A Parallel and Adaptive Space-Time Discontinuous Galerkin Method for Visco-Elastic and Visco-Acoustic Waves

Zur Erlangung des akademischen Grades eines

DOKTORS DER NATURWISSENSCHAFTEN

von der KIT-Fakultät für Mathematik des
Karlsruher Instituts für Technologie (KIT)
genehmigte

DISSERTATION

von

M.Sc. Daniel Alexander Ziegler

Tag der mündlichen Prüfung: 20. November 2019
1. Referent: Prof. Dr. Christian Wieners
2. Referent: Prof. Dr. Willy Dörfler
3. Referent: Prof. Dr. Andreas Rieder

CONTENTS

CHAPTER

ONE

INTRODUCTION

Wave phenomena and their understanding are a challenging task in numerical sciences. A typical prototype of such wave problems are linear acoustic waves, which are subject to a wide field of research. They model the interaction of pressure waves with air or a gas at fixed temperature. Waves in solid cannot be described by the acoustic wave equation, because additional shear waves are observed. In this case we have to consider the elastic wave equations. Focusing on real materials, the energy of a seismic wave in a solid is partially transformed into heat. This attenuation effect is modeled by the Generalized Standard Linear Solid model (GSLS) and described by the visco-elastic wave equations. For this equations, we derive a variational setting and prove existence and uniqueness of the solution.

In chapter 3 we focus on the discretization. Since modern computer facilities are designed with an enormous number of processor cores, parallel realization of conventional methods becomes inefficient. The classical methods for solving time depending partial differential equations (PDEs) are the method of lines or Rothe's method. They use first a discretization in space or in time and then apply standard techniques for the other variable. Since the time can be interpreted as another spatial dimension, our method treats space and time simultaneously in a variational manner. We discretize the space with discontinuous Galerkin (dG) finite elements and construct upwind fluxes by solving a general Riemann problem. In time, we discretize either with discontinuous ansatz and test functions – resulting in what we call the dG-dG space-time

method – or continuous ansatz functions but discontinuous test functions – resulting in the discontinuous Galerkin - continuous Galerkin space-time discretization, a Petrov–Galerkin method which we abbreviate with dG-cPG.

Space-time discretizations result in huge linear systems. To master this challenges we introduce adaptive finite element techniques in chapter 4. In applications, as the solution is of interest only in a certain region, we use dual weighted residual estimators (DWR) to reduce the total amount of degrees of freedom without losing accuracy in the region of interest (RoI). We focus on error indicators which are efficient to compute. To solve the still huge linear system, we introduce a space-time multilevel preconditioner.

Finally, we perform in chapter 5 several numerical experiments. We begin with a simple almost homogeneous material, where we can construct an analytical solution. With this setting, we can investigate the convergence rates for the dG-dG and dG-cPG methods. Then we focus on heterogeneous materials inspired by geophysics. The Marmousi II benchmark gives a heterogeneous material distribution resulting in various velocities for the primary and secondary wave. Since in applications the wave is measured at certain points resulting in seismograms, we evaluate the numerical simulations by such point measurements.

1.1 Prepublication

Parts of the results of this work have been published in advance together with Prof. Dr. Willy Dörfler, Prof. Dr. Christian Wieners and Dr. Stefan Findeisen in “Parallel adaptive discontinuous Galerkin discretizations in space and time for linear elastic and acoustic waves”, see [DFWZ19].

The idea for the discontinuous Galerkin discretization in time has been published together with Dr. Fernando Gaspoz, Prof. Dr. Kunibert Siebert and Prof. Dr. Christian Kreuzer in “A convergent time-space adaptive dG(s) finite element method for parabolic problems motivated by equal error distribution”, see [GKSZ18].

1.2 Acknowledgement

The author gratefully acknowledges the support of the German Research Foundation (DFG) via Collaborative Research Center (CRC) 1173 »Wave phenomena: analysis and numerics«.

This work was performed on the supercomputer ForHLR II funded by the Ministry of Science, Research and the Arts Baden-Württemberg and by the Federal Ministry of Education and Research.

1.3 State of the art

The construction of space-time methods for time-dependent partial differential equations and their numerical analysis is an active and fast developing field, see [NSW17] for an overview of recent results.

The contributions include a broad spectrum of space-time methods. Approximation results and solution techniques for parabolic equations are presented in [Ste15, LMN16], time integration methods with parallelisation in time in [Gan15] and multigrid in time in [FFK$^+$14, VLN$^+$18]. One realization of space-time methods are Trefftz–discontinuous Galerkin methods. For acoustic wave problems of this method see [KMPS15, MP18] and for Maxwell's equations [EKSW15]. Variational space-time methods for the wave equation are treated in [KB14, BKRS18]. Space-time discontinuous Petrov–Galerkin (DPG) methods for the Schrödinger equation [DGNS17] and for acoustic waves [GS19] as well as a tent pitching scheme for hyperbolic systems [GSW17] can be found in [NSW17]. Regularity results in space and time for linear wave equations are considered in [MS16].

1.4 Notation and basic terms

Let $\mathbb{N}$ denote the natural numbers and $\mathbb{R}$ the real numbers.

We consider functions $u\colon \mathbb{R}^{\dim} \times \mathbb{R} \to \mathbb{R}$ with $\dim \in \{1, 2, 3\}$, where the last variable is the time variable t and the remaining variables are space variables $\mathbf{x} \in \mathbb{R}^{\dim}$. Such functions are elements of the so called *Bochner spaces*. We consider u as a function $u(t) = u(\cdot, t)$, which attains a value $u(t)$ that is a function of $\mathbf{x}$ and belongs to a suitable space of functions depending on $\mathbf{x}$. This means that $u(t)$ represents the mapping $\mathbf{x} \mapsto (u(t))(\mathbf{x}) = u(\mathbf{x}, t)$.

We denote the partial derivatives of a function u by

$$\partial_t := \frac{\partial}{\partial t}\,, \qquad \partial_d := \frac{\partial}{\partial x_d}\,.$$

Let Ω be a bounded domain in $\mathbb{R}^{\dim}$. We define the inner $\mathrm{L}_2(\Omega)$-product by

$$\langle \mathbf{v},\, \mathbf{w} \rangle_\Omega = \int_\Omega \mathbf{v} \cdot \mathbf{w}\, \mathrm{d}\mathbf{x} \qquad \text{for all } \mathbf{v}, \mathbf{w} \in \mathrm{L}_2(\Omega; \mathbb{R}^J)\,.$$

This can be extended to a space-time domain $Q := \Omega \times (0, T)$ by

$$\langle \mathbf{v},\, \mathbf{w} \rangle_Q = \int_0^T \int_\Omega \mathbf{v} \cdot \mathbf{w}\, \mathrm{d}\mathbf{x}\, \mathrm{d}t \qquad \text{for all } \mathbf{v}, \mathbf{w} \in \mathrm{L}_2(Q; \mathbb{R}^J)\,.$$

We use the induced norms

$$\|\cdot\|_\Omega^2 = \langle \cdot,\, \cdot \rangle_\Omega \qquad \text{and} \qquad \|\cdot\|_Q^2 = \langle \cdot,\, \cdot \rangle_Q\,.$$

With $\delta_{j,k}$ we denote the Kronecker delta

$$\delta_{j,k} = \begin{cases} 1 & \text{if } j = k \\ 0 & \text{else}\,. \end{cases}$$

The function $\mathbb{1}_I(\varphi)$ with an arbitrary interval $I = (a, b)$ denotes the characteristic function

$$\mathbb{1}_I(\varphi) = \begin{cases} 1 & \text{if } \varphi \in I \\ 0 & \text{else}\,. \end{cases}$$

CHAPTER

TWO

HYPERBOLIC SYSTEMS

2.1 General linear hyperbolic systems

Let Ω be a bounded polyhedral domain in $\mathbb{R}^{\dim}$ with $\dim \in \{1, 2, 3\}$ and $(0, T)$ a fixed time interval. This yields the space-time cylinder $Q = \Omega \times (0, T)$. We consider first order evolution equations of the following type

$$L\mathbf{u} = M\partial_t \mathbf{u} + A\mathbf{u} = \mathbf{b} \qquad \text{in } Q\,. \tag{2.1}$$

Here $M \in \mathrm{L}_\infty(\Omega; \mathbb{R}^{J\times J})$ is a symmetric and uniformly positive matrix, i.e., there exists a constant $c > 0$ such that for all $0 \neq \mathbf{v} \in \mathrm{L}_2(\Omega; \mathbb{R}^J)$ we get $\langle M\mathbf{v}, \mathbf{v}\rangle_\Omega \geq c\|\mathbf{v}\|_\Omega^2 > 0$. We assume, that the operator A can be written as

$$A\mathbf{v} = \sum_{d=1}^{\dim} \partial_d(B_d\mathbf{v}) = \sum_{d=1}^{\dim} B_d(\partial_d\mathbf{v}) \in \mathrm{L}_2(\Omega; \mathbb{R}^J)\,, \qquad \mathbf{v} \in \mathcal{D}(A) \subset \mathrm{L}_2(\Omega; \mathbb{R}^J)\,, \tag{2.2}$$

with symmetric matrices $B_d \in \mathbb{R}^{J\times J}_{\mathrm{sym}}$. The following definition of hyperbolic systems is given e.g. in [Eva10, Sec. 7.3].

Definition 2.1 (Hyperbolic)**.** A linear system of the form (2.1) is called *hyperbolic*, if for every $\mathbf{n} = (n_1, \cdots, n_{\dim})^T \in \mathbb{R}^{\dim}$ the $J \times J$ matrix

$$B = \sum_{d=1}^{\dim} n_d B_d \tag{2.3}$$

is diagonalizable with real eigenvalues.

2.2 Variational setting

Including attenuation effects requires to extend (2.1) by damping parameters into the operator D. We consider now the space-time differential operator L defined by

$$L\mathbf{u}(t) = M\partial_t \mathbf{u}(t) + A\mathbf{u}(t) + D\mathbf{u}(t)\,. \tag{2.4}$$

We assume that $M, D \in \mathrm{L}_\infty(\Omega; \mathbb{R}^{J\times J}_{\mathrm{sym}})$ and M is uniformly positive definite, whereas D is positive semi-definite. The analysis of the wave problems will be considered with homogeneous boundary conditions on $\partial\Omega$ which are realized by the choice of a suitable domain $\mathcal{D}(A)$. We assume that the hyperbolic differential operator A is skew-adjoint in the domain, i.e.,

$$\langle A\mathbf{v},\, \mathbf{w}\rangle_\Omega = -\langle \mathbf{v},\, A\mathbf{w}\rangle_\Omega \qquad \mathbf{v}, \mathbf{w} \in \mathcal{D}(A)\,. \tag{2.5}$$

The domain of the space-time operator L is given as

$$V = \mathcal{D}(L) = \overline{\{\mathbf{v} \in C^1(0,T;\mathrm{L}_2(\Omega;\mathbb{R}^J) \cap C^0(0,T;\mathcal{D}(A)) :\ \mathbf{v}(0) = \mathbf{0}\}}\,,$$

where the closure is taken with respect to the weighted graph norm

$$\|\mathbf{v}\|_V^2 = \langle M\mathbf{v},\, \mathbf{v}\rangle_Q + \left\langle M^{-1}L\mathbf{v},\, L\mathbf{v}\right\rangle_Q\,.$$

Then we define $W = \overline{L(V)} \subseteq \mathrm{L}_2(Q;\mathbb{R}^J)$ with the weighted norm

$$\|\mathbf{w}\|_W^2 = \langle M\mathbf{w},\, \mathbf{w}\rangle_Q\,.$$

We obtain the variational formulation by multiplying $L\mathbf{v}$ with a test function $\mathbf{w} \in W$ and integrate over the space-time domain Q. This defines the bilinear form $\mathcal{B}:\ V \times W \to \mathbb{R}$ with

$$\mathcal{B}(\mathbf{v},\mathbf{w}) = \langle L\mathbf{v},\, \mathbf{w}\rangle_Q\,. \tag{2.6}$$

We can establish the standard Babuška setting for this bilinear form which gives us the following theorem.

Theorem 2.1 (Nečas Theorem)**.** *Let $\mathcal{B} : V \times W \to \mathbb{R}$ be a continuous bilinear form and $W^* = \mathrm{L}_2(\Omega;\mathbb{R}^J)$ the dual space of W. Then the variational problem*

$$\textit{find } \mathbf{u} \in V: \quad \mathcal{B}(\mathbf{u},\mathbf{w}) = \langle \mathbf{b},\, \mathbf{w}\rangle_Q \qquad \forall\ \mathbf{w} \in W, \tag{2.7}$$

admits a unique solution $\mathbf{u} \in V$ *for all* $\mathbf{b} \in W^*$*, which depends continuously on* $\mathbf{b}$*, if and only if the bilinear form* $\mathcal{B}$ *satisfies one of the equivalent inf-sup conditions:*

1. *There exists* $\beta > 0$ *such that*

$$\begin{aligned} &\forall\ \mathbf{v} \in V \quad \sup_{\mathbf{w}\in W} \frac{\mathcal{B}(\mathbf{v},\mathbf{w})}{\|\mathbf{w}\|_W} \geq \beta \|\mathbf{v}\|_V\,; \\ &\forall\ 0 \neq \mathbf{w} \in W \quad \exists\ \mathbf{v} \in V: \quad \mathcal{B}(\mathbf{v},\mathbf{w}) \neq 0\,. \end{aligned} \tag{2.8}$$

2. *There holds*

$$\inf_{\mathbf{v}\in V} \sup_{\mathbf{w}\in W} \frac{b(\mathbf{v},\mathbf{w})}{\|\mathbf{v}\|_V \|\mathbf{w}\|_W} > 0\,, \quad \inf_{\mathbf{w}\in W} \sup_{\mathbf{v}\in V} \frac{b(\mathbf{v},\mathbf{w})}{\|\mathbf{v}\|_V \|\mathbf{w}\|_W} > 0\,. \tag{2.9}$$

3. *There exists* $\beta > 0$ *such that*

$$\inf_{\mathbf{v}\in V} \sup_{\mathbf{w}\in W} \frac{b(\mathbf{v},\mathbf{w})}{\|\mathbf{v}\|_V \|\mathbf{w}\|_W} = \inf_{\mathbf{w}\in W} \sup_{\mathbf{v}\in V} \frac{b(\mathbf{v},\mathbf{w})}{\|\mathbf{v}\|_V \|\mathbf{w}\|_W} = \beta\ . \tag{2.10}$$

In addition, the solution $\mathbf{u}$ *of* (2.7) *satisfies the stability estimate*

$$\|\mathbf{u}\|_V \leq \beta^{-1} \|\mathbf{b}\|_{W^*}\,. \tag{2.11}$$

The theorem and proof can be found in [NSV09, Theorem 2.2].

We can prove that the bilinear form (2.6) satisfies the first condition of Thm. 2.1 by using [DFW16, Lem. 1].

Lemma 2.1 (Continuity of $\mathcal{B}$)**.** *The bilinear form* $\mathcal{B}(\mathbf{v},\mathbf{w}) = \langle L\mathbf{v},\, \mathbf{w}\rangle_Q$ *with the space-time operator* L *defined by* (2.4) *is continuous on* $V \times W$*.*

Proof. We show that $\mathcal{B}$ is bounded and hence continuous, by using Cauchy–Schwarz inequality

$$\begin{aligned} |\mathcal{B}(\mathbf{v},\mathbf{w})|^2 = \langle L\mathbf{v},\, \mathbf{w}\rangle_Q^2 = \left\langle MM^{-1}L\mathbf{v},\, \mathbf{w}\right\rangle_Q^2 &\leq \|M^{-1}L\mathbf{v}\|_W^2\ \|\mathbf{w}\|_W^2 \\ &\leq (\|\mathbf{v}\|_W^2 + \|M^{-1}L\mathbf{v}\|_W^2)\ \|\mathbf{w}\|_W^2 = \|\mathbf{v}\|_V^2\ \|\mathbf{w}\|_W^2\,. \end{aligned}$$

□

Lemma 2.2 (Inf-sup condition for $\mathcal{B}$)**.** *Assume that A and D are positive semi-definite. Then, the bilinear form $\mathcal{B}(\mathbf{v}, \mathbf{w}) = \langle L\mathbf{v}, \mathbf{w}\rangle_Q$ with the space-time operator L defined by* (2.4) *satisfies the inf-sup condition.*

Proof. Since the conditions in Thm. 2.1 are equivalent, we show that the bilinear form fulfills the first condition. We first note that for all $\mathbf{v} \in C^1(0,T;\mathrm{L}_2(\Omega;\mathbb{R}^J)\cap C^0(0,T;\mathcal{D}(A))$ with $\mathbf{v}(0) = \mathbf{0}$ we have

$$\begin{aligned}
\|\mathbf{v}\|_W^2 &= \int_0^T \langle M\mathbf{v}(t),\, \mathbf{v}(t)\rangle_\Omega \,\mathrm{d}t \\
&= \int_0^T \left(\langle M\mathbf{v}(t),\, \mathbf{v}(t)\rangle_\Omega + \langle M\mathbf{v}(0),\, \mathbf{v}(0)\rangle_\Omega\right) \mathrm{d}t \\
&= \int_0^T \int_0^t \partial_t \langle M\mathbf{v}(s),\, \mathbf{v}(s)\rangle_\Omega \,\mathrm{d}s\,\mathrm{d}t = 2\int_0^T \int_0^t \langle M\partial_t\mathbf{v}(s),\, \mathbf{v}(s)\rangle_\Omega \,\mathrm{d}s\,\mathrm{d}t \\
&\le 2\int_0^T \int_0^t \langle M\partial_t\mathbf{v}(s) + A\mathbf{v}(s) + D\mathbf{v}(s),\, \mathbf{v}(s)\rangle_\Omega \,\mathrm{d}s\,\mathrm{d}t \\
&\le 2\int_0^T \int_0^t \left\langle M^{-1}L\mathbf{v}(s),\, L\mathbf{v}(s)\right\rangle_\Omega^{1/2} \langle M\mathbf{v}(s),\, \mathbf{v}(s)\rangle_\Omega^{1/2} \,\mathrm{d}s\,\mathrm{d}t \\
&\le 2T\|M^{-1}L\mathbf{v}\|_W \|\mathbf{w}\|_W \,.
\end{aligned}$$

This yields $\|\mathbf{v}\|_W \le 2T\|M^{-1}L\mathbf{v}\|_W$. For $\|\mathbf{v}\|_V$ we get

$$\begin{aligned}
\|\mathbf{v}\|_V^2 &= \|\mathbf{v}\|_W^2 + \|M^{-1}L\mathbf{v}\|_W^2 \le 4T^2\|M^{-1}L\mathbf{v}\|_W^2 + \|M^{-1}L\mathbf{v}\|_W^2 \\
&= (4T^2+1)\|M^{-1}L\mathbf{v}\|_W^2 \,.
\end{aligned}$$

By inserting the special choice $\mathbf{w} = M^{-1}L\mathbf{v} \in W\backslash\{\mathbf{0}\}$ into (2.8) we get

$$\begin{aligned}
\sup_{\mathbf{w}\in W} \frac{\mathcal{B}(\mathbf{v},\mathbf{w})}{\|\mathbf{w}\|_W} \ge \frac{\mathcal{B}(\mathbf{v}, M^{-1}L\mathbf{v})}{\|M^{-1}L\mathbf{v}\|_W} &= \frac{\langle L\mathbf{v},\, M^{-1}L\mathbf{v}\rangle_Q}{\|M^{-1}L\mathbf{v}\|_W} \\
&= \|M^{-1}L\mathbf{v}\|_W \ge (4T^2+1)^{-1/2}\|\mathbf{v}\|_V \,.
\end{aligned}$$

For the proof of the second condition, we refer to [Ern18, Sec. 3.2]. □

Remark 2.1. Thm. 2.1 ensures existence and uniqueness of the solution $\mathbf{u}$. The stability estimate holds with $\beta = (4T^2+1)^{-1/2}$.

The constant β of Rem. 2.1 could be improved in [EW19, Lem. 4] to the constant $\beta = (T^2+2)^{-1/2}$.

The following lemma shows that H^1-in-time regular functions $\mathbf{v}$ have point-evaluations $\mathbf{v}(t) \in \mathrm{L}_2(\Omega;\mathbb{R}^J)$. We define the space $H = \mathrm{L}_2(\Omega;\mathbb{R}^J)$ equipped with the norm $\|\cdot\|_H^2 = \langle M\cdot,\, \cdot\rangle_\Omega$.

Lemma 2.3. *For $t \in [0,T]$ the mapping $\mathrm{H}^1(0,T;\mathrm{L}_2(\Omega;\mathbb{R}^J)) \longrightarrow \mathrm{L}_2(\Omega;\mathbb{R}^J)$, $\mathbf{v} \longmapsto \mathbf{v}(t)$, is well-defined and allows the bound*

$$\begin{aligned}\left\|\mathbf{v}(t)\right\|_H &:= \sqrt{\langle M\mathbf{v}(t),\, \mathbf{v}(t)\rangle_\Omega} \\ &\leq \sqrt{\frac{2}{T}}\left\|\mathbf{v}\right\|_W + \sqrt{\frac{T}{2}}\left\|\partial_t\mathbf{v}\right\|_W, \qquad \mathbf{v} \in \mathrm{H}^1(0,T;\mathrm{L}_2(\Omega;\mathbb{R}^J))\,.\end{aligned}$$

Proof. For $t_0 \in [0,T] \setminus \{t\}$ with $|t - t_0| > T/2$ we define the scaling function $d_0(s) = (s - t_0)/(t - t_0)$. This yields for $\mathbf{v} \in \mathrm{H}^1(0,T;\mathrm{L}_2(\Omega;\mathbb{R}^J))$

$$\begin{aligned}\left\|\mathbf{v}(t)\right\|_H &= \langle d(t)M\mathbf{v}(t),\, \mathbf{v}(t)\rangle_\Omega - \langle d(t_0)M\mathbf{v}(t_0),\, \mathbf{v}(t_0)\rangle_\Omega \\ &= \int_{t_0}^{t} \partial_s \langle d(s)M\mathbf{v}(s),\, \mathbf{v}(s)\rangle_\Omega \,\mathrm{d}s \\ &= \frac{1}{t-t_0}\int_{t_0}^{t} \langle M\mathbf{v}(s),\, \mathbf{v}(s)\rangle_\Omega \,\mathrm{d}s + 2\int_{t_0}^{t} \langle d(s)M\partial_s\mathbf{v}(s),\, \mathbf{v}(s)\rangle_\Omega \,\mathrm{d}s \\ &\leq \frac{2}{T}\left\|\mathbf{v}\right\|_W^2 + 2\left\|\partial_t\mathbf{v}\right\|_W\left\|\mathbf{v}\right\|_W \\ &\leq \frac{2}{T}\Big(\left\|\mathbf{v}\right\|_W + \frac{T}{2}\left\|\partial_t\mathbf{v}\right\|_W\Big)^2.\end{aligned}$$

□

2.3 Elastic waves in solids

In dynamic models in continuum mechanics, the motion of a material point $\mathbf{x}$ in the reference configuration Ω at time t is described by the deformation vector $\boldsymbol{\varphi}(\mathbf{x},t)$. The velocity is denoted by $\mathbf{v} = \partial_t\boldsymbol{\varphi}$. Elastic waves are determined by Newton's law for the balance of momentum

$$\rho\partial_t\mathbf{v} = \operatorname{div}\boldsymbol{\sigma} + \mathbf{b}\,,$$

with the mass density ρ, acceleration $\partial_t\mathbf{v}$, and the vector of body forces $\mathbf{b}$, together with a constitutive relation for the stress $\boldsymbol{\sigma}$ depending on the deformation gradient $\mathbf{F} = \mathrm{D}\boldsymbol{\varphi}$. For elastic materials a response function $\hat{\Sigma}(\cdot)$ exists so that the stress is determined by the response $\boldsymbol{\sigma} = \hat{\Sigma}(\mathbf{F})$. Then the stress rate is given by

$$\partial_t\boldsymbol{\sigma} = \mathrm{D}\hat{\Sigma}(\mathrm{D}\boldsymbol{\varphi})(\mathrm{D}\mathbf{v})\,.$$

Assuming small strains and $\boldsymbol{\varphi} \approx \mathrm{id}$, this is approximated by its linearization

$$\partial_t\boldsymbol{\sigma} = \mathbf{C}\boldsymbol{\varepsilon}(\mathbf{v})\,, \qquad \boldsymbol{\varepsilon}(\mathbf{v}) = \operatorname{sym}(\mathrm{D}\mathbf{v})$$

with the elasticity tensor $\mathbf{C} = \mathrm{D}\hat{\Sigma}(\mathbf{I})$. The balance of torsional moments yields that the stress is symmetric and that the stress rate only depends on the symmetric strain rate. In isotropic media the elasticity tensor $\mathbf{C}\boldsymbol{\varepsilon} = 2\mu\boldsymbol{\varepsilon} + \lambda\,\mathrm{trace}(\boldsymbol{\varepsilon})\mathbf{I}$ is characterized by the Lamé parameters $\lambda \geq 0$, $\mu > 0$. Introducing the compression modulus $\kappa = \frac{2\mu+3\lambda}{3}$ and the deviatoric stress $\mathrm{dev}(\boldsymbol{\sigma}) = \boldsymbol{\sigma} - \frac{1}{3}\,\mathrm{trace}(\boldsymbol{\sigma})\mathbf{I}$ we obtain

$$\begin{aligned} \mathbf{C}(\mu,\kappa)\boldsymbol{\varepsilon} &= 2\mu\,\mathrm{dev}(\boldsymbol{\varepsilon}) + \kappa\,\mathrm{trace}(\boldsymbol{\varepsilon})\mathbf{I}\,, \\ \mathbf{C}^{-1}(\mu,\kappa)\boldsymbol{\sigma} &= \frac{1}{2\mu}\,\mathrm{dev}(\boldsymbol{\sigma}) + \frac{1}{3\kappa}\,\mathrm{trace}(\boldsymbol{\sigma})\mathbf{I}\,. \end{aligned} \tag{2.12}$$

Remark 2.2. To reduce this system form 3D to 2D we can use the approach of plain strain ($\varepsilon_{33} = 0$) or plain stress ($\sigma_{33} = 0$).

The space-time operator L for the elastic wave equation uses the mass and hyperbolic operators

$$M = \begin{pmatrix} \rho & 0 \\ 0 & \mathbf{C}^{-1} \end{pmatrix} \qquad \text{and} \qquad A = -\begin{pmatrix} 0 & \mathrm{div}(\cdot) \\ \boldsymbol{\varepsilon}(\cdot) & 0 \end{pmatrix}. \tag{2.13}$$

The solution vector is $\mathbf{u} = (\mathbf{v}, \boldsymbol{\sigma})^\top$. Since no damping is included in this model, the damping operator vanishes, i.e., $D = \mathbf{0}$.

Lemma 2.4. *The operator A defined in* (2.13) *is positive semi-definite on the domain $\mathcal{D}(A) = \mathrm{H}_0^1(\Omega;\mathbb{R}^{\dim}) \times \mathrm{H}(\mathrm{div},\Omega;\mathbb{R}^{\dim\times\dim}_{\mathrm{sym}})$ and hence Lem. 2.2 holds true.*

Proof. For all $\mathbf{u} = \begin{pmatrix} \boldsymbol{\sigma} \\ \mathbf{v} \end{pmatrix} \in \mathcal{D}(A)$ it holds that

$$\begin{aligned} \langle A\mathbf{u},\, \mathbf{u}\rangle_\Omega &= \Big\langle -\big(\mathrm{div}(\boldsymbol{\sigma})\,,\, \boldsymbol{\varepsilon}(\mathbf{v})\big)^\top,\, \big(\mathbf{v}\,,\, \boldsymbol{\sigma}\big)^\top \Big\rangle_\Omega \\ &= -\int_\Omega \left(\mathrm{div}(\boldsymbol{\sigma})\cdot\mathbf{v} + \boldsymbol{\varepsilon}(\mathbf{v}) : \boldsymbol{\sigma}\right) \mathrm{d}\mathbf{x} \\ &= -\int_\Omega \left(\mathrm{div}(\boldsymbol{\sigma}\cdot\mathbf{v}) - \mathrm{D}\mathbf{v} : \boldsymbol{\sigma} + \boldsymbol{\varepsilon}(\mathbf{v}) : \boldsymbol{\sigma}\right) \mathrm{d}\mathbf{x} \\ &= -\int_\Omega \left(\mathrm{div}(\boldsymbol{\sigma}\cdot\mathbf{v}) - \boldsymbol{\varepsilon}(\mathbf{v}) : \boldsymbol{\sigma} + \boldsymbol{\varepsilon}(\mathbf{v}) : \boldsymbol{\sigma}\right) \mathrm{d}\mathbf{x} \\ &= -\int_\Omega \mathrm{div}(\boldsymbol{\sigma}\cdot\mathbf{v})\,\mathrm{d}\mathbf{x} = -\int_{\partial\Omega} \mathbf{v}\cdot\boldsymbol{\sigma}\mathbf{n}\,\mathrm{d}\mathbf{x} = 0\,. \end{aligned}$$

□

2.4 Acoustic waves

Assuming that shear forces can be neglected, i.e., we consider the limit $\mu \to 0$. Then, the stress $\boldsymbol{\sigma} = p\mathbf{I}$ is isotropic with hydrostatic pressure $p = \frac{1}{3}\operatorname{trace}\boldsymbol{\sigma}$, and compressional waves are described by the system

$$\partial_t p = \kappa \operatorname{div} \mathbf{v}\,, \qquad \rho \partial_t \mathbf{v} = \nabla p + \mathbf{b}\,.$$

In particular this applies to acoustic waves in air or in a gas at fixed temperature. Note that this is only a formal derivation of the acoustic wave equation using the setting of continuum mechanics of solids.
The space-time operator L for the acoustic wave equation uses the mass and hyperbolic operators

$$M = \begin{pmatrix} \rho & 0 \\ 0 & \kappa^{-1} \end{pmatrix} \qquad \text{and} \qquad A = -\begin{pmatrix} 0 & \nabla \\ \operatorname{div} & 0 \end{pmatrix}. \tag{2.14}$$

Lemma 2.5. *The operator A defined in* (2.14) *is positive semi-definite on the domain $\mathcal{D}(A) = \mathrm{H}_0(\operatorname{div}, \Omega) \times \mathrm{H}^1(\Omega)$ (Neumann boundary condition for the velocity component) and hence Lem. 2.2 holds true.*

Proof. For all $\mathbf{u} \in \mathcal{D}(A)$ it holds that

$$\langle A\mathbf{u},\, \mathbf{u}\rangle_\Omega = -\int_\Omega \operatorname{div}(\mathbf{v})\, p + \nabla p \cdot \mathbf{v}\, \mathrm{d}\mathbf{x} = \int_{\partial\Omega} \mathbf{n} \cdot \mathbf{v}\, p\, \mathrm{d}\mathbf{x} = 0\,.$$

□

2.5 Nonlocal material laws

General linear material laws for visco-elasticity have the form

$$\boldsymbol{\sigma}(t) = \boldsymbol{\sigma}(0) + \int_0^t \mathbf{C}(t-s)\boldsymbol{\varepsilon}(\mathbf{v}(s))\,\mathrm{d}s\,,$$

i.e., the stress $\boldsymbol{\sigma}$ depends on the strain rate $\boldsymbol{\varepsilon}(\mathbf{v})$ by a convolution kernel $\mathbf{C}(\cdot)$ in time. This yields with $\mathbf{C} = \mathbf{C}(0)$ for the stress rate

$$\partial_t \boldsymbol{\sigma}(t) = \mathbf{C}\boldsymbol{\varepsilon}(\mathbf{v}(t)) + \int_0^t \dot{\boldsymbol{C}}(t-s)\boldsymbol{\varepsilon}(\mathbf{v}(s))\,\mathrm{d}s\,,$$

with $\dot{\boldsymbol{C}}$ denoted as the relaxation tensor. In applications, the relaxation tensor is adapted to measurements of the wave propagation within a fixed frequency

range. For the case of generalized standard linear solids [Fic11, Chap. 5], a model for a spring combined with G Maxwell bodies (see Fig. 2.1 for a sketch) can be calibrated to velocity and attenuation of time-harmonic waves for a number of sample frequencies ($f_g = (2\pi\tau_g)^{-1}$) using a least-squares approach, see [BRS95] for details. The corresponding relaxation tensor is then given by

$$\dot{\boldsymbol{C}}(s) = -\sum_{g=1}^{G} \frac{1}{\tau_g} \exp\Big(-\frac{s}{\tau_g}\Big)\mathbf{C}_g \tag{2.15}$$

depending on a decomposition $\mathbf{C} = \mathbf{C}_0 + \mathbf{C}_1 + \cdots + \mathbf{C}_G$ and relaxation parameters $\tau_l > 0$. Introducing the corresponding stress decomposition $\boldsymbol{\sigma} = \boldsymbol{\sigma}_0 + \cdots + \boldsymbol{\sigma}_G$ with

$$\boldsymbol{\sigma}_g(t) = \int_0^t \exp\left(\frac{s-t}{\tau_g}\right) \mathbf{C}_g\boldsymbol{\varepsilon}(\mathbf{v}(s))\,\mathrm{d}s\,, \qquad g = 1, \ldots, G$$

results in the system

$$\rho\,\partial_t\mathbf{v} = \nabla\cdot\boldsymbol{\sigma}_0 + \cdots + \nabla\cdot\boldsymbol{\sigma}_G + \mathbf{b}\,, \tag{2.16a}$$

$$\partial_t\boldsymbol{\sigma}_0 = \mathbf{C}_0\boldsymbol{\varepsilon}(\mathbf{v})\,, \tag{2.16b}$$

$$\partial_t\boldsymbol{\sigma}_g = \mathbf{C}_g\boldsymbol{\varepsilon}(\mathbf{v}) - \frac{1}{\tau_g}\boldsymbol{\sigma}_g\,, \qquad g = 1, \ldots, G\,. \tag{2.16c}$$

The space-time operator L for the visco-elastic wave equation in isotropic materials uses the special choice $\mathbf{C}_0 = \mathbf{C}(\mu, \kappa)$ defined by (2.12) and $\mathbf{C}_1 = \ldots = \mathbf{C}_G = \mathbf{C}(\mu\tau_{\mathrm{S}}, \kappa\tau_{\mathrm{P}})$ with given attenuation parameters for the shear wave τ_{S} and for the compressional wave τ_{P} (cf. [Zel19, Chap. 2]). Summing up, the

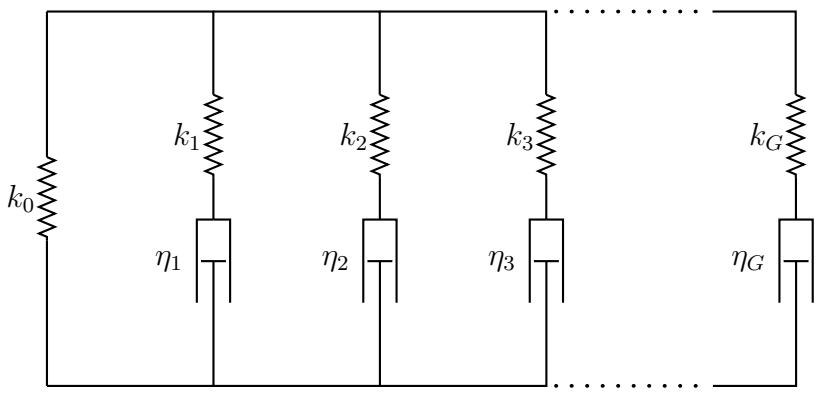

Figure 2.1: Schematic diagram of generalized standard linear solids (GSLS) with G relaxation mechanisms / Maxwell bodies.

system depends on the parameters $\rho,\ \mu,\ \kappa,\ \tau_{\mathrm{S}},\ \tau_{\mathrm{P}}$ and $\tau_1,\dots,\tau_G$. In [Zel19, Chap. 5] the existence and uniqueness of a solution is proven. We transfer the visco-elastic wave equations in our setting and get the generalized mass operator

$$\begin{aligned} M &= \begin{pmatrix} \rho & 0 & \cdots & \cdots & 0 \\ 0 & \mathbf{C}(\mu,\kappa)^{-1} & \ddots & & \vdots \\ \vdots & \ddots & \mathbf{C}(\mu\tau_{\mathrm{S}},\kappa\tau_{\mathrm{P}})^{-1} & \ddots & 0 \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & \mathbf{C}(\mu\tau_{\mathrm{S}},\kappa\tau_{\mathrm{P}})^{-1} \end{pmatrix} \\ &= \operatorname{diag}\left(\rho,\ \mathbf{C}(\mu,\kappa)^{-1},\ \mathbf{C}(\mu\tau_{\mathrm{S}},\kappa\tau_{\mathrm{P}})^{-1},\dots,\mathbf{C}(\mu\tau_{\mathrm{S}},\kappa\tau_{\mathrm{P}})^{-1}\right), \end{aligned} \tag{2.17}$$

the hyperbolic operator

$$A = -\begin{pmatrix} 0 & \operatorname{div} & \cdots & \operatorname{div} \\ \boldsymbol{\varepsilon} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{\varepsilon} & 0 & \cdots & 0 \end{pmatrix} \tag{2.18}$$

and the damping operator

$$D = \operatorname{diag}\left(0,\ 0,\ \frac{1}{\tau_1}\mathbf{C}(\mu\tau_{\mathrm{S}},\kappa\tau_{\mathrm{P}})^{-1},\dots,\frac{1}{\tau_G}\mathbf{C}(\mu\tau_{\mathrm{S}},\kappa\tau_{\mathrm{P}})^{-1}\right). \tag{2.19}$$

Lemma 2.6. *The operator A defined in* (2.18) *is positive semi-definite on the domain $\mathcal{D}(A) = \mathrm{H}_0^1(\Omega;\mathbb{R}^{\dim}) \times \mathrm{H}(\operatorname{div},\Omega;\mathbb{R}^{\dim\times\dim}_{sym})^{G+1}$ and hence Lem. 2.2 holds true.*

Proof. For all $\mathbf{u} = (\mathbf{v},\boldsymbol{\sigma}_0,\dots,\boldsymbol{\sigma}_G)^\top \in \mathcal{D}(A)$ it holds that

$$\begin{aligned} \langle A\mathbf{u},\,\mathbf{u}\rangle_\Omega &= -\sum_{g=0}^{G}\int_\Omega \operatorname{div}(\boldsymbol{\sigma}_g)\cdot\mathbf{v} + \boldsymbol{\varepsilon}(\mathbf{v}) : \boldsymbol{\sigma}_g \,\mathrm{d}\mathbf{x} \\ &= -\sum_{g=0}^{G}\int_\Omega \operatorname{div}(\boldsymbol{\sigma}_g\cdot\mathbf{v}) - \mathrm{D}\mathbf{v} : \boldsymbol{\sigma}_g + \boldsymbol{\varepsilon}(\mathbf{v}) : \boldsymbol{\sigma}_g \,\mathrm{d}\mathbf{x} \\ &= -\sum_{g=0}^{G}\int_\Omega \operatorname{div}(\boldsymbol{\sigma}_g\cdot\mathbf{v}) - \boldsymbol{\varepsilon}(\mathbf{v}) : \boldsymbol{\sigma}_g + \boldsymbol{\varepsilon}(\mathbf{v}) : \boldsymbol{\sigma}_g \,\mathrm{d}\mathbf{x} \\ &= -\sum_{g=0}^{G}\int_\Omega \operatorname{div}(\boldsymbol{\sigma}_g\cdot\mathbf{v})\,\mathrm{d}\mathbf{x} = -\sum_{g=0}^{G}\int_{\partial\Omega} \mathbf{v}\cdot\boldsymbol{\sigma}_g\mathbf{n}\,\mathrm{d}\mathbf{x} = 0\,. \end{aligned}$$

□

The visco-acoustic wave equations are

$$\begin{aligned} \rho\,\partial_t \mathbf{v} &= \sum_{g=0}^{G} \nabla p_g + \mathbf{b}\,, \\ \partial_t p_0 &= \kappa \nabla \cdot \mathbf{v}\,, \\ \partial_t p_g &= \kappa \tau_{\mathrm{P}} \nabla \cdot \mathbf{v} - \frac{1}{\tau_g} p_g, \qquad g = 1, \ldots, G\,. \end{aligned} \tag{2.20}$$

These equations fit also into the setting of the space-time operator L with the operators

$$\begin{aligned} M(\mathbf{v}, p_0, p_1, \cdots, p_G) &= (\rho\mathbf{v}, \kappa^{-1} p_0, (\kappa\tau_{\mathrm{P}})^{-1} p_1, \cdots, (\kappa\tau_{\mathrm{P}})^{-1} p_G)\,, \\ A(\mathbf{v}, p_0, p_1, \cdots, p_G) &= -(\nabla(p_0 + \cdots + p_G), \nabla \cdot \mathbf{v}, \cdots, \nabla \cdot \mathbf{v})\,, \\ D(\mathbf{v}, p_0, p_1, \cdots, p_G) &= (\mathbf{0}, 0, (\tau_1 \kappa \tau_{\mathrm{P}})^{-1} p_1, \cdots, (\tau_G \kappa \tau_{\mathrm{P}})^{-1} p_G)\,. \end{aligned} \tag{2.21}$$

Lemma 2.7. *The operator A defined in* (2.21) *is positive semi-definite on the domain $\mathcal{D}(A) = \mathrm{H}_0(\mathrm{div}, \Omega) \times \mathrm{H}^1(\Omega)^{G+1}$ (Neumann boundary condition for the velocity component) and hence Lem. 2.2 holds true.*

Proof. For all $\mathbf{u} \in \mathcal{D}(A)$ it holds that

$$\langle A\mathbf{u},\, \mathbf{u} \rangle_\Omega = -\sum_{g=0}^{G} \int_\Omega \mathrm{div}(\mathbf{v})\, p_g + \nabla p_g \cdot \mathbf{v}\, \mathrm{d}\mathbf{x} = \sum_{g=0}^{G} \int_{\partial\Omega} \mathbf{n} \cdot \mathbf{v}\, p_g\, \mathrm{d}\mathbf{x} = 0\,.$$

□

Remark 2.3. Choosing $G = 0$ reduces the visco-elastic and visco-acoustic equation to the simple elastic and acoustic equations.

CHAPTER

THREE

THE DISCRETIZATION

We start from the continuous variational formulation: find $\mathbf{u} \in V$ such that for all $\mathbf{v} \in W$ holds

$$\mathcal{B}(\mathbf{u}, \mathbf{v}) = \langle \mathbf{b}, \mathbf{v} \rangle .$$

The idea of the Galerkin method for approximating the solution of this problem is using a finite dimensional space V_h and W_h. The discrete problem reads as follows: find $\mathbf{u}_h \in V_h$ such that for all $\mathbf{v}_h \in W_h$ holds

$$\mathcal{B}(\mathbf{u}_h, \mathbf{v}_h) = \langle \mathbf{b}, \mathbf{v}_h \rangle .$$

If ansatz space and test space do not coincide, we obtain a Petrov–Galerkin method. If the ansatz and test space are subspaces of the corresponding continuous spaces, the method is called *conforming*, otherwise it is called a *non-conforming* method. In the case of a non-conforming method, the bilinear form has to be extended to a discrete bilinear form

$$\mathcal{B}_h(\mathbf{u}_h, \mathbf{v}_h) = \langle \mathbf{b}, \mathbf{v}_h \rangle .$$

One possibility for an implementation of such a method is the finite element method (FEM). For an introduction into FEM we refer to [Cia02, Bra13].

3.1 Discontinuous Galerkin semi-discretization in space

The semi-discretization in space will be done with the nodal discontinuous Galerkin method [HW08]. We assume that Ω is a bounded polyhedral Lipschitz domain decomposed into a finite number of open elements $K \subset \Omega$ such that $\overline{\Omega} = \bigcup_{K\in\mathcal{K}} \overline{K}$, where $\mathcal{K}$ is the set of elements in space. Let $\mathcal{F}_K$ be the set of faces of $K \in \mathcal{K}$. For inner faces $f \in \mathcal{F}_K$ let K_f be the neighboring cell such that $f = \partial K \cap \partial K_f$, and let $\mathbf{n}_{K,f}$ be the outer unit normal vector on ∂K. The outer unit normal vector field on $\partial\Omega$ is denoted by $\mathbf{n}$.

In every time slice, we select polynomial degrees p_K, and define the local spaces $H_{h,K} = \mathbb{P}_{p_K}(K;\mathbb{R}^J)$ and the global discontinuous Galerkin space

$$H_h = \Big\{\mathbf{v}_h \in \mathrm{L}_2(\Omega;\mathbb{R}^J)\colon \mathbf{v}_h|_K \in H_{h,K} \text{ for all } K \in \mathcal{K}\Big\}.$$

For $\mathbf{v}_h \in H_h$ we define $\mathbf{v}_{h,K} = \mathbf{v}_h|_K \in H_{h,K}$ for the restriction to K. In the semi-discrete problem

$$L_h\mathbf{u}_h(t) = M_h\partial_t\mathbf{u}_h(t) + A_h\mathbf{u}_h(t) + D_h\mathbf{u}_h(t) = \mathbf{b}_h(t)\,, \qquad t \in (0,T)\,, \tag{3.1}$$

the discrete operators $M_h, D_h \in \mathcal{L}(H_h, H_h)$ and the right-hand side $\mathbf{b}_h(t) \in H_h$ are the Galerkin approximations of M, D and $\mathbf{b}(\cdot)$ defined by

$$\begin{aligned}
\langle M_h\mathbf{v}_h,\, \mathbf{w}_h\rangle_\Omega &= \langle M\mathbf{v}_h,\, \mathbf{w}_h\rangle_\Omega \qquad & \mathbf{v}_h, \mathbf{w}_h \in H_h\,,\\
\langle D_h\mathbf{v}_h,\, \mathbf{w}_h\rangle_\Omega &= \langle D\mathbf{v}_h,\, \mathbf{w}_h\rangle_\Omega \qquad & \mathbf{v}_h, \mathbf{w}_h \in H_h\,,\\
\langle \mathbf{b}_h,\, \mathbf{w}_h\rangle_\Omega &= \langle \mathbf{b}(\cdot),\, \mathbf{w}_h\rangle_\Omega \qquad & \mathbf{w}_h \in H_h\,.
\end{aligned} \tag{3.2}$$

Note that M_h is represented by a block diagonal positive definite matrix and D_h is a block diagonal positive semi-definite matrix.

Remark 3.1. We aim to obtain a fully adaptive space-time method combined with a multilevel preconditioner. To avoid further issues, we restrict our problems to the case of cellwise constant material parameter. As a result, the matrix M_h does not depend on time.

The discrete operator $A_h \in \mathcal{L}(H_h, H_h)$ is constructed as follows: integration by parts on $K \in \mathcal{K}$ yields for smooth ansatz functions $\mathbf{v}$ and smooth test functions $\boldsymbol{\phi}_K$

$$\begin{aligned}\langle A\mathbf{v},\, \boldsymbol{\phi}_K\rangle_K &= \langle \operatorname{div} \mathbf{F}(\mathbf{v}),\, \boldsymbol{\phi}_K\rangle_K \\ &= -\langle \mathbf{F}(\mathbf{v}),\, \nabla\boldsymbol{\phi}_K\rangle_K + \sum_{f\in\mathcal{F}_K} \langle \mathbf{n}_{K,f}\cdot\mathbf{F}(\mathbf{v}),\, \boldsymbol{\phi}_K\rangle_f \,.\end{aligned}$$

We then define for $\mathbf{v}_h \in H_h$ and $\boldsymbol{\phi}_{h,K} \in H_{h,K}$

$$\langle A_h\mathbf{v}_h,\, \boldsymbol{\phi}_{h,K}\rangle_K = -\langle \mathbf{F}(\mathbf{v}_{h,K}),\, \nabla\boldsymbol{\phi}_{h,K}\rangle_K + \sum_{f\in\mathcal{F}_K} \langle \mathbf{n}_{K,f}\cdot\mathbf{F}_K^{\mathrm{num}}(\mathbf{v}_h),\, \boldsymbol{\phi}_{h,K}\rangle_f \,,$$

where $\mathbf{n}_{K,f}\cdot\mathbf{F}_K^{\mathrm{num}}(\mathbf{v}_h)$ is the upwind flux obtained from local solutions of Riemann problems (cf. Sec. 3.2). Again using integration by parts, we obtain

$$\begin{aligned}\langle A_h\mathbf{v}_h,\, \boldsymbol{\phi}_{h,K}\rangle_K &= \langle \operatorname{div} \mathbf{F}(\mathbf{v}_{h,K}),\, \boldsymbol{\phi}_{h,K}\rangle_K \\ &\quad + \sum_{f\in\mathcal{F}_K} \langle \mathbf{n}_{K,f}\cdot(\mathbf{F}_K^{\mathrm{num}}(\mathbf{v}_h) - \mathbf{F}(\mathbf{v}_{h,K})),\, \boldsymbol{\phi}_{h,K}\rangle_f \,.\end{aligned}$$

On inner faces $f = \partial K \cap \partial K_f$ the difference $\mathbf{n}_{K,f}\cdot(\mathbf{F}_K^{\mathrm{num}}(\mathbf{v}_h) - \mathbf{F}(\mathbf{v}_{h,K}))$ only depends on the jump term $[\mathbf{v}_h]_{K,f} = \mathbf{v}_{h,K_f} - \mathbf{v}_{h,K}$, so that $\mathbf{n}_{K,f}\cdot(\mathbf{F}_K^{\mathrm{num}}(\mathbf{v}) - \mathbf{F}(\mathbf{v})) = 0$ on all faces $f \in \mathcal{F}_K$ for $\mathbf{v} \in \mathcal{D}(A)$. On boundary faces, we define the jump term $[\mathbf{v}_h]_{K,f}$ depending on the boundary conditions. On H_h we define the operator A_h by

$$\langle A_h\mathbf{v}_h,\, \boldsymbol{\phi}_h\rangle_{\mathcal{K}} = \sum_{K\in\mathcal{K}} \langle A_h\mathbf{v}_h,\, \boldsymbol{\phi}_{h,K}\rangle_K \,, \qquad \mathbf{v}_h, \boldsymbol{\phi}_h \in H_h \,.$$

By construction, the operator A_h satisfies the consistency condition

$$\langle A\mathbf{v},\, \boldsymbol{\phi}_h\rangle_\Omega = \langle A_h\mathbf{v},\, \boldsymbol{\phi}_h\rangle_\Omega \,, \qquad \mathbf{v} \in \mathcal{D}(A)\cap \mathrm{H}^1(\Omega;\mathbb{R}^J)\,,\ \boldsymbol{\phi}_h \in H_h \,, \tag{3.3}$$

since the numerical flux $\mathbf{F}^{\mathrm{num}}$ satisfies

$$\sum_{K\in\mathcal{K}} \langle \mathbf{n}_{K,f}\cdot\mathbf{F}_K^{\mathrm{num}}(\mathbf{v}_{h,K}),\, \mathbf{v}\rangle_{\partial K} = 0 \,, \qquad \mathbf{v} \in \mathcal{D}(A)\cap \mathrm{H}^1(\Omega;\mathbb{R}^J)$$

for $\mathbf{v}_h \in H_h$.

For our applications we can show in the next section that the upwind flux together with the correct choice of the boundary flux guarantees that the discrete operator is non-negative and controls the nonconformity, i.e., a constant

$C_{\mathrm{A}} > 0$ exists such that

$$\langle A_h \mathbf{v}_h,\, \mathbf{v}_h \rangle_\Omega \geq C_{\mathrm{A}} \sum_{f \in \mathcal{F}_K} \left\| \mathbf{n}_{K,f} \cdot (\mathbf{F}_K^{\mathrm{num}}(\mathbf{v}_h) - \mathbf{F}(\mathbf{v}_{h,K})) \right\|_f^2 \geq 0 \tag{3.4}$$

for all $\mathbf{v}_h \in H_h$.

3.2 Upwind flux

We decided to discretize the hyperbolic operator A_h by an upwind flux scheme. The main ideas are presented in [LeV02, Chap. 3.8 and 9.9] and summarized in [HPS+15, Sec. 3.1]. The upwind flux is defined by the solution of the Riemann problem.

Definition 3.1 (Riemann problem)**.** Let $\mathbf{n} \in \mathbb{R}^{\dim}$ be a given unit vector. Then $\mathbb{R}^{\dim}$ is divided into two open subsets $\Omega_L = \{\mathbf{x} \in \mathbb{R}^{\dim} \colon \mathbf{n} \cdot \mathbf{x} < 0\}$ and $\Omega_R = \{\mathbf{x} \in \mathbb{R}^{\dim} \colon \mathbf{n} \cdot \mathbf{x} > 0\}$. The Riemann problem reads as follows: find a weak solution $\mathbf{u}$ to the discontinuous initial function

$$\mathbf{u}_0(\mathbf{x}) = \begin{cases} \mathbf{u}_L & \text{for all } \mathbf{x} \in \Omega_L\,, \\ \mathbf{u}_R & \text{for all } \mathbf{x} \in \Omega_R\,, \end{cases}$$

with $\mathbf{u}_L\,,\ \mathbf{u}_R \in \mathbb{R}^J$ and piecewise constant $M|_{\Omega_L} = M_L$ and $M|_{\Omega_R} = M_R$.

Following the steps in [HPS+15, Sec. 3.1], we define by $(\lambda_{j,L}, \mathbf{w}_{j,L})$ and $(\lambda_{j,R}, \mathbf{w}_{j,R})$ the corresponding M-orthogonal eigenpairs of the matrix B defined in (2.3), i.e.,

$$\begin{aligned} B\mathbf{w}_{j,L} &= \lambda_{j,L} M_L \mathbf{w}_{j,L} \qquad \text{with} \quad \mathbf{w}_{j,L} \cdot M_L \mathbf{w}_{k,L} = \delta_{j,k}\,, \\ B\mathbf{w}_{j,R} &= \lambda_{j,R} M_R \mathbf{w}_{j,R} \qquad \text{with} \quad \mathbf{w}_{j,R} \cdot M_R \mathbf{w}_{k,R} = \delta_{j,k}\,. \end{aligned}$$

A general solution of the Riemann problem is given by

$$\mathbf{u}(\mathbf{x}, t) = \begin{cases} \mathbf{u}_L + \displaystyle\sum_{\mathbf{x}\cdot\mathbf{n} - \lambda_{j,L} t > 0} b_{j,L} \mathbf{w}_{j,L} & \text{for all } \mathbf{x} \in \Omega_L\,, \\ \mathbf{u}_R + \displaystyle\sum_{\mathbf{x}\cdot\mathbf{n} - \lambda_{j,R} t < 0} b_{j,R} \mathbf{w}_{j,R} & \text{for all } \mathbf{x} \in \Omega_R\,, \end{cases} \tag{3.5}$$

for arbitrary coefficients $b_{j,L}\,,\ b_{j,R} \in \mathbb{R}$. To obtain a weak solution in $\mathbb{R}^{\dim}$, continuity of the flux on the interface $\partial\Omega_L \cap \partial\Omega_R = \{\mathbf{x} \in \mathbb{R}^{\dim} \colon \mathbf{x}\cdot\mathbf{n} = 0\}$ is required, i.e.,

$$B\left(\mathbf{u}_L + \sum_{\lambda_{j,L}<0} b_{j,L}\mathbf{w}_{j,L}\right) = B\left(\mathbf{u}_R + \sum_{\lambda_{j,R}>0} b_{j,R}\mathbf{w}_{j,R}\right).$$

This condition is the so called *Rankine–Hugoniot condition*.
The coefficients $b_{j,L}$ are determined from the jump $[\mathbf{u}_0] = \mathbf{u}_R - \mathbf{u}_L$ solving the equations

$$\mathbf{w}_{k,R}\cdot B[\mathbf{u}_0] = \mathbf{w}_{k,R}\cdot \sum_{\lambda_{j,L}<0} b_{j,L}B\mathbf{w}_{j,L} \qquad \text{for all}\ \ \lambda_{k,R} < 0\,.$$

The solution of the Riemann problem defines the upwind flux on $\partial\Omega_L \cap \partial\Omega_R$ by

$$\mathbf{n}\cdot\mathbf{F}^{\text{num}}(\mathbf{u}_0) = B\left(\mathbf{u}_L + \sum_{\lambda_{j,L}<0} b_{j,L}\mathbf{w}_{j,L}\right). \tag{3.6}$$

Upwind flux for visco-acoustic waves

We use the formulation (2.20) and thus have

$$\operatorname{div}\mathbf{F}(\mathbf{v}, p_0, \dots, p_G) = -\left(\nabla\sum_{g=0}^{G} p_g,\ \operatorname{div}\mathbf{v},\ \cdots,\ \operatorname{div}\mathbf{v}\right)^{\top},$$
$$\mathbf{n}\cdot\mathbf{F}(\mathbf{v}, p_0, \dots, p_G) = -\left(\mathbf{n}\sum_{g=0}^{G} p_g, \mathbf{v}\cdot\mathbf{n}, \cdots, \mathbf{v}\cdot\mathbf{n}\right)^{\top}$$

together with the mass matrix

$$M = \begin{pmatrix} \rho\mathbf{I}_D & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{0} & \kappa^{-1} & 0 & \cdots & 0 \\ \vdots & 0 & (\kappa\tau_{\mathrm{P}})^{-1} & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ \mathbf{0} & 0 & \cdots & 0 & (\kappa\tau_{\mathrm{P}})^{-1} \end{pmatrix}.$$

In a first step, we focus on the acoustic case ($G = 0$). This reduces the problem to find eigenvalues and eigenvectors of $B\mathbf{w}_{\pm} = \pm cM\mathbf{w}_{\pm}$ with

$$M = \begin{pmatrix} \rho\mathbf{I}_D & 0 \\ 0 & \kappa^{-1} \end{pmatrix} \quad \text{and} \quad \mathbf{n}\cdot\mathbf{F}(\mathbf{v}, p) = B\begin{pmatrix}\mathbf{v}\\ p\end{pmatrix} = -\begin{pmatrix}\mathbf{n}p\\ \mathbf{v}\cdot\mathbf{n}\end{pmatrix}.$$

The solution is given by the velocity of sound $c = \pm\sqrt{\kappa/\rho}$ together with $\mathbf{w}_\pm = \begin{pmatrix} \mp c\mathbf{n} \\ \kappa \end{pmatrix}$. Inserting them into (3.6) gives the upwind flux

$$\begin{aligned}
\mathbf{n}\cdot\mathbf{F}^{\text{num}}(\mathbf{u}) &= B\mathbf{u}_L + \frac{\begin{pmatrix} c_R\mathbf{n} \\ \kappa_R \end{pmatrix} \cdot B \begin{pmatrix} [\mathbf{v}] \\ [p] \end{pmatrix}}{\begin{pmatrix} c_R\mathbf{n} \\ \kappa_R \end{pmatrix} \cdot B \begin{pmatrix} c_L\mathbf{n} \\ \kappa_L \end{pmatrix}} B \begin{pmatrix} c_L\mathbf{n} \\ \kappa_L \end{pmatrix} \\
&= B\mathbf{u}_L - \frac{c_R[p] + \kappa_R[\mathbf{v}]\cdot\mathbf{n}}{c_R\kappa_L + c_L\kappa_R} \begin{pmatrix} \kappa_L\mathbf{n} \\ c_L \end{pmatrix} \\
&= B\mathbf{u}_L - \left(\alpha_1 \begin{pmatrix} 0 \\ [p] \end{pmatrix} + \alpha_2 \begin{pmatrix} ([\mathbf{v}]\cdot\mathbf{n})\mathbf{n} \\ 0 \end{pmatrix} + \alpha_3 \begin{pmatrix} 0 \\ [\mathbf{v}]\cdot\mathbf{n} \end{pmatrix} + \alpha_4 \begin{pmatrix} [p]\mathbf{n} \\ 0 \end{pmatrix} \right)
\end{aligned}$$

with the coefficients defined with the impedance $Z = \sqrt{\kappa\rho}$

$$\begin{aligned}
\alpha_1 &= \frac{1}{Z_L + Z_R}\,, & \alpha_2 &= \frac{Z_L Z_R}{Z_L + Z_R}\,, \\
\alpha_3 &= \frac{Z_R}{Z_L + Z_R}\,, & \alpha_4 &= \frac{Z_L}{Z_L + Z_R}\,.
\end{aligned}$$

This upwind flux can be now extended to the visco-acoustic case. On inner boundaries $K \cap K_f = f \subset \Omega$ with $\kappa_L = \kappa|_K(1 + G\tau_{\text{P}})\,,\ \kappa_R = \kappa|_{K_f}(1 + G\tau_{\text{P}})$ $\rho_L = \rho|_K$ and $\rho_L = \rho|_{K_f}$ we obtain

$$\begin{aligned}
&\langle A_h(\mathbf{v}_h, p_{0,h}, \cdots, p_{G,h}),\, (\boldsymbol{\psi}_{K,h}, \phi_{0,K,h}, \cdots, \phi_{G,K,h})\rangle_K \\
&\quad = -\left\langle \operatorname{div}\mathbf{v}_{K,h},\, \sum_{g=0}^{G} \phi_{g,K,h} \right\rangle_K - \left\langle \sum_{g=0}^{G} \nabla p_{g,K,h},\, \boldsymbol{\psi}_{K,h} \right\rangle_K \\
&\qquad - \sum_{f\in\mathcal{F}_K} \frac{1}{Z_K + Z_{K_f}} \\
&\qquad\quad \left\langle \sum_{g=0}^{G} [p_{g,h}]_{K,f} + Z_{K_f}\ \mathbf{n}_{K,f}\cdot[\mathbf{v}_h]_{K,f},\, \sum_{g=0}^{G} \phi_{g,K,h} + Z_K\ \boldsymbol{\psi}_{K,h}\cdot\mathbf{n}_{K,f} \right\rangle_f .
\end{aligned}$$

On boundary faces $f \subset \partial\Omega$ we want to use the same definition of the upwind flux as on interior cell faces. The general solution of the Riemann problem at the boundary is

$$\mathbf{u}(t,\mathbf{x}) = \begin{cases} u_L + b_L\ \mathbf{w}_L & \text{by (3.5)}\,, \\ (p_{0,\partial\Omega}, \ldots, p_{G,\partial\Omega})^\top & \text{Dirichlet boundary data}\,, \end{cases}$$

or

$$\mathbf{u}(t,\mathbf{x}) = \begin{cases} u_L + b_{1,L}\ \mathbf{w}_{1,L} & \text{by (3.5)}\,, \\ g_{\mathrm{N}} & \text{Neumann boundary data.}\,. \end{cases}$$

For given Dirichlet values in the pressure component we get the system

$$(p_{0,L} + b_L\kappa|_K,\ p_{1,L} + b_L\kappa|_K\tau_{\mathrm{P}}, \ldots, p_{G,L} + b_L\kappa|_K\tau_{\mathrm{P}})^\top = (p_{0,\partial\Omega}, \ldots, p_{G,\partial\Omega})^\top\,.$$

Summing over all entries

$$p_L = \sum_{g=0}^{G} p_{g,L}\,, \qquad p_{\partial\Omega} = \sum_{g=0}^{G} p_{g,\partial\Omega}\,, \qquad \kappa_L = \kappa|_K(1 + G\tau_{\mathrm{P}})\,,$$

defines $b_L = \dfrac{p_{\partial\Omega} - p_L}{\kappa_L}$. Since on the boundary no K_f exists, we define $K_f := K$ and obtain

$$\begin{aligned} & B\mathbf{u}_L - \frac{c_R[p] + \kappa_R[\mathbf{v}]\cdot\mathbf{n}}{c_R\kappa_L + c_L\kappa_R}\begin{pmatrix}\kappa_L\mathbf{n}\\ c_L\end{pmatrix} = B\left(\mathbf{u}_L + b_L\begin{pmatrix}c_L\mathbf{n}\\ \kappa_L\end{pmatrix}\right) \\ \Longrightarrow \quad & \frac{c_R[p] + \kappa_R[\mathbf{v}]\cdot\mathbf{n}}{c_R\kappa_L + c_L\kappa_R}\begin{pmatrix}\kappa_L\mathbf{n}\\ c_L\end{pmatrix} = \frac{p_{\partial\Omega} - p_L}{\kappa_L}\begin{pmatrix}\kappa_L\mathbf{n}\\ c_L\end{pmatrix} \\ \Longrightarrow \quad & \frac{[p] + \kappa_L/c_L[\mathbf{v}]\cdot\mathbf{n}}{2\kappa_L} = \frac{p_{\partial\Omega} - p_L}{\kappa_L}\,. \end{aligned}$$

For homogeneous Dirichlet boundary conditions in the pressure component we have to set $[p_{g,h}]_{K,f} = -2p_{g,h}$ and $[\mathbf{v}_h]_{K,f}\cdot\mathbf{n}_{K,f} = 0$. For Neumann boundary conditions in $\mathbf{v}$ we obtain the equation $g_{\mathrm{N}} = \mathbf{n}\cdot\mathbf{v}_L + b_L c_L$ which we compare to the definition of the upwind flux on interior boundaries

$$\begin{aligned} & B\mathbf{u}_L - \frac{c_R[p] + \kappa_R[\mathbf{v}]\cdot\mathbf{n}}{c_R\kappa_L + c_L\kappa_R}\begin{pmatrix}\kappa_L\mathbf{n}\\ c_L\end{pmatrix} = B\left(\mathbf{u}_L + b_L\begin{pmatrix}c_L\mathbf{n}\\ \kappa_L\end{pmatrix}\right) \\ \Longrightarrow \quad & \frac{c_R[p] + \kappa_R[\mathbf{v}]\cdot\mathbf{n}}{c_R\kappa_L + c_L\kappa_R}\begin{pmatrix}\kappa_L\mathbf{n}\\ c_L\end{pmatrix} = \frac{g_{\mathrm{N}} - \mathbf{v}_L\cdot\mathbf{n}}{c_L}\begin{pmatrix}\kappa_L\mathbf{n}\\ c_L\end{pmatrix} \\ \Longrightarrow \quad & \frac{c_L/\kappa_L[p] + [\mathbf{v}]\cdot\mathbf{n}}{2c_L} = \frac{g_{\mathrm{N}} - \mathbf{v}_L\cdot\mathbf{n}}{c_L}\,. \end{aligned}$$

to obtain the correct choice for homogeneous Neumann boundaries $[p_{g,h}]_{K,f} = 0$ and $[\mathbf{v}_h]_{K,f}\cdot\mathbf{n}_{K,f} = -2\mathbf{v}_{K,h}\cdot\mathbf{n}_{K,f}$.

Lemma 3.1. *Using a discontinuous Galerkin discretization with upwind flux for visco-acoustic equations guarantees that the discrete hyperbolic operator is non-negative, i.e.,* (3.4) *holds.*

Proof. Using the notation $p_h = \sum_{g=0}^{G} p_{g,h}$, $Z_K = \sqrt{\kappa_K\ \rho_K}$ and $Z_{K_f} = \sqrt{\kappa_{K_f}\ \rho_{K_f}}$ we have

$$
\begin{aligned}
&\langle A_h(\mathbf{v}_h, p_{0,h}, \cdots, p_{G,h}),\ (\mathbf{v}_h, p_{0,h}, \cdots, p_{G,h})\rangle_\Omega \\
&\quad = \sum_K - \int_K \operatorname{div} \mathbf{v}_{K,h} p_{K,h} + \mathbf{v}_{K,h} \nabla p_{K,h}\, \mathrm{d}\mathbf{x} \\
&\qquad - \sum_{f\in\mathcal{F}_K} \frac{1}{Z_K + Z_{K_f}} \Big\langle [p_h]_{K,f} + Z_{K_f} \mathbf{n}_{K,f} \cdot [\mathbf{v}_h]_{K,f},\ p_{K,h} + Z_K \mathbf{v}_{K,h} \cdot \mathbf{n}_{K,f} \Big\rangle_f \\
&\quad = \sum_K \sum_{f\in\mathcal{F}_K} - \langle p_{K,h},\ \mathbf{v}_{K,h} \cdot \mathbf{n}_{K,f}\rangle_f \\
&\qquad - \frac{1}{Z_K + Z_{K_f}} \Big\langle [p_h]_{K,f} + Z_{K_f} \mathbf{n}_{K,f} \cdot [\mathbf{v}_h]_{K,f},\ p_{K,h} + Z_K \mathbf{v}_{K,h} \cdot \mathbf{n}_{K,f} \Big\rangle_f \\
&\quad = \frac{1}{2} \sum_K \sum_{f\in\mathcal{F}_K} \frac{1}{Z_K + Z_{K_f}} \Big(Z_{K_f} Z_K \|[v_h]_{K,f} \cdot \mathbf{n}_{K,f}\|_f^2 + \|[p_h]_{K,f}\|_f^2 \Big) \geq 0\,.
\end{aligned}
$$

Here, we use $[\mathbf{v}_h]_{K,f} = -[\mathbf{v}_h]_{K_f,f}$, $[p_h]_{K,f} = -[p_h]_{K_f,f}$, $\mathbf{n}_{K,f} = -\mathbf{n}_{K_f,f}$,

$$
\begin{aligned}
\|[v_h]_{K,f} \cdot \mathbf{n}_{K,f}\|_f^2 = &- \langle [\mathbf{v}_h]_{K,f} \cdot \mathbf{n}_{K,f},\ \mathbf{v}_{K,h} \cdot \mathbf{n}_{K,f}\rangle_f \\
&- \Big\langle [\mathbf{v}_h]_{K_f,f} \cdot \mathbf{n}_{K_f,f},\ \mathbf{v}_{K_f,h} \cdot \mathbf{n}_{K_f,f} \Big\rangle_f\ , \\
\|[p_h]_{K,f}\|_f^2 = &- \langle [p_h]_{K,f},\ p_{K,h}\rangle_f - \Big\langle [p_h]_{K_f,f},\ p_{K_f,h} \Big\rangle_f\ ,
\end{aligned}
$$

and

$$
\begin{aligned}
&\sum_K \sum_{f\in\mathcal{F}_K\cap\Omega} - \langle p_{K,h},\ \mathbf{v}_{K,h} \cdot \mathbf{n}_{K,f}\rangle_f - \frac{Z_K}{Z_K + Z_{K_f}} \langle [p_h]_{K,f},\ \mathbf{v}_{K,h} \cdot \mathbf{n}_{K,f}\rangle_f \\
&\qquad\qquad - \frac{Z_{K_f}}{Z_K + Z_{K_f}} \langle \mathbf{n}_{K,f} \cdot [\mathbf{v}_h]_{K,f},\ p_{K,h}\rangle_f \\
&= \sum_K \sum_{f\in\mathcal{F}_K\cap\Omega} - \frac{Z_K}{Z_K + Z_{K_f}} \langle p_{K,h} + [p_h]_{K,f},\ \mathbf{v}_{K,h} \cdot \mathbf{n}_{K,f}\rangle_f \\
&\qquad\qquad - \frac{Z_{K_f}}{Z_K + Z_{K_f}} \langle \mathbf{n}_{K,f} \cdot (\mathbf{v}_{K,h} + [\mathbf{v}_h]_{K,f}),\ p_{K,h}\rangle_f \\
&= \sum_K \sum_{f\in\mathcal{F}_K\cap\Omega} - \frac{Z_K}{Z_K + Z_{K_f}} \Big\langle p_{K_f,h},\ \mathbf{v}_{K,h} \cdot \mathbf{n}_{K,f} \Big\rangle_f - \frac{Z_{K_f}}{Z_K + Z_{K_f}} \Big\langle \mathbf{n}_{K,f} \cdot \mathbf{v}_{K_f,h},\ p_{K,h} \Big\rangle_f \\
&= 0
\end{aligned}
$$

and on the boundary

$$
\begin{aligned}
&\sum_{f\in\mathcal{F}_K\cap\partial\Omega} -\frac{1}{2}\left\langle p_{K,h}+[p_h]_{K,f},\, \mathbf{v}_{K,h}\cdot\mathbf{n}_{K,f}\right\rangle_f - \frac{1}{2}\left\langle \mathbf{n}_{K,f}\cdot(\mathbf{v}_{K,h}+[\mathbf{v}_h]_{K,f}),\, p_{K,h}\right\rangle_f \\
&= \sum_{\substack{f\in\mathcal{F}_K\cap\partial\Omega \\ \text{Dirichlet b.c. in } p}} -\frac{1}{2}\left\langle p_{K,h}-2p_{K,h},\, \mathbf{v}_{K,h}\cdot\mathbf{n}_{K,f}\right\rangle_f - \frac{1}{2}\left\langle \mathbf{n}_{K,f}\cdot\mathbf{v}_{K,h},\, p_{K,h}\right\rangle_f \\
&\quad + \sum_{\substack{f\in\mathcal{F}_K\cap\partial\Omega \\ \text{Neumann b.c. in } \mathbf{v}}} -\frac{1}{2}\left\langle p_{K,h},\, \mathbf{v}_{K,h}\cdot\mathbf{n}_{K,f}\right\rangle_f - \frac{1}{2}\left\langle \mathbf{n}_{K,f}\cdot(\mathbf{v}_{K,h}-2\mathbf{v}_{K,h}),\, p_{K,h}\right\rangle_f \\
&= \sum_{\substack{f\in\mathcal{F}_K\cap\partial\Omega \\ \text{Dirichlet b.c. in } p}} +\frac{1}{2}\left\langle p_{K,h},\, \mathbf{v}_{K,h}\cdot\mathbf{n}_{K,f}\right\rangle_f - \frac{1}{2}\left\langle \mathbf{n}_{K,f}\cdot\mathbf{v}_{K,h},\, p_{K,h}\right\rangle_f \\
&\quad + \sum_{\substack{f\in\mathcal{F}_K\cap\partial\Omega \\ \text{Neumann b.c. in } \mathbf{v}}} -\frac{1}{2}\left\langle p_{K,h},\, \mathbf{v}_{K,h}\cdot\mathbf{n}_{K,f}\right\rangle_f + \frac{1}{2}\left\langle \mathbf{n}_{K,f}\cdot\mathbf{v}_{K,h},\, p_{K,h}\right\rangle_f \\
&= 0\,.
\end{aligned}
$$

□

Upwind flux for visco-elastic waves

We again start with the elastic case ($G = 0$). We denote by $\pm c_{\mathrm{S}} = \sqrt{\frac{\mu}{\rho}}$ the velocity of shear waves and by $\pm c_{\mathrm{P}} = \sqrt{\frac{2\mu/3+\lambda}{\rho}}$ the velocity of pressure waves, which are both eigenvalues. The corresponding eigenvectors are $\begin{pmatrix} \pm c_{\mathrm{S}}\boldsymbol{\tau} \\ \mu(\boldsymbol{\tau}\,\mathbf{n}^\top + \mathbf{n}\,\boldsymbol{\tau}^\top) \end{pmatrix}$ and $\begin{pmatrix} \pm c_{\mathrm{P}}\mathbf{n} \\ 2\mu\mathbf{n}\,\mathbf{n}^\top + \lambda\mathbf{I} \end{pmatrix}$, where $\boldsymbol{\tau}$ is a unit tangent vector.

Remark 3.2. In 3D there exist two unit tangent vectors, consequently the corresponding eigenspace for the shear wave has dimension two. The following proofs will be restricted to 2D.

We follow the steps as in the visco-acoustic case and finally lump the hyperbolic operator with upwind flux for the visco-elastic wave equation together with

$$
\begin{aligned}
Z_{\mathrm{P},K} &= \rho_K c_{\mathrm{P},K}\,, & Z_{\mathrm{P},K_f} &= \rho_{K_f} c_{\mathrm{P},K_f}\,, \\
Z_{\mathrm{S},K} &= \rho_K c_{\mathrm{S},K}\,, & Z_{\mathrm{S},K_f} &= \rho_{K_f} c_{\mathrm{S},K_f}
\end{aligned}
$$

as

$$
\begin{aligned}
&\langle A_h(\mathbf{v}_h, \boldsymbol{\sigma}_{0,h}, \cdots, \boldsymbol{\sigma}_{G,h}),\, (\boldsymbol{\psi}_{K,h}, \boldsymbol{\varphi}_{K,0,h}, \cdots, \boldsymbol{\varphi}_{K,G,h})\rangle_K \\
&= -\left\langle \sum_{g=0}^{G} \operatorname{div} \boldsymbol{\sigma}_{g,h},\, \boldsymbol{\psi}_{K,h} \right\rangle_K - \left\langle \boldsymbol{\varepsilon}(\mathbf{v}_{K,h}),\, \sum_{g=0}^{G} \boldsymbol{\varphi}_{K,g,h} \right\rangle_K \\
&\quad - \sum_{f \in \mathcal{F}_K} \frac{1}{Z_{\mathrm{P},K} + Z_{\mathrm{P},K_f}} \left\langle \mathbf{n}_{K,f} \cdot \sum_{g=0}^{G} [\boldsymbol{\sigma}_{g,h}]_{K,f} \mathbf{n}_{K,f},\, \mathbf{n}_{K,f} \cdot \sum_{g=0}^{G} \boldsymbol{\varphi}_{K,g,h} \mathbf{n}_{K,f} \right\rangle_f \\
&\qquad + \frac{Z_{\mathrm{P},K}}{Z_{\mathrm{P},K} + Z_{\mathrm{P},K_f}} \left\langle \mathbf{n}_{K,f} \cdot \sum_{g=0}^{G} [\boldsymbol{\sigma}_{g,h}]_{K,f} \mathbf{n}_{K,f},\, \mathbf{n}_{K,f} \cdot \boldsymbol{\psi}_{K,h}) \right\rangle_f \\
&\qquad + \frac{Z_{\mathrm{P},K_f}}{Z_{\mathrm{P},K} + Z_{\mathrm{P},K_f}} \left\langle \mathbf{n}_{K,f} \cdot [\mathbf{v}_h]_{K,f},\, \mathbf{n}_{K,f} \cdot \sum_{g=0}^{G} \boldsymbol{\varphi}_{K,g,h} \mathbf{n}_{K,f} \right\rangle_f \\
&\qquad + \frac{Z_{\mathrm{P},K}\, Z_{\mathrm{P},K_f}}{Z_{\mathrm{P},K} + Z_{\mathrm{P},K_f}} \left\langle \mathbf{n}_{K,f} \cdot [\mathbf{v}_h]_{K,f},\, Z_{\mathrm{P},K} \mathbf{n}_{K,f} \cdot \boldsymbol{\psi}_{K,h} \right\rangle_f \\
&\quad - \sum_{f \in \mathcal{F}_K} \frac{1}{Z_{\mathrm{S},K} + Z_{\mathrm{S},K_f}} \left\langle \boldsymbol{\tau}_{K,f} \cdot \sum_{g=0}^{G} [\boldsymbol{\sigma}_{g,h}]_{K,f} \mathbf{n}_{K,f},\, \boldsymbol{\tau}_{K,f} \cdot \sum_{g=0}^{G} \boldsymbol{\varphi}_{K,g,h} \mathbf{n}_{K,f} \right\rangle_f \\
&\qquad + \frac{Z_{\mathrm{S},K}}{Z_{\mathrm{S},K} + Z_{\mathrm{S},K_f}} \left\langle \boldsymbol{\tau}_{K,f} \cdot \sum_{g=0}^{G} [\boldsymbol{\sigma}_{g,h}]_{K,f} \mathbf{n}_{K,f},\, \boldsymbol{\tau}_{K,f} \cdot \boldsymbol{\psi}_{K,h} \right\rangle_f \\
&\qquad + \frac{Z_{\mathrm{S},K_f}}{Z_{\mathrm{S},K} + Z_{\mathrm{S},K_f}} \left\langle \boldsymbol{\tau}_{K,f} \cdot [\mathbf{v}_h]_{K,f},\, \boldsymbol{\tau}_{K,f} \cdot \sum_{g=0}^{G} \boldsymbol{\varphi}_{K,g,h} \mathbf{n}_{K,f} \right\rangle_f \\
&\qquad + \frac{Z_{\mathrm{S},K}\, Z_{\mathrm{S},K_f}}{Z_{\mathrm{S},K} + Z_{\mathrm{S},K_f}} \left\langle \boldsymbol{\tau}_{K,f} \cdot [\mathbf{v}_h]_{K,f},\, \boldsymbol{\tau}_{K,f} \cdot \boldsymbol{\psi}_{K,h} \right\rangle_f .
\end{aligned}
$$

We can follow the same steps as for the visco-acoustic case and conclude that on boundary faces $f = \partial K \cap \partial\Omega$, we set $[\mathbf{v}_h]_{K,f} = -2\mathbf{v}_{K,h}$ and $[\boldsymbol{\sigma}_{g,h}]_{K,f} = 0$ for Dirichlet boundary conditions in the velocity component.

Lemma 3.2. *Using a discontinuous Galerkin discretization with upwind flux for visco-elastic equations guarantees that the discrete hyperbolic operator is non-negative, i.e.,* (3.4) *holds.*

Proof. The proof will be done for 2D.
To simplify the notation we define

$$
\begin{aligned}
\alpha_1^K &= \frac{1}{Z_{\mathrm{P},K} + Z_{\mathrm{P},K_f}}, & \alpha_2^K &= \frac{Z_{\mathrm{P},K}}{Z_{\mathrm{P},K} + Z_{\mathrm{P},K_f}}, \\
\alpha_3^K &= \frac{Z_{\mathrm{P},K_f}}{Z_{\mathrm{P},K} + Z_{\mathrm{P},K_f}}, & \alpha_4^K &= \frac{Z_{\mathrm{P},K}\, Z_{\mathrm{P},K_f}}{Z_{\mathrm{P},K} + Z_{\mathrm{P},K_f}}, \\
\alpha_5^K &= \frac{1}{Z_{\mathrm{S},K} + Z_{\mathrm{S},K_f}}, & \alpha_6^K &= \frac{Z_{\mathrm{S},K}}{Z_{\mathrm{S},K} + Z_{\mathrm{S},K_f}}, \\
\alpha_7^K &= \frac{Z_{\mathrm{S},K_f}}{Z_{\mathrm{S},K} + Z_{\mathrm{S},K_f}}, & \alpha_8^K &= \frac{Z_{\mathrm{S},K}\, Z_{\mathrm{S},K_f}}{Z_{\mathrm{S},K} + Z_{\mathrm{S},K_f}}.
\end{aligned}
$$

We want to note that $\alpha_2^K = \alpha_3^{K_f}$ and $\alpha_6^K = \alpha_7^{K_f}$.

On each K it holds using $\boldsymbol{\sigma}_h = \sum_{g=0}^{G} \boldsymbol{\sigma}_{g,h}$ and $1 = \alpha_2^K + \alpha_3^K = \alpha_6^K + \alpha_7^K$

$$
\begin{aligned}
&\langle \boldsymbol{\sigma}_{K,h},\, \boldsymbol{\varepsilon}(\mathbf{v}_{K,h}) \rangle_K + \langle \operatorname{div} \boldsymbol{\sigma}_{K,h},\, \mathbf{v}_{K,h} \rangle_K = \langle \boldsymbol{\sigma}_{K,h},\, \nabla \mathbf{v}_{K,h} \rangle_K + \langle \operatorname{div} \boldsymbol{\sigma}_{K,h},\, \mathbf{v}_{K,h} \rangle_K \\
&\quad = \sum_{f \in \mathcal{F}_K} \langle \boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f},\, \mathbf{v}_{K,h} \rangle_f \\
&\quad = \sum_{f \in \mathcal{F}_K} \langle \mathbf{n}_{K,f} \cdot (\boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f}),\, \mathbf{n}_{K,f} \cdot \mathbf{v}_{K,h} \rangle_f \\
&\qquad\quad + \langle \boldsymbol{\tau}_{K,f} \cdot (\boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f}),\, \boldsymbol{\tau}_{K,f} \cdot \mathbf{v}_{K,h} \rangle_f \\
&\quad = \sum_{f \in \mathcal{F}_K} \alpha_2^K \langle \mathbf{n}_{K,f} \cdot (\boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f}),\, \mathbf{n}_{K,f} \cdot \mathbf{v}_{K,h} \rangle_f \\
&\qquad\quad + \alpha_2^{K_f} \langle \mathbf{n}_{K,f} \cdot (\boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f}),\, \mathbf{n}_{K,f} \cdot \mathbf{v}_{K,h} \rangle_f \\
&\qquad\quad + \alpha_6^K \langle \boldsymbol{\tau}_{K,f} \cdot (\boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f}),\, \boldsymbol{\tau}_{K,f} \cdot \mathbf{v}_{K,h} \rangle_f \\
&\qquad\quad + \alpha_6^{K_f} \langle \boldsymbol{\tau}_{K,f} \cdot (\boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f}),\, \boldsymbol{\tau}_{K,f} \cdot \mathbf{v}_{K,h} \rangle_f \\
&\quad = \sum_{f \in \mathcal{F}_K} \alpha_2^K \langle \mathbf{n}_{K,f} \cdot (\boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f}),\, \mathbf{n}_{K,f} \cdot \mathbf{v}_{K,h} \rangle_f \\
&\qquad\quad + \alpha_3^K \langle \mathbf{n}_{K,f} \cdot (\boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f}),\, \mathbf{n}_{K,f} \cdot \mathbf{v}_{K,h} \rangle_f \\
&\qquad\quad + \alpha_6^K \langle \boldsymbol{\tau}_{K,f} \cdot (\boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f}),\, \boldsymbol{\tau}_{K,f} \cdot \mathbf{v}_{K,h} \rangle_f \\
&\qquad\quad + \alpha_7^K \langle \boldsymbol{\tau}_{K,f} \cdot (\boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f}),\, \boldsymbol{\tau}_{K,f} \cdot \mathbf{v}_{K,h} \rangle_f
\end{aligned}
$$

and therefore

$$
\begin{aligned}
&\langle A_h \mathbf{u}_h,\, \mathbf{u}_h \rangle_\Omega \\
&= -\sum_K \langle \boldsymbol{\sigma}_{K,h},\, \boldsymbol{\varepsilon}(\mathbf{v}_{K,h}) \rangle_K + \langle \operatorname{div} \boldsymbol{\sigma}_{K,h},\, \mathbf{v}_{K,h} \rangle_K \\
&\quad + \sum_{f \in \mathcal{F}_K} \alpha_1^K \langle \mathbf{n}_{K,f} \cdot ([\boldsymbol{\sigma}_h]_{K,f} \mathbf{n}_{K,f}),\, \mathbf{n}_{K,f} \cdot \boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f} \rangle_f \\
&\qquad\qquad + \alpha_2^K \langle \mathbf{n}_{K,f} \cdot ([\boldsymbol{\sigma}_h]_{K,f} \mathbf{n}_{K,f}),\, \mathbf{n}_{K,f} \cdot \mathbf{v}_{K,h} \rangle_f \\
&\qquad\qquad + \alpha_3^K \langle \mathbf{n}_{K,f} \cdot [\mathbf{v}_h]_{K,f},\, \mathbf{n}_{K,f} \cdot \boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f} \rangle_f \\
&\qquad\qquad + \alpha_4^K \langle \mathbf{n}_{K,f} \cdot [\mathbf{v}_h]_{K,f},\, \mathbf{n}_{K,f} \cdot \mathbf{v}_{K,h} \rangle_f \\
&\qquad\qquad + \alpha_5^K \langle \boldsymbol{\tau}_{K,f} \cdot ([\boldsymbol{\sigma}_h]_{K,f} \mathbf{n}_{K,f}),\, \boldsymbol{\tau}_{K,f} \cdot \boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f} \rangle_f \\
&\qquad\qquad + \alpha_6^{K_f} \langle \boldsymbol{\tau}_{K,f} \cdot ([\boldsymbol{\sigma}_h]_{K,f} \mathbf{n}_{K,f}),\, \boldsymbol{\tau}_{K,f} \cdot \mathbf{v}_{K,h} \rangle_f \\
&\qquad\qquad + \alpha_7^K \langle \boldsymbol{\tau}_{K,f} \cdot [\mathbf{v}_h]_{K,f},\, \boldsymbol{\tau}_{K,f} \cdot \boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f} \rangle_f \\
&\qquad\qquad + \alpha_8^K \langle \boldsymbol{\tau}_{K,f} \cdot [\mathbf{v}_h]_{K,f},\, \boldsymbol{\tau}_{K,f} \cdot \mathbf{v}_{K,h} \rangle_f \\
&= -\sum_K \sum_{f \in \mathcal{F}_K} \alpha_1^K \langle \mathbf{n}_{K,f} \cdot ([\boldsymbol{\sigma}_h]_{K,f} \mathbf{n}_{K,f}),\, \mathbf{n}_{K,f} \cdot \boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f} \rangle_f \\
&\qquad\qquad + \alpha_2^K \left\langle \mathbf{n}_{K,f} \cdot (\boldsymbol{\sigma}_{K_f,h} \mathbf{n}_{K,f}),\, \mathbf{n}_{K,f} \cdot \mathbf{v}_{K,h} \right\rangle_f \\
&\qquad\qquad + \alpha_3^K \left\langle \mathbf{n}_{K,f} \cdot \mathbf{v}_{K_f,h},\, \mathbf{n}_{K,f} \cdot \boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f} \right\rangle_f \\
&\qquad\qquad + \alpha_4^K \langle \mathbf{n}_{K,f} \cdot [\mathbf{v}_h]_{K,f},\, \mathbf{n}_{K,f} \cdot \mathbf{v}_{K,h} \rangle_f \\
&\qquad\qquad + \alpha_5^K \langle \boldsymbol{\tau}_{K,f} \cdot ([\boldsymbol{\sigma}_h]_{K,f} \mathbf{n}_{K,f}),\, \boldsymbol{\tau}_{K,f} \cdot \boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f} \rangle_f \\
&\qquad\qquad + \alpha_6^K \left\langle \boldsymbol{\tau}_{K,f} \cdot (\boldsymbol{\sigma}_{K_f,h} \mathbf{n}_{K,f}),\, \boldsymbol{\tau}_{K,f} \cdot \mathbf{v}_{K,h} \right\rangle_f \\
&\qquad\qquad + \alpha_7^K \left\langle \boldsymbol{\tau}_{K,f} \cdot \mathbf{v}_{K_f,h},\, \boldsymbol{\tau}_{K,f} \cdot \boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f} \right\rangle_f \\
&\qquad\qquad + \alpha_8^K \langle \boldsymbol{\tau}_{K,f} \cdot [\mathbf{v}_h]_{K,f},\, \boldsymbol{\tau}_{K,f} \cdot \mathbf{v}_{K,h} \rangle_f \,.
\end{aligned}
$$

On inner faces $f \in \mathcal{F}_K \cap \Omega$ we have $\mathbf{n}_{K,f} = -\mathbf{n}_{K_f,f}$ together with $\alpha_2^K = \alpha_3^{K_f}$ and $\alpha_6^K = \alpha_7^{K_f}$ we get

$$
\begin{aligned}
0 =& \alpha_2^K \left\langle \mathbf{n}_{K,f} \cdot (\boldsymbol{\sigma}_{K_f,h} \mathbf{n}_{K,f}),\, \mathbf{n}_{K,f} \cdot \mathbf{v}_{K,h} \right\rangle_f + \alpha_3^K \left\langle \mathbf{n}_{K,f} \cdot \mathbf{v}_{K_f,h},\, \mathbf{n}_{K,f} \cdot \boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f} \right\rangle_f \\
&+ \alpha_6^K \left\langle \boldsymbol{\tau}_{K,f} \cdot (\boldsymbol{\sigma}_{K_f,h} \mathbf{n}_{K,f}),\, \boldsymbol{\tau}_{K,f} \cdot \mathbf{v}_{K,h} \right\rangle_f + \alpha_7^K \left\langle \boldsymbol{\tau}_{K,f} \cdot \mathbf{v}_{K_f,h},\, \boldsymbol{\tau}_{K,f} \cdot \boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f} \right\rangle_f \\
&+ \alpha_2^{K_f} \left\langle \mathbf{n}_{K_f,f} \cdot (\boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f}),\, \mathbf{n}_{K_f,f} \cdot \mathbf{v}_{K_f,h} \right\rangle_f + \alpha_3^{K_f} \left\langle \mathbf{n}_{K_f,f} \cdot \mathbf{v}_{K,h},\, \mathbf{n}_{K_f,f} \cdot \boldsymbol{\sigma}_{K_f,h} \mathbf{n}_{K_f,f} \right\rangle_f \\
&+ \alpha_6^{K_f} \left\langle \boldsymbol{\tau}_{K_f,f} \cdot (\boldsymbol{\sigma}_{K,h} \mathbf{n}_{K,f}),\, \boldsymbol{\tau}_{K_f,f} \cdot \mathbf{v}_{K_f,h} \right\rangle_f + \alpha_7^{K_f} \left\langle \boldsymbol{\tau}_{K_f,f} \cdot \mathbf{v}_{K,h},\, \boldsymbol{\tau}_{K_f,f} \cdot \boldsymbol{\sigma}_{K_f,h} \mathbf{n}_{K_f,f} \right\rangle_f \,.
\end{aligned}
$$

For boundary faces $f \in \mathcal{F}_K \cap \partial\Omega$ with Dirichlet boundary conditions for the

velocity component we get

$$\begin{aligned}
0 =&\alpha_2^K \left\langle \mathbf{n}_{K,f} \cdot (([\boldsymbol{\sigma}_h]_{K,f} + \boldsymbol{\sigma}_{K,h})\mathbf{n}_{K,f}),\, \mathbf{n}_{K,f} \cdot \mathbf{v}_{K,h} \right\rangle_f \\
&+ \alpha_3^K \left\langle \mathbf{n}_{K,f} \cdot ([\mathbf{v}_h]_{K,f} + \mathbf{v}_{K,h}),\, \mathbf{n}_{K,f} \cdot \boldsymbol{\sigma}_{K,h}\mathbf{n}_{K,f} \right\rangle_f \\
&+ \alpha_6^K \left\langle \boldsymbol{\tau}_{K,f} \cdot (([\boldsymbol{\sigma}_h]_{K,f} + \boldsymbol{\sigma}_{K,h})\mathbf{n}_{K,f}),\, \boldsymbol{\tau}_{K,f} \cdot \mathbf{v}_{K,h} \right\rangle_f \\
&+ \alpha_7^K \left\langle \boldsymbol{\tau}_{K,f} \cdot ([\mathbf{v}_h]_{K,f} + \mathbf{v}_{K,h}),\, \boldsymbol{\tau}_{K,f} \cdot \boldsymbol{\sigma}_{K,h}\mathbf{n}_{K,f} \right\rangle_f \\
=&\frac{1}{2} \left\langle \mathbf{n}_{K,f} \cdot (\boldsymbol{\sigma}_{K,h}\mathbf{n}_{K,f}),\, \mathbf{n}_{K,f} \cdot \mathbf{v}_{K,h} \right\rangle_f - \frac{1}{2} \left\langle \mathbf{n}_{K,f} \cdot \mathbf{v}_{K,h},\, \mathbf{n}_{K,f} \cdot \boldsymbol{\sigma}_{K,h}\mathbf{n}_{K,f} \right\rangle_f \\
&+ \frac{1}{2} \left\langle \boldsymbol{\tau}_{K,f} \cdot (\boldsymbol{\sigma}_{K,h}\mathbf{n}_{K,f}),\, \boldsymbol{\tau}_{K,f} \cdot \mathbf{v}_{K,h} \right\rangle_f - \frac{1}{2} \left\langle \boldsymbol{\tau}_{K,f} \cdot \mathbf{v}_{K,h},\, \boldsymbol{\tau}_{K,f} \cdot \boldsymbol{\sigma}_{K,h}\mathbf{n}_{K,f} \right\rangle_f .
\end{aligned}$$

With

$$\begin{aligned}
&\|\mathbf{n}_{K,f} \cdot [\mathbf{v}_h]_{K,f}\|_f^2 \\
&\quad = - \left\langle \mathbf{n}_{K,f} \cdot [\mathbf{v}_h]_{K,f},\, \mathbf{n}_{K,f} \cdot \mathbf{v}_{K,h} \right\rangle_f - \left\langle \mathbf{n}_{K_f,f} \cdot [\mathbf{v}_h]_{K_f,f},\, \mathbf{n}_{K_f,f} \cdot \mathbf{v}_{K_f,h} \right\rangle_f \\
&\|\boldsymbol{\tau}_{K,f} \cdot [\mathbf{v}_h]_{K,f}\|_f^2 \\
&\quad = - \left\langle \boldsymbol{\tau}_{K,f} \cdot [\mathbf{v}_h]_{K,f},\, \boldsymbol{\tau}_{K,f} \cdot \mathbf{v}_{K,h} \right\rangle_f - \left\langle \boldsymbol{\tau}_{K_f,f} \cdot [\mathbf{v}_h]_{K_f,f},\, \boldsymbol{\tau}_{K_f,f} \cdot \mathbf{v}_{K_f,h} \right\rangle_f
\end{aligned}$$

and

$$\begin{aligned}
&\|\mathbf{n}_{K,f} \cdot ([\boldsymbol{\sigma}_h]_{K,f}\mathbf{n}_{K,f})\|_f^2 \\
&\quad = - \left\langle \mathbf{n}_{K,f} \cdot ([\boldsymbol{\sigma}_h]_{K,f}\mathbf{n}_{K,f}),\, \mathbf{n}_{K,f} \cdot (\boldsymbol{\sigma}_{K,h}\mathbf{n}_{K,f}) \right\rangle_f \\
&\qquad\quad - \left\langle \mathbf{n}_{K_f,f} \cdot ([\boldsymbol{\sigma}_h]_{K_f,f}\mathbf{n}_{K_f,f}),\, \mathbf{n}_{K_f,f} \cdot (\boldsymbol{\sigma}_{K_f,h}\mathbf{n}_{K_f,f}) \right\rangle_f \\
&\|\boldsymbol{\tau}_{K,f} \cdot ([\boldsymbol{\sigma}_h]_{K,f}\mathbf{n}_{K,f})\|_f^2 \\
&\quad = - \left\langle \boldsymbol{\tau}_{K,f} \cdot ([\boldsymbol{\sigma}_h]_{K_f,f}\mathbf{n}_{K_f,f}),\, \boldsymbol{\tau}_{K_f,f} \cdot (\boldsymbol{\sigma}_{K_f,h}\mathbf{n}_{K_f,f}) \right\rangle_f \\
&\qquad\quad - \left\langle \boldsymbol{\tau}_{K_f,f} \cdot ([\boldsymbol{\sigma}_h]_{K_f,f}\mathbf{n}_{K_f,f}),\, \boldsymbol{\tau}_{K_f,f} \cdot (\boldsymbol{\sigma}_{K_f,h}\mathbf{n}_{K_f,f}) \right\rangle_f
\end{aligned}$$

we conclude and get

$$\begin{aligned}
&\langle A_h\mathbf{u}_h,\, \mathbf{u}_h \rangle_\Omega \\
&\quad = \frac{1}{2} \sum_K \sum_{f \in \mathcal{F}_K} \alpha_1^K \|\mathbf{n}_{K,f} \cdot ([\boldsymbol{\sigma}_h]_{K,f}\mathbf{n}_{K,f})\|_f^2 + \alpha_4^K \|\mathbf{n}_{K,f} \cdot [\mathbf{v}_h]_{K,f}\|_f^2 \\
&\qquad\qquad + \alpha_5^K \|\boldsymbol{\tau}_{K,f} \cdot ([\boldsymbol{\sigma}_h]_{K,f}\mathbf{n}_{K,f})\|_f^2 + \alpha_8^K \|\boldsymbol{\tau}_{K,f} \cdot [\mathbf{v}_h]_{K,f}\|_f^2 \\
&\quad \geq 0 .
\end{aligned}$$

□

3.3 Space–time discretizations

We extend the spatial discretization of the previous section to space-time discretizations based on tensor product space-time meshes. The first method uses ansatz functions which are discontinuous in space and time. The second discretization uses discontinuous ansatz functions in space, but continuous in time, combined with test functions discontinuous in space and time leading to a Petrov–Galerkin method.

3.3.1 Full discretization: discontinuous Galerkin in space and time

Let $\overline{Q} = \bigcup_{R\in\mathcal{R}} \overline{R}$ be a decomposition of the space-time cylinder into space-time cells $R = K \times I$ with $K \in \mathcal{K}$ and $I \subset [0,T]$ an interval; $\mathcal{R}$ denotes the set of space-time cells. For a fixed mesh $\mathcal{K}$ in space and a time series $0 = t_0 < t_1 < \cdots < t_N = T$, the space-time mesh is defined by

$$\mathcal{R} = \bigcup_{n=1,\ldots,N} \mathcal{R}_n\,, \qquad \mathcal{R}_n = \Big\{K \times I_n \colon I_n := (t_{n-1}, t_n],\ K \in \mathcal{K}\Big\}\,.$$

For every $R \in \mathcal{R}$ we choose local ansatz and test spaces

$$V_{h,R} = W_{h,R} = \mathbb{P}_{p_R}(K;\mathbb{R}^J) \otimes \mathbb{P}_{q_R}(I_n;\mathbb{R}^J) \subset \mathrm{L}_2(R;\mathbb{R}^J)$$

and define the global space

$$V_h = W_h = \Big\{\mathbf{v}_h \in \mathrm{L}_2((0,T);\mathrm{L}_2(\Omega;\mathbb{R}^J)) \colon \mathbf{v}_{h,R} = \mathbf{v}_h|_R \in V_{h,R}\Big\}\,.$$

In the following, we introduce the discontinuous Galerkin space-time scheme dG(q) of degree q, where dG(0) corresponds to the well known implicit Euler scheme.

Starting with the continuous variational formulation (2.7), i.e.,

$$\langle M\partial_t \mathbf{u} + A\mathbf{u} + D\mathbf{u},\, \mathbf{z}\rangle_Q = \langle \mathbf{b},\, \mathbf{z}\rangle_Q \qquad \mathbf{u} \in V,\, \mathbf{z} \in W$$

we get for smooth $\mathbf{z}$ with $\mathbf{z}(T) = \mathbf{0}$ and using partial integration in time

$$\begin{aligned} -\langle M\mathbf{u},\, \partial_t \mathbf{z}\rangle_Q - \langle M\mathbf{u}(0),\, \mathbf{z}(0)\rangle_\Omega + \langle A\mathbf{u} + D\mathbf{u},\, \mathbf{z}\rangle_Q \\ = \langle \mathbf{b},\, \mathbf{z}\rangle_Q\,. \end{aligned} \tag{3.7}$$

For $\mathbf{u}_h \in V_h$ we use the notation $\mathbf{u}_{h,n} = \mathbf{u}_h|_{I_n}$. By $[\![\mathbf{u}_h]\!]_{n-1}$ for $n = 1, \dots, N$ we denote the jump $[\![\mathbf{u}_h]\!]_{n-1} := \mathbf{u}^+_{h,n-1} - \mathbf{u}^-_{h,n-1}$ of $\mathbf{u}_h$ across t_{n-1}, where we use $\mathbf{u}^+_{h,n-1} := \lim_{t \downarrow t_{n-1}} \mathbf{u}_h|_{I_n}(t)$ and $\mathbf{u}^-_{h,n} := \mathbf{u}_h|_{I_n}(t_n)$. We choose to use homogeneous initial conditions $\mathbf{u}(0) = \mathbf{0}$. Therefore we have to define the value $\mathbf{u}^-_{h,0} := \mathbf{0}$.

Together with the smooth $\mathbf{z}$ mentioned above with $\mathbf{z}(t_N) = \mathbf{z}(T) = \mathbf{0}$ and the discrete version of the operators M_h, D_h and A_h from the last sections, we have

$$\begin{aligned}
&\int_0^T - \langle M_h\, \mathbf{u}_h,\, \partial_t \mathbf{z} \rangle_\Omega \, \mathrm{d}t \\
&\quad = \sum_{n=1}^N \int_{t_{n-1}}^{t_n} - \langle M_h \mathbf{u}_{h,n},\, \partial_t \mathbf{z} \rangle_\Omega \, \mathrm{d}t \\
&\quad = \sum_{n=1}^N \int_{t_{n-1}}^{t_n} \langle M_h \partial_t \mathbf{u}_{h,n},\, \mathbf{z} \rangle_\Omega \, \mathrm{d}t - \left\langle M_h \mathbf{u}^-_{h,n},\, \mathbf{z}(t_n) \right\rangle_\Omega + \left\langle M_h \mathbf{u}^+_{h,n-1},\, \mathbf{z}(t_{n-1}) \right\rangle_\Omega \\
&\quad = \sum_{n=1}^N \int_{t_{n-1}}^{t_n} \langle M_h \partial_t \mathbf{u}_{h,n},\, \mathbf{z} \rangle_\Omega \, \mathrm{d}t + \left\langle M_h \, [\![\mathbf{u}_h]\!]_{n-1} \,,\, \mathbf{z}(t_{n-1}) \right\rangle_\Omega \\
&\quad = \int_0^T \langle M_h \partial_t \mathbf{u}_h,\, \mathbf{z} \rangle_\Omega \, \mathrm{d}t + \sum_{n=2}^N \left\langle M_h \, [\![\mathbf{u}_h]\!]_{n-1} \,,\, \mathbf{z}(t_{n-1}) \right\rangle_\Omega + \left\langle M_h \mathbf{u}^+_{h,0},\, \mathbf{z}(t_0) \right\rangle_\Omega .
\end{aligned} \tag{3.8}$$

Combining (3.7) and (3.8) we define the discrete bilinear form

$$\begin{aligned}
\mathcal{B}^d_h(\mathbf{u}_h, \mathbf{w}_h) := {} & \langle M_h \partial_t \mathbf{u}_h + A_h \mathbf{u}_h + D_h \mathbf{u}_h,\, \mathbf{w}_h \rangle_Q \\
& + \sum_{n=1}^N \left\langle M_h \, [\![\mathbf{u}_h]\!]_{n-1} \,,\, \mathbf{w}_{h,n}(t_{n-1}) \right\rangle_\Omega
\end{aligned} \tag{3.9}$$

and the variational problem: find $\mathbf{u}_h \in V_h$ such that

$$\mathcal{B}^d_h(\mathbf{u}_h, \mathbf{w}_h) = \langle \mathbf{b},\, \mathbf{w}_h \rangle_Q \qquad \text{for all } \ \mathbf{w}_h \in W_h \,. \tag{3.10}$$

Remark 3.3. This formulation can be generalized to arbitrary initial conditions by adding them to the right hand side resulting in the variational problem: find $\mathbf{u}_h \in V_h$ such that

$$\mathcal{B}^d_h(\mathbf{u}_h, \mathbf{w}_h) = \langle \mathbf{b},\, \mathbf{w}_h \rangle_Q + \langle M \mathbf{u}(0),\, \mathbf{w}_h(0) \rangle_\Omega \,.$$

Conforming reconstruction in time

$\mathbf{u}_h$ is allowed to be discontinuous in time across the nodal points $t_0, \dots, t_{N-1}$ and hence in general $\mathbf{u}_h \notin V$, cf. Lem. 2.3. In order to construct from $\mathbf{u}_h$

a conforming function, we recall that the dG(q) schemes are closely related to Runge–Kutta–Radau IIA collocation methods, see [MN06, Lem. 2.3]. The corresponding Radau IIA quadrature formula with abscissae $c_1, \dots, c_{q+1}$ and weights $w_1, \dots, w_{q+1}$ is exact of degree $2q$, cf. Sec. A.1. In fact, we have

$$\sum_{j=1}^{q+1} w_j P(c_j) = \int_0^1 P(t)\,\mathrm{d}t \qquad \text{for all } P \in \mathbb{P}_{2q}. \tag{3.11}$$

We define $\widehat{\mathbf{u}} \in V$, $\widehat{\mathbf{u}}_R = \widehat{\mathbf{u}}|_R \in \mathbb{P}_{p_R}(K;\mathbb{R}^J) \otimes \mathbb{P}_{q_R+1}(I_n;\mathbb{R}^J)$ as the piecewise interpolation of $\mathbf{u}_h$ at the local Radau IIA points $t_n^j := t_{n-1} + c_j\tau_n$ with $\tau_n = t_n - t_{n-1}$, i.e.,

$$\widehat{\mathbf{u}}_R(t_n^j) = \mathbf{u}_R(t_n^j), \qquad j = 1, \dots, q+1. \tag{3.12a}$$

The continuous embedding of V in $C^0(0,T;\mathcal{D}(A))$ (cf. Lem. 2.3) additionally enforces $\widehat{\mathbf{u}}(t_{n-1}) = \mathbf{u}_{h,n-1}^-$. We relax this request for the adaptive case to

$$\widehat{\mathbf{u}}(t_{n-1}) = \Pi_n \mathbf{u}_{h,n-1}^- \tag{3.12b}$$

where $\Pi_n \colon H_{h,n-1} \to H_{h,n}$ is the L_2-projection in space with

$$H_{h,n} = \left\{ \mathbf{w}_h \in \mathrm{L}_2(\Omega;\mathbb{R}^J) \colon \mathbf{w}_h|_K \in \mathbb{P}_{p_R}(K;\mathbb{R}^J) \text{ for all } R = I_n \times K \in \mathcal{R}^n \right\}$$

defined by

$$\langle M_h \Pi_n \mathbf{z}_{n-1},\, \mathbf{z}_n \rangle_\Omega = \langle M_h \mathbf{z}_{n-1},\, \mathbf{z}_n \rangle_\Omega \tag{3.13}$$

for all $\mathbf{z}_n \in H_{h,n}$ and $\mathbf{z}_{n-1} \in H_{h,n-1}$. Hence $\widehat{\mathbf{u}}$ is uniquely defined by

$$\widehat{\mathbf{u}}|_{I_n} := \sum_{j=0}^{q+1} \mathcal{L}_j^{q+1}\left(\frac{t - t_{n-1}}{\tau_n}\right) \Pi_n \mathbf{u}_h(t_n^j) \qquad \text{for } t \in I_n \tag{3.14}$$

with the Lagrange polynomials

$$\mathcal{L}_j^{q+1}(s) := \prod_{\substack{i=0 \\ i\neq j}}^{s+1} \frac{s - c_j}{c_i - c_j} \in \mathbb{P}_{q+1}, \qquad j = 0, \dots, q+1 \tag{3.15}$$

and $c_0 := 0$ defining $t_n^0 = t_{n-1}$.

Lemma 3.3 (Lem. 2.2 in [MN06])**.** *The following representation of* $\widehat{\mathcal{I}}\mathbf{v}_h - \mathbf{v}_h$ *is valid*

$$\left(\widehat{\mathcal{I}}\mathbf{v}_h - \mathbf{v}_h\right)|_{I_n}(t) = [\![\Pi_n \mathbf{v}_h]\!]_{n-1}\, \mathcal{L}_0\left(\frac{t - t_{n-1}}{t_n - t_{n-1}}\right) \qquad \textit{for all } \ t \in I_n\,.$$

Proof. $\left(\widehat{\mathcal{I}}\mathbf{v}_h - \mathbf{v}_h\right)(t)|_{I_n}$ is a polynomial in time of degree $q+1$ vanishing at the Radau points t_n^j for $j = 1, \ldots, q+1$ since $\Pi_n \mathbf{v}_{h,n} = \mathbf{v}_{h,n}$. The claim follows from

$$\left(\widehat{\mathcal{I}}\mathbf{v}_h - \mathbf{v}_h\right)(t_n^0) = \Pi_n \mathbf{v}_{h,n-1}^- - \mathbf{v}_{h,n-1}^+ = \Pi_n \mathbf{v}_{h,n-1}^- - \Pi_n \mathbf{v}_{h,n-1}^+ .$$

□

The reconstruction operator $\widehat{\mathcal{I}}$ with the restriction on a space-time cell $\widehat{\mathcal{I}}|_R \colon V_{h,R} \to \widehat{V}_{h,R} = \mathbb{P}_{p_R}(K;\mathbb{R}^J) \otimes \mathbb{P}_{q_R+1}(I_n;\mathbb{R}^J)$ as introduced in [MN06, Lem. 2.1] has the properties mentioned above.

Lemma 3.4. *$\widehat{\mathbf{u}}|_R := \widehat{\mathcal{I}}\mathbf{u}_h|_R \in \widehat{V}_{h,R}$ satisfies in $R = K \times (t_{n-1}, t_n)$*

$$\widehat{\mathbf{u}}_h(t_{n-1}) = \Pi_n \mathbf{u}_h(t_{n-1})$$

and

$$\begin{aligned} &\int_{t_{n-1}}^{t_n} \langle M_h \partial_t \widehat{\mathbf{u}}_h,\, \mathbf{w}_h \rangle_K \,\mathrm{d}t \\ &\quad = \int_{t_{n-1}}^{t_n} \langle M_h \partial_t \mathbf{u}_h,\, \mathbf{w}_h \rangle_K \,\mathrm{d}t + \left\langle M_h(\mathbf{u}_h^+(t_{n-1}) - \mathbf{u}_h^-(t_{n-1})),\, \mathbf{w}_h^+(t_{n-1}) \right\rangle_K \end{aligned} \tag{3.16}$$

for all $\mathbf{w}_h \in W_{h,R}$.

Proof. Integrating (3.16) by parts we get

$$\begin{aligned} &\int_{t_{n-1}}^{t_n} \langle M_h \partial_t \widehat{\mathbf{u}}_h,\, \mathbf{w}_h \rangle_K \,\mathrm{d}t \\ &= -\int_{t_{n-1}}^{t_n} \langle M_h \widehat{\mathbf{u}}_h,\, \partial_t \mathbf{w}_h \rangle_K \,\mathrm{d}t \\ &\qquad + \left\langle M_h \widehat{\mathbf{u}}_h^-(t_n),\, \mathbf{w}_h^-(t_n) \right\rangle_K - \left\langle M_h \widehat{\mathbf{u}}_h^+(t_{n-1}),\, \mathbf{w}_h^+(t_{n-1}) \right\rangle_K \\ &= -\int_{t_{n-1}}^{t_n} \langle M_h \widehat{\mathbf{u}}_h,\, \partial_t \mathbf{w}_h \rangle_K \,\mathrm{d}t \\ &\qquad + \left\langle M_h \mathbf{u}_h^-(t_n),\, \mathbf{w}_h^-(t_n) \right\rangle_K - \left\langle M_h \Pi_n \mathbf{u}_h(t_{n-1}),\, \mathbf{w}_h^+(t_{n-1}) \right\rangle_K \\ &= -\int_{t_{n-1}}^{t_n} \langle M_h \widehat{\mathbf{u}}_h,\, \partial_t \mathbf{w}_h \rangle_K \,\mathrm{d}t \\ &\qquad + \left\langle M_h \mathbf{u}_h^-(t_n),\, \mathbf{w}_h^-(t_n) \right\rangle_K - \left\langle M_h \mathbf{u}_h(t_{n-1}),\, \mathbf{w}_h^+(t_{n-1}) \right\rangle_K \end{aligned}$$

and

$$\begin{aligned}
&\int_{t_{n-1}}^{t_n} \langle M_h \partial_t \mathbf{u}_h,\, \mathbf{w}_h \rangle_K \,\mathrm{d}t + \Big\langle M_h(\mathbf{u}_h^+(t_{n-1}) - \mathbf{u}_h^-(t_{n-1})),\, \mathbf{w}_h^+(t_{n-1}) \Big\rangle_K \\
&= -\int_{t_{n-1}}^{t_n} \langle M_h \mathbf{u}_h,\, \partial_t \mathbf{w}_h \rangle_K \,\mathrm{d}t \\
&\qquad + \Big\langle M_h \mathbf{u}_h^-(t_n),\, \mathbf{w}_h^-(t_n) \Big\rangle_K - \Big\langle M_h \mathbf{u}_h^+(t_{n-1}),\, \mathbf{w}_h^+(t_{n-1}) \Big\rangle_K \\
&\qquad + \Big\langle M_h(\mathbf{u}_h^+(t_{n-1}) - \mathbf{u}_h^-(t_{n-1})),\, \mathbf{w}_h^+(t_{n-1}) \Big\rangle_K \\
&= -\int_{t_{n-1}}^{t_n} \langle M_h \mathbf{u}_h,\, \partial_t \mathbf{w}_h \rangle_K \,\mathrm{d}t \\
&\qquad + \Big\langle M_h \mathbf{u}_h^-(t_n),\, \mathbf{w}_h^-(t_n) \Big\rangle_K - \Big\langle M_h \mathbf{u}_h^-(t_{n-1}),\, \mathbf{w}_h^+(t_{n-1}) \Big\rangle_K .
\end{aligned}$$

Note that $\mathbf{u}_h(t_{n_1}) = \mathbf{u}_h^-(t_{n_1})$ and with the exactness of the Radau integration rule on I_n we get

$$\begin{aligned}
&\int_{t_{n-1}}^{t_n} \langle M_h \mathbf{u}_h,\, \partial_t \mathbf{w}_h \rangle_K \,\mathrm{d}t = \tau_n \sum_{j=1}^{q+1} \Big\langle M_h \mathbf{u}_h(t_n^j),\, \partial_t \mathbf{w}_h(t_n^j) \Big\rangle_K \\
&= \tau_n \sum_{j=1}^{q+1} \Big\langle M_h \widehat{\mathbf{u}}_h(t_n^j),\, \partial_t \mathbf{w}_h(t_n^j) \Big\rangle_K = \int_{t_{n-1}}^{t_n} \langle M_h \widehat{\mathbf{u}}_h,\, \partial_t \mathbf{w}_h \rangle_K .
\end{aligned}$$

□

Using Lem. 3.4 we observe that (3.9) can be formulated equivalently by defining a new space-time operator $\widehat{L}_h$ by

$$\Big\langle \widehat{L}_h \mathbf{u}_h,\, \mathbf{v}_h \Big\rangle_Q = \Big\langle M_h \partial_t \widehat{\mathcal{I}} \mathbf{u}_h + A_h \mathbf{u}_h + D_h \mathbf{u}_h,\, \mathbf{v}_h \Big\rangle_Q . \tag{3.17}$$

This defines the variational problem: find $\mathbf{u}_h \in V_h$ such that

$$\Big\langle \widehat{L}_h \mathbf{u}_h,\, \mathbf{w}_h \Big\rangle_Q = \langle \mathbf{b},\, \mathbf{w}_h \rangle_Q \qquad \text{for all } \ \mathbf{w}_h \in W_h . \tag{3.18}$$

We use the norms

$$\|\cdot\|_{W_h}^2 = \langle M_h \cdot,\, \cdot \rangle_Q \quad \text{and} \quad \|\cdot\|_{V_h}^2 = \|\cdot\|_{W_h}^2 + \|M_h^{-1} \widehat{L}_h \cdot\|_{W_h}$$

on the discrete spaces. With this, we can prove discrete inf-sup stability, but at first we need an auxiliary result.

Lemma 3.5 (Lem. 4.4 in [Fin16])**.** *For every $\phi \in \mathrm{L}_1(0,T)$ it holds that*

$$\int_0^T \int_0^t \phi(s) \,\mathrm{d}s \,\mathrm{d}t = \int_0^T d_T(t) \phi(t) \,\mathrm{d}t$$

with weight function $d_T(t) = T - t \geq 0$.

This can be verified by Fubini's theorem.

Theorem 3.1 (discrete inf-sup condition)**.** *Assume that*

$$\langle M_h \partial_t \widehat{\mathbf{v}}_h,\, d_T(\mathbf{v}_h - \widehat{\mathbf{v}}_h)\rangle_Q \geq 0 \qquad \textit{for all } \mathbf{v}_h \in V_h \tag{3.19}$$

and

$$\|\mathbf{v}_h\|_{W_h} \leq C_q\, \|\widehat{\mathcal{I}}\mathbf{v}_h\|_{W_h} \qquad \textit{for all } \mathbf{v}_h \in V_h\,. \tag{3.20}$$

Then the bilinear form

$$\mathcal{B}_h^c(\mathbf{u}_h, \mathbf{w}_h) = \left\langle \widehat{L}_h \mathbf{u}_h,\, \mathbf{w}_h \right\rangle_Q$$

is bounded and inf-sup stable in $V_h \times W_h$ with $\beta = (1 + 4T^2 C_q^2)^{-1/2}$ and hence for given $\mathbf{b} \in \mathrm{L}_2(Q;\mathbb{R}^J)$ there exists a unique solution $\mathbf{u}_h \in V_h$ solving the variational problem (3.18)*.*

Proof. First we have a closer look at the conforming reconstruction regarding the jumps in time, i.e.,

$$\begin{aligned}
&\left\langle M_h \widehat{\mathbf{v}}_h^-(t_{n-1}),\, \widehat{\mathbf{v}}_h^-(t_{n-1})\right\rangle_\Omega - \left\langle M_h \widehat{\mathbf{v}}_h^+(t_{n-1}),\, \widehat{\mathbf{v}}_h^+(t_{n-1})\right\rangle_\Omega \\
&= \langle M_h \mathbf{v}_{h,n-1}(t_{n-1}),\, \mathbf{v}_{h,n-1}(t_{n-1})\rangle_\Omega - \langle M_h \Pi_n \mathbf{v}_{h,n-1}(t_{n-1}),\, \Pi_n \mathbf{v}_{h,n-1}(t_{n-1})\rangle_\Omega \\
&= \|M_h^{1/2} \mathbf{v}_{h,n-1}(t_{n-1})\|_\Omega^2 - \|M_h^{1/2} \Pi_n \mathbf{v}_{h,n-1}(t_{n-1})\|_\Omega^2\,.
\end{aligned} \tag{3.21}$$

Since Π_n is a projection, we have for the case $H_{h,n-1} \subset H_{h,n}$

$$\|M_h^{1/2} \mathbf{v}_{h,n-1}(t_{n-1})\|_\Omega = \|M_h^{1/2} \Pi_n \mathbf{v}_{h,n-1}(t_{n-1})\|_\Omega$$

but in general it holds that

$$\|M_h^{1/2} \mathbf{v}_{h,n-1}(t_{n-1})\|_\Omega \geq \|M_h^{1/2} \Pi_n \mathbf{v}_{h,n-1}(t_{n-1})\|_\Omega\,.$$

With the estimate for the jumps in time (3.21) and without loss of generality

we assume $t \in I_{\widehat{n}}$ and get

$$\begin{aligned}
\frac{1}{C_q}\|\mathbf{v}_h\|_{W_h}^2 &\leq \|\widehat{\mathcal{I}}\mathbf{v}_h\|_{W_h}^2 \\
&= \int_0^T \langle M_h\widehat{\mathbf{v}}_h(t),\, \widehat{\mathbf{v}}_h(t)\rangle_\Omega \;\mathrm{d}t \\
&= \int_0^T \int_{t_{\widehat{n}-1}}^t \partial_s \langle M_h\widehat{\mathbf{v}}_h(s),\, \widehat{\mathbf{v}}_h(s)\rangle_\Omega \,\mathrm{d}s + \left\langle M_h\widehat{\mathbf{v}}_h^+(t_{\widehat{n}-1}),\, \widehat{\mathbf{v}}_h^+(t_{\widehat{n}-1})\right\rangle_\Omega \mathrm{d}t \\
&= \int_0^T \int_{t_{\widehat{n}-1}}^t \partial_s \langle M_h\widehat{\mathbf{v}}_h(s),\, \widehat{\mathbf{v}}_h(s)\rangle_\Omega \,\mathrm{d}s + \left\langle M_h\widehat{\mathbf{v}}_h^+(t_{\widehat{n}-1}),\, \widehat{\mathbf{v}}_h^+(t_{\widehat{n}-1})\right\rangle_\Omega \\
&\quad + \sum_{n=1}^{\widehat{n}-1} \int_{t_{n-1}}^{t_n} \partial_s \langle M_h\widehat{\mathbf{v}}_h(s),\, \widehat{\mathbf{v}}_h(s)\rangle_\Omega \,\mathrm{d}s \\
&\qquad - \left\langle M_h\widehat{\mathbf{v}}_h^-(t_n),\, \widehat{\mathbf{v}}_h^-(t_n)\right\rangle_\Omega + \left\langle M_h\widehat{\mathbf{v}}_h^+(t_{n-1}),\, \widehat{\mathbf{v}}_h^+(t_{n-1})\right\rangle_\Omega \;\mathrm{d}t \\
&= \int_0^T \int_0^t \partial_s \langle M_h\widehat{\mathbf{v}}_h(s),\, \widehat{\mathbf{v}}_h(s)\rangle_\Omega \,\mathrm{d}s \\
&\quad + \sum_{\substack{n=1 \\ t_n<t}}^{N} - \left\langle M_h\widehat{\mathbf{v}}_h^-(t_n),\, \widehat{\mathbf{v}}_h^-(t_n)\right\rangle_\Omega + \left\langle M_h\widehat{\mathbf{v}}_h^+(t_n),\, \widehat{\mathbf{v}}_h^+(t_n)\right\rangle_\Omega \;\mathrm{d}t \\
&\leq \int_0^T \int_0^t \partial_s \langle M_h\widehat{\mathbf{v}}_h(s),\, \widehat{\mathbf{v}}_h(s)\rangle_\Omega \,\mathrm{d}s \;\mathrm{d}t \\
&= 2\int_0^T \int_0^t \langle M_h\partial_t\widehat{\mathbf{v}}_h(s),\, \widehat{\mathbf{v}}_h(s)\rangle_\Omega \,\mathrm{d}s \;\mathrm{d}t \\
&= 2\,\langle M_h\partial_t\widehat{\mathbf{v}}_h,\, d_T\widehat{\mathbf{v}}_h\rangle_Q \,.
\end{aligned}$$

In the next step, we apply assumption (3.19). Since D_h is positive semi-definite and the hyperbolic operator is non-negative, guaranteed by the use of the upwind flux (cf.(3.4)), we can additionally insert $0 \leq \langle (A_h + D_h)\mathbf{v}_h,\, \mathbf{v}_h\rangle_\Omega$. This yields

$$\begin{aligned}
\|\mathbf{v}_h\|_{W_h}^2 &\leq 2C_q\,\langle M_h\partial_t\widehat{\mathbf{v}}_h,\, d_T\mathbf{v}_h\rangle_Q \\
&\leq 2C_q\left\langle M_h\partial_t\widehat{\mathcal{I}}\mathbf{v}_h(t) + A_h\mathbf{v}_h + D_h\mathbf{v}_h,\, d_T\mathbf{v}_h\right\rangle_Q \\
&\leq 2TC_q\|M_h^{-1}\widehat{L}_h\mathbf{v}_h\|_{W_h}\|\mathbf{v}_h\|_{W_h} \,.
\end{aligned}$$

Hence we achieve $\|\mathbf{v}_h\|_{W_h} \leq 2TC_q\|M_h^{-1}\widehat{L}_h\mathbf{v}_h\|_{W_h}$ and for the discrete norm in V_h we get $\|\mathbf{v}_h\|_{V_h}^2 = \|\mathbf{v}_h\|_{W_h}^2 + \|M_h^{-1}\widehat{L}_h\mathbf{v}_h\|_{W_h}^2 \leq (1 + 4T^2C_q^2)\|M_h^{-1}\widehat{L}_h\mathbf{v}_h\|_{W_h}^2$.

Inserting the special choice of $\mathbf{w}_h = M_h^{-1}\widehat{L}_h\mathbf{v}_h$ into the estimate results in

$$\sup_{\mathbf{w}_h\in\mathbf{W}_h}\frac{\mathcal{B}(\mathbf{v}_h,\mathbf{w}_h)}{\left\|\mathbf{w}_h\right\|_{W_h}} = \sup_{\mathbf{w}_h\in\mathbf{W}_h}\frac{\left\langle\widehat{L}_h\mathbf{v}_h,\ \mathbf{w}_h\right\rangle_Q}{\left\|\mathbf{w}_h\right\|_{W_h}} \geq \frac{\left\langle\widehat{L}_h\mathbf{v}_h,\ M_h^{-1}\widehat{L}_h\mathbf{v}_h\right\rangle_Q}{\left\|M_h^{-1}\widehat{L}_h\mathbf{v}_h\right\|_{W_h}}$$
$$= \frac{\left\|M_h^{-1}\widehat{L}_h\mathbf{v}_h\right\|_{W_h}^2}{\left\|M_h^{-1}\widehat{L}_h\mathbf{v}_h\right\|_{W_h}} \geq \beta\left\|\mathbf{v}_h\right\|_{V_h}, \quad \mathbf{v}_h\in V_h\,.$$

The bilinear form is bounded by

$$|\mathcal{B}_h^c(\mathbf{u}_h,\mathbf{w}_h)| = \left|\left\langle\widehat{L}_h\mathbf{u}_h,\ \mathbf{w}_h\right\rangle_Q\right| = \left|\left\langle M_hM_h^{-1}\widehat{L}_h\mathbf{u}_h,\ \mathbf{w}_h\right\rangle_Q\right|$$
$$\leq \|M_h^{-1}\widehat{L}_h\mathbf{u}_h\|_{W_h}\ \|\mathbf{w}_h\|_{W_h} \leq \|\mathbf{u}_h\|_{V_h}\ \|\mathbf{w}_h\|_{W_h}\,.$$

□

Lemma 3.6. *Let $\widehat{\mathbf{v}}_h$ be the conforming reconstruction of $\mathbf{v}_h$. Then the assumption* (3.20) *holds.*

Proof. We will prove, that the constant C_q depends only on the polynomial order in time q of the discretization, but not on the mesh refinement and therefore not on N.
We observe that the integrals can be decomposed into space and time, i.e.,

$$\langle M_h\mathbf{w}_h,\ \mathbf{w}_h\rangle_R = \left\langle M_h\sum_{j=1}^{q_R+1}\mathcal{L}_{n,j}^{q_R}\mathbf{w}_h(t_n^j),\ \sum_{k=1}^{q_R+1}\mathcal{L}_{n,k}^{q_R}\mathbf{w}_h(t_n^k)\right\rangle_{R=K\times I_n}$$
$$= \sum_{j=1}^{q_R+1}\sum_{k=1}^{q_R+1}\int_{I_n}\mathcal{L}_{n,j}^{q_R}\mathcal{L}_{n,k}^{q_R}\,\mathrm{d}t\ \left\langle M_h\mathbf{w}_h(t_n^j),\ \mathbf{w}_h(t_n^k)\right\rangle_K$$

for a space-time cell $R = K\times I_n$ and

$$\langle M_h\widehat{\mathbf{w}}_h,\ \widehat{\mathbf{w}}_h\rangle_R = \left\langle M_h\sum_{j=0}^{q_R+1}\mathcal{L}_{n,j}^{q_R+1}\mathbf{w}_h(t_n^j),\ \sum_{k=0}^{q_R+1}\mathcal{L}_{n,k}^{q_R+1}\mathbf{w}_h(t_n^k)\right\rangle_{R=K\times I_n}$$
$$= \sum_{j=0}^{q_R+1}\sum_{k=0}^{q_R+1}\int_{I_n}\mathcal{L}_{n,j}^{q_R+1}\mathcal{L}_{n,k}^{q_R+1}\,\mathrm{d}t\ \left\langle M_h\mathbf{w}_h(t_n^j),\ \mathbf{w}_h(t_n^k)\right\rangle_K\,.$$

This motivates the definition of the vectors

$$\underline{v_R} = \left(M_h^{1/2}\mathbf{v}_h(t_n^1),\ldots,M_h^{1/2}\mathbf{v}_h(t_n^{q+1})\right)^\top,$$
$$\underline{\widehat{v}_R} = \left(M_h^{1/2}\Pi_n\mathbf{v}_h(t_n^0),\ M_h^{1/2}\mathbf{v}_h(t_n^1),\ldots,M_h^{1/2}\mathbf{v}_h(t_n^{q+1})\right)^\top$$

and the matrices $A^R,\ B^R$ defined on $R = K \times I_n$ by

$$\begin{aligned} A^R_{jk} &= \int_{I_n} \mathcal{L}^{q_R}_{n,j}(t)\, \mathcal{L}^{q_R}_{n,k}(t)\, \mathrm{d}t \\ &= \tau_n \int_0^1 \mathcal{L}^{q_R}_{j}(t)\, \mathcal{L}^{q_R}_{k}(t)\, \mathrm{d}t = \tau_n A^{q_R}_{jk} \qquad \in \mathbb{R}^{(q_R+1)\times(q_R+1)}\,, \\ B^R_{jk} &= \int_{I_n} \mathcal{L}^{q_R+1}_{n,j}(t)\, \mathcal{L}^{q_R+1}_{n,k}(t)\, \mathrm{d}t \\ &= \tau_n \int_0^1 \mathcal{L}^{q_R+1}_{j}(t)\, \mathcal{L}^{q_R+1}_{k}(t)\, \mathrm{d}t = \tau_n B^{q_R}_{jk} \qquad \in \mathbb{R}^{(q_R+2)\times(q_R+2)}\,, \end{aligned}$$

with the transformed Lagrange polynomials

$$\mathcal{L}^q_{n,j}(t) = \mathcal{L}^q_j\left(\frac{t - t_{n-1}}{t_n - t_{n-1}}\right).$$

B^R is the local mass matrix and therefore positive definite. We obtain

$$\begin{aligned} \|\mathbf{v}_h\|^2_{W_h} &= \sum_{R\in\mathcal{R}} \int_K \underline{v_R}^\top A^R \underline{v_R}\, \mathrm{d}\mathbf{x} \le \sum_{R\in\mathcal{R}} \lambda_{\max}(A^R) \sum_{k=1}^{q+1} \left\langle \underline{v_R},\, \underline{v_R} \right\rangle_K \\ &\le \sum_{R\in\mathcal{R}} \lambda_{\max}(A^R) \sum_{k=0}^{q+1} \left\langle \underline{\widehat{v}_R},\, \underline{\widehat{v}_R} \right\rangle_K \\ &\le \sum_{R\in\mathcal{R}} \lambda_{\max}(A^R)\lambda_{\min}(B^R)^{-1} \int_K \underline{\widehat{v}_R}^\top B^R \underline{\widehat{v}_R}\, \mathrm{d}\mathbf{x} \\ &\le \max_{R\in\mathcal{R}} \lambda_{\max}(A^R)\lambda_{\min}(B^R)^{-1} \|\widehat{\mathbf{v}}_h\|^2_{W_h}\,. \end{aligned}$$

Since

$$\lambda_{\max}(A^R)\lambda_{\min}(B^R)^{-1} = \lambda_{\max}(\tau_n A^{q_R})\lambda_{\min}(\tau_n B^{q_R})^{-1} = \lambda_{\max}(A^{q_R})\lambda_{\min}(B^{q_R})^{-1}$$

we conclude that the constant C_q is given by

$$C_q^2 = \max_{q=q_{\min},\dots,q_{\max}} \lambda_{\min}(A^q)\lambda_{\max}(B^q)^{-1}\,.$$

Here $q_{\min}$ denotes the lowest polynomial degree in time and $q_{\max}$ the highest polynomial degree in time used by the discretization.

□

Lemma 3.7. *Let $\widehat{\mathbf{v}}_h$ be the conforming reconstruction of $\mathbf{v}_h$. Then the assumption* (3.19) *holds for $q = 0, \dots, 5$.*

Proof. In a first step, we prove the case $q = 0$. In this case we have

$$\begin{aligned}
\mathbf{v}_h|_{I_n} &= \mathbf{v}_{h,n}\,, \\
\widehat{\mathbf{v}}_h|_{I_n} &= \frac{t_n - t}{t_n - t_{n-1}} \Pi_n \mathbf{v}_{h,n-1} + \frac{t - t_{n-1}}{t_n - t_{n-1}} \mathbf{v}_{h,n}\,, \\
\partial_t \widehat{\mathbf{v}}_h|_{I_n} &= \frac{1}{t_n - t_{n-1}} (\mathbf{v}_{h,n} - \Pi_n \mathbf{v}_{h,n-1})\,.
\end{aligned}$$

This results in

$$\begin{aligned}
&\langle M_h \partial_t \widehat{\mathbf{v}}_h,\, d_T(\mathbf{v}_h - \widehat{\mathbf{v}}_h)\rangle_Q = \sum_{R\in\mathcal{R}} \langle M_h \partial_t \widehat{\mathbf{v}}_h,\, d_T(\mathbf{v}_h - \widehat{\mathbf{v}}_h)\rangle_R \\
&= \sum_{R\in\mathcal{R}} \left\langle M_h \frac{1}{t_n - t_{n-1}} (\mathbf{v}_{h,n} - \Pi_n \mathbf{v}_{h,n-1}),\, d_T \left(\mathbf{v}_{h,n-1} - \widehat{\mathbf{v}}_h|_{I_n}\right) \right\rangle_R \\
&= \sum_{R\in\mathcal{R}} \left\langle M_h \frac{1}{t_n - t_{n-1}} (\mathbf{v}_{h,n} - \Pi_n \mathbf{v}_{h,n-1}),\, d_T \left(\Pi_n \mathbf{v}_{h,n-1} - \widehat{\mathbf{v}}_h|_{I_n}\right) \right\rangle_R \\
&= \sum_{R\in\mathcal{R}} \left\langle M_h \frac{1}{t_n - t_{n-1}} (\mathbf{v}_{h,n} - \Pi_n \mathbf{v}_{h,n-1}),\, d_T \frac{t_n - t}{t_n - t_{n-1}} (\mathbf{v}_{h,n} - \Pi_n \mathbf{v}_{h,n-1}) \right\rangle_R \\
&= \sum_{R\in\mathcal{R}} \langle M_h (\mathbf{v}_{h,n} - \Pi_n \mathbf{v}_{h,n-1}),\, \mathbf{v}_{h,n} - \Pi_n \mathbf{v}_{h,n-1} \rangle_K \int_{I_n} \frac{1}{t_n - t_{n-1}} d_T \frac{t_n - t}{t_n - t_{n-1}}\, \mathrm{d}t \\
&= \sum_{R\in\mathcal{R}} \langle M_h (\mathbf{v}_{h,n} - \Pi_n \mathbf{v}_{h,n-1}),\, \mathbf{v}_{h,n} - \Pi_n \mathbf{v}_{h,n-1} \rangle_K \frac{1}{6} (3T - t_n - 2t_{n-1}) \geq 0\,.
\end{aligned}$$

We use that $\langle M_h \mathbf{w},\, \mathbf{w}\rangle_\Omega \geq 0$ and $T \geq t_n > t_{n-1}$.

For a more general approach we transfer the estimate from an arbitrary space-time cell $R = K \times I_n$ with interval $I_n = (t_{n-1}, t_n)$ to the reference time interval $(0, 1)$. For polynomial order $q = q_R$ in time we define by $\mathcal{L}_j^q$, $j = 1, \ldots, q+1$, the j-th Lagrange polynomial of degree q with respect to the quadrature points of the Radau IIA integration rule with $q+1$ points shown in Tab. A.1. By $\mathcal{L}_j^{q+1}$ we denote the Lagrange polynomial of order $q+1$ by adding the quadrature point at zero.

A function $\mathbf{v}_h$ can be written with $t_n^j := t_{n-1} + c_j \tau_n$ and $\tau_n = t_{n+1} - t_n$ as

$$\mathbf{v}_h(\mathbf{x}, t)|_{I_n} = \sum_{j=1}^{q+1} \mathbf{v}_h(t_n^j) \mathcal{L}_j^q \left(\frac{t - t_{n-1}}{\tau_n} \right)$$

and the conforming reconstruction by adding the integration point $t_n^0 = t_n$

$$\widehat{\mathbf{v}}_h(\mathbf{x}, t)|_{I_n} = \sum_{j=0}^{q+1} \Pi_n \mathbf{v}_h(t_n^j) \mathcal{L}_j^{q+1} \left(\frac{t - t_{n-1}}{\tau_n} \right)\,.$$

We start with the argument by writing

$$\langle M_h \partial_t \widehat{\mathbf{v}}_h,\, d_T(\mathbf{v}_h - \widehat{\mathbf{v}}_h)\rangle_R = \sum_{i,j=0}^{q+1} a_{ij} b_{ij}$$

with $b_{ij} = \langle M_h \Pi_n \mathbf{v}_h(t_n^i),\, \Pi_n \mathbf{v}_h(t_n^j)\rangle_K$ and

$$\begin{aligned} a_{ij} &= \int_{t_{n-1}}^{t_n} d_T(t) \partial_t \mathcal{L}_i^{q+1}\left(\frac{t-t_{n-1}}{t_n - t_{n-1}}\right)\left(\mathcal{L}_j^q - \mathcal{L}_j^{q+1}\right)\left(\frac{t-t_{n-1}}{t_n - t_{n-1}}\right) \mathrm{d}t \\ &= \int_0^1 \tau_n d_T(\tau_n s + t_{n-1}) \partial_t \mathcal{L}_i^{q+1}(s)\left(\mathcal{L}_j^q - \mathcal{L}_j^{q+1}\right)(s)\,\mathrm{d}s\,. \end{aligned}$$

Here we set $\mathcal{L}_0^q(\cdot) = 0$. Considering for $0 \le s \le 1$

$$0 \le \tau_n d_T(\tau_n s + t_{n-1}) = \tau_n (T - t_{n-1} - \tau_n s) = \frac{\tau_n}{T - t_{n-1}}\left(1 - \frac{\tau_n}{T - t_{n-1}} s\right) \le 1$$

with

$$0 < \frac{\tau_n}{T - t_{n-1}} = \frac{t_n - t_{n-1}}{T - t_{n-1}} \le 1$$

we interpret the first factor as a scaling of the matrix and conclude, based of the continuity of the integral, that we have to investigate the two matrices

$$(a_{ij}^1) = \begin{pmatrix} \int_0^1 (1-t)(\partial_t \mathcal{L}_0^{q+1})(-\mathcal{L}_0^{q+1})\,\mathrm{d}t & \int_0^1 (1-t)(\partial_t \mathcal{L}_0^{q+1})(\mathcal{L}_1^q - \mathcal{L}_1^{q+1})\,\mathrm{d}t & \cdots \\ \int_0^1 (1-t)(\partial_t \mathcal{L}_1^{q+1})(-\mathcal{L}_0^{q+1})\,\mathrm{d}t & \int_0^1 (1-t)(\partial_t \mathcal{L}_1^{q+1})(\mathcal{L}_1^q - \mathcal{L}_1^{q+1})\,\mathrm{d}t & \\ \vdots & & \ddots \end{pmatrix}$$

and

$$(a_{ij}^2) = \begin{pmatrix} \int_0^1 (\partial_t \mathcal{L}_0^{q+1})(-\mathcal{L}_0^{q+1})\,\mathrm{d}t & \int_0^1 (\partial_t \mathcal{L}_0^{q+1})(\mathcal{L}_1^q - \mathcal{L}_1^{q+1})\,\mathrm{d}t & \cdots \\ \int_0^1 (\partial_t \mathcal{L}_1^{q+1})(-\mathcal{L}_0^{q+1})\,\mathrm{d}t & \int_0^1 (\partial_t \mathcal{L}_1^{q+1})(\mathcal{L}_1^q - \mathcal{L}_1^{q+1})\,\mathrm{d}t & \\ \vdots & & \ddots \end{pmatrix}$$

based on the identity

$$(a_{ij}) = \frac{\tau_n}{T - t_{n-1}}(a_{ij}^1) + \left(1 - \frac{\tau_n}{T - t_{n-1}}\right)(a_{ij}^2) =: A\,.$$

With the symmetric positive semi-definite matrix $B = (b_{ij})$ we get $A : B = A : B^\top = A^\top : B = \frac{1}{2}(A + A^\top) : B$. If the matrix A is positive semi-definite,

we can use singular value decomposition of A and B and get by the Frobenius inner product and its induced norm

$$\begin{aligned}\frac{1}{2}(A+A^\top) : B &= U_A^\top \Sigma_A U_A : U_B^\top \Sigma_B U_B = U_B U_A^\top \Sigma_A : \Sigma_B U_B U_A^\top \\ &= \sqrt{\Sigma_B} U_B U_A^\top \sqrt{\Sigma_A} : \sqrt{\Sigma_B} U_B U_A^\top \sqrt{\Sigma_A} = \|\sqrt{\Sigma_B} U_B U_A^\top \sqrt{\Sigma_A}\|_F^2 \geq 0\,.\end{aligned}$$

Hence, for the proof of $\langle M_h \partial_t \widehat{\mathbf{v}},\, d_T(\mathbf{v}-\widehat{\mathbf{v}})\rangle_Q = \sum_{R\in\mathcal{R}} \langle M_h \partial_t \widehat{\mathbf{v}},\, d_T(\mathbf{v}-\widehat{\mathbf{v}})\rangle_R \geq 0$ it is sufficient to prove that the matrices (a_{ij}^1) and (a_{ij}^2) are positive semi-definite.

For the case $q = 1$:
The Radau IIA quadrature rule has the integration points $c_0 = 1/3$ and $c_1 = 1$. Therefore we have

$$\begin{aligned}
\mathcal{L}_1^1(t) &= \frac{t-1}{1/3-1}\,, & \mathcal{L}_0^2(t) &= \frac{t-1/3}{0-1/3}\cdot\frac{t-1}{0-1}\,, & \partial_t \mathcal{L}_0^2(t) &= \quad 6t-4\,,\\
\mathcal{L}_2^1(t) &= \frac{t-1/3}{1-1/3}\,, & \mathcal{L}_1^2(t) &= \frac{t-0}{1/3-0}\cdot\frac{t-1}{1/3-1}\,, & \partial_t \mathcal{L}_1^2(t) &= -9t+4.5\,,\\
& & \mathcal{L}_2^2(t) &= \frac{t-0}{1-0}\cdot\frac{t-1/3}{1-1/3}\,, & \partial_t \mathcal{L}_2^2(t) &= \quad 3t-0.5\,.
\end{aligned}$$

We solve the integrals with Maple, a computer algebra system (CAS), resulting for $q = 1$ in the matrices

$$(a_{ij}^1) = \begin{pmatrix} \frac{13}{30} & -\frac{13}{20} & \frac{13}{60} \\ -\frac{21}{40} & \frac{63}{80} & -\frac{21}{80} \\ \frac{11}{120} & -\frac{11}{80} & \frac{11}{240} \end{pmatrix} \quad \text{and} \quad (a_{ij}^2) = \begin{pmatrix} \frac{1}{2} & -\frac{3}{4} & \frac{1}{4} \\ -\frac{3}{4} & \frac{9}{8} & -\frac{3}{8} \\ \frac{1}{4} & -\frac{3}{8} & \frac{1}{8} \end{pmatrix}.$$

The matrix $(a_{i,j}^1)$ has the eigenvalues $\lambda_0^1 = \lambda_1^1 = 0$ and $\lambda_2^1 = 19/15$ and the matrix $(a_{i,j}^2)$ the eigenvalues $\lambda_0^2 = \lambda_1^2 = 0$ and $\lambda_2^2 = 7/4$. Therefore, both are positive semi-definite.
The proof for polynomials with higher degree is done with the use on a computer algebra system and are available online at [Sub]. □

Lemma 3.8 (Galerkin orthogonality)**.** *Let* $\mathbf{u} \in V$ *be the exact solution of problem* (2.4) *and let* $\mathbf{u}_h \in V_h$ *be the discrete solution of problem* (3.18)*. Then the* Galerkin orthogonality

$$\mathcal{B}_h^c(\mathbf{u}-\mathbf{u}_h, \mathbf{w}_h) = 0 \tag{3.22}$$

holds for all $\mathbf{w}_h \in W_h$.

Proof. The Galerkin approximation (3.2) and the consistency of the discontinuous Galerkin method (3.3) together with the cellwise constant material parameters (cf. Rem. 3.1) yield that

$$\begin{aligned}\langle M_h\mathbf{u},\, \mathbf{w}_h\rangle_\Omega &= \langle M\mathbf{u},\, \mathbf{w}_h\rangle_\Omega \ ,\\ \langle D_h\mathbf{u},\, \mathbf{w}_h\rangle_\Omega &= \langle D\mathbf{u},\, \mathbf{w}_h\rangle_\Omega\end{aligned}$$

and

$$\langle A_h\mathbf{u},\, \mathbf{w}_h\rangle_\Omega = \langle A\mathbf{u},\, \mathbf{w}_h\rangle_\Omega \ .$$

Hence we conclude that

$$\begin{aligned}\mathcal{B}_h^c(\mathbf{u},\mathbf{w}_h) &= \mathcal{B}_h^d(\mathbf{u},\mathbf{w}_h)\\ &= \langle M_h\partial_t\mathbf{u} + A_h\mathbf{u} + D_h\mathbf{u},\, \mathbf{w}_h\rangle_Q + \sum_{n=1}^{N}\left\langle M_h\left[\!\left[\mathbf{u}\right]\!\right]_{n-1},\, \mathbf{w}_{h,n}(t_{n-1})\right\rangle_\Omega\\ &= \langle M_h\partial_t\mathbf{u} + A_h\mathbf{u} + D_h\mathbf{u},\, \mathbf{w}_h\rangle_Q\\ &= \langle M\partial_t\mathbf{u} + A\mathbf{u} + D\mathbf{u},\, \mathbf{w}_h\rangle_Q\\ &= \mathcal{B}(\mathbf{u},\mathbf{w}_h) = \langle \mathbf{b},\, \mathbf{w}_h\rangle_Q = \mathcal{B}_h^c(\mathbf{u}_h,\mathbf{w}_h)\,.\end{aligned}$$

□

Theorem 3.2. *Let* $\mathbf{u} \in V$ *be the solution of* (2.7) *and* $\mathbf{u}_h \in V_h$ *its approximation solving* (3.18)*. Then it holds that*

$$\|\mathbf{u}-\mathbf{u}_h\|_{V_h} \le (1+\beta^{-1}) \inf_{\mathbf{v}_h\in V_h} \|\mathbf{u}-\mathbf{v}_h\|_{V_h}\,.$$

If in addition the solution is sufficiently smooth, we obtain the a priori error estimate

$$\|\mathbf{u}-\mathbf{u}_h\|_{V_h} \le C\ \left(\triangle\mathbf{x}^p + \triangle t^q\right)\ \left(\|\partial_t^{q+1}\mathbf{u}\|_Q + \|\mathrm{D}^{p+1}\mathbf{u}\|_Q\right)$$

for $\triangle\mathbf{x} \ge \max\limits_{K\in\mathcal{K}} \operatorname{diam}(K)\,,\ \triangle t \ge \max\limits_{n\le N}\ t_n - t_{n-1}\,,\ p \le \min\limits_{R\in\mathcal{R}}\ p_R$ *and* $q \le \min\limits_{R\in\mathcal{R}}\ q_R\,.$

Proof. With Galerkin orthogonality (3.22) and that the bilinear form is bounded, we achieve that for all $\mathbf{v}_h \in V_h$ and $\mathbf{w}_h \in W_h$

$$\begin{aligned}\beta\|\mathbf{u}_h-\mathbf{v}_h\|_{V_h} &\le \sup_{\mathbf{w}_h\in W_h} \frac{\mathcal{B}_h^c(\mathbf{u}_h-\mathbf{v}_h,\mathbf{w}_h)}{\|\mathbf{w}_h\|_{W_h}}\\ &\le \sup_{\mathbf{w}_h\in W_h} \frac{\mathcal{B}_h^c(\mathbf{u}-\mathbf{v}_h,\mathbf{w}_h)}{\|\mathbf{w}_h\|_{W_h}} \le \|\mathbf{u}-\mathbf{v}_h\|_{V_h}\,.\end{aligned}$$

The last step is a triangle inequality, i.e.,

$$\|\mathbf{u} - \mathbf{u}_h\|_{V_h} \leq \|\mathbf{u} - \mathbf{v}_h\|_{V_h} + \|\mathbf{v}_h - \mathbf{u}_h\|_{V_h} .$$

Now we assume that the solution is regular, i.e.,

$$\mathbf{u} \in \mathrm{H}^{q+1}(0,T;\mathrm{L}_2(\Omega;\mathbb{R}^J)) \cap \mathrm{L}_2(0,T;\mathrm{H}^{p+1}(\Omega;\mathbb{R}^J)) .$$

The proof of the a priori estimate for the special case of a spatial mesh with triangles is provided in [Bra13, Sec. II.6].

□

Remark 3.4. Since the problems (3.9) and (3.18) are equivalent, the first version should be implemented. The computation of the conforming reconstruction is done only in a postprocessing step.

3.3.2 Full discretization: continuous Petrov–Galerkin in time and discontinuous Galerkin in space

This discretization uses again the same decomposition of the space-time cylinder Q into the tensor product space-time mesh $\mathcal{R}$. The discretization is discontinuous in space but uses continuous ansatz functions in time. In contrast to this, we use a test space which is discontinuous in space and time. Therefore we name this discretization discontinuous Galerkin – continuous Petrov Galerkin method (dG-cPG).

We choose local test spaces $W_{h,R} = \mathbb{P}_{p_K}(K;\mathbb{R}^J) \otimes \mathbb{P}_{q_R-1}(I_n;\mathbb{R}^J)$ and define the global test space

$$W_h = \left\{ \mathbf{w}_h \in \mathrm{L}_2(Q;\mathbb{R}^J) \colon \mathbf{w}_{h,R} = \mathbf{w}_h|_R \in W_{h,R} \right\}$$

which is discontinuous in space and time. The global ansatz space

$$\widehat{V}_h = \left\{ \mathbf{v}_h \in \mathrm{H}^1(0,T;\mathrm{L}_2(\Omega;\mathbb{R}^J)) \colon \mathbf{v}_{h,R} = \mathbf{v}_h|_R \in \widehat{V}_{h,R} \right\}$$

uses the local ansatz spaces

$$\begin{aligned}\widehat{V}_{h,R} = \Big\{ & \mathbf{v}_{h,R} \in \ \mathrm{L}_2(R;\mathbb{R}^J) \ : \\ & \mathbf{v}_{h,R}(\mathbf{x},t) = \frac{t_n - t}{t_n - t_{n-1}} \mathbf{v}_h(\mathbf{x},t_{n-1}) + \frac{t - t_{n-1}}{t_n - t_{n-1}} \mathbf{w}_{h,R}(\mathbf{x},t) , \\ & \mathbf{v}_h \in V_h|_{[0,t_{n-1}]} , \ \mathbf{w}_{h,R} \in W_{h,R} , \ (\mathbf{x},t) \in R = K \times I_n \Big\} .\end{aligned}$$

Hence $\widehat{V}_h$ is continuous in time and $\mathbf{v}_{h,R} \in \mathbb{P}_{p_K}(K;\mathbb{R}^J) \otimes \mathbb{P}_{q_R}(I_n;\mathbb{R}^J)$ for all $\mathbf{v}_{h,R} \in \widehat{V}_{h,R}$.

We define the discrete bilinear form $\widehat{\mathcal{B}}_h(\cdot,\cdot)$ on $\widehat{V}_h \times W_h$ by the discrete space-time operator L_h

$$\widehat{\mathcal{B}}_h(\mathbf{v}_h, \mathbf{w}_h) = \langle L_h \mathbf{v}_h,\, \mathbf{w}_h \rangle_Q = \langle M_h \partial_t \mathbf{v}_h + A_h \mathbf{v}_h + D_h \mathbf{v}_h,\, \mathbf{w}_h \rangle_Q \tag{3.23}$$

which is inf-sup stable with respect to the discrete norm

$$\|\mathbf{v}_h\|^2_{\widehat{V}_h} = \|\mathbf{v}_h\|^2_{W_h} + \|M_h^{-1} L_h \mathbf{v}_h\|^2_{W_h}\,.$$

By construction, the bilinear form $\widehat{\mathcal{B}}_h(\cdot,\cdot)$ is bounded in $\widehat{V}_h \times W_h$, i.e.,

$$\begin{aligned}\widehat{\mathcal{B}}_h(\mathbf{v}_h, \mathbf{w}_h) &= \langle L_h \mathbf{v}_h,\, \mathbf{w}_h \rangle_Q \\ &\leq \|M_h^{-1} L_h \mathbf{v}_h\|_{W_h} \|\mathbf{w}_h\|_{W_h} \leq \|\mathbf{v}_h\|_{\widehat{V}_h} \|\mathbf{w}_h\|_{W_h} \qquad \mathbf{v}_h \in \widehat{V}_h\,,\ \mathbf{w}_h \in W_h\,.\end{aligned}$$

Lemma 3.9 (Lem. 4.1 in [DFW16])**.** *With a tensor product space-time discretizations the bilinear form $\widehat{\mathcal{B}}_h(\cdot,\cdot)$ is inf-sup stable in $\widehat{V}_h \times W_h$ with $\beta = 1/\sqrt{1+4T^2}$, i.e.,*

$$\sup_{\mathbf{w}_h \in W_h} \frac{\widehat{\mathcal{B}}_h(\mathbf{v}_h, \mathbf{w}_h)}{\|\mathbf{w}_h\|_{W_h}} \geq \beta \|\mathbf{v}_h\|_{\widehat{V}_h}\,, \qquad \mathbf{v}_h \in \widehat{V}_h\,.$$

The proof can be found in [DFW16]. For a more detailed version, we refer to [Fin16, Thm. 4.1, Lem. 4.5, Lem. 4.6].

Remark 3.5. Thm. 2.1 asserts existence and uniqueness of a solution $\mathbf{u}_h \in \widehat{V}_h$ for a given $\mathbf{b} \in \mathrm{L}_2(Q;\mathbb{R}^J)$ to the variational problem.

Theorem 3.3 (Thm. 4.3 in [DFW16])**.** *Let $\mathbf{u} \in V$ be the solution of* (2.6) *and $\mathbf{u}_h \in \widehat{V}_h$ its approximation solving $\widehat{\mathcal{B}}_h(\mathbf{u}_h, \mathbf{w}_h) = \langle \mathbf{u}_h,\, \mathbf{w}_h \rangle_Q$, $\mathbf{w}_h \in W_h$. Then, we have*

$$\|\mathbf{u} - \mathbf{u}_h\|_{\widehat{V}_h} \leq (1 + \beta^{-1}) \inf_{\mathbf{v}_h \in \widehat{V}_h} \|\mathbf{u} - \mathbf{v}_h\|_{\widehat{V}_h}\,.$$

If in addition the solution is sufficiently smooth, we obtain the a priori error estimate

$$\|\mathbf{u} - \mathbf{u}_h\|_{\widehat{V}_h} \leq C\ \left(\triangle \mathbf{x}^p + \triangle t^q\right)\ \left(\|\partial_t^{q+1} \mathbf{u}\|_Q + \|\mathrm{D}^{p+1} \mathbf{u}\|_Q\right)$$

for $\triangle \mathbf{x} \geq \max\limits_{K \in \mathcal{K}} \operatorname{diam}(K)$, $\triangle t \geq \max\limits_{n \leq N}\ t_n - t_{n-1}$, $p \leq \min\limits_{R \in \mathcal{R}}\ p_K$ and $q \leq \min\limits_{R \in \mathcal{R}}\ q_R$.

Remark 3.6. The polynomial order in space is fixed for every $K \in \mathcal{K}$ and therefore $p_R = p_K$ for $n = 1, \ldots, N$. The extension to arbitrary p_R needs an additional projection in space.

3.3.3 Difference in time between discontinuous Galerkin and continuous Petrov–Galerkin

We illustrate the difference between the two methods by comparing the lowest order implementations in time: dG(p)-dG(0) and dG(p)-cPG(1) for a fixed polynomial order p in space.

Starting with dG-dG(0) we have ansatz functions which are constant in time

$$\mathbf{v}_h(t)|_{I_n} = \mathbf{v}_{h,n}$$

on an interval $I_n = (t_{n-1}, t_n]$ with the length $\tau_n = t_n - t_{n-1}$. The conforming reconstruction that is linear in time

$$\widehat{\mathcal{I}}\mathbf{v}_h(t)|_{I_n} = \frac{t_n - t}{t_n - t_{n-1}}\mathbf{v}_{h,n-1} + \frac{t - t_{n-1}}{t_n - t_{n-1}}\mathbf{v}_{h,n}\,.$$

Applying the space-time operator $\widehat{L}_h$ and integrating over the time interval I_n gives

$$\begin{aligned}
&\int_{I_n} \widehat{L}_h\mathbf{v}_h \,\mathrm{d}t \\
&\quad= \int_{I_n} M_h\partial_t\widehat{\mathcal{I}}\mathbf{v}_{h,n} + (A_h + D_h)\mathbf{v}_{h,n} \,\mathrm{d}t \\
&\quad= \int_{I_n} M_h\partial_t \left(\frac{t_n - t}{t_n - t_{n-1}}\mathbf{v}_{h,n-1} + \frac{t - t_{n-1}}{t_n - t_{n-1}}\mathbf{v}_{h,n}\right) + (A_h + D_h)\mathbf{v}_{h,n} \,\mathrm{d}t \\
&\quad= \int_{I_n} \mathbf{b}_{h,n} \,\mathrm{d}t \\
&\qquad\Longrightarrow M_h(\mathbf{v}_{h,n} - \mathbf{v}_{h,n-1}) + \tau_n(A_h + D_h)\mathbf{v}_{h,n} = \tau_n\mathbf{b}_{h,n}\,.
\end{aligned}$$

The lowest order in time for dG-cPG are linear ansatz functions in time, e.g.,

$$\mathbf{w}_h(t)|_{I_n} = \frac{t_n - t}{t_n - t_{n-1}}\mathbf{w}_{h,n-1} + \frac{t - t_{n-1}}{t_n - t_{n-1}}\mathbf{w}_{h,n}\,.$$

The dG-cPG method needs application of the space-time operator L_h

$$\begin{aligned}
&\int_{I_n} L_h \mathbf{w}_h \,\mathrm{d}t \\
&\quad = \int_{I_n} M_h \partial_t \left(\frac{t_n - t}{t_n - t_{n-1}} \mathbf{w}_{h,n-1} + \frac{t - t_{n-1}}{t_n - t_{n-1}} \mathbf{w}_{h,n} \right) \\
&\qquad + A_h \left(\frac{t_n - t}{t_n - t_{n-1}} \mathbf{w}_{h,n-1} + \frac{t - t_{n-1}}{t_n - t_{n-1}} \mathbf{w}_{h,n} \right) \\
&\qquad + D_h \left(\frac{t_n - t}{t_n - t_{n-1}} \mathbf{w}_{h,n-1} + \frac{t - t_{n-1}}{t_n - t_{n-1}} \mathbf{w}_{h,n} \right) \,\mathrm{d}t \\
&\quad = \int_{I_n} \mathbf{b}_{h,n} \,\mathrm{d}t
\end{aligned}$$

$$\implies M_h(\mathbf{w}_{h,n} - \mathbf{w}_{h,n-1}) + \frac{\tau_n}{2}(A_h + D_h)(\mathbf{w}_{h,n} + \mathbf{w}_{h,n-1}) = \tau_n \mathbf{b}_{h,n} \,.$$

The dG-dG(0) method corresponds to the backward Euler method in time in contrast to the dG-cPG method, which is equivalent to the implicit midpoint rule as time integrator. Focusing the evaluation of the right hand side in the variational formulation, both methods test the continuous function $\mathbf{b}$ with test functions constant in time. This implies that $\mathbf{b}_n$ is the mean value in time of $\mathbf{b}$ in both methods.

In the tensor-product case with fixed polynomial degrees $p_R = p$ in time and $p_R = p_K$ in space only depending on $K \subset \Omega$, the discontinuous Galerkin in space continuous Petrov–Galerkin in time method is equivalent to the Gauss collocation method, where as the discontinuous Galerkin in space and time method is equivalent to the Radau IIA collocation method, see [Huy09].

CHAPTER

FOUR

ADAPTIVE FINITE ELEMENT TECHNIQUES

The introduced space-time discretizations are an extension of classical finite element methods by interpreting the time as an additional variable. This results in very large systems which must be solved. We engage the problem of reducing the computational effort with adaptive techniques. The standard adaptive finite element method iterates the steps

$$\texttt{SOLVE} \rightarrow \texttt{ESTIMATE} \rightarrow \texttt{MARK} \rightarrow \texttt{REFINE} \tag{4.1}$$

giving a sequence of discrete solutions converging to the exact one. We will introduce the steps in reverse order.

4.1 General principle of adaptivity

h-adaptivity

The h-adaptivity is the most widely used adaptive method. The mesh is locally modified whereas the polynomial degree is kept fixed. The mesh modification can either be refinement of the cells or coarsening.

The typically used techniques are red refinement (allowing hanging nodes), red-green refinement (red refinement with conform closure of hanging nodes) or in case of simplices the use of bisection. Which grid refinement is used depends on constrain to the quality of the mesh (degeneration of elements and allowing hanging nodes or the need of nested meshes) and construction effort.

Adaptive h-refinement is superior to uniform h-refinement except for "nice" problems with smooth solutions [Mit89, Sec. 5]. The rate of convergence (in energy) with respect to the number of degrees of freedom for smooth solution is limited by a fixed polynomial degree [BSK81, Sec. 6.4].

p-adaptivity

This method is introduced in [BSK81]: "In the p-version of the finite element method, the triangulation is fixed and the degree p, of the piecewise polynomial approximation, is progressively increased until some desired level of precision is reached."
For analytic solutions, the rate of convergence with respect to the number of degrees of freedom is not limited by a fixed polynomial degree. For nonsmooth solutions, the p-refinement has at least the same rate of convergence as the h-version [BSK81, Sec. 6.4].

hp-adaptivity

The h-adaptivity and p-adaptivity can be combined to the hp-adaptivity. The idea is to use h-refinement, where the solution is estimated to be rough. This could be for example near discontinuities. If the solution is estimated to be analytic, the polynomial degree is increased. The combination of both methods allows to achieve exponential convergence rate with respect to the number of degrees of freedom [MM14].

Remark 4.1. To avoid the disadvantage of hanging nodes and deformation (cf. Fig. 4.1) as well as the fact, that h-refinement produces more degrees of freedom than p-refinement (cf. Fig. 4.2), we aim to the second method and develop an adaptive strategy for the selection of the local polynomial degrees in space and time (p_R, q_R) to reduce the total degrees of freedom without loosing accuracy.

Increasing the polynomial degree yields the problem of using the correct quadrature rules with a main focus on the faces of the cells. Therefore we implemented an adaptive quadrature rule selection depending on the maximum polynomial degree in space and time.

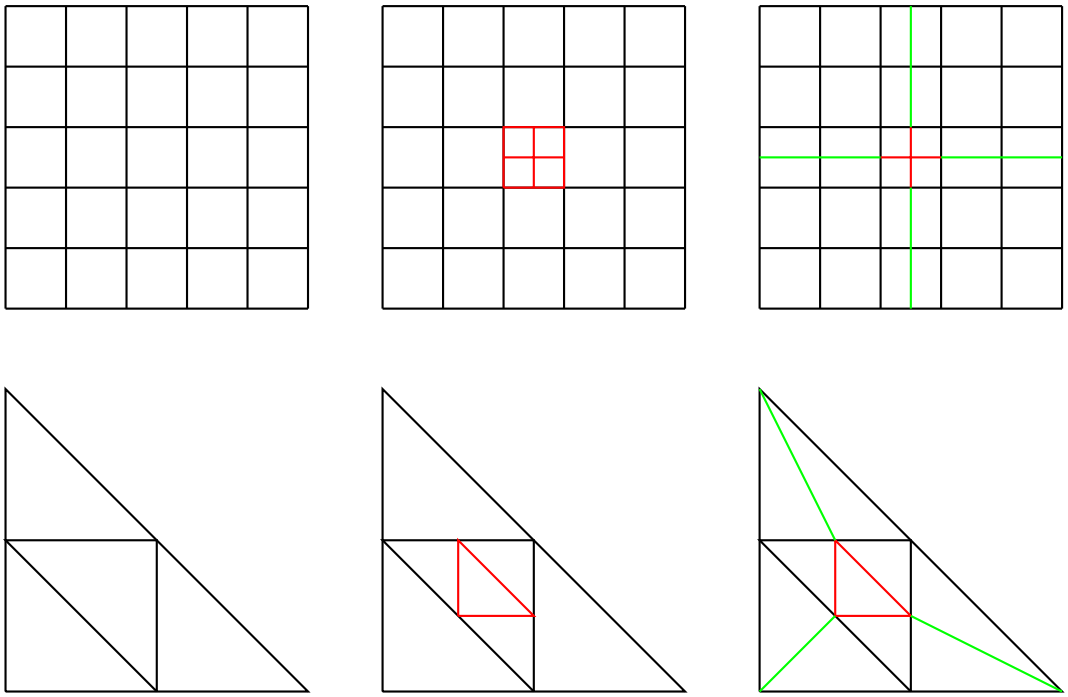

Figure 4.1: Illustration of h-adaptive mesh refinement in 2D with squares (top) and triangles (bottom). Starting with a uniform mesh (left) the central cell is marked for refinement. Using red refinement results in a mesh with hanging nodes (middle) or red-green refinement (right) bisecting cells with hanging nodes.

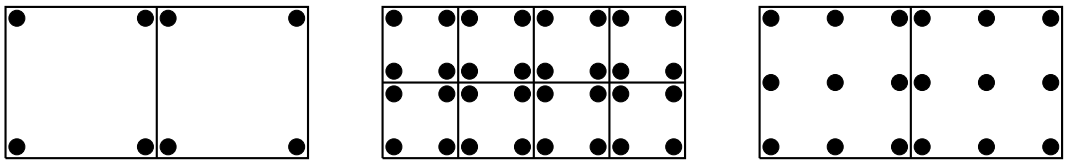

Figure 4.2: Difference between h-adaptivity and p-adaptivity in 2D: starting on the left with linear dG elements the mesh in the middle results from uniform h-refinement and on the right by p-refinement. Every dot represents one degree of freedom.

Remark 4.2. Common practice are implementations using adaptive time step size control. Since we deal with space-time discretizations, they can be interpreted as a special case of 'h-adaptivity' in space-time.

4.2 Marking strategies

Marking strategies base the decision of marking an element for refinement or coarsening on given error estimators or indicator for each element η_R. Three different marking strategies are implemented. These are in particular:

Maximum marking strategy

The maximum strategy marks a set $\mathcal{M} \subset \mathcal{R}$ depending on the maximum estimator $\eta_{\max} = \max_{R\in\mathcal{R}} \eta_R$ and a fixed given value $\theta \in (0,1)$ such that all elements are marked for refinement, if the estimator is greater than the critical value $\eta_{\text{crit}} = \theta\eta_{\max}$:

$$\forall R \in \mathcal{M} : \ \eta_R > \eta_{\text{crit}} \qquad \text{and} \qquad \forall R \in \mathcal{R}\backslash\mathcal{M} : \ \eta_R \leq \eta_{\text{crit}} \, .$$

Elements are marked for coarsening, if the estimator is sufficiently small. This means for a fixed given value $\widetilde{\theta} \in (0,1)$ we have $\eta_R < \widetilde{\theta}\eta_{\text{crit}}$.

Equidistribution strategy

This strategy uses the same algorithm as the maximum marking strategy but with the difference in computing the critical value. This is done here by computing the mean value of all estimates, i.e.,

$$\eta_{\text{crit}} = \frac{1}{|\mathcal{R}|} \sum_{R\in\mathcal{R}} \eta_R \, .$$

Dörfler marking

The idea of this marking strategy is presented in [Dör96]. A set $\mathcal{M}$ is marked, such that a certain ratio $\theta \in (0,1)$ of the total estimation is marked, i.e.,

$$\sum_{R\in\mathcal{M}} \eta_R \geq \theta \sum_{R\in\mathcal{R}} \eta_R \, .$$

Additionally it is demanded that $\mathcal{M}$ has the lowest possible cardinality.

4.3 Error indicators and error estimators

There are various types of error indicators or estimators. The simplest is a gradient-based indicator, which is used if more rigorous a posteriori error estimators are too difficult or even impossible. The indicator computes the L_2-norm of the gradient of the discrete solution on each element. This indicator is the simplest to implement but gives poor benefits for the numerical analysis of convergence properties.

Hierarchical error estimators rate the local error by the difference to a second discrete solution. This one is computed on a finer mesh or with higher order polynomial degree. Hence, the computation of the estimators has a larger computational cost than solving the partial differential equation itself.

An alternative are residual estimators. The local error estimators are computed with a suitable norm of its residual with respect to the strong form of the differential equation. Residual estimators are efficient and have been proven to lead to an error reduction on the whole computational domain for a number of problem classes.

Since we are only interested in a small part of the solution we use a goal oriented method. This kind of methods are based on duality techniques introduced in [BR96].

4.3.1 Duality based goal-oriented error estimation

We follow the framework in [Fin16, DFW16, DFWZ19] and are interested in a linear goal functional $E \in W'$ with compact support within a certain region of interest (RoI), define the adjoint problem and solve the dual problem. Then, the error is estimated in terms of the local residual and the dual weight leading to the dual weighted residual estimators (DWR).

Definition 4.1 (Adjoint operator)**.** Let $L\colon V \to V^* \subset W$ be the bounded, linear space-time operator defined by (2.4). The adjoint space of V can be

defined by

$$V^* = \{\mathbf{w} \in W : \text{there exists a unique } \mathbf{z} \in W \text{ such that } \langle L\mathbf{v}, \mathbf{w}\rangle_Q = \langle \mathbf{v}, \mathbf{z}\rangle_Q \quad \text{for all } \mathbf{v} \in V \}\,.$$

The operator $L^*\colon V^* \to V$, which satisfies

$$\langle L\mathbf{v}, \mathbf{w}\rangle_Q = \langle \mathbf{v}, L^*\mathbf{w}\rangle_Q \qquad \text{for all } \mathbf{v} \in V \text{ and } \mathbf{w} \in V^*$$

is called adjoint operator of L.

Note that the space V has incorporated homogeneous initial conditions. They are transferred to homogeneous final conditions in V^*, i.e., $\mathbf{v}^*(T) = \mathbf{0}$ for all $\mathbf{v}^* \in V^*$.
For hyperbolic evolution equations with space-time operator $L = M\partial_t + A$, the adjoint operator L^* is given by

$$L^* = -M\partial_t + A^*\,,$$

where A^* is the adjoint spatial operator of A. Moreover, it yields that

$$A^* = -A \qquad \text{on } \mathcal{D}(A) \cap \mathcal{D}(A^*)$$

and therefore $L^* = -L$, cf. [Fin16, Lem. 5.1].

If we consider attenuation effects, we have to handle the space-time operator $L = M\partial_t + A + D$. The operator D is symmetric and therefore

$$L^* = -M\partial_t - A + D \neq -L\,.$$

Let $\mathbf{u} \in V$ be the *primal solution* of a given problem $L\mathbf{u} = \mathbf{b}$ and $\mathbf{u}_h$ the discrete approximation obtained either by the dG-dG or dG-cPG method.

Definition 4.2 (Dual problem)**.** Let $\mathbf{u} \in V$ be the *primal solution* of a given problem $L\mathbf{u} = \mathbf{b}$. The dual problem is defined as: find the *dual solution* $\mathbf{u}^* \in V^*$ of

$$\langle L^*\mathbf{u}^*, \mathbf{w}\rangle_Q = \langle E, \mathbf{w}\rangle_Q \qquad \text{for all } \mathbf{w} \in W$$

for a given (linear) functional E (represented in L_2).

Remark 4.3. We assume that the dual solution $\mathbf{u}^*$ is sufficiently smooth for the following arguments. In particular we assume that $\mathbf{u}^*(\cdot,t)|_f \in \mathrm{L}_2(f;\mathbb{R}^J)$ for all faces.

Inserting the consistency of the numerical flux yields for all $\mathbf{w}_h \in W_h \cap V^*$

$$\begin{aligned}
\langle E,\, \mathbf{u}-\mathbf{u}_h\rangle_Q &= \langle \mathbf{u}-\mathbf{u}_h,\, L^*\mathbf{u}^*\rangle_Q = \langle \mathbf{u},\, L^*\mathbf{u}^*\rangle_Q - \langle \mathbf{u}_h,\, L^*\mathbf{u}^*\rangle_Q \\
&= \langle L\mathbf{u},\, \mathbf{u}^*\rangle_Q - \langle \mathbf{u},\, \mathbf{n}\cdot\mathbf{F}(\mathbf{u}^*)\rangle_{\partial Q} \\
&\qquad - \sum_{\substack{R=I_n\times K\in\mathcal{R}\\ I_n=(t_{n-1},t_n)}} \Big(\langle L\mathbf{u}_h,\, \mathbf{u}^*\rangle_R - \langle \mathbf{u}_h,\, \mathbf{n}_R\cdot\mathbf{F}(\mathbf{u}^*)\rangle_{\partial R} \\
&\qquad\qquad + \Big\langle M\,[\![\mathbf{u}_h]\!]_{n-1}\,,\, \mathbf{u}^*\Big\rangle_K \Big) \\
&= \langle \mathbf{f},\, \mathbf{u}^*\rangle_Q - \sum_{\substack{R=I_n\times K\in\mathcal{R}\\ I_n=(t_{n-1},t_n)}} \Big(\langle L\mathbf{u}_{h,R},\, \mathbf{u}^*\rangle_R - \langle \mathbf{u}_h,\, \mathbf{n}_K\cdot\mathbf{F}(\mathbf{u}^*)\rangle_{I_n\times\partial K} \\
&\qquad\qquad + \Big\langle M\,[\![\mathbf{u}_h]\!]_{n-1}\,,\, \mathbf{u}^*\Big\rangle_K \Big) \\
&= \sum_{\substack{R=I_n\times K\in\mathcal{R}\\ I_n=(t_{n-1},t_n)}} \Big(\langle \mathbf{f}-L\mathbf{u}_{h,R},\, \mathbf{u}^*\rangle_R + \langle \mathbf{n}_K\cdot\mathbf{F}(\mathbf{u}_{h,R}),\, \mathbf{u}^*\rangle_{I\times\partial K} \\
&\qquad\qquad + \Big\langle M\,[\![\mathbf{u}_h]\!]_{n-1}\,,\, \mathbf{u}^*\Big\rangle_K \Big) \\
&= \sum_{\substack{R=I_n\times K\in\mathcal{R}\\ I_n=(t_{n-1},t_n)}} \Big(\langle \mathbf{f}-L\mathbf{u}_{h,R},\, \mathbf{u}^*-\mathbf{w}_h\rangle_R + \langle \mathbf{n}_K\cdot\mathbf{F}(\mathbf{u}_{h,R}),\, \mathbf{u}^*-\mathbf{w}_h\rangle_{I\times\partial K} \\
&\qquad\qquad + \Big\langle M\,[\![\mathbf{u}_h]\!]_{n-1}\,,\, \mathbf{u}^*-\mathbf{w}_h\Big\rangle_K \Big)\,.
\end{aligned}$$

We insert as special choice the discrete solution $\mathbf{w}_h = \mathbf{u}_h^*$ of the dual problem and we obtain the estimate

$$\begin{aligned}
|\,\langle E,\, \mathbf{u}-\mathbf{u}_h\rangle_Q\,| \le \sum_{\substack{R=I_n\times K\in\mathcal{R}\\ I_n=(t_{n-1},t_n)}} \Big(&\|\mathbf{f}-L\mathbf{u}_{h,R}\|_R \|\mathbf{u}^*-\mathbf{u}_h^*\|_R \\
&+ \|\mathbf{n}_K\cdot\mathbf{F}(\mathbf{u}_{h,R})\|_{I\times\partial K} \|\mathbf{u}^*-\mathbf{u}_h^*\|_{I\times\partial K} \\
&+ \|M\,[\![\mathbf{u}_h]\!]_{n-1}\,\|_{\{t_{n-1}\}\times K} \|\mathbf{u}^*-\mathbf{u}_h^*\|_{\{t_{n-1}\}\times K} \Big)\,.
\end{aligned} \tag{4.2}$$

4.3.2 Computational error indicators for DWR

The identity (4.2) cannot be evaluated numerically since the function $\mathbf{u}^*$ is unknown. Let $\mathbf{u}_h^* \in W_h$ be a numerical approximation of the dual solution of

$$\mathcal{B}_h(\mathbf{v}_h, \mathbf{u}_h^*) = \langle E,\, \mathbf{v}_h\rangle_Q\,, \qquad \mathbf{v}_h \in V_h$$

to a bilinear form $\mathcal{B}_h\colon V_h \times W_h \to \mathbb{R}$ or $\mathcal{B}_h\colon \widehat{V}_h \times W_h \to \mathbb{R}$.
Since the quantities of $\|\mathbf{u}^*-\mathbf{u}_h^*\|_R$ and $\|\mathbf{u}^*-\mathbf{u}_h^*\|_{I\times\partial K}$ cannot be evaluated, we need local error indicators η_R which approximate them by using the discrete

solution and combine it with a projection or interpolation operator $\mathcal{I}\colon W \to W_h$. This leads to the local indicators

$$\begin{aligned}\eta_R =& \|\mathbf{f} - L\mathbf{u}_{h,R}\|_R \|\mathbf{u}_h^* - \mathcal{I}\mathbf{u}_h^*\|_R \\ &+ \|\mathbf{n}_K \cdot \mathbf{F}(\mathbf{u}_{h,R})\|_{I_n \times \partial K} \|\mathbf{u}_h^* - \mathcal{I}\mathbf{u}_h^*\|_{I_n \times \partial K} \\ &+ \|M \left[\!\left[\mathbf{u}_h\right]\!\right]_{n-1}\|_{\{t_{n-1}\}\times\partial K} \|\mathbf{u}_h^* - \mathcal{I}\mathbf{u}_h^*\|_{\{t_{n-1}\}\times\partial K}\,.\end{aligned}$$

Remark 4.4. Since the dG-cPG method is continuous in time, there are no jumps in time, i.e., $M\left[\!\left[\mathbf{u}_h\right]\!\right]_{n-1} = \mathbf{0}$ for all $n = 0, \ldots, N$. This reduces the local error estimators to

$$\eta_R = \|\mathbf{f} - L\mathbf{u}_{h,R}\|_R \|\mathbf{u}_h^* - \mathcal{I}\mathbf{u}_h^*\|_R + \|\mathbf{n}_K \cdot \mathbf{F}(\mathbf{u}_{h,R})\|_{I_n \times \partial K} \|\mathbf{u}_h^* - \mathcal{I}\mathbf{u}_h^*\|_{I_n \times \partial K}$$

Remark 4.5. Let the adjoint problem be defined by $L^* = -L$. Then the adjoint discrete solution can be obtained by using the negative transposed system matrix of the primal discrete variational problem.

Higher-order approximation

One possibility to estimate the interpolation error $\mathbf{u}^* - \mathcal{I}\mathbf{u}^*$ is to compute the dual problem on a refined mesh or with higher order polynomials in space and time. The operator $\mathcal{I}$ would then project or interpolate the dual solution back to the space W_h.
The downside of this method is, that the computation of the dual solution becomes very costly. Since higher order polynomials are used, the computational effort can exceed the effort for the primal problem.

Higher-order interpolation on patches

The dual solution is computed on the same discretization as $\mathbf{u}_h$ and has therefore the same computational effort. Furthermore the system matrix of the primal problem can be reused.
The idea of this method is to coarse the mesh $\mathcal{R}_h$ in space and time giving the mesh $\mathcal{R}_H$. On the coarse mesh we use a discretization with higher order polynomials. A patch is defined as all cells of the fine mesh which are a subdomain of one cell of the coarse mesh. The interpolation operator can then be defined locally on every patch. Further details and numerical experiments are presented in [Fin16].

Mean value error indicators

For the purpose of estimating the interpolation error $\mathbf{u}^* - \mathcal{I}\mathbf{u}^*$, the face jumps are meaningful even for the case of piecewise constant approximations. This motivates the local error indicators using the spatial L_2-projection Q_h in K for the dG-cPG method:

$$\eta_R = \|\mathbf{f} - L\mathbf{u}_h\|_R h_K^{1/2} \left\| [Q_h \mathbf{u}_h^*]_K \cdot \mathbf{n}_K \right\|_{I\times\partial K} + \frac{1}{2} \sum_{f\in\mathcal{F}_K} \left(\left\| [\mathbf{u}_h]_{K,f} \mathbf{n}_K \right\|_{I\times f} \left\| [Q_h \mathbf{u}_h^*]_{K,f} \cdot \mathbf{n}_K \right\|_{I\times f} \right).$$

Q_h denotes the piecewise L_2-projection in space to $\mathbb{P}_0(K;\mathbb{R}^J)$. The terms of the error indicators contain the given data function $\mathbf{f}$ and M and are computed by a quadrature formula. Alternatively, a term $\|\mathbf{f}-\mathbf{f}_h-(M-M_h)\partial_t \mathbf{u}_h\|_R$ could be separated for the control of this data error, but usually, this error contribution is of minor importance. This is especially the case in our numerical examples. Numerical experiments with this indicator are shown in Sec. 5.2.4. Main advantage of this method is, that only one discretization and computational mesh is needed.

Remark 4.6. We decided to use the mean value error indicators due on their numerical efficiency. The higher-order approximation is excluded because of the high computational cost. The method of higher-order interpolation on patches restricts the number of adaptive steps. The number of adaptive p-refinement steps must be less than the highest order of implemented shape functions. The mean value error indicators overcome this constraint.

4.4 Solving the space-time system

In our numerical examples we use the p-adaptive strategy described in Algorithm 1 including computation of the mean value error indicators η_R combined with a maximum marking strategy following the iteration steps (4.1). The marking depends on the parameters θ, $\widetilde{\theta} \in (0,1)$, for the adaptive selection criterion for increasing or decreasing the polynomial degree in space and time.

Remark 4.7. It is highly suggested to write dump files of the polynomial orders in space and time. We implemented this at the beginning of every loop

Algorithm 1 Adaptive algorithm.

1: choose low order polynomial degrees on the mesh, i.e., $(p, q) = (1, 1)$
2: **while** $\max_R(p_R) < p_{\max}$ and $\max_R(q_R) < q_{\max}$ **do**
3: compute $\mathbf{u}_h$
4: compute $\mathbf{u}_h^*$ and the projection $Q_h\mathbf{u}_h^*$
5: compute η_R on every cell R
6: if the error is small enough STOP
7: mark space-time cells R based on maximum strategy
8: increase or decrease polynomial degrees on marked cells by one in space and time
9: redistribute cells on processes for better load balancing
10: **end while**

before allocating the memory for the system matrix. The memory consumption of space time methods can be very high. In our numerical experiments, this could go up to several terabytes. These dump files give the possibility to restart the algorithm on hardware with more memory without recomputing the previous steps.

First we will give an idea of the structure of the system matrix. Since we use the space-time multilevel preconditioner introduced in [Fin16, Chap. 6] to solve the primal and the dual problem, it will be presented in the next section. Afterwards we explain the used load balancer.

4.4.1 Structure of the system matrix

Now we consider the structure of the linear system in the special case of a tensor product space-time mesh. Using the time slices $\mathcal{R}^n = \left\{ I_n \times K \colon K \in \mathcal{K} \right\}$ gives the total space-time mesh $\mathcal{R} = \bigcup_{n=1}^N \mathcal{R}^n$. On this mesh we use variable polynomial degrees p_R, q_R in every space-time cell R. Let $\{\boldsymbol{\psi}_{R,j}^n\}_{j=1,\dots,\dim W_{h,R}}$ be a basis of $W_{h,R}$ and define $W_h^n = \operatorname{span}\left\{ \bigcup_{R\in\mathcal{R}^n} \bigcup_{j=1}^{\dim W_{h,R}} \boldsymbol{\psi}_{R,j}^n \right\}$. The solution $\mathbf{u}_h \in V_h$ is represented by finite element functions $\mathbf{u}_h^n \in W_h^n$, $n = 1, \dots, N$. Together with $\mathbf{u}_h^0 = \mathbf{0}$ we obtain for the dG-cPG discretization

$$\mathbf{u}_h(t, \mathbf{x}) = \frac{t_n - t}{t_n - t_{n-1}} \mathbf{u}_h^{n-1}(t_{n-1}, \mathbf{x}) + \frac{t - t_{n-1}}{t_n - t_{n-1}} \mathbf{u}_h^n(t, \mathbf{x}) \,, \qquad (t, \mathbf{x}) \in I_n \times K \,.$$

By $\underline{u} = (\underline{u}^1, \dots, \underline{u}^N)^\top$ we denote the corresponding coefficient vector of the solution, where $\underline{u}^n \in \mathbb{R}^{\dim W_h^n}$ is the coefficient vector of

$$\mathbf{u}_h^n = \sum_{R \in \mathcal{R}^n} \sum_{j=1}^{\dim W_{h,R}} \underline{u}_{R,j}^n \boldsymbol{\psi}_{R,j}^n \,.$$

With respect to this basis, the discrete space-time system for dG-dG (3.18) and dG-cPG (3.23) have the matrix representation $\underline{L}\,\underline{u} = \underline{b}$ with the block matrix

$$\underline{L} = \begin{pmatrix} \underline{D}^1 & & & \\ \underline{C}^1 & \underline{D}^2 & & \\ & \ddots & \ddots & \\ & & \underline{C}^{N-1} & \underline{D}^N \end{pmatrix} .$$

The matrix entries for dG-dG are

$$\underline{D}_{R',k,R,j}^n = \int_{t_{n-1}}^{t_n} \int_\Omega \widehat{L}_h \Big(\boldsymbol{\psi}_{R,j}^n(t,\mathbf{x}) \Big) \boldsymbol{\psi}_{R',k}^n(t,\mathbf{x}) \,\mathrm{d}\mathbf{x}\,\mathrm{d}t \,, \qquad R, R' \in \mathcal{R}^n$$

$$\underline{C}_{R',k,R,j}^n = \int_\Omega M_h \Big(\boldsymbol{\psi}_{R',k}^n(t_{n-1},\mathbf{x}) - \boldsymbol{\psi}_{R,j}^{n-1}(t_{n-1},\mathbf{x}) \Big) \boldsymbol{\psi}_{R',k}^n(t,\mathbf{x}) \,\mathrm{d}\mathbf{x} \,,$$
$$R \in \mathcal{R}^{n-1} \,,\ R' \in \mathcal{R}^n$$

and for the dG-cPG method

$$\underline{D}_{R',k,R,j}^n = \int_{t_{n-1}}^{t_n} \int_\Omega L_h \Big(\frac{t - t_{n-1}}{t_n - t_{n-1}} \boldsymbol{\psi}_{R,j}^n(t,\mathbf{x}) \Big) \boldsymbol{\psi}_{R',k}^n(t,\mathbf{x}) \,\mathrm{d}\mathbf{x}\,\mathrm{d}t \,, \qquad R, R' \in \mathcal{R}^n$$

$$\underline{C}_{R',k,R,j}^n = \int_{t_{n-1}}^{t_n} \int_\Omega L_h \Big(\frac{t_n - t}{t_n - t_{n-1}} \boldsymbol{\psi}_{R,j}^{n-1}(t_{n-1},\mathbf{x}) \Big) \boldsymbol{\psi}_{R',k}^n(t,\mathbf{x}) \,\mathrm{d}\mathbf{x}\,\mathrm{d}t \,,$$
$$R \in \mathcal{R}^{n-1} \,,\ R' \in \mathcal{R}^n \,.$$

The right-hand side is in both cases $\underline{b} = (\underline{b}^1, \dots, \underline{b}^N)^\top$ with $\underline{b}_{j,R}^n = (\mathbf{b}, \boldsymbol{\psi}_{R,j}^n)_{0,R}$.

Remark 4.8. The matrix entries $\underline{C}^n$ in the case of a dG-dG discretization reduce to an integral only in space. Since only base functions of space-time cells connected by a face in time have a common support, there is no coupling with neighboring cells in space. Therefore it results in a matrix, which is more sparse and gives a speedup in comparison to the dG-cPG discretization.

Sequentially, this system can be solved by a block-Gauss–Seidel method (corresponding to implicit time integration)

$$\underline{D}^1 \underline{u}^1 = \underline{b}^1 \,, \quad \underline{D}^2 \underline{u}^2 = \underline{b}^2 - \underline{C}^1 \underline{u}^1 \,, \quad \dots, \quad \underline{D}^N \underline{u}^N = \underline{b}^N - \underline{C}^{N-1} \underline{u}^{N-1} \,,$$

provided that $\underline{D}^n$ can be inverted efficiently.

4.4.2 Space-time multilevel preconditioner

For space-time multilevel preconditioners we consider hierarchies in space and time. We define by $\mathcal{R}_{0,0}$ the coarse space-time mesh. By $l = 1, \ldots, l_{\max}$ uniform refinements in space and $k = 1, \ldots, k_{\max}$ refinements in time we obtain the space-time mesh $\mathcal{R}_{l,k}$. Let $V_{l,k}$ be the approximation spaces on $\mathcal{R}_{l,k}$ with arbitrary polynomial degrees p_R and q_R Let $\underline{L}_{l,k}$ be the corresponding matrix representation of the discrete operator.
The multilevel preconditioner combines smoothing operations on different levels and requires a transfer between the levels. Since the spaces are nested, we can define prolongation matrices $\underline{P}^{l,k}_{l-1,k}$ and $\underline{P}^{l,k}_{l,k-1}$ representing the natural injections $V_{l-1,k} \subset V_{l,k}$ in space and $V_{l,k-1} \subset V_{l,k}$ in time. Correspondingly, the restriction matrices $\underline{R}^{l,k}_{l-1,k}$ and $\underline{R}^{l,k}_{l,k-1}$ represent the L_2-projections or interpolations in space and in time of the test spaces $W_{l,k} \supset W_{l-1,k}$ and $W_{l,k} \supset W_{l,k-1}$.

Remark 4.9. In contrast to the dG-cPG discretization, the restriction and prolongation can be done locally on the single patches for the dG-dG discretization. The transfer can be done matrix-free. Such a version is implemented. The transfer matrix on the finest level would need nearly as much memory as the system matrix. This is avoided by using a matrix-free version using nodal interpolation for prolongation and restriction. If during the restriction a node lies on a face of the child cells, we build the mean value there.

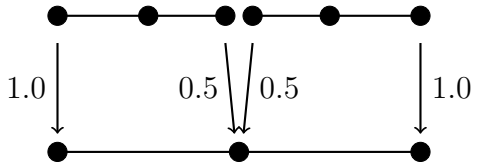

Figure 4.3: Restriction as interpolation operation: sketch in one dimension for a polynomial with degree $p = 2$.

Remark 4.10. The restriction and prolongation for dG-dG can be done locally on every patch. This is an important advantage concerning the computational effort.

For the smoothing operations on level (l, k) we consider the block-Jacobi preconditioner $B^{\mathrm{SM}}_{l,k} = B^{\mathrm{J}}_{l,k}$ or the block-Gauss–Seidel preconditioner $B^{\mathrm{SM}}_{l,k} = B^{\mathrm{GS}}_{l,k}$

(where all components corresponding to a space-time cell R build a block)

$$\underline{B}_{l,k}^{\mathrm{J}} = \theta_{l,k}\,\mathrm{block\_diag}(\underline{L}_{l,k})^{-1}\,,$$
$$\underline{B}_{l,k}^{\mathrm{GS}} = \theta_{l,k}\left(\mathrm{block\_lower}(\underline{L}_{l,k}) + \mathrm{block\_diag}(\underline{L}_{l,k})\right)^{-1}$$

with damping parameter $\theta_{l,k} \in (0,1]$. The iteration matrices are given by $\underline{S}_{l,k}^{\mathrm{J}} = \underline{\mathrm{Id}}_{l,k} - \underline{B}_{l,k}^{\mathrm{J}}\underline{L}_{l,k}$ for the block-Jacobi preconditioner and $\underline{S}_{l,k}^{\mathrm{GS}} = \underline{\mathrm{Id}}_{l,k} - \underline{B}_{l,k}^{\mathrm{GS}}\underline{L}_{l,k}$ for the block-Gauss–Seidel preconditioner. The number of pre- and post-smoothing steps are denoted by $\nu_{l,k}^{\mathrm{pre}}$ and $\nu_{l,k}^{\mathrm{post}}$.

The multilevel preconditioner $\underline{B}_{l,k}^{\mathrm{ML}}$ is defined recursively. On the coarse level, we use a parallel direct linear solver $\underline{B}_{0,0}^{\mathrm{ML}} = \left(\underline{L}_{0,0}\right)^{-1}$, see [MW11, MW16], or GMRES with a block-Gauss–Seidel preconditioner. Then, we have two options: restricting in space defines $\underline{B}_{l,k}^{\mathrm{ML}}$ by

$$\begin{aligned}&\underline{\mathrm{Id}}_{l,k} - \underline{B}_{l,k}^{\mathrm{ML}}\underline{L}_{l,k}\\ &\quad= \left(\underline{\mathrm{Id}}_{l,k} - \underline{B}_{l,k}^{\mathrm{GS}}\underline{L}_{l,k}\right)^{\nu_{l,k}^{\mathrm{pre}}}\left(\underline{\mathrm{Id}}_{l,k} - \underline{P}_{l-1,k}^{l,k}\underline{B}_{l-1,k}^{\mathrm{ML}}\underline{R}_{l-1,k}^{l,k}\underline{L}_{l,k}\right)\left(\underline{\mathrm{Id}}_{l,k} - \underline{B}_{l,k}^{\mathrm{GS}}\underline{L}_{l,k}\right)^{\nu_{l,k}^{\mathrm{post}}}\end{aligned}$$

with Gauss–Seidel smoothing and restricting in time yields

$$\begin{aligned}&\underline{\mathrm{Id}}_{l,k} - \underline{B}_{l,k}^{\mathrm{ML}}\underline{L}_{l,k}\\ &\quad= \left(\underline{\mathrm{Id}}_{l,k} - \underline{B}_{l,k}^{\mathrm{SM}}\underline{L}_{l,k}\right)^{\nu_{l,k}^{\mathrm{pre}}}\left(\underline{\mathrm{Id}}_{l,k} - \underline{P}_{l,k-1}^{l,k}\underline{B}_{l,k-1}^{\mathrm{ML}}\underline{R}_{l,k-1}^{l,k}\underline{L}_{l,k}\right)\left(\underline{\mathrm{Id}}_{l,k} - \underline{B}_{l,k}^{\mathrm{SM}}\underline{L}_{l,k}\right)^{\nu_{l,k}^{\mathrm{post}}}\end{aligned}$$

where we must decide which smoother to use. [Fin16] suggests Jacobi smoothing. The numerical experiments prove them to be the correct choice for the dG-cPG discretization. If the choice was made for the dG-dG discretization, we made numerical tests and suggest to use again Gauss–Seidel for smoothing in time.

Tests in [DFW16] indicate that it is advantageous to start with refinement in time and then refinement in space, i.e., we use the sequence of meshes $\mathcal{R}_{0,0}, \mathcal{R}_{0,1}, \ldots, \mathcal{R}_{0,k_{\max}}, \mathcal{R}_{1,k_{\max}}, \ldots, \mathcal{R}_{l_{\max},k_{\max}}$ (see Algorithm 2 for the recursive realization of the multilevel preconditioner).

Remark 4.11. The smoother on the different space-time levels (l,k) and the base solver depend on the corresponding matrix representation of the discrete

Algorithm 2 Multilevel preconditioner $\underline{c}_{l,k} = \underline{B}^{\mathrm{ML}}_{l,k}\underline{r}_{l,k}$ with Gauss–Seidel smoother $\underline{B}^{\mathrm{SM}}_{l,k} = \underline{B}^{\mathrm{GS}}_{l,k}$ in space for $l > 0$ or Jacobi smoother $\underline{B}^{\mathrm{SM}}_{0,k} = \underline{B}^{\mathrm{J}}_{0,k}$ in time for dG-cPG. The dG-dG discretization uses always Gauss–Seidel smoothing.

1: **if** $l == 0$ **and** $k == 0$ **then**
2: $\quad \underline{c}_{0,0} = \underline{B}^{\mathrm{ML}}_{0,0}\underline{r}_{0,0}$ solve
3: $\quad$ **return** $\underline{c}_{0,0}$
4: **end if**
5: *p*re-smoothing
6: **for** $\nu = 1, \ldots, \nu^{\mathrm{pre}}_{lk}$ **do**
7: $\quad \underline{w}_{l,k} = \underline{B}^{\mathrm{SM}}_{l,k}\underline{r}_{l,k}$
8: $\quad \underline{c}_{l,k} := \underline{c}_{l,k} + \underline{w}_{l,k}$ and $\underline{r}_{l,k} := \underline{r}_{l,k} - \underline{L}_{l,k}\underline{w}_{l,k}$
9: **end for**
10: $\underline{r}_{l-1,k} = \underline{R}^{l,k}_{l-1,k}\underline{r}_{l,k}$ for $l > 0$ or $\underline{r}_{0,k-1} = \underline{R}^{l,k}_{0,k-1}\underline{r}_{0,k}$ restriction
11: $\underline{c}_{l-1,k} = \underline{B}^{\mathrm{ML}}_{l-1,k}\underline{r}_{l-1,k}$ for $l > 0$ or $\underline{c}_{0,k-1} = \underline{B}^{\mathrm{ML}}_{0,k-1}\underline{r}_{0,k-1}$ PC-cycle
12: $\underline{w}_{l,k} = \underline{P}^{l,k}_{l-1,k}\underline{c}_{l-1,k}$ for $l > 0$ or $\underline{w}_{0,k} = \underline{P}^{l,k}_{0,k-1}\underline{c}_{0,k-1}$ prolongation
13: $\underline{c}_{l,k} := \underline{c}_{l,k} + \underline{w}_{l,k}$ and $\underline{r}_{l,k} := \underline{r}_{l,k} - \underline{L}_{l,k}\underline{w}_{l,k}$ correction
14: *p*ost-smoothing
15: **for** $\nu = 1, \ldots, \nu^{\mathrm{post}}_{lk}$ **do**
16: $\quad \underline{w}_{l,k} = \underline{B}^{\mathrm{SM}}_{l,k}\underline{r}_{l,k}$
17: $\quad \underline{c}_{l,k} := \underline{c}_{l,k} + \underline{w}_{l,k}$ and $\underline{r}_{l,k} := \underline{r}_{l,k} - \underline{L}_{l,k}\underline{w}_{l,k}$
18: **end for**
19: **return** $\underline{c}_{l,k}$

operator. Since we will handle heterogeneous materials, this leads to some problems. The resolution of the given data for the material parameters could be finer than the computational resolution of the space-time mesh. This means, that different problems are treated on every level. This leads to the failure of the space-time multilevel preconditioner. To avoid this, we fix the problem on the coarsest mesh $\mathcal{R}_{0,0}$ by cell wise constant material parameters. This problem could be addressed using homogenization techniques, which we will not handle in this work.

Since the polynomial degree in the space-time cells on the computational mesh can be arbitrary distributed based on the adaptive algorithm, one has to decide how to treat this on the coarser levels. The simplest way is to fix the polynomial degree for the different levels independent for the computational level. A more adapted version would be to use on the coarse discretization for every cell the highest polynomial degree of all cells corresponding to the patch of the finer mesh. We decided to use a low order preconditioning, viz., the polynomial degrees in space and time is coosen as $(p, q) = (0, 1)$. This allows multilevel preconditioning in space, at least some kind of, even for the case of space-time meshes only refined in time $(\mathcal{R}_{0,0}, \mathcal{R}_{0,1})$ during the adaptive refinement process.

4.4.3 Load balancing

A simple distribution and load balancing algorithm is the recursive coordinate bisection (RCB), see, e.g., [MW16]. This geometric partitioning algorithm was extended to space-time in [Fin16, Chap. 7.2]. Since every space-time cell $R \in \mathcal{R}$ has a unique geometric midpoint, we can use them to distribute a mesh $\mathcal{R}$ on $P \in \mathbb{N}$ processes.
The geometric coordinates are first partitioned into two balanced parts, based on weights. This weights can be the amount of degrees of freedom of a space-time cell. This guarantees that the total weight in each part is balanced, rather than the number of space-time cells. The partitioning continues recursively in each part until the desired number of balanced parts has been created.
The importance of weighted load balancing becomes clear when looking on the degrees of freedom of a space-time cell. Let's consider the dG-dG discretization

for the visco-elastic wave equation in 2D with three damping mechanisms. This means a cell with $(p, q) = (0, 0)$ has 14 degrees of freedom on the other hand a cell with $(p, q) = (4, 4)$ has 1 750 degrees of freedom. Not only the actual work load for the CPU should be balanced to minimize the processor idle time, but also the data and therefore the memory consumption.

The RCB algorithm partitions the domain recursively in space and time by bisecting the computational mesh as presented in Alg. 3. Therefore the possible use of total processes is restricted to $P \in \{2^0, 2^1, \cdots\}$. To overcome this issue we combined the RCB algorithm in space with a distribution onto time-slices. The total number of processes must be a multiple of a power of two, e.g., $P = p_1 \cdot 2^{p_2}$. Using the time slices $\mathcal{R}^n = \left\{ I_n \times K \colon K \in \mathcal{K} \right\}$ gives the total space-time mesh $\mathcal{R} = \bigcup_{n=1}^{N} \mathcal{R}^n$. In a first step the time slices are divided in p_1 partitions containing equal amount of cells. In Alg. 4 we present a weighted version. In a second step on every partition the weighted recursive coordinate bisection algorithm in space is applied p_2-times. This algorithm is presented in Alg. 5.

Intel designs their central processor units (CPU) apparently with an arbitrary number of computational cores. On the contrary, the number of computational cores in CPUs produced by AMD are a power of two. Since most high performance clusters rely on Intel processors, this method allows to use the high performance clusters in an efficient way. and was implemented with a focus on uniform convergence experiments.

Algorithm 3 RCB_st(cells $\mathcal{R}$, weights $\mathcal{W}$, factor m, bisections b, sort c)
recursive coordinate bisection in space and time

Require: $m, b \in \mathbb{N},\ c \in \{t, x, y, z\}$

if $b == 0$ **then**
  send cells in $\mathcal{R}$ to process m — distribute cells
  return
end if
— sort and bisect set of cells
sort $\mathcal{R}$ by coordinate c
split $\mathcal{R}$ into $\mathcal{R}_1$ and $\mathcal{R}_2$ such that
    $\sum_{R_1 \in \mathcal{R}_1} \mathcal{W}_{R_1} \approx \sum_{R_2 \in \mathcal{R}_2} \mathcal{W}_{R_2}$
— define coordinate for next bisection
if $c == z$ **then**
  $c := t$
else if $c == y$ **then**
  if dim $== 3$ **then**
    $c := z$
  else
    $c := t$
  end if
else if $c == x$ **then**
  if dim > 1 **then**
    $c := y$
  else
    $c := t$
  end if
else
  $c := x$
end if
— recursive call
RCB_st($\mathcal{R}_1$, $\mathcal{W}$, m, $b-1$, c)
RCB_st($\mathcal{R}_2$, $\mathcal{W}$, $m + 2^{b-1}$, $b-1$, c)

Algorithm 4 newLB(cells $\mathcal{R}$, weights $\mathcal{W}$, processes p, space_processes s)
start load balancer with stripes in time and RCB in space

Require: $s \in \{2^0, 2^1, 2^2, \ldots\}$ **and** $p/s \in \mathbb{N}$

1: $n := p/s$
2: $b := \log(s)/\log(2)$
3: sort $\mathcal{R}$ by coordinate t
4: split $\mathcal{R}$ into $\mathcal{R}_1, \ldots, \mathcal{R}_n$ such that
5: $\quad \sum_{R_1 \in \mathcal{R}_1} \mathcal{W}_{R_1} \approx \ldots \approx \sum_{R_n \in \mathcal{R}_n} \mathcal{W}_{R_n}$
6: **for** $i = 1, \ldots, n$ **do**
7: $\quad$ RCB_space($\mathcal{R}_i$, $\mathcal{W}$, $(i-1)s$, b, x)
8: **end for**

Algorithm 5 RCB_space(cells $\mathcal{R}$, weights $\mathcal{W}$, factor m, bisections b, sort c)
recursive coordinate bisection only in space

Require: $m, b \in \mathbb{N},\ c \in \{x, y, z\}$

1: **if** $b == 0$ **then**
2: $\quad$ send cells in $\mathcal{R}$ to process m
3: $\quad$ **return**
4: **end if**
5: sort $\mathcal{R}$ by coordinate c
6: split $\mathcal{R}$ into $\mathcal{R}_1$ and $\mathcal{R}_2$ such that
7: $\quad \sum_{R_1 \in \mathcal{R}_1} \mathcal{W}_{R_1} \approx \sum_{R_2 \in \mathcal{R}_2} \mathcal{W}_{R_2}$
8: **if** $c == z$ **then**
9: $\quad c := x$
10: **else if** $c == y$ **then**
11: $\quad$ **if** dim $== 3$ **then**
12: $\quad\quad c := z$
13: $\quad$ **else**
14: $\quad\quad c := x$
15: $\quad$ **end if**
16: **else if** $c == x$ **and** dim > 1 **then**
17: $\quad c := y$
18: **end if**
19: RCB_space-time($\mathcal{R}_1$, $\mathcal{W}$, m, $b-1$, c)
20: RCB_space-time($\mathcal{R}_2$, $\mathcal{W}$, $m + 2^{b-1}$, $b-1$, c)

CHAPTER

FIVE

NUMERICAL EXPERIMENTS

The implementation is put into practice using the software framework `M++` [Wie10]. The software is written in `C++` and provides a modular structure with access to all important parts of a finite element discretizations such as mesh-refinement, load-balancing, FEM basis functions, quadrature formulas and preconditioning.

5.1 A simple benchmark experiment for the acoustic wave equation

The first numerical example is specially designed for a convergence test and the solution can be calculated analytically. We use the time interval $(0, T) = (0, 4)$ and the spatial domain $\Omega = (-2, 4) \times (0, 2) \subset \mathbb{R}^2$ with piecewise constant parameters

$$\rho(x_1, x_2) = \begin{cases} 1 & x_1 < 0, \\ 1/2 & 0 < x_1 < 1, \\ 2 & 1 < x_1 \end{cases} \qquad \text{and} \qquad \kappa(\mathbf{x}) = 1/\rho(\mathbf{x})\,.$$

Starting with

$$\mathbf{u}_0(\mathbf{x}) = A(x_1) \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix} \quad \text{for} \quad A(x_1) = \begin{cases} \cos((x_1 - 1)\pi/2)^6 & -2 < x_1 < 0\,, \\ 0 & \text{else} \end{cases}$$

results in a plane wave solution with

$$\mathbf{u}(t, x_1, x_2) = \begin{cases} \mathbf{u}_0(x_1 - t, x_2) & x_1 \leq 0\,, \\ \mathbf{u}_0(0.5x_1 - t, x_2) & 0 < x_1 \leq 1\,, \\ \mathbf{u}_0(0.5 + 2(x_1 - 1) - t, x_2) & 1 \leq x_1\,. \end{cases}$$

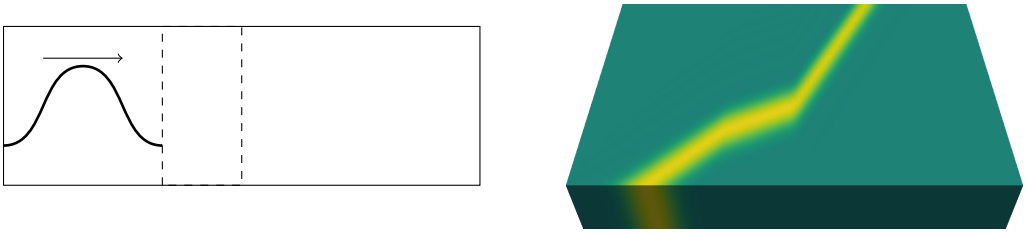

Figure 5.1: Simple benchmark experiment: The initial wave will travel from the left to the right. Sketch of the impulse (left) and pressure component of the space-time solution (right).

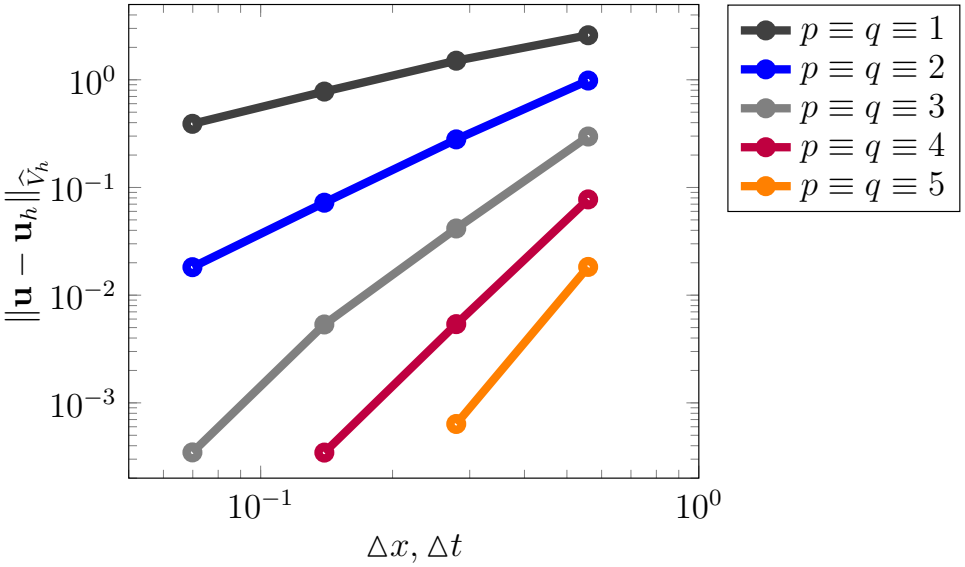

Table 5.1: Simple benchmark experiment solved with dG-cPG: Convergence of the error $\mathbf{e}_h = \mathbf{u} - \mathbf{u}_h$ with respect to the norm $\|\cdot\|_{\widehat{V}_h}$ for uniformly refined space-time meshes and different polynomial degrees.

dG-cPG with linear trial function: $p = q = 1$								
level	st-cells	st-DoF	$\|\mathbf{u} - \mathbf{u}_h\|_{\widehat{V}_h}$	EOC	$\|\mathbf{u} - \mathbf{u}_h\|_W$	EOC	$\|\mathbf{u} - \mathbf{u}_h\|_Q$	EOC
2	1 536	18 432	2.5916		4.7499e-1		6.0851e-1	
				0.78		0.79		1.18
3	12 288	147 456	1.5041		2.7514e-1		2.6856e-1	
				0.95		1.41		1.64
4	98 304	1 179 648	7.7718e-1		1.0320e-1		8.6048e-2	
				0.99		1.83		1.92
5	786 432	9 437 184	3.9002e-1		2.9005e-2		2.2754e-2	
dG-cPG with quadratic trial functions: $p = q = 2$								
level	st-cells	st-DoF	$\|\mathbf{u} - \mathbf{u}_h\|_{\widehat{V}_h}$	EOC	$\|\mathbf{u} - \mathbf{u}_h\|_W$	EOC	$\|\mathbf{u} - \mathbf{u}_h\|_Q$	EOC
2	1 536	82 944	9.8408e-1		8.8313e-2		8.4593e-2	
				1.82		2.78		3.23
3	12 288	663 552	2.7963e-1		1.2834e-2		9.0414e-3	
				1.95		3.10		3.52
4	98 304	5 308 416	7.2221e-2		1.4956e-3		7.8550e-4	
				1.99		3.02		3.18
5	786 432	42 467 328	1.8196e-2		1.8470e-4		8.6713e-5	
dG-cPG with cubic trial functions: $p = q = 3$								
level	st-cells	st-DoF	$\|\mathbf{u} - \mathbf{u}_h\|_{\widehat{V}_h}$	EOC	$\|\mathbf{u} - \mathbf{u}_h\|_W$	EOC	$\|\mathbf{u} - \mathbf{u}_h\|_Q$	EOC
2	1 536	221 184	2.9737e-1		2.0766e-2		1.3046e-2	
				2.84		4.17		4.53
3	12 288	1 769 472	4.1620e-2		1.1517e-3		5.6661e-4	
				2.96		4.03		4.06
4	98 304	14 266 776	5.3454e-3		7.0549e-5		3.3887e-5	
dG-cPG with quartic trial functions: $p = q = 4$								
level	st-cells	st-DoF	$\|\mathbf{u} - \mathbf{u}_h\|_{\widehat{V}_h}$	EOC	$\|\mathbf{u} - \mathbf{u}_h\|_W$	EOC	$\|\mathbf{u} - \mathbf{u}_h\|_Q$	EOC
2	1 536	460 800	7.7530e-2		3.4526e-3		1.7966e-3	
				3.85		5.07		5.16
3	12 288	3 686 400	5.3845e-3		1.0275e-4		5.0259e-5	
				3.96		4.97		4.99
4	98 304	29 491 200	3.4543e-4		3.2733e-6		1.5770e-6	
dG-cPG with quintic trial functions: $p = q = 5$								
level	st-cells	st-DoF	$\|\mathbf{u} - \mathbf{u}_h\|_{\widehat{V}_h}$	EOC	$\|\mathbf{u} - \mathbf{u}_h\|_W$	EOC	$\|\mathbf{u} - \mathbf{u}_h\|_Q$	EOC
2	1 536	829 440	1.8300e-2		5.5690e-4		2.7030e-4	
				4.85		5.90		5.98
3	12 288	6 635 520	6.3672e-4		9.3380e-6		4.2904e-6	

Table 5.2: Simple benchmark experiment solved with dG(p)-cPG(p): Convergence of the error $\mathbf{e}_h = \mathbf{u} - \mathbf{u}_h$ with respect to the norms $\|\cdot\|_{V_h}$, $\|\cdot\|_W$ and $\|\cdot\|_Q$. The experimental orders of convergence (EOC) for uniformly refined space-time meshes is given for different polynomial degrees.

dG-dG with linear trial functions: $p = q = 1$						
level	st-cells	st-DoF	$\|\mathbf{u} - \mathbf{u}_h\|_W$	EOC	$\|\mathbf{u} - \widehat{\mathbf{u}}_h\|_W$	EOC
2	1 536	36 864	5.1717e-1		5.1161e-1	
				1.43		1.43
3	12 288	294 912	1.9195e-1		1.9023e-1	
				2.21		2.22
4	98 304	2 359 296	4.1348e-2		4.0853e-2	
				2.59		2.61
5	786 432	18 874 368	6.8498e-3		6.6726e-3	
				2.45		2.48
6	6 291 456	150 994 944	1.2573e-3		1.1996e-3	
dG-dG with quadratic trial functions: $p = q = 2$						
level	st-cells	st-DoF	$\|\mathbf{u} - \mathbf{u}_h\|_W$	EOC	$\|\mathbf{u} - \widehat{\mathbf{u}}_h\|_W$	EOC
2	1 536	124 416	1.2138e-1		1.2073e-1	
				3.39		3.39
3	12 288	995 328	1.1584e-2		1.1485e-2	
				3.55		3.56
4	98 304	7 962 624	9.8915e-4		9.7150e-4	
				3.09		3.10
5	786 432	63 700 992	1.1582e-4		1.1346e-4	
dG-dG with cubic trial functions: $p = q = 3$						
level	st-cells	st-DoF	$\|\mathbf{u} - \mathbf{u}_h\|_W$	EOC	$\|\mathbf{u} - \widehat{\mathbf{u}}_h\|_W$	EOC
2	1 536	294 912	1.9124e-2		1.9080e-2	
				4.58		4.58
3	12 288	2 359 296	8.0018e-4		7.9627e-4	
				4.07		4.07
4	98 304	18 874 368	4.7572e-5		4.7312e-5	
dG-dG with quartic trial functions: $p = q = 4$						
level	st-cells	st-DoF	$\|\mathbf{u} - \mathbf{u}_h\|_W$	EOC	$\|\mathbf{u} - \widehat{\mathbf{u}}_h\|_W$	EOC
2	1 536	576 000	2.5533e-3		2.5506e-3	
				5.17		5.17
3	12 288	4 608 000	7.0984e-5		7.0884e-5	
				4.90		4.90
4	98 304	36 864 000	2.3825e-6		2.3796e-6	
dG-cPG with quintic trial functions: $p = q = 5$						
level	st-cells	st-DoF	$\|\mathbf{u} - \mathbf{u}_h\|_W$	EOC	$\|\mathbf{u} - \widehat{\mathbf{u}}_h\|_W$	EOC
2	1 536	995 328	3.8184e-4		3.8171e-4	
				5.97		5.97
3	12 288	7 962 624	6.1037e-6		6.1015e-6	

Table 5.3: Simple benchmark experiment solved with dG(p)-dG(p): Convergence of the error $\mathbf{e}_h = \mathbf{u} - \mathbf{u}_h$ and the error of the conforming reconstruction $\widehat{\mathbf{e}}_h = \mathbf{u} - \widehat{\mathbf{u}}_h$. The experimental orders of convergence (EOC) for uniformly refined space-time meshes is given for different polynomial degrees $p \equiv q$.

dG-dG with linear in space and constant in time trial functions						
level	st-cells	st-DoF	$\|\mathbf{u}-\mathbf{u}_h\|_W$	EOC	$\|\mathbf{u}-\widehat{\mathbf{u}}_h\|_W$	EOC
2	1 536	18 432	1.2105		1.2155	
				0.28		0.29
3	12 288	147 456	9.9359e-1		9.9267e-1	
				0.39		0.39
4	98 304	1 179 648	7.5782e-1		7.5512e-1	
				0.52		0.52
5	786 432	9 437 184	5.2828e-1		5.2570e-1	
				0.66		0.66
6	6 291 456	75 497 472	3.3466e-1		3.3286e-1	
				0.78		0.78
7	50 331 648	603 979 776	1.9470e-1		1.9362e-1	
dG-dG with quadratic in space and linear in time trial functions						
level	st-cells	st-DoF	$\|\mathbf{u}-\mathbf{u}_h\|_W$	EOC	$\|\mathbf{u}-\widehat{\mathbf{u}}_h\|_W$	EOC
2	1 536	82 944	2.8961e-1		2.8150e-1	
				2.14		2.19
3	12 288	663 552	6.5768e-2		6.1648e-2	
				2.64		2.85
4	98 304	5 308 416	1.0523e-2		8.5736e-3	
				2.49		2.99
5	786 432	42 467 328	1.8784e-3		1.0780e-3	
				2.20		3.00
6	6 291 456	339 738 624	4.0776e-4		1.3455e-4	
dG-dG with cubic in space and quadratic in time trial functions						
level	st-cells	st-DoF	$\|\mathbf{u}-\mathbf{u}_h\|_W$	EOC	$\|\mathbf{u}-\widehat{\mathbf{u}}_h\|_W$	EOC
2	1 536	221 184	3.1559e-2		2.9442e-2	
				4.07		4.66
3	12 288	1 769 472	1.8783e-3		1.1606e-3	
				3.28		4.43
4	98 304	14 155 776	1.9357e-4		5.3787e-5	
				3.04		4.11
5	786 432	113 246 208	2.3489e-5		3.1150e-6	
dG-dG with quartic in space and cubic in time trial functions						
level	st-cells	st-DoF	$\|\mathbf{u}-\mathbf{u}_h\|_W$	EOC	$\|\mathbf{u}-\widehat{\mathbf{u}}_h\|_W$	EOC
2	1 536	460 800	3.0382e-3		2.7790e-3	
				4.83		5.28
3	12 288	3 686 400	1.0665e-4		7.1716e-5	
				4.28		4.97
4	98 304	29 491 200	5.4760e-6		2.2944e-6	
dG-cPG with quintic in space and quatric in time trial functions						
level	st-cells	st-DoF	$\|\mathbf{u}-\mathbf{u}_h\|_W$	EOC	$\|\mathbf{u}-\widehat{\mathbf{u}}_h\|_W$	EOC
2	1 536	829 440	4.0151e-4		3.8419e-4	
				5.81		5.97
3	12 288	6 635 520	7.1810e-6		6.1159e-6	

Table 5.4: Simple benchmark experiment solved with dG(p)-dG(p-1): Convergence of the error $\mathbf{e}_h = \mathbf{u} - \mathbf{u}_h$ and the error of the conforming reconstruction $\widehat{\mathbf{e}}_h = \mathbf{u} - \widehat{\mathbf{u}}_h$. The experimental orders of convergence (EOC) for uniformly refined space-time meshes are given for different polynomial degrees.

The computed experimental orders of convergence for the dG-cPG method are shown in Tab. 5.1 and plotted in Fig. 5.1. We observe the expected order of convergence as predicted in Theorem 3.3 for sufficiently smooth solutions in the $\widehat{V}_h$-norm. Using the W-norm, we gain one order.
In Tab. 5.1 we observe the convergence of the dG-dG method with same polynomial degree in space and time. The convergence order is expected to be $\min\{p+q, q+1\}$ which is confirmed by the numerical tests. We obtain the same convergence rates but a slightly smaller error, if we use the conforming reconstruction of the discrete solution.
In Tab. 5.1 the polynomials in time are one order lower than in space, i.e., dG(q)-dG($q-1$). This reduces also the order of convergence for the error $\|\mathbf{u}-\mathbf{u}_h\|_W$ to order q. If we use the conforming reconstruction, the convergence can be improved by one order, obtaining the same convergence as the dG-cPG method with the same amount of degrees of freedom.

5.2 Marmousi II: a geophysical benchmark in heterogeneous media

Marmousi II [MWM06] is an elastic upgrade of Marmousi [Ver94]. It is a benchmark problem for geophysical purposes which provides realistic structures in two space dimensions with heterogeneous media, see Fig. 5.2 for the density distribution in this benchmark configuration.
Marmousi I was created 1988 and used for acoustic finite difference with synthetic data. The extension included a water layer on the top and the data for shear wave velocity for the elastic case.
For the numerical experiments, we simulate maritime measurements in seismic exploration with a local source initiating a wave by a smooth pulse in space of width $w_\mathrm{s} = 100$ [m] located at $\mathbf{x}_\mathrm{s} \in \Omega$

$$\phi(\mathbf{x}) = \begin{cases} \cos^6\left(\dfrac{\pi|\mathbf{x}_\mathrm{s}-\mathbf{x}|}{2w_\mathrm{s}}\right) & |\mathbf{x}_\mathrm{s}-\mathbf{x}| < w_\mathrm{s}\,, \\ 0 & \text{else.} \end{cases} \tag{5.1}$$

and a Ricker wavelet in time

$$\psi(t) = \left(1 - 2\pi^2(t-t_\mathrm{s})^2 f^2\right) \exp\left(-\pi^2(t-t_\mathrm{s})^2 f^2\right)$$

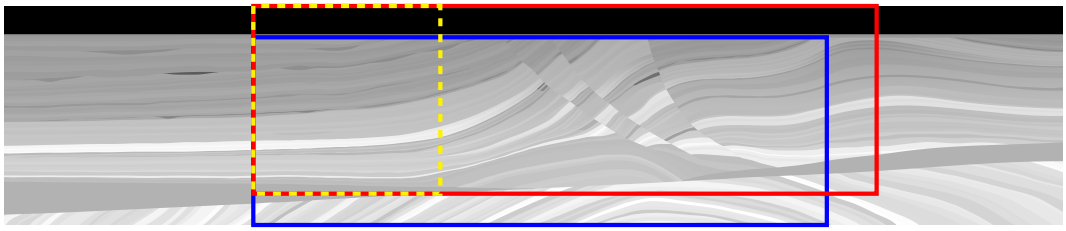

Figure 5.2: Density distribution for the Marmousi II benchmark: The graphic shows the full Marmousi II benchmark with 17 km × 3.5 km. In blue we sketch the domain of Marmousi I. The red subdomain 10 km × 3 km is used in the adaptive numerical experiments and the smaller yellow subdomain 3 km × 3 km for the convergence tests in space and time on uniform discretizations.

with frequency f and time delay $t_s > 0$. We sketched a Ricker wavelet, also called Mexican hat wavelet, with frequency $f = 10$ [Hz] and a delay $t_s = 0.15$ [s] in Fig. 5.8. This results in the right-hand side $\mathbf{b}(t, \mathbf{x}) = \psi(t)\,\phi(\mathbf{x})\,\mathbf{e}$ with $\mathbf{e} = (\mathbf{0}, 1, 0, \ldots, 0) \in \mathbb{R}^{\dim+1+L}$ in the acoustic case, and $\mathbf{e} = (\mathbf{0}, \mathbf{I}_3, \mathbf{0}, \ldots, \mathbf{0}) \in \mathbb{R}^{\dim} \times \mathbb{R}^{\dim\times\dim}_{\text{sym}} \times \cdots \times \mathbb{R}^{\dim\times\dim}_{\text{sym}}$ for elasticity.
In our tests, the solution is compared for different discretizations by the resulting pressure evaluated at the receivers positions $\mathbf{x}_{r,i} \in \Omega$, $i = 0, \ldots, N_r$. This defines a seismogram $\mathbf{s} \in L_2(0, T; \mathbb{R}^{N_r})$, i.e., $s_i(t) = p(t, \mathbf{x}_{r,i})$.

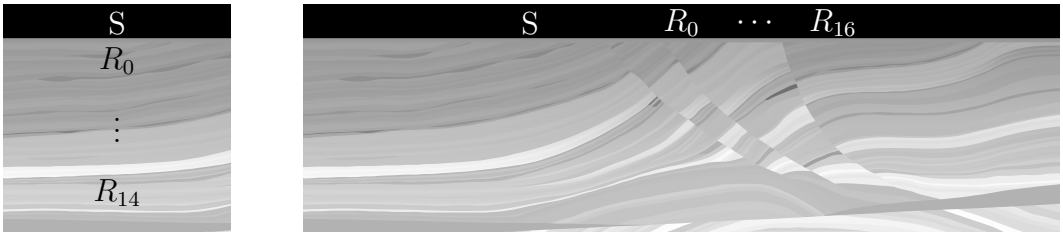

Figure 5.3: Marmousi II: Sketch of location of source and receivers for the uniform computations used in the first numerical experiment on the left and for the adaptive computations used in the second experiment on the right.

Material parameters The Marmousi model defines a density distribution $\rho \in (1010, 2627)$ [kg/m^3] (cf. Fig. 5.5) and reference values for the velocities of shear waves $v_S \in (0, 2802)$ [m/s] (cf. Fig. 5.7) and compressional waves

$v_{\mathrm{P}} \in (1028, 4700)$ [m/s] (cf. Fig. 5.6). This defines the parameters $\mu = \rho v_{\mathrm{S}}^2$ and $\kappa = \rho v_{\mathrm{P}}^2 - \frac{4}{3}\mu$ for isotropic elasticity. We fix this material parameters cellwise constant on a spatial mesh with mesh size 125 [m], cf. Rem. 4.11.

For the viscous extension with $G > 0$, we use the reference values from [Kur12, p. 168]. We set $\kappa_0 = \frac{\kappa}{1+G\tau_{\mathrm{P}}}$ and $\kappa_1 = \cdots = \kappa_G = \kappa_0\tau_{\mathrm{P}}$ with $\tau_{\mathrm{P}} = 0.1$, and we set $\mu_0 = \frac{\mu}{1+G\tau_{\mathrm{S}}}$ and $\mu_1 = \cdots = \mu_G = \mu_0\tau_{\mathrm{S}}$ with $\tau_{\mathrm{S}} = 0.1$, Furthermore, we use the relaxation time $\tau_g = \frac{1}{2\pi f_g}$ with reference frequencies $f_1 = 0.151$ [Hz], $f_2 = 1.93$ [Hz], and $f_3 = 18.9$ [Hz], cf. [Kur12, p. 115] for $G = 3$ and $f_1 = 10$ [Hz] for $G = 1$.

The quality factor is dimensionless and characterizes the damping of the generalized standard linear solid (GSLS) depending of the wave frequency. The equation for the quality factor can be found in [FOGG17, eq. (3)], i.e.,

$$Q(\omega, \tau_g, \tau_*) = \frac{1 + \sum_{g=1}^{G} \frac{\omega^2\tau_g^2}{1+\omega^2\tau_g^2}\tau_*}{\sum_{g=1}^{G} \frac{\omega\tau_g}{1+\omega^2\tau_g^2}\tau_*}.$$

Fig. 5.4 illustrates that using more damping mechanisms ensures damping for a broader frequency bandwidth.

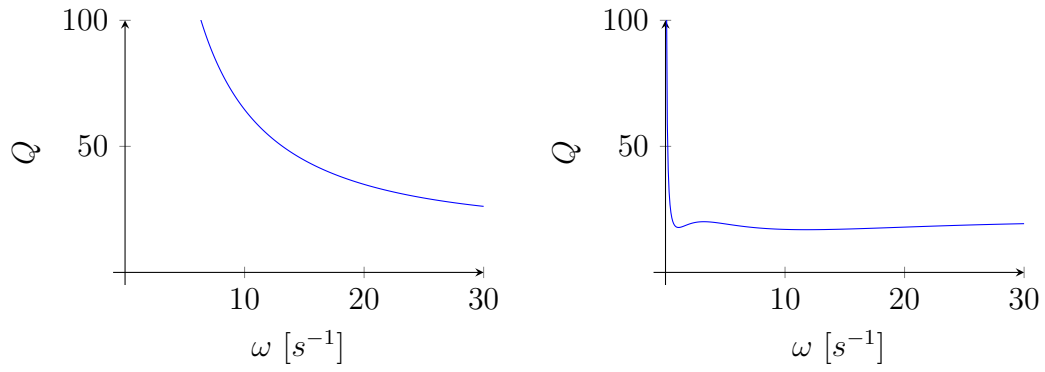

Figure 5.4: Quality factor of GSLS for $\tau_P = \tau_S = 0.1$ with one damping mechanism $G = 1$ and $\tau_1 = \frac{1}{20\pi}$ on the left and three damping mechanisms $G = 3$ with $f_1 = 0.151$, $f_2 = 1.93$ and $f_3 = 18.9$ on the right.

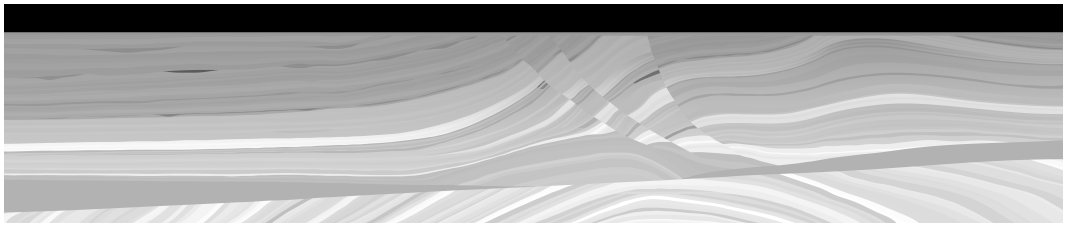

Figure 5.5: Marmousi II: density [1010kg/m^3-2627kg/m^3]

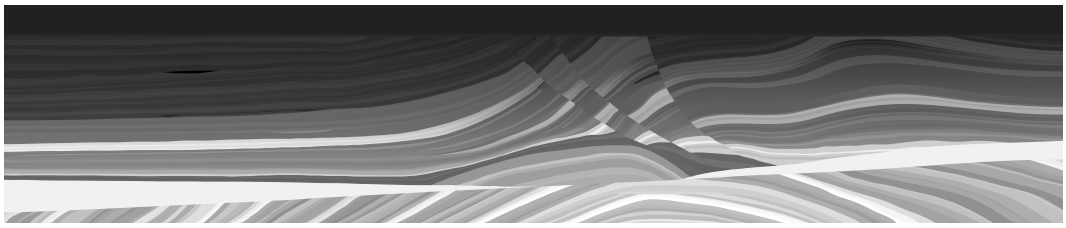

Figure 5.6: Marmousi II: velocity primary wave [1028m/s-4700m/s]

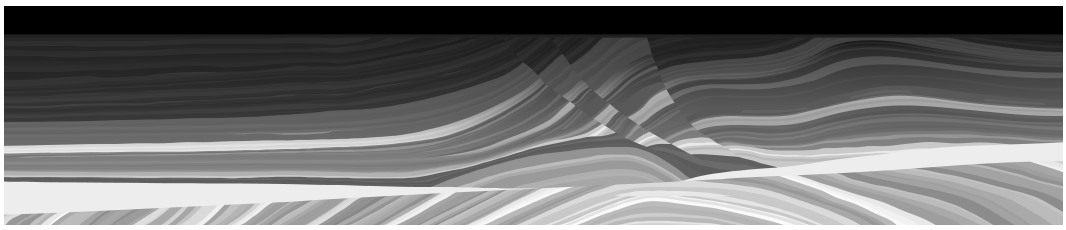

Figure 5.7: Marmousi II: velocity secondary wave [0m/s-2802m/s]

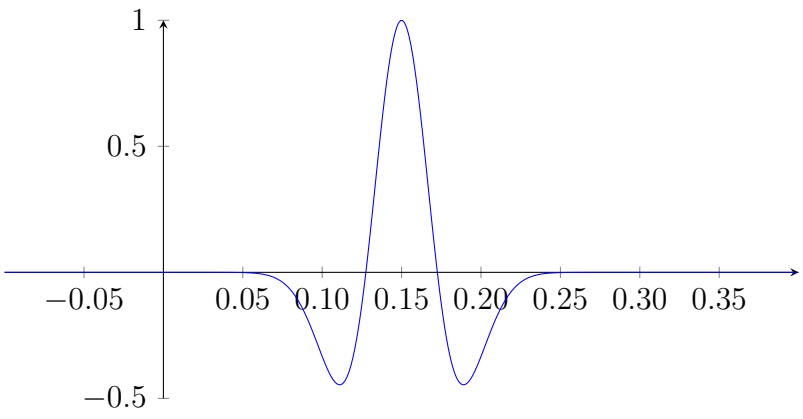

Figure 5.8: Ricker wavelet in time with $f = 10$ [Hz] and delay $t_s = 0.15$ [s]

5.2.1 Acoustic equation: convergence in space and time of the continuous Petrov–Galerkin in time method

In this numerical test we investigate the convergence properties on uniform discretizations with respect to the mesh size $h_j = 2^{-j}h_0$, the time steps size $\triangle t_k = 2^{-k}\triangle t_0$, and polynomial degrees p and q in space and time for the acoustic model ($G = 0$). Here, we use from the full Marmousi II benchmark configuration the subdomain $\Omega = (4000, 7000) \times (-3000, 0) \subset (0, 17000) \times (-3500, 0)$ [m^2] (see Fig. 5.2) and the time interval $(0, T)$ with $T = 1.5$ [s]. We use a coarse mesh in space and time with $h_0 = 1000$ [m] and $\triangle t_0 = 0.1$ [s]. The initial pulse is located at $\mathbf{x}_\mathrm{s} = (5500, -250)$. The seismograms are measured at the receivers positions $\mathbf{x}_{\mathrm{r},i} = (5500, -750 - 125i)$ for $i = 0, \ldots, 14$. The seismogram for mesh levels (j, k) in space and time and polynomial degrees (p, q) is denoted by $\mathbf{s}_{j,k,p,q}$. We estimate the convergence properties by comparing the seismograms for different discretization parameters. All quantities in this test are normalized with respect to the reference value $\|\mathbf{s}_{6,3,2,2}\|_{(0,T)}$.
The evaluation of the numerical test suite is presented in Tab. 5.5–5.8 and can be summarized as follows:

Convergence in time Asymptotically, we observe nearly forth order convergence for the two-point Gauss collocation method to the space-discrete solution and nearly sixth order convergence for the three-point Gauss collocation method (Tab. 5.5).

Convergence in space The results are summarized in Tab. 5.6 for the convergence test in space. We observe fourth order convergence with polynomials of order four in space and second order for quadratic polynomials.

Convergence in the polynomial degrees of the discretization We observe fast convergence by increasing the polynomial degrees, cf. Tab. 5.7. Here an evaluation of the convergence quality is more involved since the relation to the dimension of the ansatz space is not linear. Nevertheless, it is clearly observed that the convergence for $p = q = 2$ is very slow and that higher order ansatz spaces are much more efficient.

Estimated accuracy of the finest solution Since we observe convergence in the seismograms in space and time, we can construct a better approx-

imation of the discrete solution by extrapolation. Here we choose fixed polynomial degrees $(p, q) = (2, 2)$, so that the discretization is of second order in space and time. Then, the error of the finest solution $\mathbf{s}_{6,4,2,2}$ is estimated by first extrapolating in space

$$\mathbf{s}_k^{\text{ex}} = \frac{4}{4-1}\mathbf{s}_{6,k,2,2} - \frac{1}{4-1}\mathbf{s}_{5,k,2,2}\,, \qquad k \in \{2,3,4\}\,.$$

We can determine the convergence rate in time with the extrapolated seismograms in space by $f = \dfrac{\|\mathbf{s}_3^{\text{ex}} - \mathbf{s}_2^{\text{ex}}\|_{(0,T)}}{\|\mathbf{s}_4^{\text{ex}} - \mathbf{s}_3^{\text{ex}}\|_{(0,T)}}$. With the seismograms extrapolated in space and the convergence rate in time, we can extrapolate in time

$\|\mathbf{s}_{j,k,2,2} - \mathbf{s}^{\text{ex}}\|_{(0,T)}$	$j = 5$	$j = 6$	$\|\mathbf{s}_k^{\text{ex}} - \mathbf{s}^{\text{ex}}\|_{(0,T)}$
$k = 2$	0.7483	0.7908	0.8048
$k = 3$	0.1443	0.1528	0.1675
$k = 4$	0.1333	0.0335	0.0105

$$\mathbf{s}^{\text{ex}} = \frac{f}{f-1}\mathbf{s}_4^{\text{ex}} - \frac{1}{f-1}\mathbf{s}_3^{\text{ex}}\,.$$

Together, the extrapolated error estimate yields for the finest solution an accuracy of approximately 3%.

Efficiency of the approximation The relative error with respect to the extrapolated value is shown in Tab. 5.8. We observe an accuracy in the seismograms of 15% on space level $j = 6$ with approximately 239 Mio. DoF using 120 time slices with $(p, q) = (2, 2)$, or 179 Mio. DoF with 60 time slices with $(p, q) = (2, 3)$. On level $j = 4$ we require polynomial of order $p = 4$ in space and of order $q = 3$ in time resulting in a system with only 31 Mio. unknowns but a relative error of 11%.

An accuracy in the seismograms better than 5% is achieved only with the finest computation with approximately 478 Mio. degrees of freedom on space level $j = 6$.

$j=4$	$(p,q)=(3,1)$	$(p,q)=(3,2)$	$(p,q)=(3,3)$	$(p,q)=(4,2)$
$\|\mathbf{s}_{j,2,p,q}\|_{(0,T)}$		0.9145	0.9522	0.9485
$\|\mathbf{s}_{j,3,p,q}\|_{(0,T)}$	0.8134	0.9510	0.9537	0.9951
$\|\mathbf{s}_{j,4,p,q}\|_{(0,T)}$	0.9131	0.9535	0.9537	0.9989
$\|\mathbf{s}_{j,5,p,q}\|_{(0,T)}$	0.9456	0.9537		
$\|\mathbf{s}_{4,3,p,q}-\mathbf{s}_{4,2,p,q}\|_{(0,T)}$		0.6311	0.0474	0.7451
		3.03	5.63	2.55
$\|\mathbf{s}_{4,4,p,q}-\mathbf{s}_{4,3,p,q}\|_{(0,T)}$	1.1111	0.0774	0.0010	0.1275
	0.79	3.88		
$\|\mathbf{s}_{4,5,p,q}-\mathbf{s}_{4,4,p,q}\|_{(0,T)}$	0.6446	0.0053		

Table 5.5: Convergence in time for $k \in \{2,3,4,5\}$ for different polynomial degrees (p,q) and fixed level $j=4$ in space. The convergence rate of the seismograms is estimated by $m^k_{j,p,q} = \log_2 \dfrac{\|\mathbf{s}_{j,k,p,q}-\mathbf{s}_{j,k-1,p,q}\|_{(0,T)}}{\|\mathbf{s}_{j,k+1,p,q}-\mathbf{s}_{j,k,p,q}\|_{(0,T)}}$.

$k=3$	$(p,q)=(2,2)$	$(p,q)=(4,1)$
$\|\mathbf{s}_{3,3,p,q}\|_{(0,T)}$	0.4271	0.7460
$\|\mathbf{s}_{4,3,p,q}\|_{(0,T)}$	0.7320	0.8404
$\|\mathbf{s}_{5,3,p,q}\|_{(0,T)}$	0.9647	0.8446
$\|\mathbf{s}_{6,3,p,q}\|_{(0,T)}$	1.0000	
$\|\mathbf{s}_{4,3,p,q}-\mathbf{s}_{3,3,p,q}\|_{(0,T)}$	0.5762	0.3210
		4.02
$\|\mathbf{s}_{5,3,p,q}-\mathbf{s}_{4,3,p,q}\|_{(0,T)}$	0.4762	0.0198
	2.24	
$\|\mathbf{s}_{6,k,p,q}-\mathbf{s}_{5,3,p,q}\|_{(0,T)}$	0.1008	

Table 5.6: Convergence in space level for $j \in \{3,4,5,6\}$ for different polynomial degrees (p,q) and fixed level $k=3$ in time. The convergence rate of the seismograms is estimated by $m^j_{k,p,q} = \log_2 \dfrac{\|\mathbf{s}_{j,k,p,q}-\mathbf{s}_{j-1,k,p,q}\|_{(0,T)}}{\|\mathbf{s}_{j+1,k,p,q}-\mathbf{s}_{j,k,p,q}\|_{(0,T)}}$.

	$\|\mathbf{s}_{j,k,2,2}\|_{(0,T)}$	$\|\mathbf{s}_{j,k,3,3}\|_{(0,T)}$	$\|\mathbf{s}_{j,k,4,4}\|_{(0,T)}$
$j=3,\ k=3$	0.4271	0.6472	0.8538
$j=4,\ k=3$	0.7207	0.9522	0.9991

Table 5.7: Convergence in polynomial degrees in space and time for $p,q \in \{2,3,4\}$ on fixed space-time meshes.

j	(p,q)	$\frac{\text{DoF}}{\text{Cell}}$	$k=2$	DoF	$k=3$	DoF	$k=4$	DoF	$k=5$	DoF
4	$(3,1)$	48					85%	26 542 080	29%	53 084 160
	$(3,2)$	96			17%	26 542 080	20%	53 084 160	20%	106 168 320
	$(3,3)$	144			17%	39 813 120	20%	79 626 240	20%	159 252 480
	$(4,2)$	150	79%	20 736 000	14%	41 472 000	6%	82 944 000		
	$(4,3)$	225	11%	31 104 000	6%	62 208 000				
	$(4,4)$	300	6%	124 416 000						
5	$(2,1)$	27					88%	59 719 680	31%	119 439 360
	$(2,2)$	54	75%	29 859 840	14%	59 719 680	13%	119 439 360	14%	238 878 720
	$(2,3)$	81	13%	44 789 760	14%	89 579 520	14%	179 159 040		
	$(3,2)$	96	79%	53 084 160	15%	106 168 320				
	$(3,3)$	144	12%	79 626 240						
6	$(1,2)$	24	66%	53 084 160	29%	106 168 320				
	$(1,3)$	36	29%	79 626 240	29%	159 252 480				
	$(2,2)$	54			15%	238 878 720	3%	477 757 440		
	$(2,3)$	81	12%	179 159 040						

Table 5.8: Relative error $\|\mathbf{s}_{j,k,p,q} - \mathbf{s}^{\text{ex}}\|_{(0,T)}$ with respect to the extrapolated value $\mathbf{s}^{\text{ex}}$ together with the necessary degrees of freedom.

5.2.2 Acoustic equation: convergence of the adaptive algorithm with the Petrov–Galerkin in time method

In the second experiment with heterogeneous media, we test the efficiency of the adaptive scheme for the acoustic model with respect to a reference solution computed with a time stepping scheme on a uniform mesh.

Here we choose the domain $\Omega = (4000, 13000) \times (-3000, 0) \subset (0, 17000) \times (-3500, 0)$ [m^2] and the time interval $(0, T)$ with $T = 3$ [s]. The source is located at $\mathbf{x}_{\text{s}} = (7000, -250)$, and the receivers positions are $\mathbf{x}_{\text{r},j} = (9000 + 125j, -250)$ for $j = 0, \dots, 16$. For the adaptive simulations we use the goal functional

$$\mathcal{J}_{\text{acoustic}}(\mathbf{v}, p) = \frac{1}{|\Omega_{\text{RoI}}|} \int\limits_{\Omega_{\text{RoI}} \times \{T\}} p \, \mathrm{d}\mathbf{x}$$

evaluating the mean value in a region of interest $\Omega_{\text{RoI}} = (8750, 11250) \times (-400, -100)$ [m^2] of the pressure (cf. Def. 4.2). The adaptive algorithm uses the maximum marking strategy with the parameter $\theta = 5e{-}5$ and $\tilde{\theta} = 1e{-}2$. Hence, the polynomial degree in space and time is increased in all cells with $\eta_R > \theta \, \eta_{\max}$ and decreased, if $\eta_R < \tilde{\theta} \, \theta \, \eta_{\max}$ (cf. Sec. 4.2).

The seismogram $\mathbf{s}_{\text{ref}}$ of the reference solution is computed with a time stepping scheme using in space the mesh on level $j = 6$ and polynomials of order $p = 4$ resulting in 9 216 000 degrees of freedom in space, and 6 000 steps in time with the implicit midpoint rule.

adaptive (p, q)-refinement on mesh level 3						
r	e	e_0	DoF	%DoF	ML	uniform DoF
0	0.998	0.997	2 211 840	100%	50	2 211 840
1	0.964	0.905	3 134 184	31%	74	9 953 280
2	0.784	0.556	7 403 634	28%	92	26 542 080
3	0.424	0.263	14 780 223	27%	115	55 296 000
4	0.157	0.163	25 748 967	26%	144	99 532 800
adaptive (p, q)-refinement on mesh level 4						
r	e	e_0	DoF	%DoF	ML	uniform DoF
0	0.987	0.971	17 694 720	100%	14	17 694 720
1	0.593	0.368	20 779 680	26%	17	79 626 240
2	0.145	0.132	48 338 979	23%	26	212 336 640

Table 5.9: Acoustic waves: error of the seismograms $e = \frac{\|\mathbf{s} - \mathbf{s}_{\text{ref}}\|_{(0,2.5)}}{\|\mathbf{s}_{\text{ref}}\|_{(0,2.5)}}$ and of the first receiver $e_0 = \frac{\|s_0 - s_{0,\text{ref}}\|_{(0,2.5)}}{\|s_{0,\text{ref}}\|_{(0,2.5)}}$ on fixed space-time meshes for the steps r of the p-adaptive algorithm. In both tests we start for $r = 0$ with $(p, q) = (1, 1)$. The GMRES solver used ML-steps, which were preconditioned with the multilevel preconditioner. We use 10 smoothing steps if coarsened in time and 20 if coarsened in space. The last column gives the number of degrees of freedom obtained by uniform p-refinement, were we expect to have nearly the same accuracy.

The adaptive algorithm starts with a very coarse initial approximation using linear functions in space and time. With this initial solution, the p-adaptive algorithm starts refining the necessary cells by increasing and decreasing the polynomial degrees in space and time simultaneously based on the error indicator. We observe convergence towards the reference solution (cf. Tab. 5.9). With the adaptive algorithm we save in the final step approximately 74% on level 3 and 74% on level 4 of the degrees of freedom compared to uniform

refinement. The adaptive results on level 3 are visualized in Fig. 5.10 and on level 4 in Fig. 5.11. The seismogram of the first receiver for both level are visualized in Fig. 5.9.

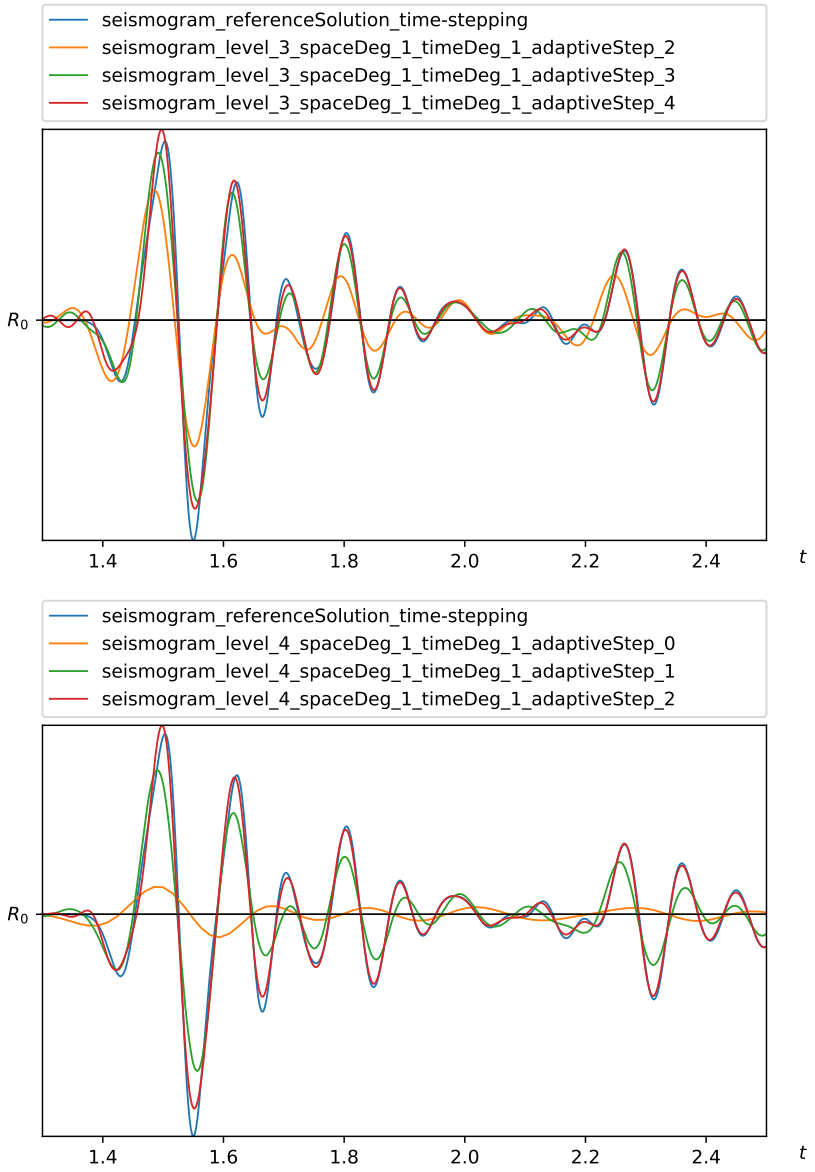

Figure 5.9: Adaptive results with focus on the first receiver: Adaptive steps $r = 2, 3, 4$ on level 3 (top) and adaptive steps $r = 0, 1, 2$ on level 4 (bottom) The seismogram $\mathbf{s}_{\text{ref}}$ of the reference solution is plotted in blue.

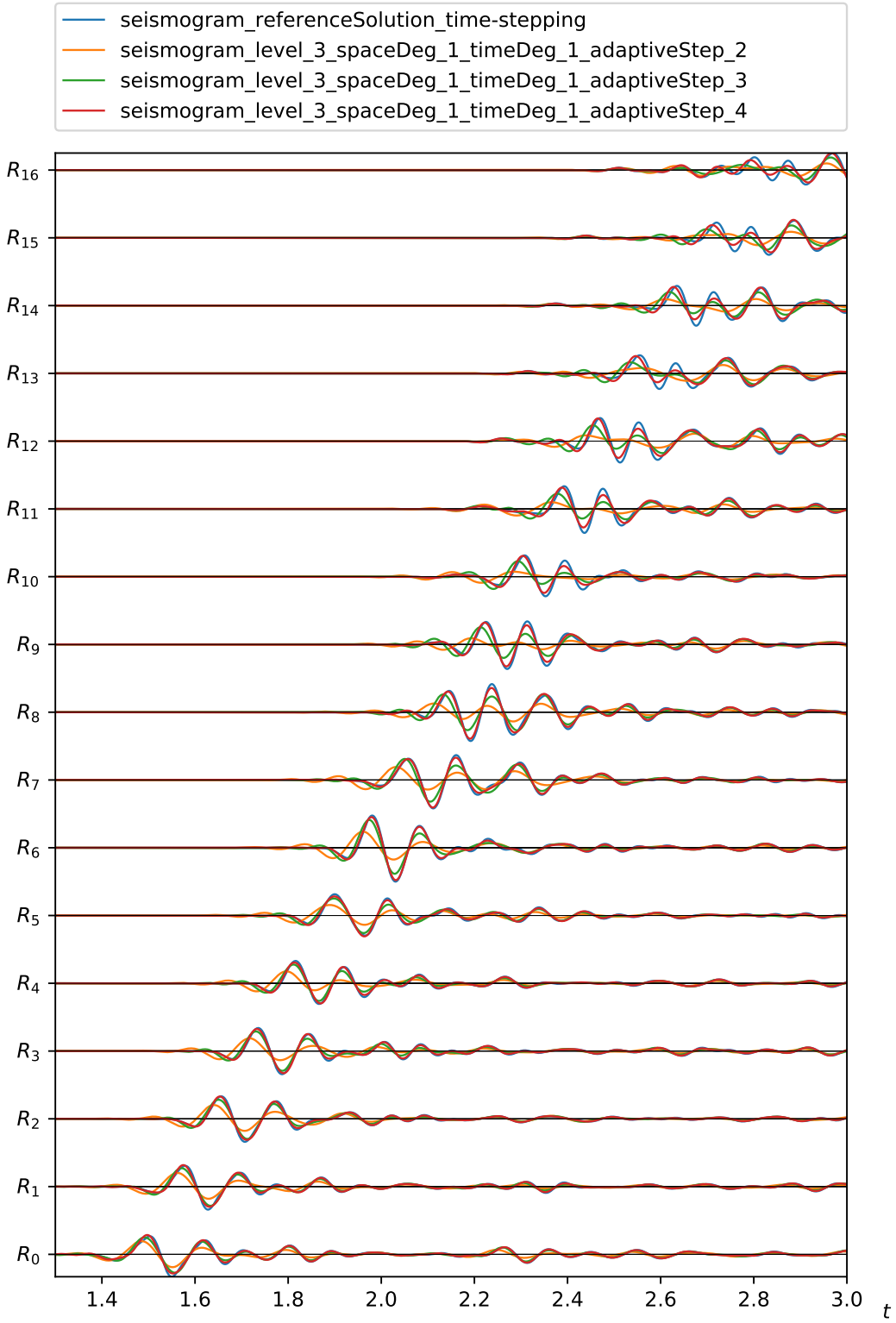

Figure 5.10: Seismogramms of the adaptive results on level 3. The initial and first step are not shown. The second step is orange and the third adaptive step is green and the fourth step is red. The seismogramm of the reference solution is computed by a time stepping scheme (blue).

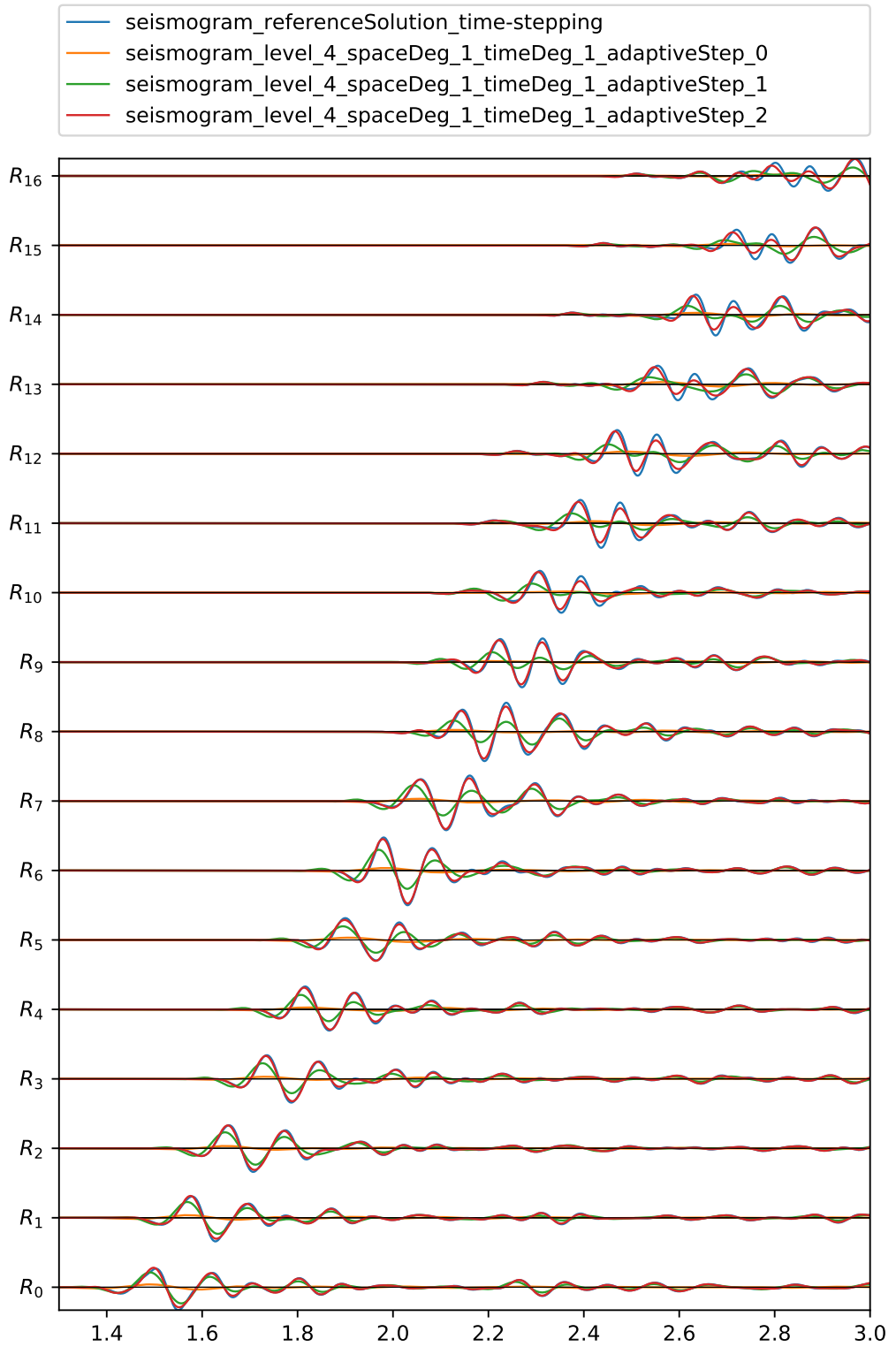

Figure 5.11: Seismogramms of the adaptive results on level 4. Starting with the orange wave, the adaptive algorithm refines to the green and finishes with the red seismograms. The seismogramm of the reference solution is computed by a time stepping scheme (blue).

5.2.3 Visco-acoustic equation with three damping mechanisms and uniform p-refinement

We compare the dG-dG method with the dG-cPG method on uniform discretizations with polynomial degrees p and q in space and time for the visco-acoustic model with three damping mechanisms ($G = 3$) in this numerical test. Here, we use from the full Marmousi II benchmark configuration the subdomain $\Omega = (4000, 7000) \times (-3000, 0) \subset (0, 17000) \times (-3500, 0)$ [m^2] (see Fig. 5.2 yellow dashed box) and the time interval $(0, T)$ with $T = 1.5$ [s]. We use a coarse mesh in space and time with $h_0 = 1000$ [m] and $\triangle t_0 = 0.25$ [s]. The initial pulse is located at $\mathbf{x}_\mathrm{s} = (5500, -250)$. The seismograms are measured at the receivers with the positions $\mathbf{x}_{\mathrm{r},i} = (5500, -750 - 125i)$ for $i = 0, \ldots, 14$. Since we have no analytical solution for the problem, we decide to compute the reference seismogram by extrapolation in space and time simultaneously. Therefore we follow the idea given in [HPS+15]. The order of convergence on the space-time mesh of level l can be estimated from the factor

$$f_l = \frac{\|\mathbf{s}_{l-1} - \mathbf{s}_{l-2}\|_{(0,T)}}{\|\mathbf{s}_l - \mathbf{s}_{l-1}\|_{(0,T)}},$$

where $\mathbf{s}_l$ denotes the seismogram on level l combined with the L_2-norm. With this factor a better approximation can be constructed by extrapolation as

$$\mathbf{s}_\mathrm{ex} = \frac{f_l}{f_l - 1}\mathbf{s}_l - \frac{1}{f_l - 1}\mathbf{s}_{l-1}.$$

Here we choose the fixed polynomial degrees $(p, q) = (3, 2)$ and the space-time levels $l = 3, \ldots, 5$ obtained by uniform refinement in space-time. All quantities in this test are normalized with respect to the reference value $\|\mathbf{s}_\mathrm{ex}\|_{(0,T)}$.
A selection of the results of this numerical experiment are shown in Tab. 5.10. At first we want to remark that the dG-cPG(q) and dG-dG(q-1) method have the same amount of degrees of freedom. The results indicate, that the cPG version gives more accurate results than the dG method with one order lower in time.
The advantage of the dG-dG method over the dG-cPG method is, that the system matrix is less dense. As a result, the total time to solve the system is less. Also less total system memory is needed especially with higher polynomials in time. The reason is the fewer coupling between the space-time cells in the matrix graph.

dG-cPG on space-time mesh level 4								
(p,q)	e		RAM	DoF	ML	time	cores	cluster
(2,2)	26.9%		387 GB	23 887 872	10	0:15:04	256	MA-PDE
(2,3)	28.7%		753 GB	35 831 808	9	0:27:35	256	MA-PDE
(3,2)	4.6%		1.0 TB	42 467 328	15	1:06:22	256	MA-PDE
(3,3)	4.8%		2.2 TB	63 700 992	15	2:21:11	256	MA-PDE
dG-dG on space-time mesh level 4								
(p,q)	e	$\widehat{e}$	RAM	DoF	ML	time	cores	cluster
(2,1)	39.4%	39.1%	248 GB	23 887 872	10	0:06:48	256	MA-PDE
(2,2)	28.8%	28.8%	473 GB	35 831 808	10	0:13:38	256	MA-PDE
(2,3)	28.7%	28.7%	768 GB	47 775 744	10	0:22:55	256	MA-PDE
(3,1)	31.1%	30.9%	636 GB	42 467 328	16	0:29:54	256	MA-PDE
(3,2)	5.1%	5.1%	1.3 TB	63 700 992	15	1:02:34	256	MA-PDE
(3,3)	4.8%	4.8%	3.9 TB	84 934 656	17	0:10:44	2048	ForHLR2
dG-cPG on space-time mesh level 5								
(p,q)	e		RAM	DoF	ML	time	cores	cluster
(2,2)	2.5%		2.7 TB	191 102 976	24	0:10:21	2048	ForHLR2
(2,3)	2.7%		5.5 TB	286 654 464	22	0:21:07	2048	ForHLR2
(3,2)	0.6%		11.7 TB	509 607 936	38	0:35:47	4096	ForHLR2
dG-dG on space-time mesh level 5								
(p,q)	e	$\widehat{e}$	RAM	DoF	ML	time	cores	cluster
(2,1)	7.7%	7.6%	1.4 TB	191 102 976	17	1:09:41	256	MA-PDE
(2,2)	2.7%	2.7%	2.8 TB	286 654 464	17	1:56:44	192	ForHLR2
(3,1)	6.7%	6.6%	5.6 TB	339 738 624	29	0:25:40	2048	ForHLR2

Table 5.10: Marmousi II dG vs. cPG: comparison of the two methods on uniform discretizations. The error $e = \frac{\|\mathbf{s} - \mathbf{s}_{\text{ex}}\|_{(0,T)}}{\|\mathbf{s}_{\text{ex}}\|_{(0,T)}}$ is given in percent. The error of the seismogram obtained by evaluation of the conforming reconstruction $\widehat{e} = \frac{\|\widehat{\mathbf{s}} - \mathbf{s}_{\text{ex}}\|_{(0,T)}}{\|\mathbf{s}_{\text{ex}}\|_{(0,T)}}$ is additionally given in tha case of the dG-dG method. ML denotes the GMRES steps with the multilevel preconditioner. We use 10 smoothing steps if coarsened in time and 20 if coarsened in space. The time to solve the space-time system is given in [hh:mm:ss].

Not all computations are comparable because different clusters were used. The Intel processors of the ForHLR2 are faster than the asymmetrical MA-PDE with AMD central processing units. Also every single process has some code overhead which needs system memory. As a consequence the total system memory of the code run on 256 cores will need less total memory as the run on 4096 cores.

In Fig. 5.12 we illustrate the reconstruction of the operator working on linear ansatz functions in time and resulting in conforming quadratic function for the wave initiated by a Ricker wavelet.

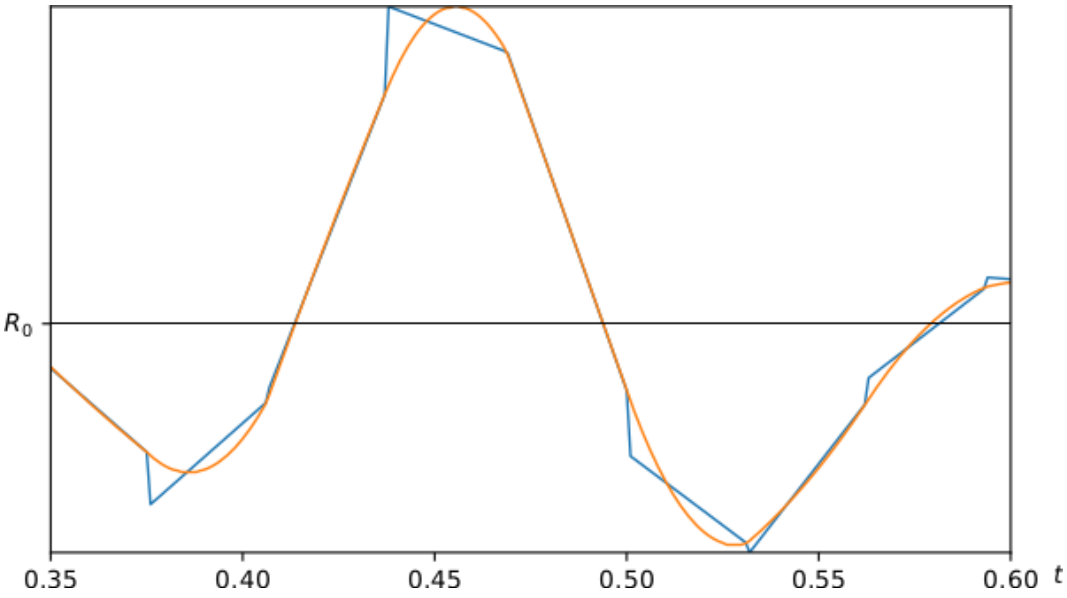

Figure 5.12: Sketch of the feature using conforming reconstruction: the solution discontinuous in time obtained by the dG(p)-dG(q) method with $(p, q) = (3, 1)$ (blue) is reconstructed with Radau IIA integration points (orange).

5.2.4 Visco-elastic adaptive computation on 8192 cores

This numerical test shows the capability of the code. The visco-elastic system with one damping mechanism ($G = 1$) is solved using one adaptive step and the dG-cPG method.

Here we choose the domain $\Omega = (4000, 13000) \times (-3000, 0) \subset (0, 17000) \times (-3500, 0)$ [m^2] (marked red in Fig. 5.2) and the time interval $(0, T)$ with $T = 3$ [s]. The source is located at $\mathbf{x}_\mathrm{s} = (7000, -250)$, and the receivers positions are $\mathbf{x}_{\mathrm{r},j} = (9000 + 125j, -250)$ for $j = 0, \ldots, 16$. For the adaptive simulations we use the goal functional

$$\mathcal{J}_\mathrm{elastic}(\mathbf{v}, \boldsymbol{\sigma}) = \frac{1}{|\Omega_\mathrm{RoI}|} \int\limits_{\Omega_\mathrm{RoI} \times \{T\}} \operatorname{trace} \boldsymbol{\sigma} \, \mathrm{d}\mathbf{x} \,, \qquad \boldsymbol{\sigma} = \boldsymbol{\sigma}_0 + \boldsymbol{\sigma}_1$$

together with the region of interest $\Omega_\mathrm{RoI} = (4750, 100) \times (7250, 400)$ (cf. Def. 4.2). We start with linear functions in space and time and solve the primal and dual problem. In all space-time cells where the error indicator η_R is greater than $\theta = 1e{-}9$ times the largest error indicator $\eta_\mathrm{max} = \max_{R \in \mathcal{R}} \eta_R$, i.e., $\eta_R > \eta_\mathrm{crit} = \theta \eta_\mathrm{max}$, the polynomial degree is increased in space and time. In contrast the polynomial degree is decreased if $\eta < 0.01 \cdot \eta_\mathrm{crit}$.

The visco-elastic adaptive space-time dG-cPG simulation tracks the propagation of the wave from the source to the receivers. The first stress component (column 1) and the distribution of the polynomial degrees (p, q) (column 2) are visualized in Fig. 5.13. In the blue area we have $(p, q) = (0, 1)$, gray $(p, q) = (1, 1)$ and red $(p, q) = (2, 2)$.

We have approximately 364 Mio. degrees of freedom and need 14 GMRES steps with the multilevel preconditioner presented in Sec. 4.4.2 (using 50 Gauss–Seidel smoothing steps in space and 25 Jacobi smoothing steps in time) for the solution of the full linear space-time system. The p-adaptive method reduces the degrees of freedom by approximately 78% compared to a uniform computation (1 968 Mio. degrees of freedom). On 4096 parallel processes the system was solved in 30 minutes and 53 seconds whereas on 8192 parallel processes the time was 15 minutes and 47 seconds. The solving time was cut nearly in half by doubling the number of processes demonstrating very good strong scaling behavior.

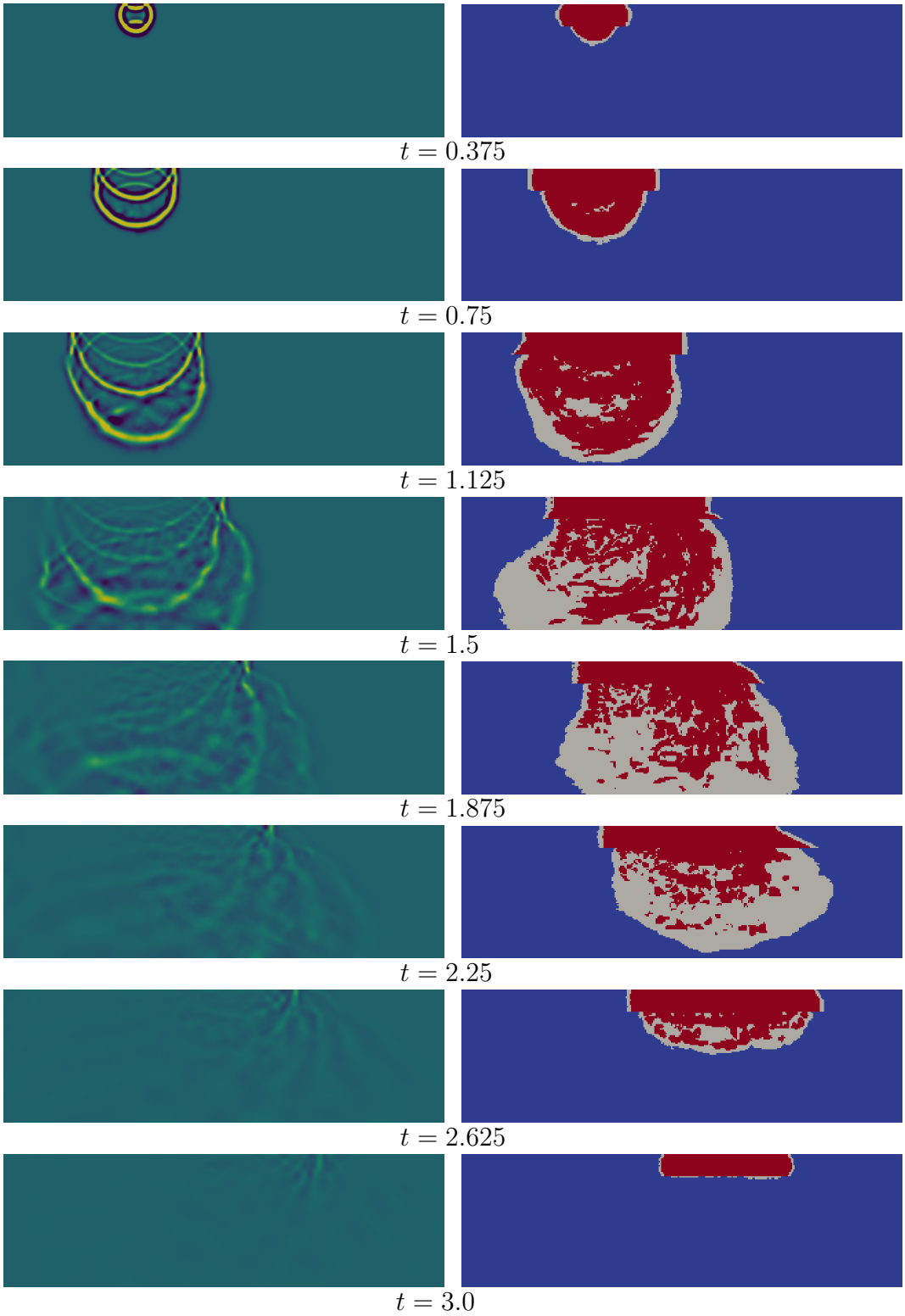

Figure 5.13: Slices through the space-time solution for the visco-elastic adaptive computation at different times. On the left is the first stress component and on the right the corresponding polynomial order in space and time.

CHAPTER

SIX

FINAL REMARKS

6.1 Conclusion

The main goal of this work consisted in developing a discretization for first order linear hyperbolic systems, where space and time are treated simultaneously in a variational manner. In particular we focused on (visco-)acoustic and (visco-)elastic waves.

We presented a space-time discretization with discontinuous ansatz functions in space and time (dG-dG). We proved existence and uniqueness of a discrete solution in case of tensor product space-time meshes for arbitrary polynomial degrees in space and time in each cell. A conforming reconstruction operator, working on the discrete solution, is introduced. For the case of constant polynomial degree in space p, the operator gives a solution, which is continuous in time.

An alternative discretization with discontinuous ansatz functions in space but continuous in time is additionally presented (dG-cPG). The inf-sup stability for this discretization had been proven only for polynomial degrees in space, which are fixed on the spatial mesh $\mathcal{K}$. We expect to generalize this proof for arbitrary polynomial degrees with the same techniques used in the proof for the dG-dG discretization.

We verified the theoretical results with numerical experiments. The simple benchmark experiment of an acoustic wave with analytical solution shows convergence for both methods of expected order. The conforming reconstruction

operator improves in a postprocessing step the convergence order of the dG-dG discretization, especially if the polynomial degree in time q is lower than the polynomial degree in space p, i.e., $p > q$.
The Marmousi II benchmark has heterogeneous material parameters. We compared the dG-dG discretization with the dG-cPG discretization. The dG-cPG method has smaller errors compared to the dG-dG method referring to the total amount of degrees of freedom in our examples. However, the dG-dG method is much faster and has less memory consumption. We show that the adaptive p-refinement allows to save a big part of the degrees of freedom. Finally, we show the capability of the numerical framework. We compute a visco-elastic wave with the adaptive algorithm. The work is distributed on 8192 computational cores showing the parallel scalability.
When we compared the time to solve the system with the two methods, we did not mention the time to assemble them. Since the dG-dG method has has less coupling between the space-time cells than the dG-cPG method, the assembly needs significant less time. For the final decision, which method performs better, additional research is necessary.

6.2 Future directions

Our parallel implementation should be prepared for exascale computing. The numerical test showed efficient scaling regarding the time to solve the linear system with several hundred million degrees of freedom. Also the time to assemble the system matrix scales with the number of computational cores. Some tasks are handled in serial, such as, e.g., the output of visualization data. The adaptive algorithm starts in every step with the initial guess zero. Here, the solution of the previous step should be used. Therefore, the load balancing module must be expanded, such that the degrees of freedom are hand over, preferably working in parallel.
The next step would be to implement the space-time discretizations in a matrix free version. Even using sparse matrices format results in enormous consumption of random access memory. Instead of storing the coefficient of the matrix explicitly, the access of the matrix are realized by evaluating matrix-vector products. The limitations of access to a large high performance cluster is a

disadvantage of space-time methods. Missing computational cores can be replaced by longer computation times, but necessary memory to store the system matrix is irreplaceable.

On further interest is the extension to three space dimensions. Although all components of our implementation support three space dimensions, the realization of matrix free methods should be the first step. This is based on the memory consumption of the system matrix.

One big challenge is still open for hyperbolic problems. [FFK+14] could show for parabolic problems, that a parallel in time and space multilevel solver can outperform the classical time stepping method. For two space dimensions several hundreds cores and for three dimensions thousands of computational cores in parallel were needed for this numerical experiments. This would be nice to obtain with our implementation. We could not verify this due to the lack of access to the necessary computational resources.

Up to now, we use structured space-time meshes of tensor product structure. This could be generalized to arbitrary triangulations in space-time. Also on part of the modeling aspect, the adaptation of the spatial mesh to the distribution of material parameters would improve the modeling error. Orienting cell faces on interfaces of different material parameters reduces additional artificial reflections.

We use the wave equations combined with homogeneous boundary conditions. In reality, the propagated waves are not reflected at the borders of the investigated domain. The implementation of transparent boundary conditions would solve this issue. This could be realized by a perfectly matched layer, an artificial absorbing layer, see [Sch15, Chap. 2].

The space-time discretizations presented within this work are designed to be applied to inverse problems such as seismic imaging. The use as forward solver for full waveform inversion is regarded.

APPENDIX

A

A.1 Integration formula

A quadrature rule is an approximation of the integral of a function stated as a weighted sum of function evaluations at specified points given on the reference interval [0,1] as

$$\int_0^1 f(t)\,\mathrm{d}t \approx \sum_{i=1}^{n} w_i f(t_i)\,.$$

Equally spaced points yield the so called Newton-Cotes formulas. These formulas can be transferred to general intervals (a, b). Typical examples are the midpoint rule

$$\int_a^b f(t)\,\mathrm{d}t \approx (b-a) f\left(\frac{a+b}{2}\right)$$

or the trapezoidal rule

$$\int_a^b f(t)\,\mathrm{d}t \approx (b-a)\frac{f(a)+f(b)}{2}\,.$$

If arbitrary integration points are allowed, the so called Gaussian quadrature formula results in more accurate integration. An overview of such integration rules are presented in Tab. A.1.

The Radau IIA quadrature rule with n integration points is exact for polynomials up to order $p = 2n-2$ and has only positive weights [But08, Thm. 344A]. The integration formulas in Tab. A.1 can be found in [DB02]. The integration points for higher order are given in Tab. A.1.

name	polynomial	root	order
Gauss	$L_n(2t-1)$	$t_i \in (0,1)$	$2n-1$
Radau IA	$L_n(2t-1) + L_{n-1}(2t-1)$	$t_i \in [0,1)\,,\ t_1 = 0$	$2n-2$
Radau IIA	$L_n(2t-1) - L_{n-1}(2t-1)$	$t_i \in (0,1]\,,\ t_n = 1$	$2n-2$
Lobatto	$L_n(2t-1) - L_{n-2}(2t-1)$	$t_i \in [0,1]\,,\ t_1 = 0\,,\ t_n = 1$	$2n-3$

Table A.1: Important Gauss quadrature rules defined by the polynomials where L_n denotes the n-th Legendre polynomial defined on the interval $(-1, 1)$, see [But08, p. 223].

t_i	1
w_i	1

t_i	1/3	1
w_i	3/4	1/4

t_i	$\frac{4-\sqrt{6}}{10}$	$\frac{4+\sqrt{6}}{10}$	1
w_i	$\frac{16-\sqrt{6}}{36}$	$\frac{16+\sqrt{6}}{36}$	$\frac{1}{9}$

Table A.2: Radau IIA quadrature with integration points t_i and weights w_i exact for polynomials of order $p_1 = 0$, $p_2 = 2$ and $p_3 = 4$.

int. pt.	roots
$n = 1$	1.000000000000000
$n = 2$	0.333333333333333
	1.000000000000000
$n = 3$	0.155051025721682
	0.644948974278318
	1.000000000000000
$n = 4$	0.088587959512704
	0.409466864440735
	0.787659461760847
	1.000000000000000
$n = 5$	0.057104196114518
	0.276843013638124
	0.583590432368917
	0.860240135656219
	1.000000000000000
$n = 6$	0.039809857051469
	0.198013417873608
	0.437974810247386
	0.695464273353636
	0.901464914201174
	1.000000000000000

Table A.3: Integration points of the Radau IIA quadrature rule computed approximately using a computer algebra system to solve the roots of corresponding polynomial, i.e., $0 = \partial_t^{n-1}\ t^{n-1}(t-1)^n$.

A.2 Specifications of computational resources

All numerical experiments were executed on one of the following clusters:

DELTA-Cluster MA-PDE The computational cluster MA-PDE is hosted by the Research Group 3: Scientific Computing in the Institute for Applied and Numerical Mathematics of the Department of Mathematics at KIT. It contains:

- 6 small nodes with 128 GB RAM and 32 cores:
  two AMD Opteron(TM) Processor 6274,
- 2 nodes with 512 GB RAM and 64 cores:
  four AMD Opteron(TM) Processor 6376,
- 2 fat nodes with 512 GB RAM and 64/128 cores:
  two AMD EPYC 7551 32-Core Processor supporting hyper-threading,
- 3 fast nodes with 128 GB RAM and 32/64 cores:
  one AMD Ryzen Threadripper 2990WX 32-Core Processor supporting hyper-threading,
- connecting network is an InfiniBand QDR Interconnect.

Forschungshochleistungsrechner ForHLR II The high-performance computer ForHLR II is hosted by the Steinbuch Centre for Computing at KIT. It contains:

- 5 login nodes with 256 GB RAM and 20 cores:
  2 Deca-Core Intel Xeon E5-2660 v3,
- 1152 thin nodes with 64 GB RAM and 20 cores:
  2 Deca-Core Intel Xeon E5-2660 v3 resulting in a top performance of 832 GFLOPS,
- 21 fat nodes with 4 NVIDIA GeForce GTX980 Ti graphics boards, 1TB RAM and 48 cores:
  4 12-core Intel Xeon E7-4830 v3,
- connecting network is an InfiniBand 4X EDR Interconnect.

bwUniCluster The cluster computer bwUniCluster is hosted by the Steinbuch Centre for Computing at KIT and gives basic supply of computational resources for all universities in Baden-Wuerttemberg. It contains:

- 2 login nodes with 64 GB RAM and 16 cores:
  2 Octa-Core Intel Xeon E5-2670,
- 2 login nodes with 128 GB RAM and 20 cores:
  2 10-Core Intel Xeon E5-2630 v4,
- 512 thin nodes with 64 GB RAM and 16 cores:
  2 Octa-Core Intel Xeon E5-2670,
- 352 thin nodes with 128 GB RAM and 28 cores:
  2 14-Core Intel Xeon E5-2660 v4,
- 8 fat nodes with 1TB RAM and 32 cores:
  4 Octa-Core Intel Xeon E5-4640,
- connecting network is an InfiniBand 4X FDR Interconnect.

BIBLIOGRAPHY

[BKRS18] M. Bause, U. Köcher, F.A. Radu, and F. Schieweck. Post-processed Galerkin approximation of improved order for wave equations. *arXiv preprint arXiv:1803.03005*, 2018.

[BR96] R. Becker and R. Rannacher. A feed-back approach to error control in finite element methods: Basic analysis and examples. *East-West J. Numer. Math*, 4:237–264, 1996.

[Bra13] D. Braess. *Finite Elemente : Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie*. Springer, Berlin, 5th edition, 2013.

[BRS95] J. Blanch, J. Robertsson, and W. Symes. Modeling of a constant Q: Methodology and algorithm for an efficient and optimally inexpensive viscoelastic technique. *GEOPHYSICS*, 60(1):176–184, 1995.

[BSK81] I. Babuska, B. Szabo, and I. Katz. The p-version of the finite element method. *SIAM Journal on Numerical Analysis*, 18(3):515–545, 1981.

[But08] J.C. Butcher. *Numerical Methods for Ordinary Differential Equations*. John Wiley & Sons, second edition, 2008.

[Cia02] P. Ciarlet. *The Finite Element Method for Elliptic Problems*. Society for Industrial and Applied Mathematics, 2002.

[DB02] P. Deuflhard and F. Bornemann. *Numerische Mathematik II - Gewöhnliche Differentialgleichungen*. De Gruyter, Berlin, 2. vollständig überarbeitete und erweiterte edition, 2002.

[DFW16] W. Dörfler, S. Findeisen, and C. Wieners. Space-time discontinuous Galerkin discretizations for linear first-order hyperbolic evolution systems. *Comput. Methods Appl. Math.*, 16(3):409–428, 2016.

[DFWZ19] W. Dörfler, S. Findeisen, C. Wieners, and D. Ziegler. Parallel adaptive discontinuous Galerkin discretizations in space and time for linear elastic and acoustic waves. In U. Langer and O. Steinbach, editors, *Space-Time Methods. Applications to Partial Differential Equations*, volume 25 of *Radon Series on Computational and Applied Mathematics*, pages 61–88. Walter de Gruyter, 2019.

[DGNS17] L.F. Demkowicz, J. Gopalakrishnan, S. Nagaraj, and P. Sepulveda. A spacetime DPG method for the Schrödinger equation. *SIAM J. Numer. Anal.*, 55(4):1740–1759, 2017.

[Dör96] W. Dörfler. A convergent adaptive algorithm for Poisson's equation. *SIAM Journal on Numerical Analysis*, 33(3):1106–1124, 1996.

[EKSW15] H. Egger, F. Kretzschmar, S. M. Schnepp, and T. Weiland. A space-time discontinuous Galerkin–Trefftz method for time dependent Maxwell's equations. *SIAM J. Sci. Comput.*, 37(5):B689–B711, 2015.

[Ern18] J. Ernesti. *Space-Time Methods for Acoustic Waves with Applications to Full Waveform Inversion*. PhD thesis, Karlsruher Institut für Technologie (KIT), 2018.

[Eva10] L.C. Evans. *Partial differential equations, 2. ed.* American Mathematical Society, Providence, RI, 2010.

[EW19] J. Ernesti and C. Wieners. Space-time discontinuous Petrov-Galerkin methods for linear wave equations in heterogeneous media. Technical Report 13, Karlsruher Institut für Technologie (KIT), 2019.

[FFK+14] R.D. Falgout, S. Friedhoff, T.V. Kolev, S.P. MacLachlan, and J.B. Schroder. Parallel time integration with multigrid. *SIAM J. Sci. Comput.*, 36(6):C635–C661, 2014.

[Fic11] A. Fichtner. *Full Seismic Waveform Modelling and Inversion*. Advances in Geophysical and Environmental Mechanics and Mathematics. Springer-Verlag Berlin Heidelberg, 2011.

[Fin16] S.M. Findeisen. *A Parallel and Adaptive Space-Time Method for Maxwell's Equations*. PhD thesis, Karlsruher Institut für Technologie (KIT), 2016.

[FOGG17] G. Fabien-Ouellet, E. Gloaguen, and B. Giroux. Time domain viscoelastic full waveform inversion. *Geophysical Journal International*, 209:1718–1734, 03 2017.

[Gan15] M.J. Gander. 50 years of time parallel time integration. In T. Carraro, M. Geiger, S. Körkel, and R. Rannacher, editors, *Multiple Shooting and Time Domain Decomposition*, pages 69–113. Springer, 2015.

[GKSZ18] F. Gaspoz, C. Kreuzer, K. Siebert, and D. Ziegler. A convergent time-space adaptive dG(s) finite element method for parabolic problems motivated by equal error distribution. *IMA Journal of Numerical Analysis*, pages 1–37, 2018.

[GS19] J. Gopalakrishnan and P. Sepulveda. A space-time DPG method for the wave equation in multiple dimensions. In U. Langer and O. Steinbach, editors, *Space-Time Methods. Applications to Partial Differential Equations*, volume 25 of *Radon Series on Computational and Applied Mathematics*, pages 117–140. Walter de Gruyter, 2019.

[GSW17] J. Gopalakrishnan, J. Schöberl, and C. Wintersteiger. Mapped tent pitching schemes for hyperbolic systems. *SIAM Journal on Scientific Computing*, 39(6):B1043–B1063, 2017.

[HPS+15] M. Hochbruck, T. Pazur, A. Schulz, E. Thawinan, and C. Wieners. Efficient time integration for discontinuous Galerkin approximations of linear wave equations. *ZAMM Z. Angew. Math. Mech.*, 95:237–259, 2015.

[Huy09] H.T. Huynh. Collocation and Galerkin time-stepping methods. In *19th AIAA Computational Fluid Dynamics*, 2009.

[HW08] J.S. Hesthaven and T. Warburton. *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications.* Texts in applied mathematics, 54. Springer, New York, NY, 2008.

[KB14] U. Köcher and M. Bause. Variational space-time methods for the wave equation. *J. Sci. Comput.*, 61(2):424–453, 2014.

[KMPS15] F. Kretzschmar, A. Moiola, I. Perugia, and S.M. Schnepp. A priori error analysis of space-time Trefftz discontinuous Galerkin methods for wave problems. *IMA J. Numer. Anal.*, pages 1599–1635, 2015.

[Kur12] A. Kurzmann. *Applications of 2D and 3D full waveform tomography in acoustic and viscoacoustic complex media.* PhD thesis, Karlsruhe Institute of Technology, 2012.

[LeV02] R.J. LeVeque. Finite volume methods for hyperbolic problems. *Meccanica*, 39:88–89, 2002.

[LMN16] U. Langer, S. E Moore, and M. Neumüller. Space-time isogeometric analysis of parabolic evolution problems. *Comput. Methods Appl. Mech. Engrg.*, 306:342–363, 2016.

[Mit89] W.F. Mitchell. A comparison of adaptive refinement techniques for elliptic problems. *ACM Trans. Math. Softw.*, 15(4):326–347, December 1989.

[MM14] W.F. Mitchell and M.A. McClain. A comparison of hp-adaptive strategies for elliptic partial differential equations. *ACM Trans. Math. Softw.*, 41(1):2:1–2:39, October 2014.

[MN06] C. Makridakis and R.H. Nochetto. A posteriori error analysis for higher order dissipative methods for evolution problems. *Numerische Mathematik*, 104:489–514, 2006.

[MP18] A. Moiola and I. Perugia. A space-time Trefftz discontinuous Galerkin method for the acoustic wave equation in first-order formulation. *Numerische Mathematik*, 138(2):389–435, 2018.

[MS16] F. Müller and C. Schwab. Finite elements with mesh refinement for elastic wave propagation in polygons. *Math. Meth. Appl. Sci.*, 39(17):5027–5042, 2016.

[MW11] D. Maurer and C. Wieners. A parallel block LU decomposition method for distributed finite element matrices. *Parallel Comput.*, 37(12):742–758, 2011.

[MW16] D. Maurer and C. Wieners. A scalable parallel factorization of finite element matrices with distributed Schur complements. *Numer. Linear Algebra Appl.*, 23(5):848–864, 2016.

[MWM06] G.S. Martin, R. Wiley, and K.J. Marfurt. Marmousi2: An elastic upgrade for Marmousi. *The Leading Edge*, 25(2):156–166, 2006.

[NSV09] R.H. Nochetto, K.G. Siebert, and A. Veeser. Theory of adaptive finite element methods: An introduction. In Ronald DeVore and Angela Kunoth, editors, *Multiscale, Nonlinear and Adaptive Approximation*, pages 409–542, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[NSW17] R. Nochetto, S. Sauter, and C. Wieners. Space-time methods for time-dependent partial differential equations, 2017. Mathematisches Forschungsinstitut Oberwolfach, Report No. 15/2017.

[Sch15] A. Schulz. *Numerical Analysis of the Electro-Magnetic Perfectly Matched Layer in a Discontinuous Galerkin Discretization.* PhD thesis, Karlsruher Institut für Technologie (KIT), 2015. Karlsruhe, KIT, Diss., 2015.

[Ste15] O. Steinbach. Space-time finite element methods for parabolic problems. *Comput. Methods Appl. Math.*, 15(4):551–566, 2015.

[Sub] Subversion. M++. `https://svn.math.kit.edu/svn/M++/SummerSchool`.

[Ver94] R. Versteeg. The Marmousi experience: Velocity model determination on a synthetic complex data set. *The Leading Edge*, 13(9):927–936, 1994.

[VLN+18] K. Voronin, C.S. Lee, M. Neumüller, P. Sepulveda, and P.S. Vassilevski. Space-time discretizations using constrained first-order system least squares (CFOSLS). *Journal of Computational Physics*, 373:863 – 876, 2018.

[Wie10] C. Wieners. A geometric data structure for parallel finite elements and the application to multigrid methods with block smoothing. *Computing and visualization in science*, 13(4):161–175, 2010.

[Zel19] U.C. Zeltmann. *The Viscoelastic Seismic Model: Existence, Uniqueness and Differentiability with Respect to Parameters*. PhD thesis, Karlsruher Institut für Technologie (KIT), 2019.

DANKSAGUNG

Zum Schluss möchte ich mich bei allen Personen bedanken, die mir die Erstellung meiner Dissertation ermöglicht haben.

Zu besonderem Dank bin ich meinem Doktorvater Prof. Dr. Christian Wieners verpflichtet, der mein Potential erkannt hat und mir die Möglichkeit gab am Karlsruher Institut für Technologie meine Dissertation anzufertigen. Er stand nicht nur mit seiner fachlichen Kompetenz zur Seite, sondern auch in schwerwiegenden Fällen von Debugging. Des Weiteren bedanke ich mich bei Prof. Dr. Willy Dörfler für die wertvolle Betreuung bei meiner Forschungsarbeit. Doch auch Prof. Dr. Andreas Rieder bin ich für sein drittes Gutachten zu Dank verpflichtet.

Meinen wissenschaftlichen Kolleginnen und Kollegen vom IANM3 danke ich für die (immer zielführenden) Diskussionen und die Stressbewältigung am Tischkicker.

Besonderer Dank gilt meiner Familie: meinen Eltern Karl und Carmen, die es mir ermöglichten zu studieren und mein Vorhaben in Karlsruhe unterstützt haben, sowie meinen Geschwistern Edward und Silke, die mir abseits der Mathematik stets zur Seite stehen.

Zum Schluss möchte ich mich noch bei der wichtigsten Person bedanken: meine Freundin Carina. Ohne ihre Unterstützung hätte ich diese Arbeit nicht anfertigen können.

Daniel A. Ziegler