = \begin{pmatrix}(\int_I g^{(d+1)}(t)f_i(x)dt)_{i=0,\ldots,\mathcal{K}^F} \\ 0\end{pmatrix}$, $\mathbf{h}^{(d+1)} = \begin{pmatrix}(0,\ldots,0)^T \\ 1\end{pmatrix}$. The CP-decomposition of the operators allows for the storage in H-Tucker format. For its usage in the reduced basis method, the affine decomposition has to be taken into account. If $\mathbf{B} = \sum_{q=1}^{Q_b} \theta_q^b(\mu)b_q(u,v)$ is the affine decomposition with respect to the parameter μ, each component $b_q(u,v)$ again has its tensor

decomposition as described above. The combination of both leads to a double sum

$$\mathbf{B} = \sum_{q=1}^{Q_b} \theta_q^b(\mu) \sum_{i=1}^{T_b^{(q)}} B_i^{q,(d+1)} \otimes \cdots \otimes B_i^{q,(1)}$$

where we collect the varying tensor components of the respective operator components in the shortened notation $B_i^{q,(j)}$. For the solution and the test function we assume tensorised basis functions and receive $u(x) = u^1(x_1)u^2(x_2)$ as well as $v(x) = v^1(x_1)v^2(x_2)$ for $x = (x_1, x_2) \in \mathbb{R}^2$. The H-Tucker rank possibly increases when we solve the system for a given parameter by reconstructing the operator out of precomputed components:

Example 4.3.2. Consider the following diffusion $\nabla \cdot \begin{pmatrix} \mu x_1 + 2 & 0 \\ 0 & 0 \end{pmatrix} \nabla u(t,x)$ for $x \in \Omega \subset \mathbb{R}^2$. Obviously $\alpha = \begin{pmatrix} \mu x_1 + 2 & 0 \\ 0 & 0 \end{pmatrix}$ is separable as required. The weak formulation for that part details in

$$\int_\Omega \left(\begin{pmatrix} \mu x_1 + 2 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} u^1_{x_1}(x_1)u^2(x_2) \\ u^1(x_1)u^2_{x_2}(x_2) \end{pmatrix} \right)^T \begin{pmatrix} v^1_{x_1}(x_1)v^2(x_2) \\ v^1(x_1)v^2_{x_2}(x_2) \end{pmatrix} dx$$
$$= \int_{\Omega_1} (\mu x_1 + 2) u^1_{x_1}(x_1)v^1_{x_1}(x_1)dx_1 \int_{\Omega_2} u^2(x_2)v^2(x_2)dx_2.$$

This part would lead to an H-Tucker representation of rank one. Because of the affine decomposition we have to separate the integral into two parts, to be able to separate the actual parameter μ from the integral that has to be precomputed offline:

$$\mu \int_{\Omega_1} x_1 u^1_{x_1}(x_1)v^1_{x_1}(x_1)dx_1 \int_{\Omega_2} u^2(x_2)v^2(x_2)dx_2 + 2 \int_{\Omega_1} x_1 u^1_{x_1}(x_1)v^1_{x_1}(x_1)dx_1 \int_{\Omega_2} u^2(x_2)v^2(x_2)dx_2.$$

Storing the two components separately in H-Tucker format, the reconstructed operator has an increased rank of two.

Note that the application of the BPX-preconditioners introduced in Section 4.2.3 in the normal equation $\mathbf{B}^T\mathbf{N}^{-1}\mathbf{B} = \mathbf{B}^T\mathbf{N}^{-1}\mathbf{f}$ further increases the rank. Of course the higher the ranks get the less efficient is the representation in low rank tensor format. We come back to this point in our numerical experiments.

4.3.2. Offline Phase with H-Tucker

There are two main reasons to use the H-Tucker format in the offline phase:

- the computational effort is reduced

- higher space dimensions can be achieved.

In this section we present two approaches.

The first approach is mainly to handle the first point for 1D or 2D problems. The Crank Nicolson time stepping scheme is used for the actual solution process. The solution $u^{\mathcal{N}} \in \mathbb{R}^{\mathcal{J}_1 \times \cdots \times \mathcal{J}_d \times \mathcal{K}^E}$ is transformed in H-Tucker format via truncation. All further precomputations and the greedy procedure are performed in low rank tensor format.

The second approach is appropriate for higher dimensional problems. The minimal residual problem is to be solved. The operators are given in H-Tucker format and an iterative solver is used for the solution process. The solution is a tensor in H-Tucker format. Again all precomputations as well as the greedy search are performed in low rank tensor format.

Both approaches have one essential drawback. The snapshot reproducibility fails in general if we apply the truncation operator as well as if we use an iterative solver. This has to taken care of in the greedy procedure. In both cases, the approximation error between the detailed solution $u^{\mathcal{N}} \in \mathbb{X}^{\mathcal{N}}$ and its H-Tucker representative $u^{\mathcal{T}} \in \mathcal{H}$ has to be bounded and the greedy tolerance chosen accordingly. Further, the operator that corresponds to the originally used norm $\|\cdot\|_{\mathbb{X}}$ (resp. $\|\cdot\|_{\mathbb{X},\mathrm{bar}}$, cf. Equation (2.2.2) and (2.2.20)) is not given in CP decomposition. Same holds for the dual norm operator required for the residual. We therefore replace $\|\cdot\|_{\mathbb{X}}$ by $\|\cdot\|_{\mathcal{M}}$ and $\|\cdot\|_{\mathbb{Y}'}$ by $\|\cdot\|_{\mathcal{N}'}$. Details of the norm equivalences are subject of Section 4.3.2.1.

Additionally we introduce a slightly different version of the BPX-preconditioner – as it was proposed in [AT14, (2.26)]:

$$\mathcal{N}_{\pm}^{*}(f \otimes v, h) := (f \otimes \sum_{\ell} q_{\ell}^{\pm 2} Q_{\ell} v, h)$$

for $(f \otimes v, h) \in \mathbb{Y}^{\mathcal{N}}$. The preconditioner $\mathcal{N}^*$ is still a norm inducing operator and – in contrary to $\mathcal{N}$ – also applicable to the finite element discretisation that leads to the Crank Nicolson time stepping scheme.

4.3.2.1. Norm Equivalences

The norm equivalences have to be taken care of separately. We have three of them:

(i) $\exists\ 0 < d_{\mathcal{M}} \leq D_{\mathcal{M}}$ such that $d_{\mathcal{M}}\|\cdot\|_{\mathcal{M}} \leq \|\cdot\|_{\mathbb{X}} \leq D_{\mathcal{M}}\|\cdot\|_{\mathcal{M}}$

(ii) $\exists\ 0 < d_{\mathcal{N}} \leq D_{\mathcal{N}}$ such that $d_{\mathcal{N}}\|\cdot\|_{\mathcal{N}} \leq \|\cdot\|_{\mathbb{Y}} \leq D_{\mathcal{N}}\|\cdot\|_{\mathcal{N}}$

(iii) $\exists\ 0 < d_{\mathcal{N}^*} \leq D_{\mathcal{N}^*}$ such that $d_{\mathcal{N}^*}\|\cdot\|_{\mathcal{N}^*} \leq \|\cdot\|_{\mathbb{Y}} \leq D_{\mathcal{N}^*}\|\cdot\|_{\mathcal{N}^*}$.

Notice further [And14, Eq. (35)], that the left-hand side given by the normal equation $\mathbf{B}^T\mathbf{N}^{-1}\mathbf{B}$, 2.3.1 (3), is a symmetric positive-definite operator on $\mathbb{X}^{\mathcal{N}}$ that satisfies

(iv) $\beta^{\mathcal{N}} d_{\mathcal{M}} d_{\mathcal{N}} \| \cdot \|_{\mathcal{M}} \leq \| \cdot \|_{\mathbf{B}^T\mathbf{N}^{-1}\mathbf{B}} \leq C_b D_{\mathcal{M}} D_{\mathcal{N}} \| \cdot \|_{\mathcal{M}}$.

The actual value of the constants depend on the choice of the Hilbert spaces V and H. The value is deduced from the norm equivalences stated in Equation (4.2.7) and the corresponding estimates for $\mathcal{N}$ given in [And12, Sec. 6.2.1]. We receive

$$\begin{aligned} d_{\mathcal{M}} &= \min\{d_{L_2(I)} d_V, d_{H^1(I)} d_{V'}\}, & D_{\mathcal{M}} &= \max\{D_{L_2(I)} D_V, D_{H^1(I)} D_{V'}\} \\ d_{\mathcal{N}} &= \min\{d_{L_2(I)} d_V, d_H\}, & D_{\mathcal{N}} &= \max\{D_{L_2(I)} D_V, D_H\} \\ d_{\mathcal{N}^*} &= \min\{d_V, d_H\}, & D_{\mathcal{N}^*} &= \max\{D_V, D_H\}. \end{aligned}$$

Remark 4.3.3. The improved quasi-optimality estimate of Theorem 2.2.23 stated in Remark 2.2.25, still holds true in $\| \cdot \|_{\mathcal{M}}$,

$$\|u - u^{\mathcal{N}}\|_{\mathcal{M}} \leq \frac{C_b^{\mathcal{N}}}{\beta^{\mathcal{N}}} \frac{D_{\mathcal{N}} D_{\mathcal{M}}}{d_{\mathcal{N}} d_{\mathcal{M}}} \inf_{w^{\mathcal{N}} \in \mathbb{X}^{\mathcal{N}}} \|u - w^{\mathcal{N}}\|_{\mathcal{M}}.$$

A change of norms further influences the error bound since

$$\|u - u^{\mathcal{N}}\|_{\mathcal{M}} \leq \frac{1}{\beta^{\mathcal{N}}} \frac{1}{d_{\mathcal{N}} d_{\mathcal{M}}} \|b(u, \cdot) - f(\cdot)\|_{\mathcal{N}'}.$$

In particular, $\|u - u^{\mathcal{N}}\|_{\mathbb{X}} \leq \frac{1}{\beta^{\mathcal{N}}} \frac{1}{d_{\mathcal{N}}} \|b(u, \cdot) - f(\cdot)\|_{\mathcal{N}'}$. If the inf-sup constant is directly computed for $\| \cdot \|_{\mathcal{M}}$ and $\| \cdot \|_{\mathcal{N}}$, we can simply measure the error by $\| \cdot \|_{\mathcal{M}}$. Using $\mathcal{N}^*$ instead of $\mathcal{N}$ gives the same results for the constants $d_{\mathcal{N}^*}$ and $D_{\mathcal{N}^*}$ instead of $d_{\mathcal{N}}$ and $D_{\mathcal{N}}$ respectively.

4.3.2.2. Crank Nicolson Scheme with Truncation

As long as the space dimensions allow, we can further compute the solution using the Crank Nicolson time stepping scheme. The H-Tucker format is than used to compute the residual norms and to perform all required precomputations in the RB offline phase. In this case we might be interested in the originally introduced norms so the norm equivalences have to be taken into account.

In Algorithm 4.3.1 the steps of the offline phase are detailed using the Crank Nicolson time stepping scheme with transformation of the solution into H-Tucker format.

Algorithm 4.3.1 Greedy Algorithm with CN Scheme and Truncation

Input: $\mathcal{D}^{\text{train}} \subset \mathcal{D}$ training set, tol > 0 RB target tolerance.
Output: $\Xi_{\mathbb{X}}^N$ RB basis.

1: For $\mu_1 \in \mathcal{D}^{\text{train}}$ compute $u^{\mathcal{N}}(\mu_1) \to \mathcal{T}(u^{\mathcal{N}}(\mu_1)) = u^{\mathcal{T}}(\mu_1) = u^1$, $\Xi_{\mathbb{X}}^1 = \{u^1\}$.
2: **while** $\max_{\mu \in \mathcal{D}^{\text{train}}} \Delta_N(\mu) >$ tol **do**
3: Compute $u^{\mathcal{N}}(\mu_i)$ for maximising μ_i.
4: $u^i := \mathcal{T}(u^{\mathcal{N}}(\mu_i))$.
5: $\Xi_{\mathbb{X}}^i := \Xi_{\mathbb{X}}^{i-1} \cup \{u^i\}$.
6: **end while**
7: $N := i$.

Truncation Truncation can be performed regarding an absolute or a relative error bound, [Tob12, Rem. 3.6]:

$$\|x - x^{\mathcal{T}}\|_2 \leq \epsilon_{\text{abs}}$$
$$\|x - x^{\mathcal{T}}\|_2 \leq \epsilon_{\text{rel}} \|x\|_2.$$

Here, $\|\cdot\|_2$ denotes the Frobenius norm. For $t \in \mathcal{T}_d \setminus \{t_{\text{root}}\}$ the rank r_t of $x^{\mathcal{T}}$ is chosen such that $\sqrt{\sum_{i=r_t+1}^{n_t} \sigma_{t,i}^2} \leq \frac{\epsilon_{\text{abs}}}{\sqrt{2d-3}}$ respectively $\leq \frac{\epsilon_{\text{rel}}\|x\|_2}{\sqrt{2d-3}}$. The quantities $\sigma_{t,i}$ are the singular values of the t-mode matricisation $x^{(t)}$ of x, cf. (4.2.3).

Orthonormalisation We have seen in the numerical experiment in Section 2.5 that orthonormalisation is important for numerical stabilisation. A standard Gram Schmidt algorithm requires a ridge walk between truncation and orthonormality, [Hac12b, Sec. 13.3.6]. We follow the same strategy as presented in Section 3.2.2.4 for the online phase. We compute the Gramian $[(u_{\mathcal{T}}^i, u_{\mathcal{T}}^j)_{\mathbb{X}}]_{i,j=1,\dots,N}$ for the reduced basis $\{u^1, \dots, u^N\}$ and apply an SVD. Here, the resulting preconditioner for the RB system is applied directly to the operator components in the offline phase. Online, we just transform the coefficients back such that the solution u^N is given w.r.t. the non-orthonormalised RB basis.

Greedy Tolerance Even though we could in theory approximate the Crank Nicolson solution $u^{\mathcal{N}} \in \mathbb{X}^{\mathcal{N}}$ exactly as tensor, the resulting ranks get infeasibly large. By truncation of $u^{\mathcal{N}}$ to $u^{\mathcal{T}}$ the Petrov-Galerkin orthogonalisation is not valid any more, i.e. the already mentioned snapshot reproducibility fails as $b(u^{\mathcal{T}}, v; \mu) \neq f(v)$ for some $v \in \mathbb{Y}^{\mathcal{N}}$. To receive an accurately working greedy procedure the greedy tolerance has to be chosen very carefully in accordance to the truncation accuracy.

Proposition 4.3.4. For a given parameter $\mu \in \mathcal{D} \subset \mathbb{R}^d$, $d \in \mathbb{N}$, let $u^{\mathcal{N}}(\mu) \in \mathbb{X}^{\mathcal{N}}$ be the detailed solution, solving

$$b(u, v; \mu) = f(v; \mu) \quad \forall\ v \in \mathbb{Y}^{\mathcal{N}} \tag{4.3.1}$$

for $b \in \mathcal{L}(\mathbb{X}, \mathbb{Y}')$ and $f \in \mathbb{Y}'$. Let $u^{\mathcal{T}}(\mu) \in \mathbb{X}_N$ be the corresponding reduced basis element, $\mathcal{T}(u^{\mathcal{N}}(\mu)) = u^{\mathcal{T}}(\mu) \in \mathcal{H}$. Let the discrete problem formulation be well-posed, i.e., the inf-sup condition holds true. Denote by $\epsilon^{\mathbb{X}}(\mu)$ the truncation error measured in the norm $\|\cdot\|_{\mathbb{X}}$, i.e.,

$$\|u^{\mathcal{N}}(\mu) - u^{\mathcal{T}}(\mu)\|_{\mathbb{X}} \leq \epsilon^{\mathbb{X}}(\mu).$$

Then, the error in the greedy procedure is bounded as follows

$$\|u_N(\mu) - u^{\mathcal{N}}(\mu)\|_{\mathbb{X}} \leq \frac{C_b^{\mathcal{N}}(\mu)}{\beta^{\mathcal{N}}(\mu)} \epsilon^{\mathbb{X}}(\mu).$$

Proof. In a first step we use the best approximation result of the Petrov-Galerkin projection. As $u^{\mathcal{T}}(\mu)$ is in $\mathbb{X}_N$ the claim follows:

$$\|u_N(\mu) - u^{\mathcal{N}}(\mu)\|_{\mathbb{X}} \leq \frac{C_b^{\mathcal{N}}(\mu)}{\beta^{\mathcal{N}}(\mu)} \inf_{u \in \mathbb{X}_N} \|u - u^{\mathcal{N}}(\mu)\|_{\mathbb{X}} \leq \frac{C_b^{\mathcal{N}}(\mu)}{\beta^{\mathcal{N}}(\mu)} \|u^{\mathcal{T}}(\mu) - u^{\mathcal{N}}(\mu)\|_{\mathbb{X}}.$$

□

Corollary 4.3.5. *With the same assumptions, the standard error estimator* $\Delta_N(\mu) = \frac{1}{\beta^{\mathcal{N}}} \|b(u_N(\mu), \cdot; \mu) - f(\cdot; \mu)\|_{\mathbb{Y}'}$ *is bounded by*

$$\Delta_N(\mu) \leq \left(\frac{C_b^{\mathcal{N}}(\mu)}{\beta^{\mathcal{N}}(\mu)} \right)^2 \epsilon^{\mathbb{X}}(\mu).$$

Proof. We consider the dual norm of the residual only. The first inequality is given by boundedness of the bilinear form $b(\cdot, \cdot)$, the second is the application of Proposition 4.3.4.

$$\|b(u_N(\mu), \cdot; \mu) - f(\cdot; \mu)\|_{(\mathbb{Y}^{\mathcal{N}})'} \leq C_b^{\mathcal{N}}(\mu) \|u_N(\mu) - u^{\mathcal{N}}(\mu)\|_{\mathbb{X}} \leq \frac{(C_b^{\mathcal{N}})^2(\mu)}{\beta^{\mathcal{N}}(\mu)} \epsilon^{\mathbb{X}}(\mu).$$

□

With the presented estimates the greedy tolerance and the truncation tolerance can be chosen accordingly. Performing e.g. a greedy procedure with given tolerance `tol` using the standard norms $\|\cdot\|_{\mathbb{X}}$ and $\|\cdot\|_{(\mathbb{Y}^{\mathcal{N}})'}$, we have to ensure that $\epsilon^{\mathbb{X}}(\mu) \geq \|u^{\mathcal{N}}(\mu) - u^{\mathcal{T}}(\mu)\|_{\mathbb{X}}$ satisfies

$$\texttt{tol} \geq \frac{C_b^{\mathcal{N}}(\mu)}{\beta_N(\mu)} \epsilon^{\mathbb{X}}(\mu) \quad \Longleftrightarrow \quad \epsilon^{\mathbb{X}}(\mu) \leq \frac{\beta_N(\mu)}{C_b^{\mathcal{N}}(\mu)} \texttt{tol}.$$

Remark 4.3.6. As the associated Riesz operators of $\|\cdot\|_{\mathbb{X}}$ and $\|\cdot\|_{\mathbb{Y}}$ are not available in CP decomposition, a change of norms is required for an efficient computation. For the error measured in $\|\cdot\|_{\mathcal{M}}$ the truncation accuracy is given by $\epsilon^{\mathcal{M}}(\mu) = \frac{\beta_N(\mu)}{\gamma^{\mathcal{N}}(\mu)} \frac{d^{\mathcal{M}}}{D^{\mathcal{M}}} \texttt{tol}$.

Given a tolerance for a weak greedy procedure using the error estimator $\Delta_N(\mu)$, the truncation has to be performed w.r.t. a very small tolerance because of the appearance of the square in the estimate.

As we have the choice what kind of truncation tolerance we consider in the first place, application of the following corollary might be more sensible.

Corollary 4.3.7. *Let the assumptions of Proposition 4.3.4 be valid. For* $\|u^{\mathcal{N}} - u^{\mathcal{T}}\|_{\mathcal{M}} \leq \epsilon^{\mathcal{M}}(\mu)$ *the true error in* $\|\cdot\|_{\mathbb{X}}$ *is bounded by*

$$\|u_N(\mu) - u^{\mathcal{N}}(\mu)\|_{\mathbb{X}} \leq \frac{C_b^{\mathcal{N}}(\mu)}{\beta^{\mathcal{N}}(\mu)} \frac{1}{d_{\mathcal{M}}} \epsilon^{\mathcal{M}}(\mu).$$

□

Of course the operator $\mathcal{N}$ can be replaced by $\mathcal{N}^*$ in all considerations.

4.3.3. Minimal Residual Approach with Iterative Solver

The second possibility is to use an iterative solver that directly solves the parabolic PDE in low rank tensor format. For higher space dimensions this is mandatory as the Crank Nicolson scheme separates the temporal and spatial dimensions only. The solution is approximately given w.r.t. some prefixed tolerance and the snapshot reproducibility fails in the greedy search. A main task is to adjust the iterative solver such that the tensor ranks stay feasibly small and the quality of the approximated solution is suitable for the application in the greedy search.

Solving linear systems in H-Tucker format was subject of [Tob12, Sec. 4] where further information on low rank iterative solvers is provided. The method of choice is the low rank tensor variant of the CGNR method proposed in [Tob12, Sec. 5]. The CGNR method is the Conjugate Gradient (CG) method applied on the system of normal equation

$$\mathcal{A}^T \mathcal{A} x = \mathcal{A}^T b$$

associated to the linear system $\mathcal{A}x = b$ for non-symmetric $\mathcal{A}$, cf. [Saa03, Ch. 8]. The algorithm is adapted such that the truncation operations are taken into account, cf. Algorithm 4.3.2. The residual $\|\mathcal{B}u^{\mathcal{T}} - f\|_{\mathcal{N}^{-1}}$ is computed in every iteration step and the algorithm stops if it achieves the desired tolerance. As the iterative method improves the solution in every time step by adding the new information, the tensor rank of the solution grows in every step. The truncation operator has to be applied, cf. line 4 in Algorithm 4.3.2. A truncation tolerance that is chosen too large may prevent a convergence towards the solution. In contrary, large ranks become unfeasible at some point. Applying the preconditoners increases the ranks again. In the actual computations,

Algorithm 4.3.2 Preconditioned CGNR for tensors

Input: $\mathcal{B}$, f, preconditioner $\mathcal{M}^{-1}$, $\mathcal{N}^{-1}$, tolerance tol.
Output: Solution $u^{\mathcal{T}}$ in H-Tucker format.

$u_0 = 0$, $R_0 = f$, $S_0 = \mathcal{B}^T(\mathcal{N}^{-1}(R_0))$, $P_0 = \mathcal{M}^{-1}(S_0)$, $\gamma_0 = \langle \mathcal{B}(P_0), \mathcal{B}(P_0)\rangle_{\mathcal{N}^{-1}}$, k = 0.
while $\|\tilde{R}_0\|_{\mathcal{N}^{-1}} >$ tol **do**
  $\alpha_k = \langle S_k, P_k\rangle / \gamma_k$
  $\tilde{u}_{k+1} = u_k + \alpha_k P_k, \quad u_{k+1} \leftarrow \mathcal{T}(\tilde{u}_k)$
  $\tilde{R}_{k+1} = f - \mathcal{B}u_{k+1}, \quad R_{k+1} \leftarrow \mathcal{T}(\tilde{R}_{k+1})$
  $\tilde{S}_{k+1} = \mathcal{B}^T(\mathcal{N}^{-1}(R_{k+1})), \quad S_{k+1} \leftarrow \mathcal{T}(\tilde{S}_{k+1})$
  $Z_{k+1} = \mathcal{M}^{-1}(S_{k+1})$
  $\beta_{k+1} = -\langle \mathcal{B}(Z_{k+1}), \mathcal{B}(P_k)\rangle_{\mathcal{N}^{-1}} / \gamma_k$
  $\tilde{P}_{k+1} = Z_{k+1} + \beta_{k+1} P_k, \quad P_{k+1} \leftarrow \mathcal{T}(\tilde{P}_{k+1})$
  $\gamma_{k+1} = \langle \mathcal{B}(P_{k+1}, \mathcal{B}(P_{k+1})\rangle_{\mathcal{N}^{-1}}$
  $k = k + 1$
end while
$u^{\mathcal{T}} := u_k$.

the preconditioners are approximated by operators of lower rank. We evaluate the error in $\|\cdot\|_{\mathcal{M}}$ and the residual in $\|\cdot\|_{\mathcal{N}'}$.

Proposition 4.3.8. Recall the setting of Proposition 4.3.4 with generalised test space $\mathbb{Y}^{\widetilde{\mathcal{N}}}$ and $\mathcal{N} \neq \widetilde{\mathcal{N}}$. Let $\|u^{\mathcal{T}}(\mu) - u^{\mathcal{N}}(\mu)\|_{\mathcal{B}^T\mathcal{N}^{-1}\mathcal{B}} \leq \epsilon$, then

$$\|u_N(\mu) - u^{\mathcal{N}}(\mu)\|_{\mathcal{M}} \leq \frac{C_b^{\mathcal{N}}(\mu)}{(\beta^{\mathcal{N}}(\mu))^2}\epsilon.$$

Here, the continuity constant $C_b^{\mathcal{N}}(\mu) := C_b^{(\mathcal{N},\widetilde{\mathcal{N}})}(\mu)$ and the inf-sup constant $\beta^{\mathcal{N}}(\mu) := \beta^{(\mathcal{N},\widetilde{\mathcal{N}})}(\mu)$ are those w.r.t. $\|\cdot\|_{\mathcal{M}}$ and $\|\cdot\|_{\mathcal{N}^{-1}}$.

Proof. Analogous to the proof of Proposition 4.3.4 we receive,

$$\|u_N(\mu) - u^{\mathcal{N}}(\mu)\|_{\mathcal{M}} \leq \frac{C_b^{\mathcal{N}}(\mu)}{\beta^{\mathcal{N}}(\mu)}\|u^{\mathcal{T}}(\mu) - u^{\mathcal{N}}(\mu)\|_{\mathcal{M}}$$

□

Corollary 4.3.9. *Recall the setting of Proposition 4.3.8. Then*

$$\|u_N(\mu) - u^{\mathcal{N}}(\mu)\|_{\mathcal{B}^T\mathcal{N}^{-1}\mathcal{B}} \leq \left(\frac{C_b^{\mathcal{N}}(\mu)}{\beta^{\mathcal{N}}(\mu)}\right)^2 \epsilon.$$

Proof. As before,

$$\begin{aligned}\|u_N - u^{\mathcal{N}}\|_{\mathcal{B}^T\mathcal{N}^{-1}\mathcal{B}} &\leq C_b^{\mathcal{N}}(\mu)\|u_N(\mu) - u^{\mathcal{N}}(\mu)\|_{\mathcal{M}} \leq \frac{C_b^{\mathcal{N}}(\mu)^2}{\beta^{\mathcal{N}}(\mu)}\|u^{\mathcal{T}}(\mu) - u^{\mathcal{N}}(\mu)\|_{\mathcal{M}} \\ &\leq \frac{C_b^{\mathcal{N}}(\mu)^2}{\beta^{\mathcal{N}}(\mu)^2}\|u^{\mathcal{T}}(\mu) - u^{\mathcal{N}}(\mu)\|_{\mathcal{B}^T\mathcal{N}^{-1}\mathcal{B}}.\end{aligned}$$

□

For given greedy tolerance `tol` w.r.t. the residual norm, the CGNR solver has to satisfy the approximation tolerance $\epsilon \leq \left(\frac{\beta^{\mathcal{N}}(\mu)}{C_b^{\mathcal{N}}(\mu)}\right)^2$ `tol`.

4.4. Numerical Experiments

We construct a reduced basis for a 2D diffusion-convection-reaction problem using the H-Tucker storage format in the offline phase. All matrix-matrix, matrix-vector and scalar multiplications with matrices are performed in H-Tucker format using the provided functions of the H-Tucker toolbox [KT12], e.g. `apply_mat_to_mat` and `apply_mat_to_vec`. Same holds for addition and (scalar) multiplication of tensors in H-Tucker format. For tensor truncation the function `truncate_std` is used – provided again in the H-Tucker Toolbox.

4.4.1. Problem Formulation

We solve the following problem:

$$\begin{aligned}\dot{u} - \alpha\Delta u + \beta\nabla u + \gamma u = 0, \quad &\text{on } I \times \Omega = [0, 0.3] \times (0, 1)^2 \\ u(0, \omega) = \sin(2\pi\omega_1)\sin(2\pi\omega_2) \quad &\forall \omega = (\omega_1, \omega_2) \in \Omega \\ u = 0 \quad &\text{on } I \times \partial\Omega.\end{aligned}$$

The parameter domain is given by $\mu = (\alpha, \beta, \gamma) \in \mathcal{D} = [0.5, 1.5] \times [0, 1] \times [0, 1]$. The space-time variational formulation is well-posed for $V = H_0^1(\Omega)$, $H = L_2(\Omega)$ as $\alpha > 0$. We apply a standard finite element discretisation in space, using linear hat functions in both spatial directions. In the temporal domain we use linear hat functions in both, ansatz and test space. The grid used for the temporal test space is refined w.r.t. the grid of the temporal ansatz space. The discrete system is inf-sup stable, cf. Section 2.2.2.3.

4.4.2. Truth Solver

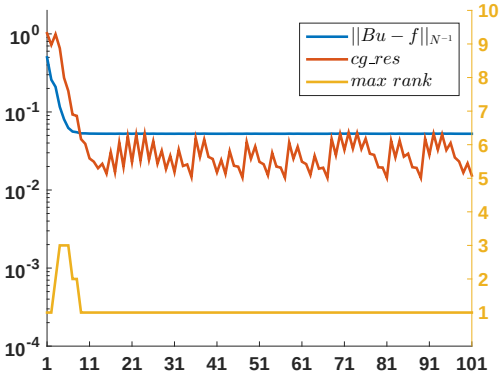
We apply the presented CGNR algorithm as truth solver. The truncation tolerance as well as the space and time discretisation level have an impact on the solver convergence. Consider the parameter $\mu = (1, 0, 0) \in \mathcal{D}$. The CP decomposition of the left-hand side operator has three summands. The first covers the time derivative of the solution, the two other belong to the diffusion. The corresponding H-Tucker representation is therefore of rank 3. We solve the PDE for different scenarios given in Table 4.4.1. We run all scenarios for 100 iterations without restriction on a stopping tolerance ϵ to compare the achieved solution qualities. The operator $\mathcal{M}^{-1}$ is represented by a truncated H-Tucker representation as proposed in [AT14, Sec. 3.3]. We do not truncate the H-Tucker representation of the operator $\mathcal{N}^{-1}$ as this is the residual norm measuring the error in the reduced basis offline phase. The CGNR residual $\|\mathcal{B}\mathcal{N}^{-1}\mathcal{B}u - \mathcal{B}\mathcal{N}^{-1}f\|_{\mathcal{M}^{-1}}$ (cg-norm) is monitored additionally as it indicates possible reasons for stagnation in the convergence. In Scenario 1, the convergence error of the CGNR algorithm

Scenario 1	$\mathcal{J}_1 = \mathcal{J}_2 = 2^3$	$\mathcal{K}^E = 2^3, \mathcal{K}^F = 2^4$	$\epsilon_{\text{rel}} = 10^{-2}$	max_rank = 50
Scenario 2	$\mathcal{J}_1 = \mathcal{J}_2 = 2^3$	$\mathcal{K}^E = 2^3, \mathcal{K}^F = 2^4$	$\epsilon_{\text{rel}} = 10^{-6}$	max_rank = 50
Scenario 3	$\mathcal{J}_1 = \mathcal{J}_2 = 2^6$	$\mathcal{K}^E = 2^6, \mathcal{K}^F = 2^7$	$\epsilon_{\text{rel}} = 10^{-2}$	max_rank = 50
Scenario 4	$\mathcal{J}_1 = \mathcal{J}_2 = 2^6$	$\mathcal{K}^E = 2^6, \mathcal{K}^F = 2^7$	$\epsilon_{\text{rel}} = 10^{-6}$	max_rank = 50
Scenario 5	$\mathcal{J}_1 = \mathcal{J}_2 = 2^9$	$\mathcal{K}^E = 2^9, \mathcal{K}^F = 2^10$	$\epsilon_{\text{rel}} = 10^{-2}$	max_rank = 50
Scenario 6	$\mathcal{J}_1 = \mathcal{J}_2 = 2^9$	$\mathcal{K}^E = 2^9, \mathcal{K}^F = 2^10$	$\epsilon_{\text{rel}} = 10^{-4}$	max_rank = 50

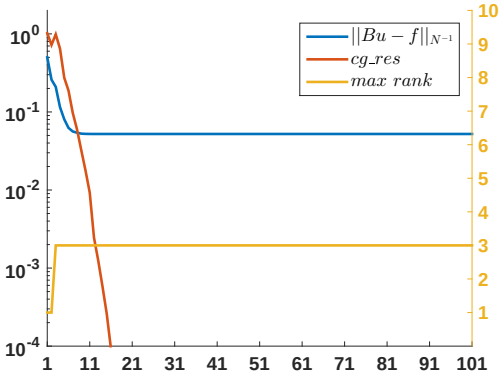
Table 4.4.1.: Scenarios 1–6 for the CGNR solver.

stagnates after about 10 iterations for both norms, the residual and the cg-norm, cf. Figure 4.4.1 (a). By increasing the relative truncation tolerance in Scenario 2, the error measured in the cg-norm decreases further down to 10^{-16}, cf. Figure 4.4.1 (b). The residual error still stagnates on the same level as in Scenario 1. As the cg-error decreases, the stagnation of the residual error is caused by the coarse spatial and temporal discretisation. The discretisation refinement in Scenario 3 brings the residual error down to 10^{-2}, cf. Figure 4.4.1 (c). The value of the cg-norm reaches a level below 10^{-3}. Increasing the truncation tolerance in Scenario 4 accelerates again the error measured in the cg-norm but does not affect the residual error, cf. Figure 4.4.1 (d). The residual error stagnates on the approximation level until the spatial and temporal discretisation is refined in Scenario 5 and 6, cf. 4.4.1 (e) and (f).

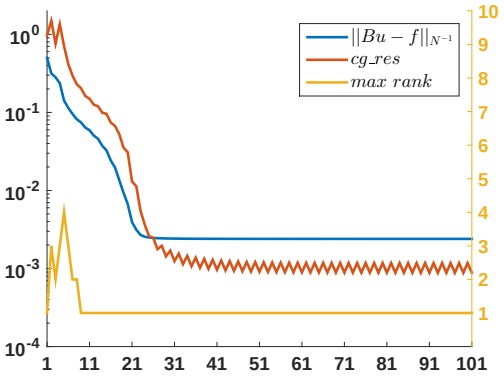
For a second parameter $\mu = (1, 0.5, 0.5)$ we repeat the computations for Scenario 3 and 5 in different variations, cf. Table 4.4.2. The H-Tucker representation of the left-hand side operator is now of tensor rank 6. Because of the higher operator rank the procedure is not computationally feasible anymore for performing Scenario 4 directly. Further, the higher rank affects the convergence of the CGNR solver: The residual error in Scenario 3 reaches again a tolerance of 10^{-2} but the absolute value of 0.0085 is worse in comparison to the results for this



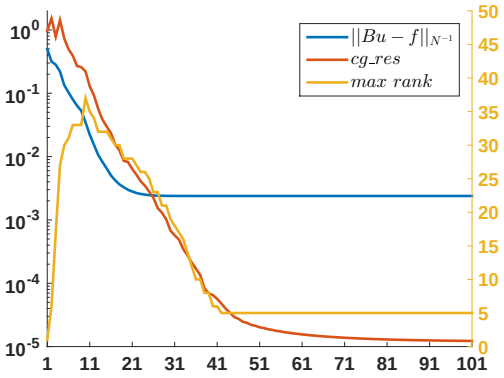
(a) Scenario 1.



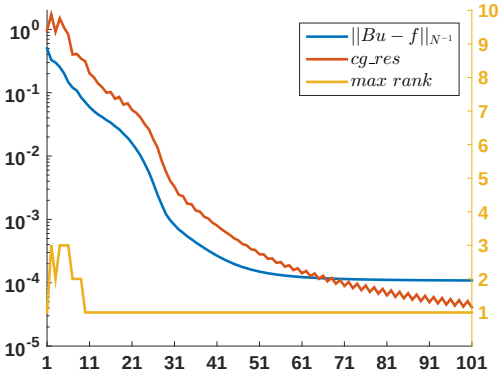
(b) Scenario 2.



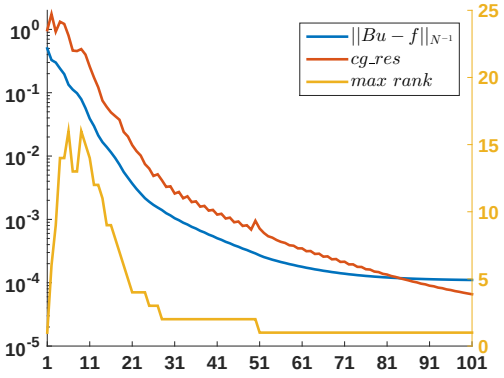
(c) Scenario 3.



(d) Scenario 4.



(e) Scenario 5.



(f) Scenario 6.

Figure 4.4.1.: Convergence of the CGNR solver.

scenario for $\mu = (1, 0, 0)$, cf. Figure 4.4.1 (c). In Scenario 3b the relative truncation tolerance is increased to 10^{-4} and the maximal tensor rank decreased to 10 at the same time. The truncation algorithm tries to reach an approximation accuracy of 10^{-4} allowing for at most a tensor of rank 10. The residual error improves, cf. Figure 5.4.5 (b). An increasing of the maximal rank to 20 in Scenario 3*c* does not significantly change the results, cf. Figure 5.4.5 (c). The finer spatial and temporal discretisation in Scenario 5 does not improve the approximation quality significantly, cf. Figure 5.4.5 (d). Again increasing the relative truncation tolerance to 10^{-4} and decreasing the maximal rank to 10 in Scenario 5b allows the residual error to go down to 10^{-3}.

Scenario 3	$\mathcal{J}_1 = \mathcal{J}_2 = 2^6$	$\mathcal{K}^E = 2^6, \mathcal{K}^F = 2^7$	$\epsilon_{\text{rel}} = 10^{-2}$	max_rank = 50
Scenario 3b	$\mathcal{J}_1 = \mathcal{J}_2 = 2^6$	$\mathcal{K}^E = 2^6, \mathcal{K}^F = 2^7$	$\epsilon_{\text{rel}} = 10^{-4}$	max_rank = 10
Scenario 3c	$\mathcal{J}_1 = \mathcal{J}_2 = 2^6$	$\mathcal{K}^E = 2^6, \mathcal{K}^F = 2^7$	$\epsilon_{\text{rel}} = 10^{-4}$	max_rank = 20
Scenario 5	$\mathcal{J}_1 = \mathcal{J}_2 = 2^9$	$\mathcal{K}^E = 2^9, \mathcal{K}^F = 2^10$	$\epsilon_{\text{rel}} = 10^{-2}$	max_rank = 50
Scenario 5b	$\mathcal{J}_1 = \mathcal{J}_2 = 2^9$	$\mathcal{K}^E = 2^9, \mathcal{K}^F = 2^10$	$\epsilon_{\text{rel}} = 10^{-2}$	max_rank = 10

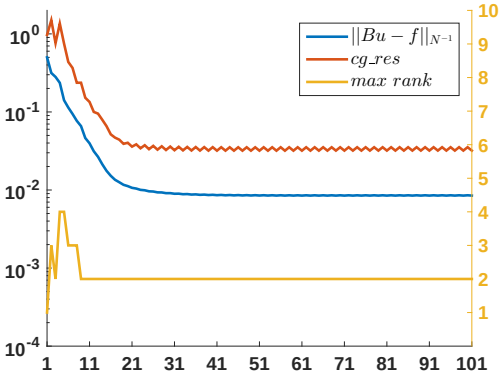
Table 4.4.2.: Scenarios 3, 3b, 3c and 5, 5b for the CGNR solver.

4.4.2.1. Greedy Procedure and Online Phase

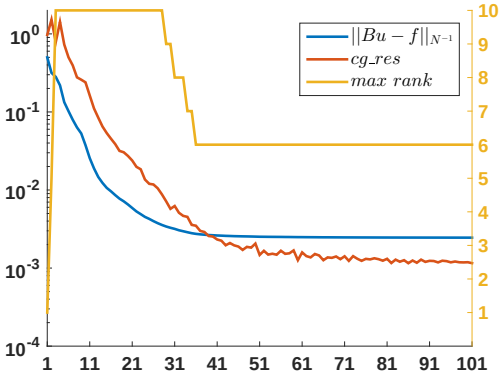
Evaluating the convergence studies of the previous section, we set $\mathcal{J}_1 = \mathcal{J}_2 = \mathcal{K}^E = 2^6$ and $\mathcal{K}^F = 2^7$. The relative truncation tolerance is given by $\epsilon = 10^{-4}$ and we allow a maximal rank of 50, i.e. we decide for Scenario 3b. We expect the CGNR solver to reach a tolerance of 0.005. We set the greedy tolerance to $\texttt{tol} = 0.008$ adding a small safety buffer. Note, that this value does not include the inf-sup and continuity constants but nevertheless leads to satisfying results. The greedy procedure follows the standard way. We restrict to the error evaluation in the residual norm as it is the same norm computed in the truth solver.

The convergence of the RB approximation error is displayed in Figure 4.4.3 (a). Online, we test the RB approximation for two test sets of the parameter domain, cf. Figure 4.4.3 (b) and (c). The greedy tolerance is not violated in the online procedure. In fact, the RB approximation of a solution for a parameter in the sample set is identic to the associated CGNR solution. We even observe that the RB approximations are of possible better quality than the CGNR solutions.

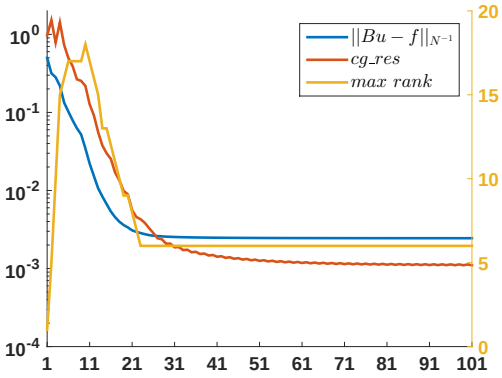
Offline we compute a problem of high dimension $\mathcal{N} = 2^6 2^6 2^6 = 262,144$. Online, we solve a small linear equation system of dimension $N = 7$. The storage capacities for the snapshots are capable as they are stored in their low rank tensor representation. For a parameter that leads to an operator with tensor representation of rank three, the CGNR solver takes about 10 seconds in mean to converge, cf. Table 4.4.3. The RB approximation, i.e. the RB simulation and the reconstruction step, takes 0.005 seconds. The reduction is of order 2000:1.



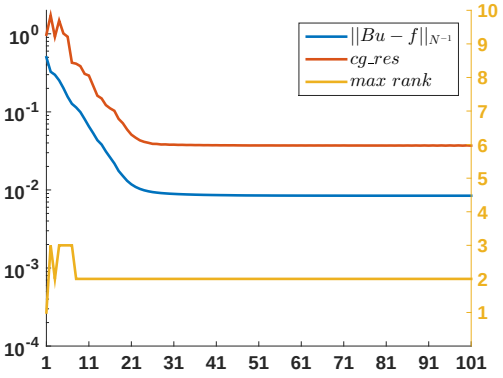
(a) Scenario 3.



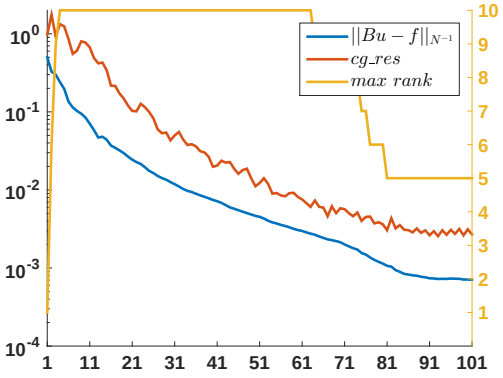
(b) Scenario 3b.



(c) Scenario 3c.

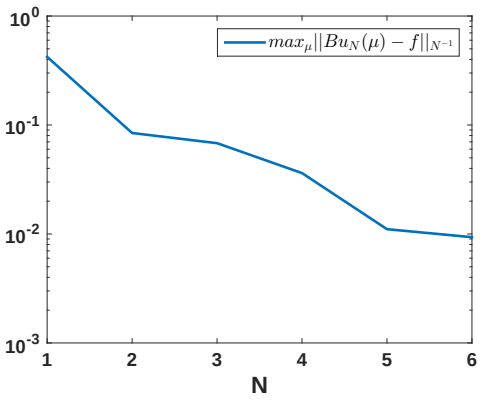


(d) Scenario 5.

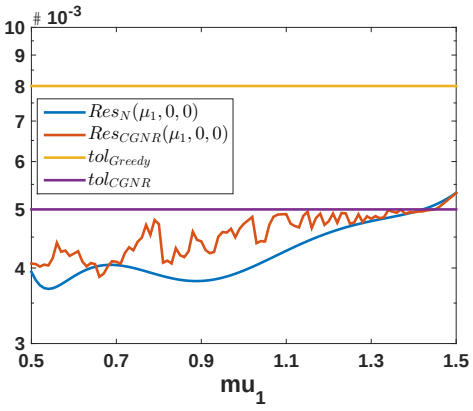


(e) Scenario 5b.

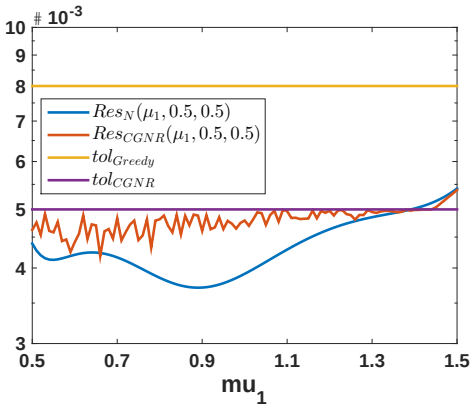
Figure 4.4.2.: Convergence of the CGNR solver.



(a) Convergency of the residual error in the greedy procedure.



(b) Online approximation for $\mu = (\mu_1, 0, 0)$.



(c) Online approximation for $\mu = (\mu_1, 0.5, 0.5)$.

Figure 4.4.3.: Greedy error decay and residual errors in the online application.

For an operator with tensor representation of rank 6 the CGNR solver takes 70 seconds in mean to converge. The RB solution is still given by solving a 7×7 linear equation system and takes as before 0.005 seconds in mean. So, the reduction is of order 14000:1. The computational times of the RBM are significantly shorter and we can achieve even better approximation results.

	CGNR solver [sec]	RB approximation [sec]
$\mu = (\mu_1, 0, 0)$	10.0529	0.0053
$\mu = (\mu_1, 0.5, 0.5)$	73.4399	0.0053

Table 4.4.3.: The mean of computational times: duration of the CGNR procedure vs. duration of the RB simulation and solution reconstruction.

4.4.3. Further Remarks

We have shown that a low rank tensor format is in general applicable in the offline phase of the RBM. The disadvantage is the requirement of an iterative solver where – depending on the application – the approximation quality may not reach the required tolerance for the use as truth solution in the RBM: For operators with more than a diffusion term, the number of summands in the CP decomposition further increases. The ranks increase very fast in the iterative solver up to a magnitude where they are computationally not feasible any more. The operator ranks therefore set a limit to the method. Despite that, the approximation quality in the online phase of the reduced basis method is not bounded by the approximation quality of the iterative solver. In fact, the reduced basis approximation can result in a better approximation of the actual discrete system.

5. Application in Finance

5.1. Introduction

The application of model reduction methods in finance can be extremely beneficial. In option pricing a fast pricing procedure is of advantage not only for pricing itself but especially in the model calibration process to speed up the therein used numerical optimisation. Pricing options in terms of partial differential equations has been proposed by F. Black and M. Scholes [BS73] as well as S. Heston [Hes93]. Even though the standard models provide explicit solution formulas, numerical methods are needed for more sophisticated models, pricing American, barrier or basket options, cf. [MNS05, San04]. Various numerical methods have been applied on PDEs derived from option pricing models – from finite differences or finite element methods to sparse grid and adaptive wavelet discretisations for the numerical solution, cf. [MNS05, MPS04, HRSW13, WAW01, San04, Kes13]. Standard numerical schemes are computational expensive especially for stochastic volatility or multivariate option pricing models where the corresponding PDE space dimensions get larger than one. Therefore, model reduction techniques are used. The proper orthogonal decomposition was applied by [SS13, SS08] in the calibration and pricing process of Lévy driven jump diffusion models solving the underlying PIDEs. Here, the original PIDE is reformulated in a so-called Dupire-type formulation, cf. [Dup94], for the calibration process to avoid expensive recomputations of the PIDE if strike prices differ. The new problem formulation is solved once based on a time stepping scheme for the starting parameter values used in the calibration process and a POD basis is computed for these solutions at every time step. The POD basis is then used in the calibration process, i.e. the solution of the small dimensional POD system replaces the finite element solver. RB-like techniques have been used to price and calibrate (basket and american) options given in terms of P(I)DEs, cf. [Pir09, Pir12, CLP11, PGB15]. Using the reduced basis method, a small dimensional linear system can again be solved in the calibration process instead of using e.g. finite elements. As the model reduction takes place w.r.t. the parameters, we can ensure a certain approximation quality of the RB solution. A standard RBM approach for parabolic PDEs based on a time-stepping scheme has been applied for American option pricing in [HSW12, BHSW15]. Here, the Dupire-type reformulation is not considered and the RB is constructed for only one fixed payoff with one fixed strike price. The

constructed RB can only be used in pricing for this specific setting. We applied the space-time RBM for parameter functions on the Heston model, [MU14]. Here, we do not have to reformulate the problem in a Dupire-type formulation as the payoff of the option is considered as the parameter function. The constructed reduced basis can thus be used in a calibration process, for different strike prices, different time to maturities and – in contrast to the Dupire-type reformulation – even for different type of options.

5.2. Preliminaries

5.2.1. Weighted Sobolev Spaces

The solution of a PDE induced from an option pricing model is in general defined on an unbounded domain. Weighted Sobolev spaces cover the boundary behaviour at infinity using so-called weights. We call $\sigma \in \mathcal{C}^\infty(\Omega)$ a weight and define the weighted Sobolev space $L_p(\Omega, \sigma)$ by

$$L_p(\Omega, \sigma) = \{u \in L_1^{\text{loc}}(\Omega) \,:\, u\sigma \in L_p(\Omega)\}.$$

The associated norm is given by

$$\|u\|_{p,\sigma} = (\int_\Omega |u(x)\sigma(x)|^p dx)^{1/p}. \tag{5.2.1}$$

Setting $\sigma(x) \equiv 1$ we obtain the classical Lebesgue spaces. Weighted Sobolev spaces are a generalisation of the classical ones. In particular in the case $p = 2$ we set $\|\cdot\|_{0,\sigma} := \|\cdot\|_{2,\sigma}$ and define the canonical scalar product on $L_2(\Omega, \sigma)$ by

$$(u, v)_{0,\sigma} = \int_\Omega \sigma^2(x)u(x)v(x)dx. \tag{5.2.2}$$

Thus, $L_2(\Omega, \sigma)$ is a Hilbert spaces as well as $H^1(\Omega, \sigma) := \{u \in L_1^{\text{loc}} \,:\, u\sigma, Du\sigma \in L_2(\Omega)\}$ equipped with the norm $\|\cdot\|_{1,\sigma}$ given by

$$\|u\|_{1,\sigma} = (\sum_{|\alpha|\leq 1} \|D^\alpha u\|_{0,\sigma}^p)^{1/p}. \tag{5.2.3}$$

Obviously $H^1(\Omega, \sigma) \hookrightarrow L_2(\Omega, \sigma)$ by definition. For a deeper insight in the theory of weighted Sobolev spaces we refer to the book of A. Kufner [Kuf80].

5.2.2. Option Pricing Models

We state the standard definition of plain vanilla options.

Definition 5.2.1. [BS73, Introduction] An option is a security giving the right to buy or sell an asset, subject to certain conditions, within a specified period of time. An 'American' option is one that can be exercised at any time up to the date the option expires. A 'European' option is one that can be exercised only on a specified future date. The price that is paid for the asset when the option is exercised is called the 'exercise price' or 'strike price'. The last day on which the option may be exercised is called the 'expiration date' or 'maturity date'.

The right to exercise the option is not an obligation and reflected in the option price. The *writer* of the option sells the *holder* the option. The underlying is one or a basket of risky asset(s) like stocks or indexes.

There are two basic types of options, the *call option* gives the holder the right to buy the underlying for the agreed strike price K. A *put option* gives the holder the right to sell the underlying to the writer of the option for the strike price K. So called *exotic options* vary in their payoff and underlying structure.

In standard option pricing models the price-dynamics of the underlying(s) are given as real valued stochastic processes. In stochastic volatility models, the volatility as well as possible correlations of the underlying are modelled as stochastic processes as well, cf. [Hes93, DFGT07].

In the next proposition a version of the Feynman-Kac formula is presented that provides a link between stochastic differential equations (SDEs) and (deterministic) PDEs that we examined so far in the present thesis.

Proposition 5.2.2. cf. [Øks03, Thm. 8.2.1] Let X be a stochastic process, following the Itô diffusion

$$dX_t = \mu(X_t, t)dt + \sigma(X_t, t)dW_t$$

with independent Wiener processes $W_t = (W_t^1, \dots, W_t^n)$ and initial condition $X_0 = x \in \mathbb{R}^n$. For $p \in \mathcal{C}_0^2(\mathbb{R}^n)$ and interest rate r the conditional expected value $u(x,0) = E(e^{-rT}p(X_T)|X_0 = x)$ can equivalently be computed solving the following PDE,

$$\frac{d}{dt}u + \sum_{i=1}^{n} \mu_i(x,t)u_{x_i} + \frac{1}{2}\sum_{i,j=1}^{n} [\sigma^T(x,t)\sigma(x,t)]_{i,j}u_{x_ix_j} - r(x,t)u = 0$$

$$u(x,T) = p(x).$$

Remark 5.2.3. For a short introduction to stochastic analysis we refer to Appendix A.2.

Remark 5.2.4. A transformation in time given by $t \mapsto T - t$ allows to solve the forward equation with payoff function p as initial condition.

Remark 5.2.5. The extension of standard diffusion models to so called jump-diffusion models provides more flexibility in the price dynamics. In the one dimensional SDE, the diffusion part

is replaced by a Lévy process, cf. [CT03]. Again, a Feynman-Kac formula deduces an associated deterministic differential equation, a partial integro differential equation (PIDE). The additional integral term represents the jump part. The integral term requires special treatment in the numerical scheme: a standard finite element discretisation leads to densely populated and ill-conditioned stiffness matrices, cf. [MNS05, Sec. 1].

5.2.3. PDE based Methods in Option Pricing

Numerical schemes work either on the given stochastic or the induced partial differential equation. Two well-known methods are Monte Carlo methods and Fourier Pricing. The first works on the underlying SDE, the second uses the corresponding characteristic function.

Standard numerical schemes are used to solve the associated PDE instead of the SDE. Finite differences and finite element methods have been applied e.g. in [San04, WAW01]. Further, wavelet methods are applied if an additional integral term appears, cf. [MNS05, MPS04, Hil09, Kes13]. For a general overview we refer to [HRSW13]. A main task is the treatment of the underlying unbounded domain. A standard numerical scheme requires a bounded domain and imposed boundary conditions.

Localisation In a localisation process, the unbounded domain is truncated and suitable boundary conditions are imposed. The boundary conditions are chosen in accordance to the underlying model and in particular the initial value. The initial value of the PDE, i.e. the payoff function of the considered option, has a major impact on the solution. As our goal is to test the numerical method for parameter functions, especially the initial value is a-priori unknown. Therefore, the boundary behaviour of the PDE solution cannot or not entirely be derived. A workaround are procedures used in option pricing for the solution of PIDEs where the integral operator as global operator requires specific treatment. Weighted Sobolev spaces are imposed for a full description of the solution behaviour towards infinity. Hence, the solution at every time point $t \in (0, T)$ is an element of a weighted Sobolev space with suitable weight σ, cf. Section 5.2.1.

For the numerical solution, we consider the PDE with log-transformations in all variables. The domain $\mathbb{R}^n$ is truncated to a bounded domain $\Omega_{\mathbf{R}} = (\Omega_{R_1}, \ldots, \Omega_{R_n}) \subset \mathbb{R}^n$, $\mathbf{R} = (R_1, \ldots, R_n)$, and $\Omega_{R_i} := (-R_i, R_i)$. Let u denote the solution of the associated PIDE and p the payoff function. In the PIDE literature [MNS05, MPS04, Hil09, Kes13], the PIDE associated with the excess-to-payoff $U = u - p$ is solved instead of the original PIDE. The advantage is that the access to payoff tends to zero for x to infinity, i.e. $U \to 0$ for $|x| \to \infty$, [MNS05, Sec. 2.3.2]. Thus, homogeneous Dirichlet boundary conditions are naturally imposed. The P(I)DE of the excess-to-payoff $U \in \mathbb{X}$ for $V = H_0^1(\mathbb{R})$ has an initial value that equals zero and a right-hand side given by the operator $\mathcal{A}$ applied on the payoff p. We do not follow this approach as for the

application of the RBM for parameter functions we prefer the parameter function to be in the initial condition, cf. Chapter 3.

Infinite Elements Infinite elements for the boundary treatment in option pricing models were proposed by S. Sanfelici in [San04]. This is a reasonable approach to avoid possibly very large domains $\Omega_{\mathbf{R}}$ that are required to ensure a certain approximation quality of the solution. If no log-transformation of the underlying variables is applied, the domain of the underlying price process is naturally bounded on the positive real numbers. A weighted Sobolev space can be introduced to cover the behaviour of the underlying price process towards infinity, cf. [San04]. Here, the weight is piecewise defined; it equals one on the first part of the domain, i.e. on $\Omega_{\mathbf{R}} \cap \mathbb{R}^n_+$ and is polynomial decreasing on $\mathbb{R}^n_+ \setminus \Omega_{\mathbf{R}}$. A standard finite element discretisation can be applied on the bounded part of the domain. For the unbounded part, the infinite elements are used.

5.3. Black-Scholes Model

5.3.1. Model

The Black-Scholes model was introduced in [BS73] and is commonly used for option pricing. In terms of PDEs it is a model of space dimension one and provides an explicit solution formula. The model is easily extendible to stochastic volatility as well as jump-diffusion models. We assume a Black-Scholes market, i.e. an arbitrage and frictional cost free market with one riskless and one risky asset where the asset price follows a geometric Brownian motion and the interest rate and the volatility are constants over time. For a full definition we refer to [Sey09, Ass. 1.2]. Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space. The spot price B_t of the riskless asset at time t follows an ordinary differential equation

$$dB_t = rB_t dt$$

with $B_0 = B$ being the current spot price and riskless constant interest rate $r > 0$. The price S_t at time t of the risky asset is given by an SDE

$$dS_t = \mu S_t dt + \sigma S_t dW_t, \quad S_0 = S,$$

that is driven by a standard Brownian motion W_t, [Sey09, Ass. 1.2, Model 1.13].

For pricing, we change to the risk neutral evaluation with the risk neutral measure Q, cf. Remark A.2.6. The stock price follows again a geometric Brownian motion [Sey09, Sec. 1.7.3]

$$dS_t = rS_t dt + \sigma S_t dW_t.$$

The associated Black-Scholes equation is well known.

Proposition 5.3.1. [Sey09, Def. 1.1] For a financial product with payoff p at the final time point T the discounted option price $u(t, S_t) = E^Q(e^{-r(T-t)}p(S_T)|\mathcal{F}_t)$ is equivalently given as the solution of

$$\begin{aligned} \frac{d}{dt}u + \frac{\sigma^2}{2}s^2\frac{d^2}{ds^2}u + rs\frac{d}{ds}u - ru = 0 \quad &\text{in } (0,T)\times(0,\infty) \\ u(T,s) = p(s) \quad &\text{in } (0,\infty). \end{aligned}$$

To overcome the degeneracy in $S = 0$ we consider the logarithmic return price $x = \ln(s)$. Further we perform a transformation in time, $t \mapsto T - t$. The Black-Scholes equation transforms to

$$\begin{aligned} \frac{d}{dt}u - \frac{\sigma^2}{2}\frac{d^2}{dx^2}u - (r - \frac{\sigma^2}{2})\frac{d}{dx}u + ru = 0 \quad &\text{in } (0,T)\times\mathbb{R} \\ u(0,x) = p(e^x) \quad &\text{in } \mathbb{R}. \end{aligned}$$

We have not yet specified the payoff function p.

Example 5.3.2. (i) For a European call option with strike K and time to maturity T, the payoff is given by $p(s) = \max(s - K, 0)$.

(ii) For a European put option, the payoff is given by $p(s) := \max(K - s, 0)$.

In fact, a payoff function is not in $L_2(\mathbb{R})$ but may polynomially grow for $|s| \to \infty$ resp. exponentially for the log-transformed payoff and $|x| \to \infty$.

5.3.2. Space-Time Variational Formulation

We introduce the space-time variational formulation of the Black-Scholes equation. As already mentioned the payoff function is in general not an element in $L_2(\mathbb{R})$. Hence, we introduce weighted Sobolev spaces to catch a possible exponential growth at infinity, following [MPS04]. We define the weighted space $H^1(\mathbb{R}, \sigma)$ for $\sigma(x) = e^{\varkappa|x|}$ with $\varkappa \in \mathbb{R}$ by

$$H^1_\varkappa(\mathbb{R}) := \{u \in L_1^{\mathrm{loc}}(\mathbb{R}) \,:\, ue^{\varkappa|x|}, u'e^{\varkappa|x|} \in L_2(\mathbb{R})\}$$

and $L_{2,\varkappa}(\mathbb{R}) := \{u \in L_1^{\mathrm{loc}}(\mathbb{R}) \,:\, ue^{\varkappa|x|} \in L_2(\mathbb{R})\}$. We restrict to payoff functions p being an element in $L_{2,-\varkappa}(\mathbb{R})$ for fixed $\varkappa > 1$. For $u \in \mathbb{X}$, $v = (v_1, v_2) \in \mathbb{Y}$ given by $V = H^1_{-\varkappa}(\mathbb{R})$ and $H = L_{2,-\varkappa}(\mathbb{R})$, cf. Section 2.2.1, and by integration by parts, the left-hand side of the space-time

variational formulation reads

$$\begin{aligned} b(u,v) = &\int_{(0,T)}\int_{\mathbb{R}} \dot{u}(x,t)v_1(x,t)e^{-2\varkappa|x|}dx - (r-\frac{\sigma^2}{2})\int_{\mathbb{R}}\nabla u(x,t)v_1(x,t)e^{-2\varkappa|x|}dx \\ &+ \frac{\sigma^2}{2}\int_{\mathbb{R}}\nabla u(x,t)\nabla v_1(x,t)e^{-2\varkappa|x|} - 2\varkappa\mathrm{sgn}(x)\nabla u(x,t)v_1(x,t)e^{-2\varkappa|x|}dx \\ &+ r\int_{\mathbb{R}} u(x,t)v_1(x,t)e^{-2\varkappa|x|}dxdt + \int_{\mathbb{R}} u(x,0)v_2(x)e^{-2\varkappa|x|}dx. \end{aligned}$$

The right-hand side is given by

$$f(v) = \int_{\mathbb{R}} p(e^x)v_2(x)e^{-2\varkappa|x|}dx.$$

For well-posedness, we require the bilinear form $a^{-\varkappa}(\cdot,\cdot) : H^1_{-\varkappa}(\mathbb{R}) \times H^1_{-\varkappa}(\mathbb{R}) \to \mathbb{R}$ defined by

$$\begin{aligned} a^{-\varkappa}(u,v_1) := &\frac{\sigma^2}{2}\int_{\mathbb{R}}\nabla u(x,t)\nabla v_1(x,t)e^{-2\varkappa|x|}dx + r\int_{\mathbb{R}} u(x,t)v_1(x,t)e^{-2\varkappa|x|}dx \\ &- \int_{\mathbb{R}}\Big(\sigma^2\varkappa\mathrm{sgn}(x) + (r-\frac{\sigma^2}{2})\Big)\nabla u(x,t)v_1(x,t)e^{-2\varkappa|x|}dx, \end{aligned}$$

to be continuous and to satisfy the Gårding inequality, cf. Theorem 2.2.15.

Proposition 5.3.3. [MPS04, Prop. 2.1] Let $\varkappa \in \mathbb{R}$ be arbitrary fixed. The bilinear form $a^{-\varkappa}(\cdot,\cdot) : H^1_{-\varkappa}(\mathbb{R}) \times H^1_{-\varkappa}(\mathbb{R}) \to \mathbb{R}$ is continuous and satisfies the Gårding inequality, i.e. there exist a constant $C_a > 0$ and $\alpha > 0$, $\lambda > 0$, such that

(i) $|a^{-\varkappa}(u,v)| \le C_a|u|_{H^1_{-\varkappa}(\mathbb{R})}|v|_{H^1_{-\varkappa}(\mathbb{R})} \quad \forall\, u,v \in H^1_{-\varkappa}(\mathbb{R})$,

(ii) $a^{-\varkappa}(u,u) + \lambda|u|^2_{L_{2,-\varkappa}(\mathbb{R})} \ge \alpha|u|^2_{H^1_{-\varkappa}(\mathbb{R})}$.

Proof. Let $a(x) := \sigma^2\varkappa\mathrm{sgn}(x) + (r - \frac{\sigma^2}{2})$.
(i) Continuity follows from the Cauchy Schwarz inequality.

$$\begin{aligned} |a^{-\varkappa}(u,v)| &\le \frac{\sigma^2}{2}|\nabla u|_{L_{2,-\varkappa}(\mathbb{R})}|\nabla v|_{L_{2,-\varkappa}(\mathbb{R})} + r|u|_{L_{2,-\varkappa}(\mathbb{R})}|v|_{L_{2,-\varkappa}(\mathbb{R})} + |a|_\infty|\nabla u|_{L_{2,-\varkappa}(\mathbb{R})}|v|_{L_{2,-\varkappa}(\mathbb{R})} \\ &\le \max(\frac{\sigma^2}{2}, r, |a|_\infty)|u|_{H^1_{-\varkappa}(\mathbb{R})}|v|_{H^1_{-\varkappa}(\mathbb{R})} \end{aligned}$$

(ii) For $u \in H^1_{-\varkappa}(\mathbb{R})$

$$\begin{aligned} a^{-\varkappa}(u,u) &\ge \frac{\sigma^2}{2}|\nabla u|^2_{L_{2,-\varkappa}(\mathbb{R})} - \int_{\mathbb{R}}(\varkappa\mathrm{sgn}(x)\sigma^2 + r - \frac{\sigma^2}{2})\nabla u(x)u(x)e^{-2\varkappa|x|}dx + r|u|^2_{L_{2,-\varkappa}(\mathbb{R})} \\ &\ge \min(\frac{\sigma^2}{2}, r)|u|^2_{H^1_{-\varkappa}(\mathbb{R})} - |a|_\infty|\nabla u|_{L_{2,-\varkappa}(\mathbb{R})}|v|_{L_{2,-\varkappa}(\mathbb{R})} \\ &\ge \min(\frac{\sigma^2}{2}, r)|u|^2_{H^1_{-\varkappa}(\mathbb{R})} - |a|_\infty\epsilon|\nabla u|^2_{L_{2,-\varkappa}(\mathbb{R})} - |a|_\infty\frac{1}{4\epsilon}|v|^2_{L_{2,-\varkappa}(\mathbb{R})}. \end{aligned}$$

For $\epsilon < \min(\frac{\sigma^2}{2}, r)/|a|_\infty$ we obtain the Gårding inequality. □

5.3.3. Localisation

Before performing the localisation procedure by truncating the domain and imposing boundary conditions we remark the following. Since the option price u is an element in $H^1_{-\varkappa}(\mathbb{R})$, the function $\tilde{u} = ue^{-\varkappa|x|}$ is an element of $\mathbb{X}$ for $V = H^1_0(\mathbb{R})$. We can reformulate the above given variational form for $\tilde{u}$ and corresponding test functions $\tilde{v}(x,t) = v(x,t)e^{-\varkappa|x|} \in \mathbb{Y}$ with $H = L_2(\mathbb{R})$.

$$\begin{aligned}\tilde{b}(\tilde{u},\tilde{v}) = &\int_{(0,T)}\int_{\mathbb{R}} \dot{\tilde{u}}(x,t)\tilde{v}_1(x,t)dxdt \\ &+ \int_{\mathbb{R}} \tilde{u}(x,0)\tilde{v}_2(x)dx \\ &+ \int_{(0,T)} \frac{\sigma^2}{2}\int_{\mathbb{R}} \nabla\tilde{u}(x,t)\nabla\tilde{v}_1(x,t)dx \\ &+ \int_{\mathbb{R}} \Big(r - \frac{\sigma^2}{2}\varkappa^2 + (\frac{\sigma^2}{2} - r)\varkappa\mathrm{sgn}(x)\Big)\tilde{u}(x,t)\tilde{v}_1(x,t)dx \\ &+ \int_{\mathbb{R}} \Big((\frac{\sigma^2}{2} - r) - \sigma^2\varkappa\mathrm{sgn}(x)\Big)\nabla\tilde{u}(x,t)\tilde{v}_1(x,t)dx\,dt.\end{aligned}$$

We define the transformed payoff by $\tilde{p}(e^x) := p(e^x)e^{-\varkappa|x|}$. The right-hand side is given by

$$\tilde{f}(v) = \int_{\mathbb{R}} \tilde{p}(e^x)\tilde{v}_2(x)dx.$$

The problem is well-posed as the embedding $V \hookrightarrow H$ is dense and the associated bilinear form

$$\begin{aligned}\tilde{a}(\tilde{u},\tilde{v}) = &\int_{(0,T)} \frac{\sigma^2}{2}\int_{\mathbb{R}} \nabla\tilde{u}(x,t)\nabla\tilde{v}_1(x,t)dx \\ &+ \int_{\mathbb{R}} \Big(r - \frac{\sigma^2}{2}\varkappa^2 + (\frac{\sigma^2}{2} - r)\varkappa\mathrm{sgn}(x)\Big)\tilde{u}(x,t)\tilde{v}(x,t)dx \\ &+ \int_{\mathbb{R}} \Big((\frac{\sigma^2}{2} - r) - \sigma^2\varkappa\mathrm{sgn}(x)\Big)\nabla\tilde{u}(x,t)\tilde{v}(x,t)dx\,dt\end{aligned}$$

is bounded and satisfies the Gårding inequality. We localise the problem by restriction to the bounded domain $\Omega_R = (-R,R)$. Further, we set $V = H^1_0(\Omega_R)$ and $H = L_2(\Omega_R)$ in the space-time variational formulation and impose homogeneous Dirichlet boundary conditions on the boundary $\partial\Omega_R = \{-R,R\}$. The localisation error can be estimated.

Proposition 5.3.4. For the bounded domain $\Omega_R = (-R,R)$, let $\mathbb{X}$ and $\mathbb{Y}$ be given by $V = H^1_0(\Omega_R)$ and $H = L_2(\Omega_R)$. For any function $w \in H^1_0(\Omega_R)$ we denote its embedding in $H^1_0(\mathbb{R})$ by the same letter. Let u_{BS} be the true solution of the Black-Scholes equation. Let $\tilde{u} \in \mathbb{X}$ be the

solution of

$$\tilde{b}(u,v) = \tilde{f}(v) \quad \forall\, v \in \mathbb{Y}$$

with initial value $\tilde{p}(e^x) = p(e^x)e^{-\varkappa|x|}$ for $p \in L_{2,-\varkappa}(\mathbb{R})$. Then, for a constant $C > 0$,

$$\|u_{\mathrm{BS}} - \tilde{u}\|_{\mathbb{X}}^2 \leq C^2 2Rp(e^{R/2})^2 e^{-\varkappa R}.$$

Proof. Recall the a-priori estimate for the solution u of a parabolic PDE, cf. [Eva99, Sec. 7.1 b. Thm 2]. Let $C > 0$ be a constant, then

$$\|u\|_{\mathbb{X}}^2 \leq \|u\|_{L_2(I;V)}^2 + \|\dot{u}\|_{L_2(I;V')}^2 + \max_{t\in[0,T]} |u(t)|_H^2 \leq C^2\Big(\|f\|_{L_2(I;V')}^2 + |u(0)|_H^2\Big).$$

For the right-hand side $f \equiv 0$ we receive $\|u\|_{\mathbb{X}} \leq C|u(0)|_H$. Define the cut-off function, [Hil09, Thm. 3.12], $\phi_{R,R/2}(x) \in \mathcal{C}^\infty(\Omega_R)$,

$$\phi_{R,R/2} \geq 0, \quad \phi_{R,R/2} \equiv 1 \text{ on } \Omega_{R/2} \text{ and } |\nabla\phi_{R,R/2}|_\infty \leq c$$

for some constant $c > 0$. The approximation error of u_{BS} and $\tilde{u}$ is then given by

$$\begin{aligned}
\|u_{\mathrm{BS}} - \tilde{u}\|_{\mathbb{X}}^2 &\leq C|p(e^x)e^{-\varkappa|x|} - \tilde{u}(0)|_{L_2(\Omega_R)}^2 \leq C^2 \inf_{w\in H_0^1(\Omega_R)} |p(e^x)e^{-\varkappa|x|} - w|_{L_2(\Omega_R)}^2 \\
&\leq C^2|p(e^x)e^{-\varkappa|x|} - \phi_{R,R/2}(x)p(e^x)e^{-\varkappa|x|}|_{L_2(\Omega_R)}^2 \\
&= C^2 \int_{\Omega_R\setminus\Omega_{R/2}} (1-\phi_{R,R/2}(x))^2 p(e^x)^2 e^{-2\varkappa|x|}dx \\
&\leq C^2 R \max_{x\in\Omega_R\setminus\Omega_{R/2}} (1 + \phi_{R,R/2}(x)^2 - 2\phi_{R,R/2}(x)) \max_{x\in\Omega_R\setminus\Omega_{R/2}} p(e^x)^2 e^{-2\varkappa|x|} \\
&= C^2 2Rp(e^{R/2})^2 e^{-\varkappa R}.
\end{aligned}$$

□

By Proposition 5.3.4, the parameter $\varkappa$ of the weight function does influence the convergence rate. We have to ensure that our choice for $\varkappa$ does not change the PDE formulation to a convection dominant problem as our numerical schemes are not suited for those. For a call function we observe that

$$p(e^{R/2})^2 e^{-\varkappa R} = (e^R - 2e^{R/2}K + K^2)e^{-\varkappa R} \longrightarrow 0 \text{ for } R \to \infty$$

given $\varkappa > 1$.

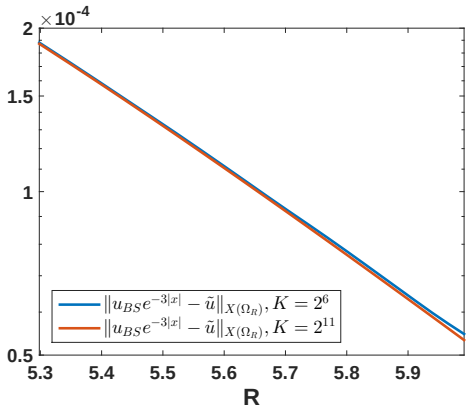


Figure 5.3.1.: Convergence of the PDE solution $\tilde{u}$ towards the transformed true solution u_{BS} w.r.t. R. The FEM solution is projected onto the finest grid given by 2^{11} intervals, the Black-Scholes formula is directly evaluated on the finest grid.

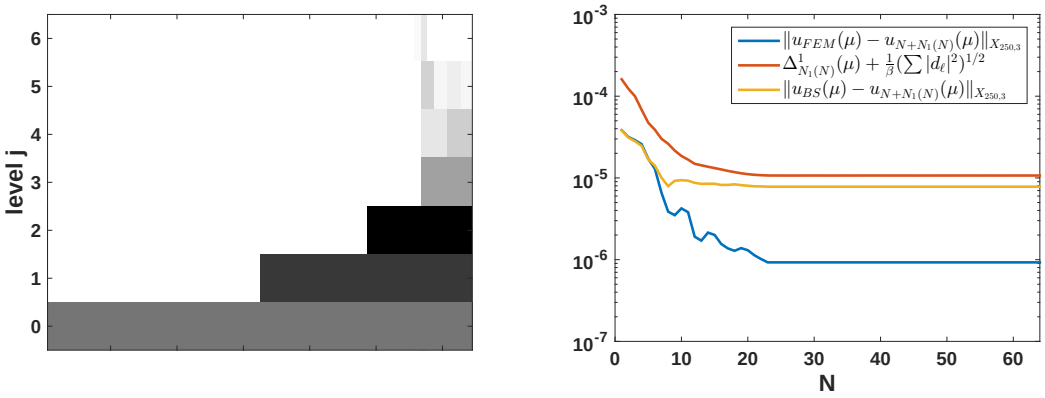
5.3.4. Numerical Experiments

We are going to price a call option with strike price $K = 60$, volatility $\sigma = 0.3$, interest rate $r = 0.04$ and a weight function given by $\varkappa = 3$ using the RBM for parameter functions presented in Section 3.2.2. The error is measured in the discrete space-time natural norm $\|\cdot\|_{\mathbb{X},\text{bar}}$, cf. Equation (2.2.20). To specify the norm dependence on the truncated domain Ω_R and the chosen weight parameter $\varkappa$, we write $\mathbb{X}_{R,\varkappa}$ for the ansatz space $\mathbb{X}$ given by $V = H^1_{-\varkappa}(\Omega_R)$. Further for $\tilde{u}(x) = u(x)e^{-\varkappa|x|}$ holds

$$\|u_{\text{BS}}(x)e^{-\varkappa|x|} - \tilde{u}(x)\|_{\mathbb{X}_{R,0},\text{bar}} = \|u_{\text{BS}}(x) - u(x)\|_{\mathbb{X}_{R,\varkappa},\text{bar}}.$$

We impose the space-time discretisation that is equivalent to the Crank Nicolson time stepping scheme and introduced in Section 2.2.2.2. We fix the number of subintervals in the time domain to $\mathcal{J} = 2^6$. In Figure 5.3.1 we see the convergence behaviour of the space-time error between the transformed Black-Scholes formula solution and the numerical solution $\tilde{u}$. A refinement w.r.t. the basis functions of the spatial domain does not significantly improve the approximation result. We achieve a space-time error below 10^{-3} for $\mathcal{K} = 2^6$, $\mathcal{J} = 2^6$ and $R = \ln(250) = 5.52$. The temporal domain is given by $[0, 0.3]$.

We assume the payoff to be a parameter function in the initial value. Analogous to the numerical experiments in Section 3.4.3, we consider Haar wavelets on the spatial domain $\Omega_R = (-R, R)$ to approximate the parameter space $L_2(\Omega_R)$. We only consider wavelets up to level 6 for the initial value approximation. We store 128 initial values in the Initial Value Library $\mathbb{L}_{\text{init}}$. The volatility $\sigma \in [0, 1]$ is the additional parameter of interest. The strike price K does not need to be included explicitly in the parameter domain as the full initial value is a parameter function.



(a) Wavelet expansion coefficients decay of $\tilde{p}(e^x)$. Dark colors correspond to high absolute coefficient values.

(b) Error and estimator for the online procedure.

Figure 5.3.2.: Online RB approximation for a call option with strike $K = 60$.

The Evolution Greedy Algorithm 3.2.2 is performed 128 times for a tolerance $\texttt{tol} = 0.001$ and a training set $\mathcal{D}_1^{\text{train}} \subset [0, 1]$ given by 17 equally distributed points. The resulting 128 small, i.e. $N_1 \in \{9, 10, 11\}$, RB bases are stored in the Evolution Library $\mathbb{L}_{\text{evol}}$.

We do not compute the corresponding inf-sup values for the greedy procedure. The decay of the residual does not change by not applying the inf-sup constant. However, the procedure may stop earlier by assuming $\beta = 1$ as $\beta < 1$ in reality.

After the library initialisation, the online procedure is performed for a call with payoff $p(e^x) = \max(e^x - 60, 0)$. We set the volatility $\sigma = 0.3$. The corresponding inf-sup constant is given by $\beta = 0.0492$. The relevant wavelet coefficients, cf. Figure 5.3.2 (a), are close to the boundary on the right-hand side where the call option payoff takes its non-zero values. The convergence of the error between the RB approximation and the FEM approximation is shown in Figure 5.3.2 (b). The error estimator, i.e., the sum of wavelet approximation error and the a-posteriori error estimator for the evolution of the solution, bounds the error as expected. A very good approximation level of 10^{-6} is reached including all $N = 23$ relevant wavelet functions. This corresponds to an RB basis for the evolutionary part of size $N_1 = 210$. But even only using the first $N = 11$ relevant wavelets, the space-time approximation error is below 10^{-5}. Here, the RB basis has only $N_1 = 101$ basis functions. An additional evaluation of the error between the RB approximation and the solution of the Black-Scholes formula in the norm of the associated weighted Sobolev spaces confirms the good approximation results. We achieve an error of about 10^{-5}. We emphasize again that our reduced basis was not constructed for the call payoff function but arbitrary payoff functions can be used.

The constructed RB can be used not only for pricing but in the calibration process. The offline

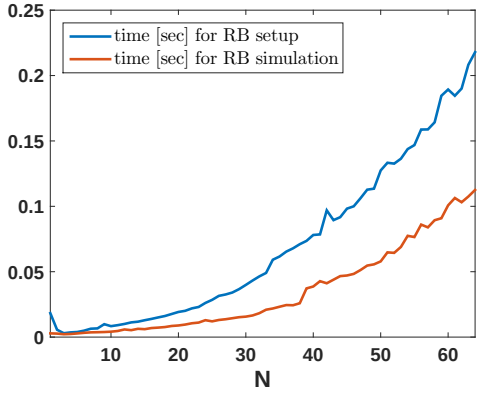


Figure 5.3.3.: Computational time in the online RB approximation.

phase of the space-time RBM for parameter functions takes 0.3205 seconds, 0.0973 seconds are needed for the initialisation of the Initial Value Library and 0.2232 seconds for the Evolution Library. In average, the Crank Nicolson truth solver takes 0.0064 seconds to solve the problem formulation. Consequently, as soon as we are interested in more than 50 evaluations – e.g. for the optimisation process of the model calibration – the offline phase pays off. For the call option that we consider online, the Crank Nicolson truth solver takes 0.005812 seconds. The RB online phase evaluation time depends on the actual number of wavelets that are considered. In Figure 5.3.3 we can observe how the time for the RB setup and the actual RB simulation increases with the number of wavelets for the approximation of the initial value. For $N = 11$ the space-time approximation error is below 10^{-5}. In that case the RB simulation takes 0.0047 seconds what is 80 % of the time the finite element truth solver takes. The time saving using the reduced system is 20 %.

5.4. Heston Model

5.4.1. Model

The Heston model is a well-known model for option pricing allowing also a non-constant volatility of the underlying.[1] It was invented 1993 by Heston, [Hes93]. The SDEs for the asset price S_t and the volatility ν_t are assumed to be

$$dS_t = rS_t dt + \sqrt{\nu_t} S_t dz_1(t), \qquad d\nu_t = \kappa[\theta - \nu_t]dt + \sigma\sqrt{\nu_t} dz_2(t), \tag{5.4.1}$$

[1] The first two paragraphs are published in [MU14, Sec. 2.1].

where z_1, z_2 are Wiener processes with correlation ρ, r is the return rate of the asset, κ the mean reversion rate to the long term variance θ and σ is the volatility of the volatility. In particular, the instantaneous variance ν_t is modeled as a CIR (Cox-Ingersoll-Ross) process, [CIR85], that is strictly positive as long as the parameters obey the so called Feller condition $2\kappa\theta > \sigma^2$. Finally, the model implies that $z_1 = \sqrt{1-\rho^2}dz_3 + \rho dz_2$ with independent Brownian motions z_2, z_3. The observed market price s_0 of the underlying asset and the (implied) volatility v_0 at time zero are the corresponding initial conditions, i.e. $S_0 = s_0$ and $\nu_0 = v_0$. We obtain global existence and uniqueness of the solution as the coefficients are locally Lipschitz continuous and of at most linear growth.

Instead of calculating the conditional expectation of the discounted payoff under a risk neutral measure we can equivalently solve the associated parabolic PDE in $\tilde{x} = (S, \nu)$, [Hes93]:

$$\dot{u} = Sr\frac{du}{dS} + (\kappa\theta - \kappa\nu)\frac{du}{d\nu} + \frac{1}{2}\nu S^2\frac{d^2u}{dS^2} + \nu\rho\sigma S\frac{d}{dS}\frac{d}{d\nu}u + \frac{1}{2}\nu\sigma^2\frac{d^2u}{d\nu^2} - ru.$$

We first apply a log-transformation regarding the asset prices S and second regarding the volatility ν. The transformed PDE in $x = (x_1, x_2) = (\log S, \log \nu)$ is given by

$$\dot{u} = (r - \frac{1}{2}e^{x_2})\frac{du}{dx_1} + ((\kappa\theta - \frac{1}{2}\sigma^2)e^{-x_2} - \kappa)\frac{du}{dx_2} + \frac{1}{2}e^{x_2}\frac{d^2u}{{dx_1}^2} + \rho\sigma\frac{d}{dx_1}\frac{d}{dx_2}u + \frac{1}{2}\sigma^2e^{-x_2}\frac{d^2u}{{dx_2}^2} - ru.$$

The PDE in non-divergence form hence reads

$$\frac{d}{dt}u(t) - \nabla(\alpha\nabla u(t)) + \beta\nabla u(t) + \gamma\, u(t) = 0 \text{ for } t \in (0,T], \quad u(0) = u_0,$$

where $u_0 = p(e^{x_1}, e^{x_2})$ is the payoff and the coefficient functions are given by

$$\alpha := \frac{1}{2}\begin{pmatrix} e^{x_2} & \sigma\rho \\ \sigma\rho & \sigma^2e^{-x_2} \end{pmatrix} \qquad \beta := -\begin{pmatrix} r - \frac{1}{2}e^{x_2} \\ \kappa\theta e^{-x_2} - \kappa \end{pmatrix}, \qquad \gamma := r. \tag{5.4.2}$$

5.4.2. Space-Time Variational Formulation

We introduce the space-time variational formulation of the Heston model. We define the weighted Sobolev spaces for $\kappa = (\kappa_1, \kappa_2) \in \mathbb{R}^2$ by

$$H^1_\varkappa(\mathbb{R}^2) := \{u \in L_1^{\mathrm{loc}}(\mathbb{R}^2) \,|\, ue^{\varkappa_1|x|}e^{\varkappa_2|y|}, \nabla ue^{\varkappa_1|x|}e^{\varkappa_2|y|} \in L_2(\mathbb{R}^2)\}$$

and $L_{2,\varkappa}(\mathbb{R}^2) := \{u \in L_1^{\mathrm{loc}}(\mathbb{R}^2) \,|\, ue^{\varkappa_1|x|}e^{\varkappa_2|y|} \in L_2(\mathbb{R}^2)\}$. We assume that the payoff functions of interest are elements in $L_{2,-\varkappa}(\mathbb{R}^2)$ for $\varkappa_1 > 1$ and $\varkappa_2 > 0$. For the ansatz space $\mathbb{X}$, cf. Equation (2.2.1), we set $V := H^1_{-\varkappa}(\mathbb{R}^2)$. Further, we choose $H = L_{2,-\varkappa}(\mathbb{R}^2)$ to define the test space $\mathbb{Y}$,

cf. Equation (2.2.5). The left-hand side of the space-time variational inequality for α, β and γ defined in Equation (5.4.2), $u \in \mathbb{X}$ and $v \in \mathbb{Y}$ is given by

$$\begin{aligned} b(u,v) = &\int_{(0,T)} \int_{\mathbb{R}^2} \dot{u}(x,t) v_1(x,t) e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)} dx \\ &+ \int_{\mathbb{R}^2} \Big(\alpha \nabla u(x,t)\Big)^T \nabla v_1(x,t) e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)} dx \\ &+ \int_{\mathbb{R}^2} (\alpha \nabla u(x,t))^T \begin{pmatrix} -2\varkappa_1 \mathrm{sgn}(x_1) \\ -2\varkappa_2 \mathrm{sgn}(x_2) \end{pmatrix} v_1(x,t) e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)} dx \\ &- \int_{\mathbb{R}^2} \beta \nabla u(x,t) v_1(x,t) e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)} dx \\ &+ \gamma \int_{\mathbb{R}^2} u(x,t) v_1(x,t) e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)} dx\, dt + \int_{\mathbb{R}^2} u(0,x) v_2(x) e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)} dx. \end{aligned}$$

The right-hand side is given by $f(v) = \int_{\mathbb{R}^2} u_0(x) v_2(x) e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)} dx$ for $v_2 \in H$.

The bilinear form $a^{-\varkappa}(\cdot,\cdot)$ given by

$$\begin{aligned} a^{-\varkappa}(u,v) = &\int_{\mathbb{R}^2} \Big(\alpha \nabla u(x,t)\Big)^T \nabla v_1(x,t) e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)} dx \\ &+ \int_{\mathbb{R}^2} (\alpha \nabla u(x,t))^T \begin{pmatrix} -2\varkappa_1 \mathrm{sgn}(x_1) \\ -2\varkappa_2 \mathrm{sgn}(x_2) \end{pmatrix} v_1(x,t) e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)} dx \\ &- \int_{\mathbb{R}^2} \beta \nabla u(x,t) v_1(x,t) e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)} dx \\ &+ \gamma \int_{\mathbb{R}^2} u(x,t) v_1(x,t) e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)} dx \end{aligned}$$

is continuous and satisfies the Gårding inequality:

Proposition 5.4.1. Let $\varkappa = (\varkappa_1, \varkappa_2) \in \mathbb{R}^2$. The bilinear form $a^{-\varkappa}(\cdot,\cdot) : H^1_{-\varkappa}(\mathbb{R}) \times H^1_{-\varkappa}(\mathbb{R}) \to \mathbb{R}$ is continuous and satisfies the Gårding inequality, i.e. there exist a constant $C_a > 0$ and $\alpha > 0$, $\lambda > 0$, such that

(i) $|a^{-\varkappa}(u,v)| \le C_a |u|_{H^1_{-\varkappa}(\mathbb{R})} |v|_{H^1_{-\varkappa}(\mathbb{R})} \quad \forall\, u, v \in H^1_{-\varkappa}(\mathbb{R})$,

(ii) $a^{-\varkappa}(u,u) + \lambda |u|^2_{L_{2,-\varkappa}(\mathbb{R})} \ge \alpha |u|^2_{H^1_{-\varkappa}(\mathbb{R})}$.

Proof. Let $a(x) = \left(\alpha^T \begin{pmatrix} -2\varkappa_1 \mathrm{sgn}(x_1) \\ -2\varkappa_2 \mathrm{sgn}(x_2) \end{pmatrix} - \beta\right)$.
(i) Continuity follows again from the Cauchy Schwarz inequality,

$$|a^{-\varkappa}(u,v)| \le |\alpha(x)|_\infty |u|_{H^1_{-\varkappa}(\mathbb{R})} |v|_{H^1_{-\varkappa}(\mathbb{R})} + |a(x)|_\infty |u|_{H^1_{-\varkappa}(\mathbb{R})} |v|_{H^1_{-\varkappa}(\mathbb{R})} + r |u|_{H^1_{-\varkappa}(\mathbb{R})} |v|_{H^1_{-\varkappa}(\mathbb{R})}.$$

(ii) Define $g(x_1,x_2) := r - \frac{1}{2}e^{x_2} - \varkappa_1 e^{-x_2}\mathrm{sgn}(x_1) - \varkappa_2\sigma\rho\,\mathrm{sgn}(x_2)$ and $h(x_1,x_2) := \kappa\theta e^{-x_2} - \kappa - \varkappa_1\sigma\rho\,\mathrm{sgn}(x_1) - \varkappa_2\sigma^2 e^{-x_2}\mathrm{sgn}(x_2)$. For $u \in H^1_{-\varkappa}(\mathbb{R})$

$$\begin{aligned}
a^{-\varkappa}(u,u) = &\int_{\mathbb{R}^2}\Big(\alpha\nabla u(x,t)\Big)^T \nabla u(x,t)e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)}dx \\
&+ \int_{\mathbb{R}^2} g(x_1,x_2)\frac{du(x,t)}{dx_1}u(x,t)e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)} \\
&+ h(x_1,x_2)\frac{du(x,t)}{dx_2}u(x,t)e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)}dx \\
&+ \int_{\mathbb{R}^2} ru^2(x,t)e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)}dx.
\end{aligned}$$

Focusing on the convection term, we receive with partial integration

$$\begin{aligned}
&\frac{1}{2}\int_{\mathbb{R}^2} g(x_1,x_2)\frac{du}{dx_1}ue^{-2(\kappa_1|x_1|+\kappa_2|x_2|)} + h(x_1,x_2)\frac{du}{dx_2}ue^{-2(\kappa_1|x_1|+\kappa_2|x_2|)}dx \\
=&\frac{1}{2}\int_{\mathbb{R}}\int_0^\infty g(x_1,x_2)\frac{du}{dx_1}ue^{-2(\kappa_1|x_1|+\kappa_2|x_2|)}dx_1 + \int_{-\infty}^0 g(x_1,x_2)\frac{du}{dx_1}ue^{-2(\kappa_1|x_1|+\kappa_2|x_2|)}dx_1dx_2 \\
&+ \frac{1}{2}\int_{\mathbb{R}}\int_0^\infty h(x_1,x_2)\frac{du}{dx_2}ue^{-2(\kappa_1|x_1|+\kappa_2|x_2|)}dx_2 + \int_{-\infty}^0 h(x_1,x_2)\frac{du}{dx_2}ue^{-2(\kappa_1|x_1|+\kappa_2|x_2|)}dx_2dx_1 \\
=&\int_{\mathbb{R}}\Big(-\frac{1}{2}\Big(r - \frac{1}{2}e^{x_2} - \varkappa_1 e^{x_2} - \varkappa_2 e^{x_2} - \varkappa_2\sigma\rho\mathrm{sgn}(x_2)\Big) \\
&+ \frac{1}{2}\Big(r - \frac{1}{2}e^{x_2} + \varkappa_1 e^{x_2} - \varkappa_2 e^{x_2} - \varkappa_2\sigma\rho\mathrm{sgn}(x_2)\Big)\Big)u^2(0,x_2) \\
&- \frac{1}{2}\int_{\mathbb{R}} g(x_1,x_2)u^2(-2\mathrm{sgn}(x_1)\varkappa_1)e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)}dx_1dx_2 \\
&+ \int_{\mathbb{R}}\Big(-\frac{1}{2}\Big(-\kappa\theta - \kappa - \varkappa_1\sigma\rho\mathrm{sgn}(x_1) - \varkappa_2\sigma^2\Big) + \frac{1}{2}\Big(-\kappa\theta - \kappa - \varkappa_1\sigma\rho\mathrm{sgn}(x_1) + \varkappa_2\sigma^2\Big)\Big)u^2(x_1,0) \\
&- \frac{1}{2}\int_{\mathbb{R}}\Big(-\kappa\theta e^{-x_2} + \varkappa_2\sigma^2 e^{-x_2}\mathrm{sgn}(x_2) - 2\mathrm{sgn}(x_2)\varkappa_2 h(x_1,x_2)\Big)u^2 e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)}dx_2dx_1 \\
=&\int_{\mathbb{R}}\varkappa_1 e^{x_2}u^2(0,x_2)dx_2 + \int_{\mathbb{R}}\varkappa_2\sigma^2 u^2(x_1,0)dx_1 \\
&- \frac{1}{2}\int_{\mathbb{R}^2} g(x_1,x_2)u^2(-2\mathrm{sgn}(x_1)\varkappa_1)e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)}dx \\
&- \frac{1}{2}\int_{\mathbb{R}^2}\Big(-\kappa\theta e^{-x_2} + \varkappa_2\sigma^2 e^{-x_2}\mathrm{sgn}(x_2) - 2\mathrm{sgn}(x_2)\varkappa_2 h(x_1,x_2)\Big)u^2 e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)}dx.
\end{aligned}$$

Thus, there exist a $\lambda > 0$ such that

$$
\begin{aligned}
&a^{-\varkappa}(u,u) + \lambda |u|^2_{L_{2,-\varkappa}(\mathbb{R})} \\
&\geq \int_{\mathbb{R}^2} \Big(\alpha \nabla u(x,t)\Big)^T \nabla u(x,t) e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)} \\
&+ (\lambda + r - \frac{1}{2}(g(x_1,x_2)(-2\mathrm{sgn}(x_1)\varkappa_1) + (-\kappa\theta e^{-x_2} + \varkappa_2\sigma^2 e^{-x_2}\mathrm{sgn}(x_2) - 2\mathrm{sgn}(x_2)\varkappa_2 h(x_1,x_2)))) \\
&\quad u^2 e^{-2(\kappa_1|x_1|+\kappa_2|x_2|)} dx \\
&\geq \alpha |u|^2_{H^1_{-\varkappa}(\mathbb{R})} \quad \text{for } \alpha > 0.
\end{aligned}
$$

□

5.4.3. Localisation

As for the Black-Scholes model, cf. Section 5.3.3, we reformulate the space-time variational formulation before the domain truncation. We consider the PDE in transformed variables $\tilde{u} = ue^{-(\varkappa_1|x_1|+\varkappa_2|x_2|)} \in \mathbb{X}$ for $V = H^1_0(\mathbb{R}^2)$ and $\tilde{v} = ve^{-(\varkappa_1|x_1|+\varkappa_2|x_2|)} \in \mathbb{X}$ for $V = H^1_0(\mathbb{R}^2)$ and $H = L_2(\mathbb{R}^2)$. The bilinear form of the left-hand side is given by

$$
\begin{aligned}
\tilde{b}(\tilde{u}(x,t),\tilde{v}(x,t)) = &\int_{(0,T)}\int_{\mathbb{R}^2} \dot{\tilde{u}}(x,t)\tilde{v}_1(x,t)dxdt \\
&+ \int_{\mathbb{R}^2} \tilde{u}(x,0)\tilde{v}_2(x)dx \\
&+ \int_{(0,T)}\int_{\mathbb{R}^2} \alpha\nabla\tilde{u}(x,t)\nabla\tilde{v}_1(x,t)dx \\
&- \int_{\mathbb{R}^2} \begin{pmatrix} r - 0.5e^{x_2} + e^{x_2}\varkappa_1\mathrm{sgn}(x_1) + \rho\sigma\varkappa_2\mathrm{sgn}(x_2) \\ \kappa\theta e^{-x_2} - \kappa + \rho\sigma\varkappa_1\mathrm{sgn}(x_1) + \sigma^2 e^{-x_2}\varkappa_2\mathrm{sgn}(x_2) \end{pmatrix} \nabla\tilde{u}(x,t)\tilde{v}_1(x,t)dx \\
&+ \int_{\mathbb{R}^2} \Big(r - (r - \frac{1}{2}e^{x_2})\varkappa_1\mathrm{sgn}(x_1) - (\kappa\theta e^{-x_2} - \kappa)\varkappa_2\mathrm{sgn}(x_2) - \varkappa_1^2\mathrm{sgn}^2(x_1)\frac{1}{2}e^{x_2} \\
&- \rho\sigma\varkappa_1\varkappa_2\mathrm{sgn}(x_1)\mathrm{sgn}(x_2) - \frac{1}{2}\sigma^2 e^{-x_2}\varkappa_2^2\mathrm{sgn}^2(x_2)\Big)\tilde{u}(x,t)\tilde{v}_1(x,t)dx\,dt.
\end{aligned}
$$

For the transformed payoff $\tilde{p}(e^{x_1},e^{x_2}) = p(e^{x_1},e^{x_2})e^{-(\varkappa_1|x_1|+\varkappa_2|x_2|)}$ the right-hand side is defined as

$$
\tilde{f}(\tilde{v}) = \int_{\mathbb{R}^2} \tilde{p}(e^x)\tilde{v}_2(x)dx.
$$

We localise the domain $\mathbb{R}^2$ by truncation onto a subdomain $\Omega_{(R_1,R_2)} = \Omega_{R_1} \times \Omega_{R_2} := (-R_1,R_1)\times(-R_2,R_2)$, $\mathbf{R} := (R_1,R_2)$. Analogous to Proposition 5.3.4, the convergence towards the true solution can be ensured.

Proposition 5.4.2. Let $\mathbb{X}$ and $\mathbb{Y}$ be given by $V = H^1_0(\Omega_{\mathbf{R}})$ and $H = L_2(\Omega_{\mathbf{R}})$. Let u_{Hes} be the

true solution of the Heston model. Let $\tilde{u} \in \mathbb{X}$ be the solution of

$$\tilde{b}(u, v) = \tilde{f}(v) \quad \forall\, v \in \mathbb{Y}$$

with initial value $\tilde{p}(e^{x_1}, e^{x_2}) = p(e^{x_1}, e^{x_2}) e^{-(\varkappa_1 |x_1| + \varkappa_2 |x_2|)}$ for $p \in L_{2,-\varkappa}(\mathbb{R}^2)$. Then,

$$\|u_{\text{Hes}} - \tilde{u}\|_{\mathbb{X}} \leq C^2 2 R_1 R_2 p(e^{R_1/2}, e^{R_2/2})^2 e^{-(\varkappa_1 R_1 + \varkappa_2 R_2)}.$$

Proof. Analogous to Proposition 5.3.4 we use the a-priori energy estimate. Let $C > 0$, then

$$\|u\|_{\mathbb{X}}^2 \leq \|u\|_{L_2(I;V)}^2 + \|\dot{u}\|_{L_2(I;V')}^2 + \max_{t \in [0,T]} |u(t)|_H^2 \leq C^2 \Big(\|f\|_{L_2(I;V')}^2 + |u(0)|_H^2 \Big).$$

For the right-hand side $f \equiv 0$ we receive $\|u\|_{\mathbb{X}} \leq C |u(0)|_H$. Recall the cut-off function $\phi_{\mathbf{R},\mathbf{R}/2}$ introduced in the proof of Proposition 5.3.4. The approximation error of u_{Hes} and $\tilde{u}$ is then given by

$$\begin{aligned}
\|u_{\text{Hes}} - \tilde{u}\|_{\mathbb{X}}^2 &\leq C^2 |p(e^{x_1}, e^{x_2}) e^{-(\varkappa_1 |x_1| + \varkappa_2 |x_2|)} - \tilde{u}(0)|_{L_2(\Omega_{\mathbf{R}})}^2 \\
&\leq C^2 \inf_{w \in H_0^1(\Omega_{\mathbf{R}})} |p(e^{x_1}, e^{x_2}) e^{-(\varkappa_1 |x_1| + \varkappa_2 |x_2|)} - w|_{L_2(\Omega_{\mathbf{R}})}^2 \\
&\leq C^2 |p(e^{x_1}, e^{x_2}) e^{-(\varkappa_1 |x_1| + \varkappa_2 |x_2|)} - \phi_{R,R/2}(x) p(e^{x_1}, e^{x_2}) e^{-(\varkappa_1 |x_1| + \varkappa_2 |x_2|)}|_{L_2(\Omega_{\mathbf{R}})}^2 \\
&= C^2 \int_{\Omega_{\mathbf{R}} \setminus \Omega_{\mathbf{R}/2}} (1 - \phi_{R,R/2}(x))^2 p(e^{x_1}, e^{x_2})^2 e^{-2(\varkappa_1 |x_1| + \varkappa_2 |x_2|)} dx \\
&\leq C^2 2 R_1 R_2 \max_{x \in \Omega_{\mathbf{R}} \setminus \Omega_{\mathbf{R}/2}} p(e^{x_1}, e^{x_2})^2 e^{-2(\varkappa_1 |x_1| + \varkappa_2 |x_2|)} \\
&= C^2 2 R_1 R_2 p(e^{R_1/2}, e^{R_2/2})^2 e^{-(\varkappa_1 R_1 + \varkappa_2 R_2)}.
\end{aligned}$$

□

5.4.4. Numerical Experiments

We are going to price a call option with $\kappa = 0.8$, $\theta = 0.2$, $\rho = 0.3$, $\sigma = 0.6$, $r = 0.001$ and strike price $K = 60$. The model parameters ρ, κ, θ and σ are subject to change when pricing different options. Moreover, we would like to determine prices for different payoff functions, i.e., we consider the initial value p as a parameter function. The parameter space is given by $[0.5, 1.5] \times [0, 1] \times [-1, 1] \times [0, 1] \times L_{2,-\varkappa}(\Omega_R)$. The weight $\varkappa = (\varkappa_1, \varkappa_2)$ has to be chosen carefully: the interaction between $\varkappa_1$ and $\varkappa_2$ in combination with the different domain sizes given by $\mathbf{R} = (R_1, R_2)$ leads to very small values of the solution that cannot be represented on the computer due to the restriction given by the machine epsilon. After first numerical tests we choose $\varkappa_1 = 1.5$ and $\varkappa_2 = 0.5$.

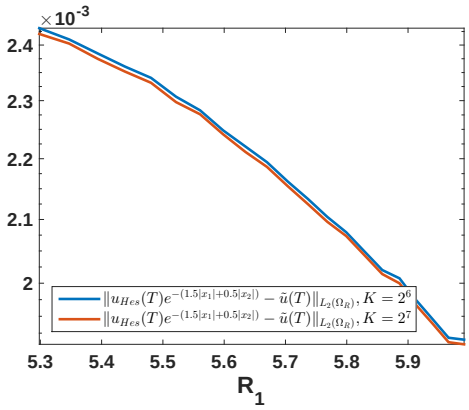


Figure 5.4.1.: Convergence of the PDE solution $\tilde{u}$ towards the closed form solution u_{Hes} w.r.t. $\mathbf{R}$. The FEM solution at final time point T is projected onto the finest grid given by 2^8 intervals, the Heston model is directly evaluated on the finest grid.

5.4.4.1. RBM for Parameter Functions

In Figure 5.4.1 the convergence error of the FEM solution at final time point T and the transformed closed form solution of the Heston model measured in the L_2-norm is given for different values of R_1. The quantity R_2 is fixed to $\ln(10)$. We set the boundaries of the truncated domain to $\mathbf{R} = (\ln(300), \ln(10))$ and $\mathcal{J}_1 = \mathcal{J}_2 = 2^6$, $\mathcal{K} = 2^5$ to obtain an L_2-approximation error at final time point T below $2.2 \cdot 10^{-3}$. We apply the RBM for parameter functions as presented in Section 3.2.2. We extend the wavelet basis functions to the 2D setting by considering the tensor product basis of the Haar wavelets.

Example 5.4.3. (Tensor Product Wavelets.) [Urb09, Sec. 8.3] Let $\phi_{[j,k]}, \psi_{[j,k]}$ be the 1D orthonormal scaling functions and wavelets on level j for $k = 0, \dots, 2^j - 1$. Let $(S_j)_{j \geq j_0}$ be a MRA, cf. Definition 3.2.7, given $\Omega \subset \mathbb{R}$. Let the set of all scaling functions build a basis for S_j. The set of all

$$\phi^{\text{2D}}_{[j,\ell]} := \phi_{[j,k]} \otimes \phi_{[j,k']}, \quad (k, k') \mapsto \ell$$

build 2D orthonormal scaling functions and a basis of $S_j^{\text{2D}} = S_j \oplus S_j$. The tensor product functions

$$\begin{aligned}
\psi^{\text{2D},1}_{[j,\ell]} &:= \psi_{[j,k]} \otimes \phi_{[j,k']} \\
\psi^{\text{2D},2}_{[j,\ell]} &:= \phi_{[j,k]} \otimes \psi_{[j,k']} \\
\psi^{\text{2D},3}_{[j,\ell]} &:= \psi_{[j,k]} \otimes \psi_{[j,k']}
\end{aligned}$$

build the corresponding orthonormal wavelet basis of W_j^{2D}, $S_{j+1}^{\text{2D}} = S_j^{\text{2D}} \otimes W_j^{\text{2D}}$. The corresponding wavelet decomposition of a 2D signal organised in a matrix is given by four parts: the coarse part is represented by the 2D scaling functions ϕ^{2D}. The horizontal detail corresponds to the

coefficients of $\psi^{2D,1}$, the vertical detail to $\psi^{2D,2}$ and the diagonal detail to $\psi^{2D,3}$. The coefficients of the wavelet expansion can be computed by performing the fast wavelet transformation on every row and column of the scaling expansion coefficients organised in a matrix.

The initial values in the initial value library $\mathbb{L}_{\text{init}}$ that correspond to the first wavelet on level 0 and on level 1 are displayed in Figure 5.4.2. In the 2D setting it is not advisable to use all wavelets on all levels up to the chosen discretisation. There are 16384 initial values to store in the initial value library and for a complete offline procedure, the computation of the same amount of reduced bases is necessary. We restrict to the computation of the 94 most relevant bases for the call option to show the resulting online approximation. However, the choice of an a-priori fix approximation space is advisable and reduces the resulting computational effort, cf. Section 3.4.2 and [MU14].

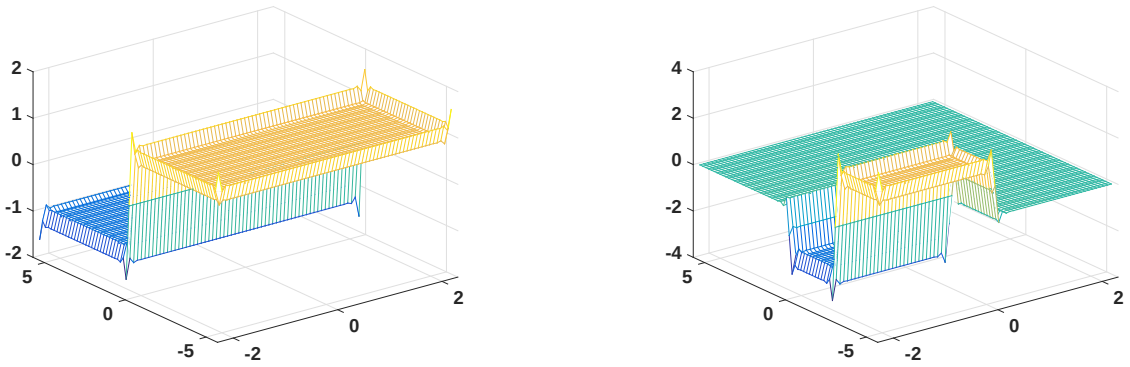
The Evolution Greedy is performed for a reduced parameter space $\rho \in \mathcal{D}_1 = [-1, 1] \subset \mathbb{R}$ to demonstrate the performance of the method with an reasonable computational effort. The training set is given by 11 equally distributed points in the parameter domain. The greedy tolerance is set to $\texttt{tol} = 0.1$. Here, $N_1 \in \{3, 4, 5\}$.

Given $r = 0.001$, we consider the test parameter set $\mu_1 = (\kappa, \theta, \rho, \sigma) = (0.8, 0.3, 0.2, 0.6)$ and a call option payoff $\mu_0 = p(e^x) = \max(e^x - 60, 0)$. There are 94 wavelet coefficients greater or equal to 10^{-4}. The online approximation error is given in Figure 5.4.3. For the estimator evaluation we assume an inf-sup constant of $\beta \geq 0.01$ – a direct numerical computation is not possible anymore because of the size of the discrete problem formulation. The estimator overestimates the error by 1.5 order of magnitude and follows the error behaviour as expected. The actual RB approximation error is below 10^{-1}, what is a good result for the before chosen Evolution Greedy tolerance.

The Crank Nicolson truth solver takes 1.274621 seconds for the evaluation. Even if we consider all 94 wavelets for the initial value approximation, the computational time of the RB simulation stays below the truth solver evaluation time, as we can see in Figure 5.4.4. E.g. for $N = 30$ we receive an RB approximation tolerance of 0.0237. The computational time of the RB simulation is given by 0.0343 seconds. In comparison to the finite element solver we only require 3 % of the computational time to achieve a solution. Consequently, if the optimisation in the model calibration process needs 50 function evaluations, we save over one hour of time replacing the finite element solver by the RB simulation.

5.4.4.2. H-Tucker Format in the Offline Procedure

We apply the H-Tucker format in the space-time offline phase as presented in Section 4.3.3. The performance of the CGNR method as truth solver highly depends on the tensor rank of the



(a) (b)

Figure 5.4.2.: Initial values w.r.t. $\psi^{2D,1}_{[0,0]}$ and $\psi^{2D,1}_{[1,0]}$.

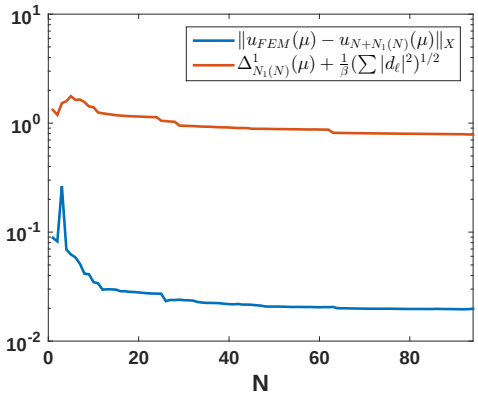


Figure 5.4.3.: Error and estimator for the online procedure.

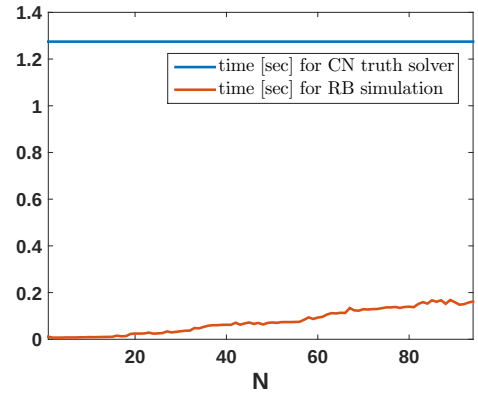


Figure 5.4.4.: Computational time in the online RB approximation.

underlying operator in CP decomposition. We fix the boundary by $R_1 = \ln(300)$ and $R_2 = \ln(10)$. For $r = 0.001$, the payoff $\tilde{p}(e^x) = \max(e^{x_1} - 60, 0)e^{-(1.5|x_1|+0.5|x_2|)}$ and the parameter μ_1 given by and $(\kappa, \theta, \rho, \sigma) = (0.8, 0.3, 0.2, 0.6)$ the test scenarios are given in Table 5.4.1. Here, the H-Tucker representation of the operator is of rank 13. For the test scenarios in Table 5.4.1 the results are limited by the computational resources due to the large tensor ranks of the operators. For the coarse discretisation in Scenario 1 and 2 the approximation error is below 10^{-1}, cf. Figure 5.4.5 (a). As expected, an increased truncation tolerance improves the convergence in cg-residual norm, Figure 5.4.5 (b). Here, the operator ranks for the preconditioners are still moderate, cf. Table 5.4.2. We only truncate the preconditioner $\mathcal{M}$ as we measure the error in the norm induced by the preconditioner $\mathcal{N}$. The spatial and temporal discretisation is increased in Scenario 3 and 4. Here, the operator ranks of the preconditioners increase additionally. The performed truncation prevents the convergence in the residual norm, cf. Figure 5.4.5 (c) and (d). The computational effort is significantly smaller when the operator is of smaller tensor

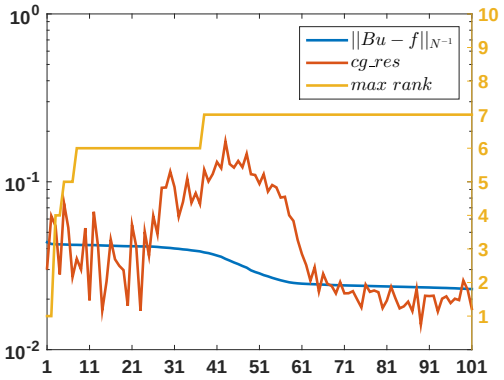
Scenario 1	$\mathcal{J}_1 = \mathcal{J}_2 = 2^3$	$\mathcal{K}^E = 2^3, \mathcal{K}^F = 2^4$	$\epsilon_{\text{rel}} = 10^{-2}$	max_rank = 50
Scenario 2	$\mathcal{J}_1 = \mathcal{J}_2 = 2^3$	$\mathcal{K}^E = 2^3, \mathcal{K}^F = 2^4$	$\epsilon_{\text{rel}} = 10^{-6}$	max_rank = 50
Scenario 3	$\mathcal{J}_1 = \mathcal{J}_2 = 2^6$	$\mathcal{K}^E = 2^6, \mathcal{K}^F = 2^7$	$\epsilon_{\text{rel}} = 10^{-2}$	max_rank = 50
Scenario 4	$\mathcal{J}_1 = \mathcal{J}_2 = 2^6$	$\mathcal{K}^E = 2^6, \mathcal{K}^F = 2^7$	$\epsilon_{\text{rel}} = 10^{-4}$	max_rank = 10

Table 5.4.1.: Scenarios 1–4 for the CGNR solver used to solve the Heston model.

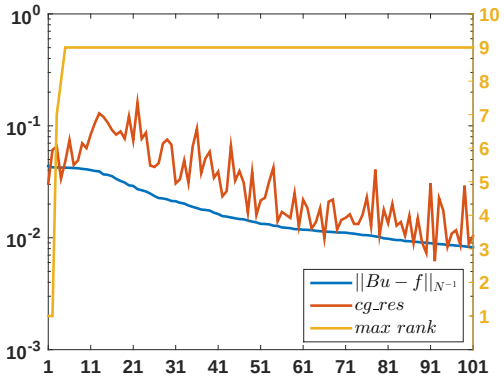
$\mathcal{J}_{1/2}$ / $\mathcal{K}^E$ / $\mathcal{K}^F$	$\mathcal{M}$	$\mathcal{N}$
2^3 / 2^3 / 2^4	1	6
2^4 / 2^4 / 2^5	1	7
2^5 / 2^5 / 2^6	5	8
2^6 / 2^6 / 2^7	5	9
2^7 / 2^7 / 2^8	8	10

Table 5.4.2.: Maximal tensor ranks for preconditioners $\mathcal{M}$ (truncated) and $\mathcal{N}$ (full) w.r.t. different spatial and temporal discretisation.

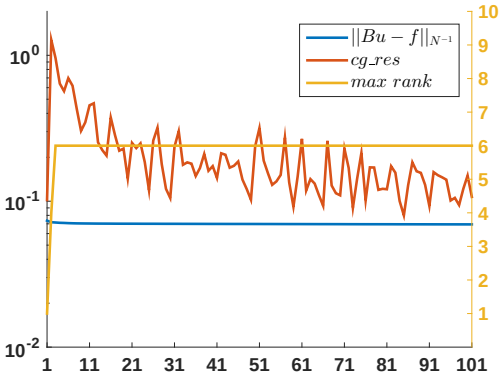
rank. We fix $\rho = 0$ as well as $r = 0$ and achieve a corresponding H-Tucker representation of the operator of rank 8. We again perform Scenario 3 and 4. A slow convergence in the residual norm is now observable, cf. Figure 5.4.6. Still, we only reach a tolerance below 10^{-1}. For the RBM we apply the setting given in Scenario 4. We reduce the parameter space to three components, κ, θ and σ. We set $\rho = r = 0$ and price a call option with strike price $K = 60$. The parameter domain is given by $(\kappa, \theta, \sigma) \in \mathcal{D} = [0.5, 1.5] \times [0, 1] \times [0, 1] \subset \mathbb{R}^3$. The training set is given by 11 equally distributed points in every parameter domain direction. Because of the large CGNR solver tolerance that we set on 0.03 (with a restriction to 100 iterations), the greedy tolerance has to be chosen accordingly and is set to $\texttt{tol} = 0.04$. This is a large tolerance and consequently the greedy only performs two steps and stops with a reduced basis of dimension $N = 3$. The first parameter in the sample set is given by $(0.5, 0, 0)$, the second is chosen to be $(1.5, 0, 0.5)$ with an



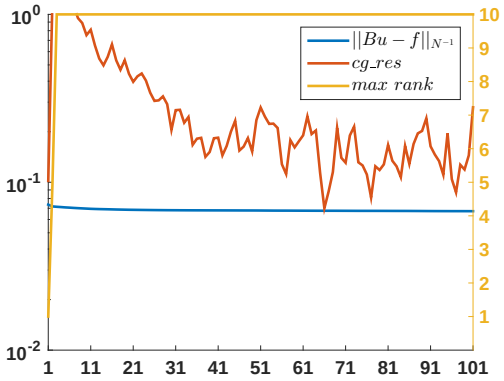
(a) Scenario 1.



(b) Scenario 2.

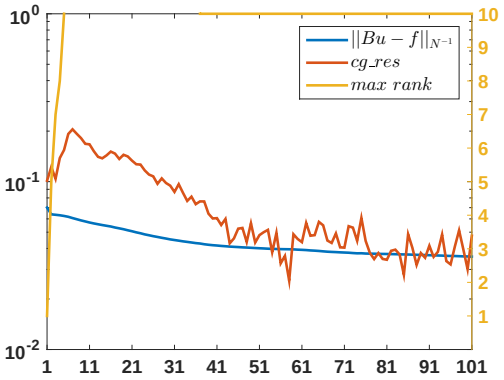


(c) Scenario 3.

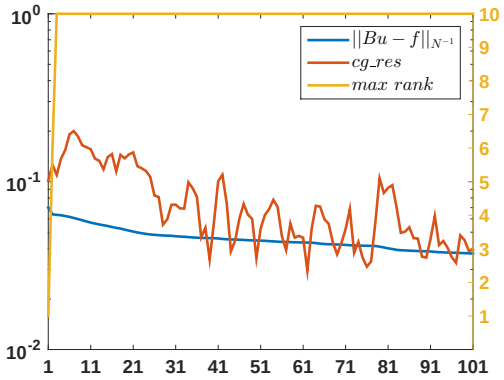


(d) Scenario 4.

Figure 5.4.5.: Convergence of the CGNR solver.



(a) Scenario 3.



(b) Scenario 4.

Figure 5.4.6.: Convergence of the CGNR solver for $\rho = r = 0$.

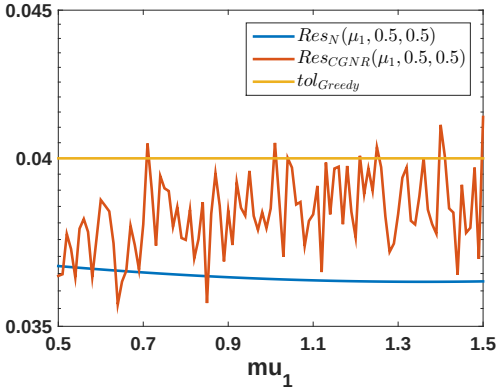


Figure 5.4.7.: Online RB approximation error in comparison to the CGNR approximation error for $\mu = (\kappa, 0.5, 0.5)$.

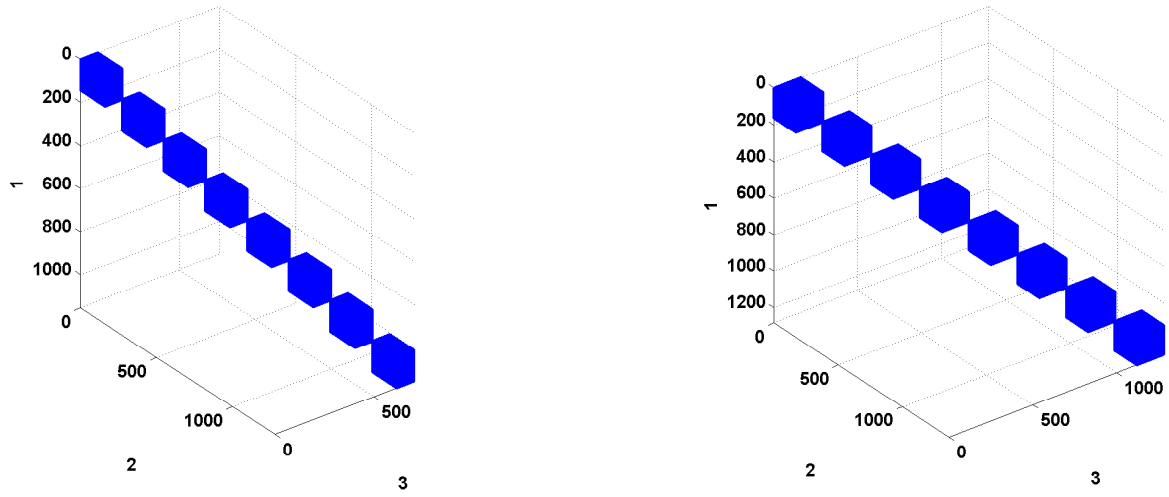
approximation error of 0.0566 and the third is $(1.5, 1, 0)$ with approximation error 0.0433.

For a test set of parameters we observe that the RB approximation errors stay below the before chosen greedy tolerance. The residual errors achieved with the CGNR solver are mainly above the RB approximation error and even above the chosen greedy tolerance, i.e., we cannot guarantee a successful greedy sampling, cf. Figure 5.4.7.

5.5. Conclusion

The developed space-time reduced basis method for parameter functions is applicable for both models and provides satisfying results. Using the RB approximation in the optimisation procedure of the model calibration process results in significant time savings. The more costly the finite element approximation is the higher is the time saving that can be achieved using the RB approximation. For the actual model calibration process we require the investigation of the PDE-constrained parameter optimisation using the RB approximation. This can be done by extending the work of M. Dihlmann and B. Haasdonk [DH15] to the space-time RBM framework. Allowing for the best N-term approximation of the initial value online, the resulting computational effort offline in a 2D setting is enormous and the use of an a-priori fix approximation space is advisable, cf. [MU14].

Further, we have seen in the numerical experiments in Section 4.4 that the application of the H-Tucker low rank tensor format in the offline phase of a standard space-time RBM is possible and can be very efficient. The application for the Heston model shows the limits of the existing methods and the underlying implementation. The tensor ranks of the operators set a clear limit in terms of computational feasibility. More precisely, for the Heston model and a discretisation



(a) $j = 1$ (b) $j = 2$

Figure 5.5.1.: Sparsity structure of transfer tensor b_0 of $\tilde{S}_j$.

given by $\mathcal{J}_1 = \mathcal{J}_2 = 2^7$, $\mathcal{K}^E = 2^7$, $\mathcal{K}^F = 2^8$ allowing for a maximal rank of 20 and a relative truncation tolerance of 10^{-4} we observe the following: in line 6 of Algorithm 4.3.2 the cg-residual is computed. The sparsity structure of the transfer tensor b_0 of $\tilde{S}_1$ and $\tilde{S}_2$ are displayed in Figure 5.5.1. The transfer tensor is stored as a full 3D matrix. In the first step, truncation can still be applied and the dimension reduces to $20 \times 20 \times 20$. In the second step the size of the transfer tensor grows again and the 3D object becomes infeasibly large for any further operation. The CGNR truth solver used in the offline phase only converges very slowly or not at all if the truncation procedure has to be applied extensively to provide the numerical feasibility. Here, an improved offline truth solver is required in the RB offline phase.

6. Conclusion and Outlook

The scope of this thesis is the space-time reduced basis method for its use in the pricing of financial products. The advantage in comparison to other reduction techniques for time dependent problems in the reduced basis framework is the additional model reduction in time. We present the underlying functional analytical framework and different RB simulation techniques.

We extend the original term of high dimensional parameter spaces to function spaces as parameter spaces. A reduced basis method was introduced, such that the initial value of a parabolic partial differential equation can be treated as a parameter in the RB setting. The method takes advantage of the underlying space-time variational formulation. The standard RB error estimator can be separated into two parts: the first that cares about the approximation quality of the initial value and the second that attends the error in the evolution of the solution. A finite decomposition of the function parameter is assumed, e.g. in terms of a (truncated) wavelet expansion. The first approach takes the same expansion for every possible parameter function online. This may result in a non-optimal initial value approximation and therefore the reduced basis approximation quality may suffer online. We overcome this problem allowing for a best N-term approximation of the parameter function online. Both methods have their justification. While the first one requires very reasonable storage capacities in the online phase to store the precomputed quantities and is therefore suitable for applications on small devices, the second requires an increased amount of storage but provides an improved parameter function approximation.

To overcome the computational costs in the offline phase of the space-time RBM and to allow for higher space dimensions, we investigate a possible use of the H-Tucker low rank tensor format in the RB offline phase. In the sampling procedure, all necessary operations are performed in low rank tensor format. The full system in space and time does not have to be established at any time. We present two approaches: The first still uses an underlying time stepping scheme as truth solver in the RBM, the second applies an iterative solver to the system in low-rank tensor format. In both versions the snapshots added in the greedy procedure to the RB are given in H-Tucker representation. Consequently, the snapshot reproducibility of the RBM fails. Both tolerances, the truncation tolerance in the truth solver and the greedy tolerance in the sampling procedure have to be chosen accordingly. We derive criteria how to determine the tolerances such that the RB offline phase can be performed. The numerical experiments show that for problems where the induced operator is of adequate tensor rank the method achieves very good results.

In fact, as soon as the iterative solver is applicable as truth solver, the RB approximation may even be of better quality by a significantly reduced computational effort.

The developed methods are applicable on diffusion models in general. To demonstrate the work in practice they are applied on two standard option pricing models, the Black-Scholes model and the Heston model. Weighted Sobolev spaces are introduced for the variational formulation. The RBM for parameter functions is tested on the Black-Scholes model. The method is applicable and we achieve good results in the online application pricing a standard call option. The Heston model is of two dimensions. Therefore, the computational effort increases in the offline phase of the space-time RBM for parameter functions. However, the method is still applicable and leads to the expected good results. Further, the use of the H-Tucker format in the offline phase of the space-time RBM of the Heston model is tested. The Heston model leads to an operator with a tensor representation of large rank. The large ranks are challenging for the CGNR truth solver – it does hardly converge to a satisfying approximation tolerance. This affects the RBM as the greedy tolerance has to be chosen accordingly.

All numerical experiments were implemented in the RBMatlab framework and an interface to the H-Tucker Toolbox is provided.

The concept of parameter functions in the initial value is expandable to parameter functions in the system operators. Here, an accurate knowledge of the application is required to provide a sophisticated error analysis. Using the H-Tucker low rank tensor format to improve the computational feasibility of the offline phase of the space-time RBM yields promising results. However, in real-world applications, the method is limited by the performance of the applied truth solver. Especially for the application in finance, the usage of the in [Kes13] developed multi-wavelets as underlying discretisation can be beneficial. We refer to [ASU14] for the treatment in the reduced basis method. Further steps have to be done towards RBMs for PIDEs, such that jump diffusion models are included.

A. Functional and Stochastic Analysis

This chapter collects shortly basic results in analysis and stochastic analysis. For the functional analysis we follow the books of D. Werner [Wer07] and W. Arendt and K. Urban [AU10]. For the stochastic analysis, we focus on the relevant results for the use in numerical/computational finance. We follow the introduction given in the monographs of R. Seydel [Sey09] and B. Øksendal [Øks03].

A.1. Functional Analysis

Let Ω be an open subset of $\mathbb{R}^n$. We denote by $\mathcal{C}^\infty(\Omega)$ the space of all infinitely often differentiable functions with values in $\mathbb{R}$. By $L_p(\Omega)$ we denote the standard real valued Lebesgue spaces of order $1 \leq p \leq \infty$, cf. [Wer07, Beispiel I.1 (h),(i)].

Definition A.1.1. [Wer07, Def. V.1.9] The set of all $\mathcal{C}^\infty$-functions with compact support is given by

$$\mathcal{C}_0^\infty(\Omega) := \{f \in \mathcal{C}^\infty(\Omega) : \operatorname{supp}(f) = \overline{\{x \in \Omega : f(x) \neq 0\}} \text{ is compact in } \Omega\}.$$

For general Sobolev spaces, the concept of weak derivatives is required and given in the next definition.

Definition A.1.2. [Wer07, Def. V.1.11] (Weak Derivative.) For a multiindex $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{N}_0^n$, $|\alpha| = \sum_{i=1}^n \alpha_i = m$, we define the weak α-th derivative of $f \in L_2(\Omega)$ by the function $D^{(\alpha)}f$ such that

$$\int_\Omega D^{(\alpha)}f(x)\,\phi(x)\,dx = (-1)^{|\alpha|}\int_\Omega f(x)\,D^{(\alpha)}\phi(x)\,dx \quad \forall \phi \in \mathcal{C}_0^\infty(\Omega)$$

with partial derivatives $D^{(\alpha)}\phi := \frac{\partial^{\alpha_1}\ldots\partial^{\alpha_n}}{\partial^{\alpha_1}x_1\cdots\partial^{\alpha_n}x_n}\phi$.

We define the Hilbert space $H^1(\Omega)$ by ([AU10, Satz 6.15])

$$H^1(\Omega) = \{f \in L_2(\Omega) \,|\, D^{(\alpha)}f \in L_2(\Omega),\ \forall\ |\alpha| \leq 1\}.$$

The associated inner product is given by

$$(u,v)_{H^1(\Omega)} = \int_\Omega u(x)v(x)dx + \int_\Omega \nabla u(x)\nabla v(x)dx$$

and the induced norm defined as

$$|u|^2_{H^1(\Omega)} = \int_\Omega u(x)^2 dx + \int_\Omega (\nabla u(x))^2 dx.$$

Further, the space $H^1_0(\Omega) := \{u \in H^1(\Omega) \,|\, \exists \phi_n \in \overline{\mathcal{C}^\infty_0(\Omega)}^{H^1(\Omega)} \text{ with } \lim_{n\to\infty}\phi_n = u \in H^1(\Omega)\}$ is a closed subspace of $H^1(\Omega)$ and itself a Hilbert space, cf. [AU10, Sec. 6.3]. We equip the space with the norm [AU10, Kor. 6.33]

$$|u|^2_{H^1_0(\Omega)} = \int_\Omega (\nabla u(x))^2 dx.$$

Proposition A.1.3. [AU10, Satz 4.21] (Riesz-Fréchet.) Let H be a Hilbert space and $\phi : H \to \mathbb{R}$ a continuous linear form. Then there exists one and only one $u \in H$ such that

$$\phi(v) = (u,v)_H \quad \forall v \in H.$$

For existence and uniqueness we are interested in densly embedded Hilbert spaces.

Definition A.1.4. Let H and V be two Hilbert spaces. We say that the space V is embedded in H if V is a subspace of H and there is a constant $c > 0$ such that

$$\|u\|_H \leq c\|u\|_V \quad \forall u \in V.$$

We denote the embedding by $V \hookrightarrow H$.

Example A.1.5. By definition,

1. $H^1(\Omega) \hookrightarrow L_2(\Omega)$ and
2. $H^1_0(\Omega) \hookrightarrow L_2(\Omega)$.

Lemma A.1.6. *[Wer07, Lemma V.1.10]* $\mathcal{C}^\infty_0(\Omega)$ *is dense in* $L_p(\Omega)$ *for* $1 \leq p < \infty$.

Corollary A.1.7. $H^1_0(\Omega)$ *is densely embedded in* $L_2(\Omega)$.

Proof. Let $\Omega \subset \mathbb{R}^n$ be open and $\partial\Omega$ a null set regarding the Lebesque measure. If $\mathcal{C}^\infty_0(\Omega)$ is dense in $L_2(\Omega)$ we know that there is for every equivalent class u in $L_2(\Omega)$ a sequence of functions in $\mathcal{C}^\infty_0$ such that $\|u - u_n\|_{L_2} \to 0$ for $n \to \infty$. By definition, $H^1_0(\Omega)$ is the closure of the set of $\mathcal{C}^\infty$-functions with compact support, regarding $|\cdot|_{H^1(\Omega)}$. Particularily, $u_n \in H^1_0(\Omega)$ for all $n \in \mathbb{N}$ so $H^1_0(\Omega) \hookrightarrow L_2(\Omega)$ dense. □

A.2. Stochastic Analysis

Definition A.2.1. [Sey09, Sec. 1.6] Let I be the interval $[0,T]$. A stochastic process X is a family of random variables X_t for a set of parameters $0 \leq t \leq T$.

Definition A.2.2. [Sey09, Def. 1.7] A Wiener process (or Brownian motion) W is a time continuous process with the properties

(a) $W_0 = 0$

(b) $W_t \sim \mathcal{N}(0,t)$ for all $t \geq 0$. That is, for each t the random variable W_t is normally distributed with mean $E(W_t) = 0$ and variance $Var(W_t) = E(W_t^2) = t$.

(c) All increments $\Delta W_t := W_{t+\Delta t} - W_t$ on non-overlapping time intervals are independent.

Definition A.2.3. For every step function b the Itô integral is defined as [Sey09, (1.29)]

$$\int_{t_0}^{t} b(s)\, dW_s := \sum_{j=1}^{N} b(t_{j-1})(W_{t_j} - W_{t_{j-1}}).$$

The Itô integral of a stochastically integrable function f is defined as

$$\int_{t_0}^{t} f(s) dW_s := \lim_{n\to\infty} \int_{t_0}^{t} b_n(s) dW_s$$

for simple processes b_n defined by [Sey09, (1.30)]

$$E\left(\int_{t_0}^{t} (f(s) - b_n(s))^2 ds\right) \to 0 \quad \text{for} \quad n \to \infty.$$

The integral of f is independent of the choice of the b_n.

Definition A.2.4. [Sey09, Def. 1.10] For $t \in [0,T]$, an Itô stochastic differential equation (SDE) is given by

$$dX_t = a(X_t, t)dt + b(X_t, t)dW_t; \tag{A.2.1}$$

this together with $X_{t_0} = X_0$ is a symbolic short form of the integral equation

$$X_t = X_{t_0} + \int_{t_0}^{t} a(X_s, s)ds + \int_{t_0}^{t} b(X_s, s)dW_s.$$

We call a solution X_t of the SDE (A.2.1) an Itô process or a stochastic diffusion. The function $a(X_t, t)$ is the *drift* term, $b(X_t, t)$ the *diffusion* of the SDE. Solutions to the SDE for a given realisation of the Wiener process are called *trajectories* or *paths*.

The same notation as in Definition A.2.4 applies if $X_t = (X_t^{(1)}, \ldots, X_t^{(n)})$ is actually an n-dimensional as well as $W_t = (W_t^{(1)}, \ldots, W_t^{(m)})$ an m-dimensional vector of stochastic processes, whereby the Wiener processes $W_t^{(i)}$ need not to be correlated, [Sey09, Sec. 1.7.5]. Here, $a(X_t, t)$ is a vector and $b(X_t, t)$ an $n \times m$-dimensional matrix. The interpretation of the SDE system is componentwise [Sey09, (1.41a)], i.e., for every $i = 1, \ldots, n$,

$$X_t^{(i)} = X_0^{(i)} + \int_{t_0}^{t} (a(X_s, s))_i ds + \sum_{k=1}^{m} \int_{t_0}^{t} (b(X_s, s))_{i,k} dW_s^{(k)}.$$

Example A.2.5. (i) The Wiener process is an Itô process given by the trivial SDE $dX_t = dW_t$.

(ii) The stochastic process S_t that solves the SDE A.2.1 for $a(S_t, t) = \mu S_t$ and $b(S_t, t) = \sigma S_t$ is called a geometric Brownian motion (GBM).

Remark A.2.6. (Risk-neutral valuation.) [Sey09, Sec. 1.7.3] Consider an SDE based option pricing model where the stock price dynamics follow a GBM. In the risk-neutral valuation, the original probability P is replaced by the adjusted risk-neutral probability Q. The (probably unknown) return rate μ is hereby replaced with the risk-free interest rate r. Additionally the original Wiener process W is removed and replaced by a standard Wiener process under the risk-neutral measure Q. Moving to the risk-neutral world implies a no-arbitrage assumption for the evaluation of the price. This leads to the computation of a fair value of the financial product.

B. Introduction to Reduced Basis Methods for Parabolic PDEs in Weak Formulation

The present thesis uses the space-time RBM as RB model reduction approach for parabolic PDEs. The RB approach that is alternatively used for parabolic PDEs is based on the weak formulation stated in Equation (2.2.4b) and uses a time stepping scheme online.

Reduced basis methods for parabolic PDEs have been introduced in the thesis of M. Grepl, [Gre05]. In comparison to the well-known elliptic case – and in contrast to the space-time reduced basis method – the additional temporal dimension is treated by imposing a time discretisation immediately. Let V be a Hilbert space. For a continuous and coercive bilinear form $a : V \times V \to \mathbb{R}$, the problem formulation reads: Find a solution u, $u(t) \in V$, such that

$$(\dot{u}(t), v)_H + \langle \mathcal{A}u(t), v\rangle_{V' \times V} = \langle g(t), v\rangle_{V' \times V} \text{ a.e. } t \in I,$$
$$u(0) = u_0 \text{ in } H.$$

A discretisation $0 = t_0 < t_1 < \ldots < t_{\mathcal{K}} = T$ of the time interval $I = [0, T]$ is assumed and the solution is obtained by solving a time stepping scheme for discrete $V^{\mathcal{N}} \subset V$. The solution of the PDE is given as set of elements $\{u_k^{\mathcal{N}}(\mu)\}_{k=0}^{\mathcal{K}} \subset V$. The RB space will also be a subset of the Hilbert space V. In particular, the RB reduction is only spatial and does not affect the time discretisation. Online, the same time stepping method is required to solve the reduced system. For some parameter $\mu \in \mathcal{D} \subset \mathbb{R}^d$, given initial value $u_N(0; \mu) = u_0^{\mathcal{N}}(\mu)$ and operators $\mathcal{L}_{\text{im}}(\mu), \mathcal{L}_{\text{ex}}(\mu) : V_N \to V_N$ as well as $b_t^N(\mu) \in \mathbb{R}^N$ for $t = 1, \ldots, \mathcal{K}$, a small linear equation system has to be solved online in every time step:

$$\mathcal{L}_{\text{im}}^N(\mu) u_N(t; \mu) = \mathcal{L}_{\text{ex}}^N(\mu) u_N(t-1; \mu) + b_t^N(\mu). \tag{B.0.1}$$

Using e.g. a Crank Nicolson scheme, the online time stepping is computationally costly since the scheme requires small time steps to satisfy the CFL-condition.

For symmetric bilinear forms, an a-posteriori error estimate for the error measured in a "spatio-temporal" energy norm $|||u(t_k; \mu)||| = \Big((\dot{u}(t_k; \mu), u(t_k; \mu))_H +$

$\sum_{k'=1}^{k} a(u(t_{k'};\mu),u(t_{k'};\mu);\mu)\Delta t\Big)^{1/2}$, [Gre05, Sec. 4.4.2 Prop.6], was introduced.

The theory was generalised to a reduced basis method for evolution equations in [HO08]. A finite volume approximation was introduced for the detailed solution, mainly caused by the considered hyperbolic problems. A generally defined parametrised evolution scheme was defined as

Definition B.0.1. [HO08, Def. 2.3] Let $\mathcal{L}_{\text{im}}(\mu), \mathcal{L}_{\text{ex}}(\mu) : V^{\mathcal{N}} \to V^{\mathcal{N}}$ and $P : L_2(\Omega) \to V^{\mathcal{N}}$ be continuous linear operators, $\mathcal{L}_{\text{im}}(\mu)$ positive definite, and $b_k(\mu) \in V^{\mathcal{N}}$. A parametrised evolution scheme is defined by

$$\begin{aligned} u_0^{\mathcal{N}}(\mu) &= P(u_0(\mu)), \\ \mathcal{L}_{\text{im}}(\mu)u_{k+1}^{\mathcal{N}}(\mu) &= \mathcal{L}_{\text{ex}}(\mu)u_k^{\mathcal{N}}(\mu) + b_k(\mu) \end{aligned}$$

and produces μ-dependent solutions $u_k^{\mathcal{N}}(\mu) \in V^{\mathcal{N}}$ for $0 = 1, \dots, \mathcal{K}$.

The reduced basis solution again is given as the set of solutions $\{u_N(k)\}_{k=0}^{\mathcal{K}} \subset V_N$ by solving (B.0.1). The introduced greedy method uses the proper orthogonal decomposition (POD), i.e. the principal component analysis (PCA) for the RB basis generation.

Definition B.0.2. (Basis-extension by PCA with fixspace.) [HO08, p. 293] The trajectory $\{u_k^{\mathcal{N}}\}_{k=0}^{\mathcal{K}}$ is projected on the orthogonal complement $(V_N)^{\perp}$ of the current RB space $V_N = \text{span}\{\phi_1, \dots, \phi_N\}$. The projected sequence is subject to a PCA. The principal component, i.e. the vector of maximum variance is chosen as ϕ_{N+1}, which is readily orthonormal to $\phi_1, \dots, \phi_N$ due to construction.

This is often referred to as the POD-Greedy method. A detailed presentation of the algorithm and the convergence analysis is presented in [Haa13]. The POD-Greedy is widely used e.g. in [EKP11] for the *hp* certified reduced basis method, to solve PDE-constrained optimisation problems [DH15], for optimal control problems, cf. [KG14], and for the solution of instationary variational inequalities, [BHSW15].

An a-posteriori estimator for the L_2-approximation error is presented.

Proposition B.0.3. [HO08, Lemma 4.1, Prop. 4.2] Let $u^{\mathcal{N}} = \{u_k^{\mathcal{N}}\}_{k=0}^{\mathcal{K}}$ denote the detailed solution of a parametrised evolution scheme and $u_N(\mu)$ denote the corresponding RB approximation. We assume that the RB space V_N contains the initial data $u_0^{\mathcal{N}}(\mu)$. For $k = 1, \dots, \mathcal{K} - 1$ we introduce the residuals

$$R^{k+1}(\mu) := \frac{1}{\Delta t_k}(\mathcal{L}_{\text{im}}^k(\mu)u_N^{k+1} - \mathcal{L}_{\text{ex}}^k(\mu)u_N^k - b^k(\mu)).$$

Then, the L_2-approximation error is bounded for all time-steps k by

$$|u_N^k(\mu) - u_k^{\mathcal{N}}(\mu)|_{L_2(\Omega)} \leq \sum_{n=0}^{k-1} \Delta t_n |R^{n+1}|_{L_2(\Omega)} (C_E)^{k-1-n} (C_I)^{k-n} =: \Delta_N^k(\mu).$$

The constants C_E and C_I are given by $\|(\mathcal{L}_{\text{im}})^{-1}\| \leq C_I$ and $\|\mathcal{L}_{\text{ex}}\| \leq C_E$ respectively.

All required components are offline-online decomposable.

Remark B.0.4. (1) An important point is that the error estimators imply the assumption $u_0(\mu) \in V_N$ for all μ. Otherwise the error in the initial condition has to be added on the right-hand side. The error is in general not offline-online decomposable and requires an $\mathcal{N}$-dependent computation. Adding the initial value directly in the RB space to not get an additional error fails when having a parameter function as initial value that is only given approximatively.

(2) The constant C_I is smaller or equal to one for a finite volume discretisation. This is not in general satisfied for a standard finite element discretisation. The error estimator might then become unusable large.

C. Scalar Multiplication and Addition in H-Tucker format

The multiplication of a tensor in H-Tucker format with a scalar value does not require an actual reconstruction of the full tensor but can directly be performed in the low rank tensor format. Same holds for the addition of two tensors in H-Tucker format.

C.1. Scalar Multiplication

A tensor $x \in \mathcal{H}$ of order d with H-Tucker representation $((b_t)_{t\in\mathcal{I}(T_d)}, (u_t)_{t\in\mathcal{L}(T_d)})$ is given by, cf. Equation (4.2.4),

$$x = \bigotimes_{i=0}^{d-1} u_{\{d-i\}} \prod_{\ell=1}^{\ell_{max}-1} \bigotimes_{t\in\mathcal{T}_d^\ell} b_t.$$

The multiplication with a scalar $c \in \mathbb{R}$ results in

$$cx = c\bigotimes_{i=0}^{d-1} u_{\{d-i\}} \prod_{\ell=1}^{\ell_{max}-1} \bigotimes_{t\in\mathcal{T}_d^\ell} b_t.$$

The standard way is to include the constant c in the root node $b_{\{1,\dots,d\}}$.

Lemma C.1.1. *[Rup14, Lemma 5.7] The tensor $y = cx \in \mathcal{H}$ is given by $u_t^y = u_t^x$ for $t \in \mathcal{L}(\mathcal{T}_d)$, $b_t^y = b_t^x$ for $t \in \mathcal{I}(\mathcal{T}_d) \setminus \{1,\dots,d\}$ and $b_{\{1,\dots,d\}}^y = cb_{\{1,\dots,d\}}^x$.*

C.2. Addition

The addition of two tensors in H-Tucker format does not require any actual computations but only a restorage.

Lemma C.2.1. *[Rup14, Lemma 5.8] Let $x, y \in \mathcal{H}$ be two tensors of order d with same dimension tree $\mathcal{T}_d$ and H-Tucker representation $((b_t^x)_{t\in\mathcal{I}(T_d)}, (u_t^x)_{t\in\mathcal{L}(T_d)})$ respectively $((b_t^y)_{t\in\mathcal{I}(T_d)}, (u_t^y)_{t\in\mathcal{L}(T_d)})$. The addition $x + y \in \mathcal{H}$ is defined by $u_t^{x+y} := [u_t^x, u_t^y] \in$*

$\mathbb{R}^{\max\{n_t^x, n_t^y\} \times r_t^x + r_t^y}$ *for all* $t \in \mathcal{L}(\mathcal{T}_d)$. *Note that the matrix* u_t *with less rows is embedded in the larger space by adding zeros. The transfer tensors are given by*

$$(b_t^{x+y})_{i,j,k} = \begin{cases} b_t^x & 1 \le i \le r_t^x, 1 \le j \le r_{t_\ell}^x, 1 \le k \le r_{t_r}^x \\ b_t^y & r_t^x + 1 \le i \le r_t^x + r_t^y, r_{t_\ell}^x + 1 \le j \le r_{t_\ell}^x + r_{t_\ell}^y, r_{t_r}^x + 1 \le k \le r_{t_r}^x + r_{t_r}^y \\ 0 & otherwise \end{cases}$$

for $t \in \mathcal{I}(\mathcal{T}_d) \setminus \{1, \dots, d\}$ *and*

$$(b_t^{x+y})_{j,k} = \begin{cases} b_t^x & 1 \le j \le r_{t_\ell}^x, 1 \le k \le r_{t_r}^x \\ b_t^y & r_{t_\ell}^x + 1 \le j \le r_{t_\ell}^x + r_{t_\ell}^y, r_{t_r}^x + 1 \le k \le r_{t_r}^x + r_{t_r}^y \\ 0 & otherwise \end{cases}$$

at $\{1, \dots, d\}$.

We do not proof the Lemma but verify the result for a 2-order tensor in the next example.

Example C.2.2. Let $x, y \in \mathcal{H}$ be 2-order tensor in hierarchical Tucker format. The dimension tree is given by $\{1, 2\}$ as root node and the splitting into $\{1\}$ and $\{2\}$, i.e.

$$x = (u_{\{1\}}^x \otimes u_{\{2\}}^x) b_{\{1,2\}}^x \quad \text{and} \quad y = (u_{\{1\}}^y \otimes u_{\{2\}}^y) b_{\{1,2\}}^y.$$

The tensor representation of $x + y$ is given by $(b_{12}^{x+y}, \{u_{\{1\}}^{x+y}, u_{\{2\}}^{x+y}\})$ described in Lemma C.2.1.

Proof.

$$\begin{aligned}
(u_{\{1\}}^{x+y} \otimes u_{\{2\}}^{x+y}) \mathrm{vec}(b_{12}^{x+y}) &= \Big([u_{\{1\}}^x, u_{\{1\}}^y] \otimes [u_{\{2\}}^x, u_{\{2\}}^y] \Big) \mathrm{vec} \begin{pmatrix} b_{\{1,2\}}^x & 0 \\ 0 & b_{\{1,2\}}^y \end{pmatrix} \\
&= \Big[u_{\{1\}}^x \otimes [u_{\{2\}}^x, u_{\{2\}}^y],\ u_{\{1\}}^y \otimes [u_{\{2\}}^x, u_{\{2\}}^y] \Big] \mathrm{vec}(\begin{pmatrix} b_{\{1,2\}}^x & 0 \\ 0 & b_{\{1,2\}}^y \end{pmatrix}) \\
&= \Big[u_{\{1\}}^x \otimes u_{\{2\}}^x,\ u_{\{1\}}^x \otimes u_{\{2\}}^y,\ u_{\{1\}}^y \otimes u_{\{2\}}^x,\ u_{\{1\}}^y \otimes u_{\{2\}}^y \Big] \begin{pmatrix} b_{\{1,2\}}^x \\ 0 \\ 0 \\ b_{\{1,2\}}^y \end{pmatrix} \\
&= (u_{\{1\}}^x \otimes u_{\{2\}}^x) b_{\{1,2\}}^x + (u_{\{1\}}^y \otimes u_{\{2\}}^y) b_{\{1,2\}}^y \\
&= x + y.
\end{aligned}$$

□

Bibliography

[And12] R. Andreev. *Stability of space-time Petrov-Galerkin discretizations for parabolic evolution equations*. PhD thesis, Eidgenössische Technische Hochschule ETH Zürich, Nr. 20842, 2012.

[And13] R. Andreev. Stability of sparse space–time finite element discretizations of linear parabolic evolution equations. *IMA J. Numer. Anal.*, 33(1):242–260, 2013.

[And14] R. Andreev. Space-time discretization of the heat equation A concise Matlab implementation. *Numerical Algorithms*, 67(4):713–731, 2014.

[ASU14] M. Ali, K. Steih, and K. Urban. Reduced basis methods based upon adaptive snapshot computations. *arXiv:1407.1708*, 2014.

[AT14] R. Andreev and Ch. Tobler. Multilevel preconditioning and low–rank tensor iteration for space–time simultaneous discretizations of parabolic PDEs. *Numerical Linear Algebra with Applications*, 2014.

[AU10] W. Arendt and K. Urban. *Partielle Differenzialgleichungen*. Springer-Verlag, 2010.

[BA72] I. Babuska and A. K. Aziz. The mathematical foundations of the finite element method with applications to partial differential equations. Technical report, 1972.

[Bab71] I. Babuška. Error-bounds for finite element method. *Numerische Mathematik*, 16(4):322–333, 1971.

[BCD+11] P. Binev, A. Cohen, W. Dahmen, R. DeVore, G. Petrova, and P. Wojtaszczyk. Convergence Rates for Greedy Algorithms in Reduced Basis Methods. *SIAM J. Math.l Anal.*, 43(3):1457–1472, 2011.

[BHSW15] O. Burkovska, B. Haasdonk, J. Salomon, and B. Wohlmuth. Reduced basis methods for pricing options with the Black-Scholes and Heston model. *SIAM J. Financial Math.*, 6(1):685–712, 2015.

[BMNP04] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. *C. R. Acad. Sci. Math.*, Ser. I 339:667–672, 2004.

[BPX90] J. H. Bramble, J. E. Pasciak, and J. Xu. Parallel multilevel preconditioners. *Mathematics of Computation*, 55(191):1–22, 1990.

[BR95] A Barrett and G Reddien. On the reduced basis method. *ZAMM - J. of Appl. Math. Mech./Z. Angew. Math. Mech.*, 75(7):543–549, 1995.

[Bra07] D. Braess. *Finite elements: Theory, fast solvers, and applications in solid mechanics*. Cambridge University Press, 4 edition, 2007.

[BS73] F. Black and M. Scholes. The pricing of options and corporate liabilities. *J. Pol. Economy*, 81:637–654, 1973.

[BS94] S. C. Brenner and R. Scott. *The mathematical theory of finite element methods*, volume 15. Springer Science & Business Media, 9 edition, 1994.

[BTWG08] T. Bui-Thanh, K. Willcox, and O. Ghattas. Model reduction for large-scale systems with high-dimensional parametric input space. *SIAM J. Sci. Comput.*, 30(6):3270–3288, 2008.

[CIR85] J. C. Cox, J. E. Ingersoll, and S. A. Ross. A theory of the term structure of interest rates. *Econometrica*, 53(2):385–407, 1985.

[CLP11] R. Cont, N. Lantos, and O. Pironneau. A reduced basis for option pricing. *SIAM J. Financial Math.*, 2(1):287–316, 2011.

[CT03] R. Cont and P. Tankov. *Financial Modelling with Jump Processes*, volume 1. Chapman and Hall/CRC, 2003.

[Dah03] W. Dahmen. Multiscale and wavelet methods for operator equations. In *Multiscale problems and methods in numerical simulations*, pages 31–96. Springer, 2003.

[DFGT07] J. Da Fonseca, M. Grasselli, and C. Tebaldi. Option pricing when correlations are stochastic: an analytical framework. *Review of Derivatives Research*, 10(2):151–180, 2007.

[DH15] M. A. Dihlmann and B. Haasdonk. Certified PDE-constrained parameter optimization using reduced basis surrogate models for evolution problems. *Comp. Opt. and App.*, 60(3):753–787, 2015.

[DK92] W. Dahmen and A. Kunoth. Multilevel preconditioning. *Numerische Mathematik*, 63(1):315–344, 1992.

[DL92] R. Dautray and J.-L. Lions. *Mathematical Analysis and Numerical Methods for Science and Technology: Volume 5 Evolution Problems I*. Springer Verlag, 1992.

[DLDMV00] L. De Lathauwer, B. De Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.*, 21(4):1253–1278, 2000.

[DMR13] D. Devaud, A. Manzoni, and G. Rozza. A combination between the reduced basis method and the ANOVA expansion: On the computation of sensitivity indices. *Comptes Rendus Mathematique*, 351(15):593–598, 2013.

[DPW14] W. Dahmen, Ch. Plesken, and G. Welper. Double greedy algorithms: reduced basis methods for transport dominated problems. *ESAIM: Math.l Model. Numer. Anal.*, 48(03):623–663, 2014.

[Dup94] B. Dupire. Pricing with a smile. *Risk*, 7(1):18–20, 1994.

[EKP11] J. L. Eftang, D. J. Knezevic, and A. T. Patera. An hp certified reduced basis method for parametrized parabolic partial differential equations. *Mathematical and Computer Modelling of Dynamical Systems*, 17(4):395–422, 2011.

[Eva99] L. C. Evans. Partial Differential Equations (Graduate Studies in Mathematics), 1999.

[FR83] J. P. Fink and W. C. Rheinboldt. On the Error Behavior of the Reduced Basis Technique for Nonlinear Finite Element Approximations. *ZAMM - J. Appl. Math. Mech./Z. Angew. Math. Mech.*, 63:21–28, 1983.

[GH11] L. Grasedyck and W. Hackbusch. An introduction to hierarchical (H-) rank and TT-rank of tensors with examples. *Comput. Methods Appl. Math.*, 11(3):291–304, 2011.

[GKT13] L. Grasedyck, D. Kressner, and C. Tobler. A literature survey of low-rank tensor approximation techniques. *GAMM-Mitteilungen*, 36(1):53–78, 2013.

[GO95] M. Griebel and P. Oswald. Tensor product type subspace splittings and multilevel iterative methods for anisotropic problems. *Advances in Comp. Math.*, 4(1):171–206, 1995.

[GO07] M. Griebel and D. Oeltz. A sparse grid space-time discretization scheme for parabolic problems. *Computing*, 81(1):1–34, 2007.

[Gra10] L. Grasedyck. Hierarchical singular value decomposition of tensors. *SIAM J. Matrix Anal. Appl.*, 31(4):2029–2054, 2010.

[Gre05] M. A. Grepl. *Reduced-basis approximation and a posteriori error estimation for parabolic partial differential equations.* PhD thesis, Massachusetts Institute of Technology, 2005.

[Gri15] F. Grimmer. Hierarchical Tensor Structures for High-Dimensional Parabolic Partial Differential Equations in Finance. Master's thesis, Universität Ulm, 2015.

[GU14] S. Glas and K. Urban. On Non-Coercive Variational Inequalities. *SIAM J. Numer. Anal.*, 52(5):2250–2271, 2014.

[GV12] A.-L. Gerner and K. Veroy. Certified reduced basis methods for parametrized saddle point problems. *SIAM J. Sci. Comput.*, 34(5):A2812–A2836, 2012.

[Haa13] B. Haasdonk. Convergence rates of the pod–greedy method. *ESAIM: Math. Model. Numer. Anal.*, 47(03):859–873, 2013.

[Haa14] B. Haasdonk. Reduced Basis Methods for Parametrized PDEs – A Tutorial Introduction for Stationary and Instationary Problems. *Reduced Order Modelling. Luminy Book series*, 2014.

[Hac12a] W. Hackbusch. *Tensor spaces and numerical tensor calculus*, volume 42. Springer Science & Business Media, 2012.

[Hac12b] W. Hackbusch. *Tensor spaces and numerical tensor calculus*, volume 42. Springer Science & Business Media, 2012.

[Hes93] S.L. Heston. A Closed-Form Solution for Options with Stochastic Volatility with Applications to Bond and Currency Options. *The Review of Financial Studies*, 6(2):327–343, 1993.

[Hie13] Hierarchical Tucker Toolbox. *Version 1.2, February 2013.* D. Kressner and Ch. Tobler, http://anchp.epfl.ch/htucker, 2013.

[Hil09] N. W. Hilber. *Stabilized wavelet methods for option pricing in high dimensional stochastic volatility models.* PhD thesis, ETH Zürich, Nr. 18176, 2009.

[HO08] B. Haasdonk and M. Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM: Math. Model. Numer. Anal.*, 42(02):277–302, 2008.

[HRS15] J. S. Hesthaven, G. Rozza, and B. Stamm. Certified Reduced Basis Methods for Parametrized Partial Differential Equations. *SpringerBriefs in Mathematics*, 2015.

[HRSP07] D. B. P. Huynh, G. Rozza, S. Sen, and A. T. Patera. A successive constraint linear optimization method for lower bounds of parametric coercivity and inf–sup stability constants. *Comptes Rendus Mathematique*, 345(8):473–478, 2007.

[HRSW13] N. Hilber, O. Reichmann, C. Schwab, and C. Winter. *Computational Methods for Quantitative Finance: Finite Element Methods for Derivative Pricing.* Springer Finance. Springer Berlin Heidelberg, 2013.

[HSW12] B. Haasdonk, J. Salomon, and B. Wohlmuth. A reduced basis method for the simulation of american options. In *Numerical Mathematics and Advanced Applications 2011*, pages 821–829. Springer, 2012.

[HSZ14] J. S. Hesthaven, B. Stamm, and S. Zhang. Efficient greedy algorithms for high-dimensional parameter spaces with applications to empirical interpolation and reduced basis methods. *ESAIM: Math. Model. Numer. Anal.*, 48(01):259–283, 2014.

[HZ11] J. S. Hesthaven and S. Zhang. On the use of ANOVA expansions in reduced basis methods for high-dimensional parametric partial differential equations. Technical report, Springer Verlag, 2011.

[Kat60] T. Kato. Estimation of iterated matrices, with application to the von Neumann condition. *Numerische Mathematik*, 2(1):22–29, 1960.

[KB09] T. G. Kolda and B. W. Bader. Tensor Decompositions and Applications. *SIAM Review*, 51(3):455–500, 2009.

[Kes13] S. Kestler. *On the adaptive tensor product wavelet Galerkin method with applications in finance.* PhD thesis, Universität Ulm, 2013.

[KG14] M. Kärcher and M. A. Grepl. *A Posteriori Error Estimation for Reduced Order Solutions of Parametrized Parabolic Optimal Control Problems.* ESAIM: Math. Model. Numer. Anal., 2014.

[KNP11] D. J. Knezevic, N. C. Nguyen, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for the parametrized unsteady Boussinesq equations. *Math. Models and Meth. in Appl. Sci.*, 21(07):1415–1442, 2011.

[KT12] D. Kressner and C. Tobler. htucker — A MATLAB toolbox for tensors in hierarchical Tucker format. *Mathicse, EPF Lausanne*, 2012.

[Kuf80] A. Kufner. *Weighted sobolev spaces*, volume 31. BG Teubner Stuttgart, 1980.

[LM72] J.-L. Lions and E Magenes. Non-homogeneous boundary value problems, Vol. I. *Grund. Math. Wiss*, 181, 1972.

[MAT14] MATLAB. *Version R2014b (8.4.0 150421).* The MathWorks, Inc., Natick, Massachusetts, United States, 2014.

[MNS05] A.-M. Matache, P.-A. Nitsche, and Ch. Schwab. Wavelet Galerkin Pricing of American Options on Lévy Driven Assets. *Quantitative Finance*, 5(4):403–424, 2005.

[MPR01] Y. Maday, A. T. Patera, and D. V. Rovas. A blackbox reduced-basis output bound method for noncoercive linear problems. *Studies in Math. and its Appl.*, 31:533–569, 2001.

[MPR02] Y. Maday, A. T. Patera, and D. V. Rovas. A blackbox reduced-basis output bound method for noncoercive linear problems. *Studies in Math. and its Appl.*, 31:533–569, 2002.

[MPS04] A.-M. Matache, T. von Petersdorff, and Ch. Schwab. Fast Deterministic Pricing of Options on Lévy Driven Assets. *ESAIM: Math. Model. Numer. Anal.*, 38(1):37–71, 2004.

[MU14] A. Mayerhofer and K. Urban. A Reduced Basis Method for Parabolic Partial Differential Equations with Parameter Functions and Application to Option Pricing. *J. Comp. Fin.*, to appear, 2014.

[Neč64] J. Nečas. Sur la coercivité des formes sesquilinéaires elliptiques. *Rev. Roumaine de Math. Pures et Appl.*, 9:47–69, 1964.

[Nou14] A. Nouy. Low-rank tensor approximation methods and Proper Generalized Decompositions. *Reduced Order Modelling. Luminy Book series*, 2014.

[NRP09] N. C. Nguyen, G. Rozza, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for the time-dependent viscous Burgers' equation. *Calcolo*, 46(3):157–185, 2009.

[Øks03] B. Øksendal. *Stochastic differential equations.* Springer, 2003.

[PGB15] B. Peherstorfer, P. Gómez, and H.-J. Bungartz. Reduced models for sparse grid discretizations of the multi-asset Black-Scholes equation. *Advances in Computational Mathematics*, pages 1–25, 2015.

[Pir09] O. Pironneau. Calibration of options on a reduced basis. *J. Comput. Appl. Math.*, 232(1):139–147, 2009.

[Pir12] O. Pironneau. Reduced Basis for Vanilla and Basket Options. *Risk and Decision Analysis*, 2(4):185–194, 2012.

[PR07] A. T. Patera and G. Rozza. Reduced Basis Approximation and A Posteriori Error Estimation for Parametrized Partial Differential Equations, 2007.

[PRV+02] Ch. Prud'Homme, D. V. Rovas, K. Veroy, L. Machiels, Y. Maday, A. Patera, and G. Turinici. Reduced–Basis Output Bound Methods for Parametrized Partial Differential Equations. In *Proceedings SMA Symposium*, 2002.

[QMN15] A. Quarteroni, A. Manzoni, and F. Negri. *Reduced Basis Methods for Partial Differential Equations: An Introduction*, volume 92. Springer, 2015.

[RBM13] RBMatlab. *Version as of 05/13/13.* M. Dihlmann, M. Drohmann, B. Haasdonk, M. Ohlberger and M. Schaefer, www.morepas.org, 2013.

[Rhe93] W. C. Rheinboldt. On the Theory and Error Estimation of the Reduced Basis Method for Multi-Parameter Problems. *Nonlin. Anal. Theory Methods Appl.*, 21(11):849–858, 1993.

[Rup14] A. Rupp. *High dimensional wavelet methods for structured financial products.* PhD thesis, Universität Ulm, 2014.

[RV07] G. Rozza and K. Veroy. On the stability of the reduced basis method for stokes equations in parametrized domains. *Comput. Methods in Appl. Mech. and Engrg.*, 196(7):1244–1260, 2007.

[Saa03] Y. Saad. *Iterative Methods for Sparse Linear Systems.* SIAM, 2003.

[San04] S. Sanfelici. Galerkin infinite element approximation for pricing barrier options and options with discontinuous payoff. *Decisions in Economics and Finance*, 27(2):125–151, 2004.

[Sey09] R. Seydel. *Tools for Computational Finance.* Springer-Verlag, Berlin, 2009.

[Sho97] R. E. Showalter. *Monotone operators in Banach Space and Nonlinear Partial Differential Equations*, volume 49. American Mathematical Soc., 1997.

[SS08] E. W. Sachs and M. Schu. *Reduced order models (POD) for calibration problems in finance.* Springer, 2008.

[SS09] Ch. Schwab and R. Stevenson. Space-time adaptive wavelet methods for parabolic evolution problems. *Mathematics of Computation*, 78(267):1293–1318, 2009.

[SS13] E. W. Sachs and M. Schu. A priori error estimates for reduced order models in finance. *ESAIM: Math. Model. Numer. Anal.*, 47(02):449–469, 2013.

[SU12] K. Steih and K. Urban. Space-time reduced basis methods for time-periodic parametric partial differential equations. *preprint, Ulm University*, 2012.

[Tob12] C. Tobler. *Low-rank tensor methods for linear systems and eigenvalue problems.* PhD thesis, ETH Zürich, 2012.

[Tuc63] L. R. Tucker. Implications of factor analysis of three-way matrices for measurement of change. *Problems in Measuring Change*, pages 122–137, 1963.

[Tuc66] L. R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31(3):279–311, 1966.

[UP14] K. Urban and A. T. Patera. An improved error bound for reduced basis approximation of linear parabolic problems. *Mathematics of Computation*, 83(288):1599–1615, 2014.

[Urb09] K. Urban. *Wavelet methods for elliptic partial differential equations*. Oxford University Press, 2009.

[UVZ14] K. Urban, S. Volkwein, and O. Zeeb. Greedy sampling using nonlinear optimization. In *Reduced order methods for modeling and computational reduction*, volume 9 of *MS&A. Model. Simul. Appl.*, pages 137–157. Springer, Cham, 2014.

[VPRP03] K. Veroy, Ch. Prud'homme, D. V. Rovas, and A. T. Patera. 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, 2003.

[WAW01] G. Winkler, T. Apel, and U. Wystup. Valuation of options in Heston's stochastic volatility model using finite element methods. *Foreign Exchange Risk*, pages 283–303, 2001.

[Wer07] D. Werner. *Funktionalanalysis*. Springer-Verlag, 6 edition, 2007.

[WZ76] J. Wloka and H.-C. Zapp. *Partielle Differentialgleichungen: Vorlesung von Prof. Dr. J. Wloka im SS 1975 an der Univ. Kiel*. Kiel, 1976.

[XZ03] J. Xu and L. Zikatanov. Some observations on Babuška and Brezzi theories. *Numerische Mathematik*, 94(1):195–202, 2003.

[Yan14] M. Yano. A Space-Time Petrov–Galerkin Certified Reduced Basis Method: Application to the Boussinesq Equations. *SIAM J. Sci. Comput.*, 36(1):A232–A266, 2014.

[YPU14] M. Yano, A. T. Patera, and K. Urban. A Space-Time Certified Reduced Basis Method For Burgers' Equation. *Math. Models and Meth. in Appl. Sci.*, 24(09):1903–1935, 2014.

List of Figures

List of Tables

List of Algorithms

List of Symbols

V	Hilbert space
H	Hilbert space
$\mathcal{N}$	dimension of detailed solution
$\mathcal{J}$	dimension of spatial domain discretisation
$\mathcal{K}$	dimension of temporal domain discretisation
$\mathcal{L}$	parameter function approximation space dimension
N	dimension of reduced space
$\mathbb{N}$	set of natural numbers
$\mathbb{R}$	set of real numbers
$\mathbb{Z}$	integers
$\mathcal{H}$	space of tensors in H-Tucker format
X	arbitrary ansatz space
$\mathsf{X}^{\mathcal{N}}$	discrete ansatz space
X_N	RB ansatz space
Y	arbitrary test space
$\mathsf{Y}^{\mathcal{N}}$	discrete test space
Y_N	RB test space
$\mathbb{X}$	ansatz space for the space-time variational formulation
$\mathbb{Y}$	test space for the space-time variational formulation
u	arbitrary function in X
$u^{\mathcal{N}}$	detailed solution
$u^{\mathcal{T}}$	truncated detailed solution
S_u	supremizing operator w.r.t. u
u^i	i-th RB element
u_N	RB solution
$\mathcal{D}$	parameter space
$\mathcal{D}^{\text{train}}$	parameter training set
d	parameter space dimension
$\mathcal{D}_0$	parameter function space
$\mathcal{D}_0^{\mathcal{L}}$	parameter function space approximation
$\mathcal{D}_1$	standard parameter space

Ξ_{X}^N	reduced basis for X_N
Ξ_{Y}^N	reduced basis for Y_N
μ	parameter, $\mu \in \mathcal{D}$
L_2	space of square integrable functions
L_1^{loc}	space of locally integrable functions
H^1	Hilbert space of weakly differentiable functions
H_0^1	subset of H^1 with zero boundary condition
$\mathcal{L}(U, W)$	space of bounded linear operators from U to W
$b(\cdot, \cdot)$	bilinear form on $\mathsf{X} \times \mathsf{Y}$
$a(\cdot, \cdot)$	bilinear form on $V \times V$
β	inf-sup constant
$\beta^{\mathcal{N}}$	discrete inf-sup constant
β_{LB}	lower bound of the inf-sup constant
C_b	continuity constant of bilinear form $b(\cdot, \cdot)$
$C_b^{\mathcal{N}}$	discrete continuity constant
C, c	constants
Ω	spatial domain
I	time interval
$\Delta_N(\mu)$	RB error bound
$\Delta_N^0(\mu)$	RB error bound for initial value approximation
$\Delta_N^1(\mu)$	RB error bound for evolution of the solution
$\otimes$	matrix tensor product
$<<$	much smaller
vec	vec-operator
$\mathcal{T}$	truncation operator
$\varkappa$	weight parameter
$H_{-\varkappa}^1$	exponentially weighted space containing H^1
$L_{2,-\varkappa}$	exponentially weighted space containing L_2
$E(\cdot)$	expected value

List of Acronyms

BRR	Brezzi-Rappaz-Raviart
CG	conjugate gradients
CGNR	conjugate gradient on the normal equation
EIM	Empirical Interpolation Method
FE	finite element
FEM	finite element method
FWT	Fast Wavelet Transform
GBM	geometric Brownian motion
MRA	multiresolution analysis
ONB	orthonormal basis
PDE	partial differential equation
POD	proper orthogonal decomposition
PPDE	parametrised partial differential equation
RB	reduced basis
RBM	reduced basis method
SCM	successive constraint method
SDE	stochastic differential equation
SVD	singular value decomposition

Danksagung

Da diese Doktorarbeit nicht ohne Unterstützung entstanden ist, möchte ich mich bei all denen bedanken, die zum Gelingen beigetragen haben.

Zuerst möchte ich mich bei Herrn Prof. Dr. Karsten Urban bedanken, der mir das Vertrauen entgegengebracht hat mich als Doktorvater bei meiner Arbeit zu betreuen und mich in dieser Position die letzten Jahre unterstützt hat: durch zahlreiche Denkanstöße in vielen Diskussionen, durch die wöchentliche RB Runde und mit Hilfe von vielen geduldigen Antworten auf meine Fragen. Mein Dank gilt auch Herrn Prof. Dr. Robert Stelzer, für die Unterstützung in vielen offenen Gesprächen, die Anregungen bezüglich der finanzmathematischen Anwendung und das entgegengebrachte Vertrauen in meine Fähigkeiten.

Die Arbeit ist zu einem großen Teil in meiner dreijährigen Zeit als Stipendiatin im DFG Graduiertenkolleg 1100 zur "Modellierung, Analyse und Simulation in der Wirtschaftsmathematik" entstanden. Ich bedanke mich daher bei der Deutschen Forschungsgemeinschaft für die finanzielle Unterstützung die mir eine sehr sorgenlose Promotionszeit ermöglicht hat. Ebenfalls bei Herrn Prof. Dr. Karsten Urban und Herrn Prof. Dr. Robert Stelzer bedanke ich mich für die Stellen als wissenschaftliche Mitarbeiterin am Institut für Numerische Mathematik und am Institut für Finanzmathematik, durch die ich meine Arbeit in meinem gewohnten wissenschaftlichen Umfeld fertig stellen konnte.

Ich hatte das große Glück mich gleich drei Einrichtungen/Instituten zugehörig fühlen zu dürfen: Vielen Dank an alle GKler für die schöne gemeinsame Promotions- und Ulmer Zeit. Ebenso Danke den Mitgliedern des Instituts für Numerische Mathematik und des Instituts für Finanzmathematik. Ihr alle habt dafür gesorgt, dass Ulm ein zweites Zuhause geworden ist. Besonders danke ich Silke Glas, Mladjan Radic, Mazen Ali, Stefan Hain, Kristina Steih und Martin Pietsch für das Korrekturlesen der Arbeit.

Meinen Eltern Werner und Christine sowie meinen Schwestern Marita und Cordula bin ich sehr dankbar für die Unterstützung auch und besonders im letzten halben Jahr, das immer-da-sein, die Geborgenheit und natürlich das Korrekturlesen. Bei Dorothee bedanke ich mich für unsere Freundschaft und das gemeinsame Promovieren.

Und Frederik – danke, dass du immer da bist und für unser Leben!

Erklärung

Ich versichere hiermit, dass ich die Arbeit selbständig angefertigt habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie die wörtlich oder inhaltlich übernommenen Stellen als solche kenntlich gemacht habe.

Ulm, den 15.01.2016

Lebenslauf

Der Lebenslauf wurde aus Gründen des Datenschutzes entfernt.