FAKULTÄT FÜR MATHEMATIK

DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Bachelorarbeit in Mathematik

Runge-Kutta and ADER discontinuous Galerkin schemes for hyperbolic partial differential equations.

Runge-Kutta und ADER discontinuous Galerkin Verfahren zur Lösung hyperbolischer Differentialgleichungen.

Autor:	Leonhard Andreas Rannabauer
Aufgabensteller:	Prof. Dr. Hans-Joachim Bungartz
Betreuer:	Dipl.-Math. Alexander Breuer
Datum:	7. August 2014

FAKULTÄT FÜR MATHEMATIK

DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Bachelorarbeit in Mathematik

Runge-Kutta and ADER discontinuous Galerkin schemes for hyperbolic partial differential equations.

Runge-Kutta und ADER discontinuous Galerkin Verfahren zur Lösung hyperbolischer Differentialgleichungen.

Autor:	Leonhard Andreas Rannabauer
Aufgabensteller:	Prof. Dr. Hans-Joachim Bungartz
Betreuer:	Dipl.-Math. Alexander Breuer
Datum:	7. August 2014

Ich versichere, dass ich diese Bachelorarbeit selbständig verfasst und nur die angegebenen
Quellen und Hilfsmittel verwendet habe.

München, den 7. August 2014 Leonhard Andreas Rannabauer

Abstract

Additionally to the widely used Finite volume and continuous Galerkin methods, discontinuous Galerkin methods provide a set of tools to numerically solve partial differential equations. In advantage to finite volume methods they offer higher-order accuracy. Compared to continuous Galerkin methods mass and stiffness matrices can be kept small, due to their local definition, where single elements only communicate with their direct neighbors. A common way to solve the time integrator of discontinuous Galerkin methods is to use Runge-Kutta solvers for ordinary differential equations. The order of convergence of these solvers is bound through the Butcher barrier, making a higher order convergence hardly realizable. To allow higher orders of convergence an alternative approach has been proposed, translating the principle of using the weak form of a partial differential equation, from the space dimension to the time dimension additionally. The resulting discontinuous Galerkin time predictor schema is similar to the semi discrete discontinuous Galerkin schema and provides a new set of mass, stiffness and flux matrices. In this thesis I will give an introduction to the ADER approach and compare it to the well known finite volume and the discontinuous Galerkin methods with Runge-Kutta time stepping and discuss the sparsity patterns of the matrices of these methods.

Contents

1 Introduction

The main topic of this thesis is solving hyperbolic partial differential equations through discontinuous Galerkin methods. The importance of partial differential equations in physics and other sciences is undeniable. In fluid mechanics, for example, the well known Navier-Stokes equations model the stream of Newtonian fluids and gases. Finding an explicit solution to these equations has been declared one of the most essential tasks for current mathematics.

For equations where the solution is not known, there's the need to solve them numerically instead.

The widely known finite volume methods [6] are a reliable way to approximate the solutions. By dividing the domain of the problem in distinct subdomains and approximating the average of the solution over this subdomains, the numeric solution mostly converges to the real solution (In fact the theorem of Lax and Wendroff states that the method always converges to a weak solution, which we will introduce in this thesis). A disadvantage of finite volume methods is that they only converge linearly to the solution.

To find methods that converge in higher order is the motivation of continuous and discontinuous Galerkin finite element methods, as introduced in [5] and [10].

After a general introduction to hyperbolic partial differential equations in section 2, we will develop the theory behind discontinuous Galerkin methods in section 3. The resulting time integral will be solved by Runge-Kutta methods.

As these methods are naturally bound to the Butcher barrier, which makes orders of convergence higher than 4 expensive to realize, we will regard an alternative approach, the ADER time prediction, proposed in [3], in chapter 4.

The ADER time prediction, develops a less expensive way to reach high orders of convergence than the Runge-Kutta methods.

While we acquire these theories we will encounter the known numeric problem of polynomial interpolation for the choice of nodal basis functions, which will be discussed in chapter 5.

As we additionally wish to implement the methods on high performance at state-of-the-art hardware, the sparsity patterns of the underlying mass, stiffness and flux matrices are analyzed in chapter 5.

In the final chapter 6 an analysis of the assumptions we made on the convergence order is performed by running an error analysis for the different methods.

2 Hyperbolic partial differential equations

In this thesis we discuss two related numerical methods seeking for a approximate solution to hyperbolic partial differential equations.

A general definition of a non-linear partial differential equation in N space dimensions of first order is given by,

$$\frac{\partial}{\partial t} q(\vec{x}, t) + \sum_{i=1}^{N} \frac{\partial}{\partial x_i} f_i(q) = S(q), \tag{1}$$

defined on a domain $\Omega \in \mathbb{R}^n$ with boundary $\delta\Omega$. $\vec{x}$ denotes the vector of all variables in space x_i, t is the variable in time. The solution q, the flux-terms f_i as well as the source term S are M dimensional vectors, i.e. the number of physical quantities.

To obtain a general overview of partial differential equations (PDEs), we will take a look at two different examples. At first the advection equation, which states a very simple problem and second the Shallow-Water equations which are hyperbolic non-linear differential equations and allow the calculation of explicit solutions.

The advection equation states an linear scalar PDE which is denoted by

$$\frac{\partial}{\partial t} q + \frac{\partial}{\partial x} (uq) = 0, \tag{2}$$

where $u \in \mathbb{R}$ is called the wave-speed and $q : I \times \mathbb{R}^+ \to R$ for some interval $I = [a, b] \subset \mathbb{R}$.

To observe a PDE we take a look at the characteristic curves of the equation, which are generally defined in one dimension by

$$X'(t) = f'(X(t), t), \tag{3}$$

with f' being the derivative of the flux.

Along characteristic curves the solution is equal to the source term

$$\frac{\partial}{\partial t} q\left(X(t), t\right) = X'(t) \frac{\partial}{\partial x} q + \frac{\partial}{\partial t} q = S, \tag{4}$$

thus for $S = 0$, the solution is constant along the characteristics in time.

By looking at the characteristic functions we have a view on how the solution will evolve in time.

For the problem described in equation (2) we obtain

$$X'(t) = u \Rightarrow X(t) = t \cdot u + c, \tag{5}$$

with $c \in I$.

As we see, the solution stays constant along lines in time.

With a initial condition at time zero $q(x, 0) = \mathring{q}(x)$, assuming periodic boundaries i.e $q(a, t) = q(b, t)$ and assuming a positive wave speed $u > 0$, we can describe the solution recursively by

$$q(x, t) = \begin{cases} q\left(b, t - (x - a)/u\right) & \text{, for } t \cdot u > x \\ \mathring{q}\left(x - (t \cdot u)\right) & \text{, for } t \cdot u < x \end{cases} \tag{6}$$

In figure 1 is a plot of a Gaussian normal distribution propagating at $u = 0.05$ with periodic boundaries. Due to it's continuity we will use this solution in the convergence analysis in chapter 6.

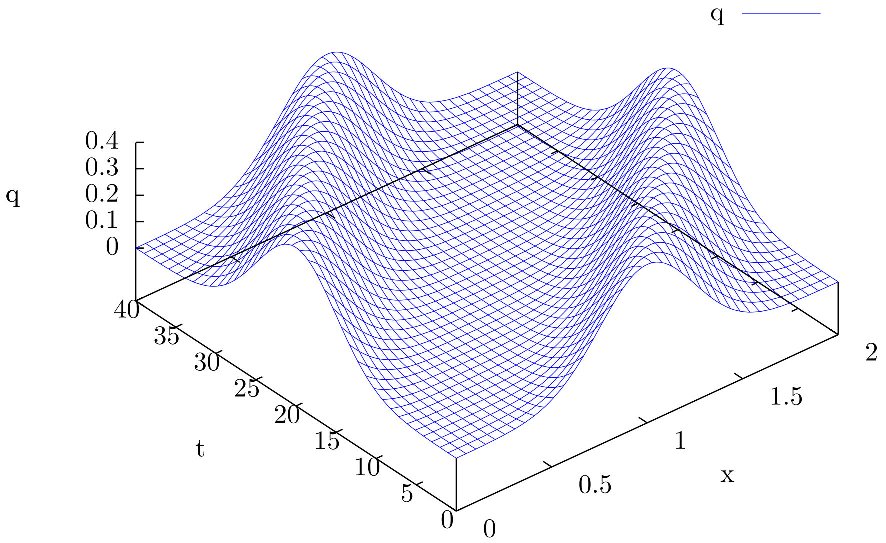


Figure 1: Explicit solution to the partial differential equation (2) with a Gaussian normal distribution as initial distribution

As second set of partial differential equations we're looking at the Shallow Water equations. An introduction to the physical background is given in [6, p. 253–259].

The equations are given by the term

$$\frac{\partial}{\partial t}q(x,t) + \frac{\partial}{\partial x}f(q) = 0, \tag{7}$$

with

$$q(x,t) = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix}, \text{ and } f(q) = \begin{pmatrix} q_2 \\ \frac{q_2^2}{q_1} + \frac{1}{2}g \cdot q_1^2 \end{pmatrix}, \tag{8}$$

where g is the gravitational constant and $q_1 \geq 0$.

By looking at the Jacobian matrix of the flux term we can observe two properties of the PDE:

$$f'(q) = \begin{pmatrix} 0 & 1 \\ -\left(\frac{q_2}{q_1}\right) + gq_1 & 2\frac{q_2}{q_1} \end{pmatrix}. \tag{9}$$

As the derived flux-term still depends on q the PDE is called non-linear. PDEs where the derived flux term is diagonalizable with distinct real eigenvalues are called hyperbolic.

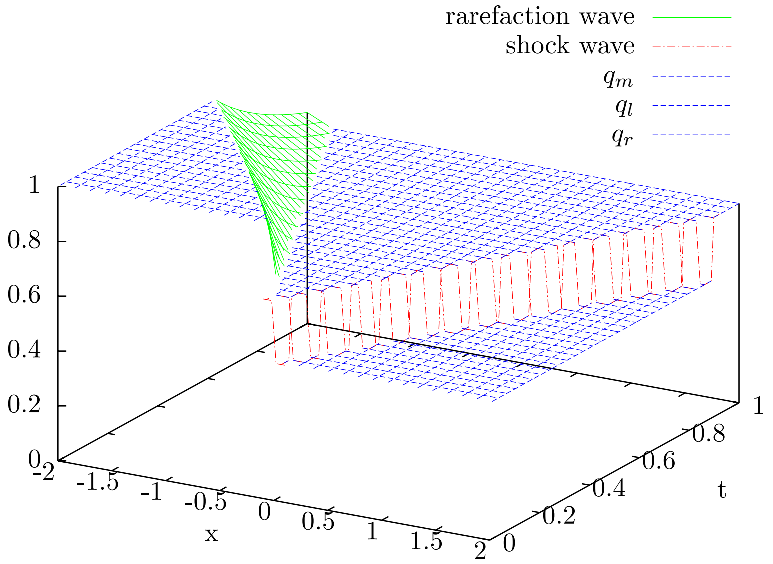


Figure 2: Solution to the dam break problem for the initial values $q_l = 1$ and $q_r = 0.5$

For the derived flux in equation (9) the eigenvalues are

$$\begin{aligned} \lambda^1 &= \frac{q_2}{q_1} - \sqrt{gq_1} \\ \lambda^2 &= \frac{q_2}{q_1} + \sqrt{gq_1} \end{aligned} \tag{10}$$

By defining the initial conditions

$$\begin{aligned} q_1 &= \begin{cases} q_l & x \leq 0 \\ q_r & x > 0 \end{cases} \\ q_2 &= 0, \end{aligned} \tag{11}$$

with $q_l > q_r$ we state the dam-break problem. A PDE together with piecewise constant initial conditions as in equation (11) are called Riemann problems.

How one can obtain the solution to a Riemann problem for nonlinear hyperbolic PDE is explained in detail in [6, p. 263–290].

q_1 of the solution for explicit values for q_l and q_r, as evaluated in [6, p. 279] is plotted in figure 2. In this example solution we can see a few characteristic properties of Riemann problems for hyperbolic non-linear equations:

- The solution is always constant along rays $\frac{x}{t}$.
- For $\frac{x}{t} < \lambda^1(q_l)$ the solution remains in the initial value q_l and behaves equivalently with $\frac{x}{t} > \lambda^2(q_r)$ and q_r.

- Along the ray $\frac{x}{t} = \lambda^2(q_r)$ we obtain a discontinuity, called shock wave.
- A new intermediate constant state q_m evolves between $\lambda_1(q_m)$ and $\lambda_2(q_r)$.
- Between $\lambda^1(q_l)$ and $\lambda^1(q_m)$ we see a rarefaction wave propagating in time.

In fact, there's a third type of wave, the contact discontinuity described in [6, p. 301]. Solutions to Riemann problems for hyperbolic non-linear equations with M eigenvalues have M waves of one of the tree types. The key to the solution itself is the calculation of the state q_m which has to fulfill certain equations when lying between the rarefaction and the shock wave.

3 The Discontinuous Galerkin method

To obtain a general formulation of the discontinuous Galerkin method we examine the general non-linear partial differential equation (1) of first order of chapter 2,

$$\frac{\partial}{\partial t} q(x,t) + \sum_{i=1}^{N} \frac{\partial}{\partial x_i} f_i(q) = S(q). \tag{12}$$

The discontinuous Galerkin method seeks for a solution piecewise defined on a disjunct set of subdomains $\Omega^{(e)}$ of Ω,

$$\Omega = \bigcup_{e \in \{1,..,m\}} \Omega^{(e)}. \tag{13}$$

The choice of how the domain is divided is arbitrary, for example an often used method in three space dimensions is the division by tetrahedrons as explained in [5, p. 409–418]. For reasons of simplicity in this thesis we will define the local subdomains as N dimensional rectangles,

$$\Omega^{(e)} = \bigotimes_{d \in \{1,\ldots,N\}} [a_d^{(e)}, b_d^{(e)}], \text{ with } a_d^{(e)}, b_d^{(e)} \in \mathbb{R}. \tag{14}$$

The global solution q is then the direct sum of the local solutions $q^{(e)}$ defined on $\Omega^{(e)}$,

$$q(\vec{x},t) = \bigoplus_{e \in \{1,..,m\}} q^{(e)}(\vec{x},t). \tag{15}$$

For a locally defined solution $q^{(e)}$ the differential equation (12) still holds.

The general approach in finite element methods, as defined in [10, p. 40], is to introduce the weak integral form of the partial differential equation on the elements,

$$\int_{\Omega^{(e)}} \phi_j \left(\frac{\partial}{\partial t} q^{(e)} + \sum_{i=1}^{N} \frac{\partial}{\partial x_i} f_i\left(q^{(e)}\right) - S\left(q^{(e)}\right) \right) d\Omega^{(e)} = 0, \, \forall j = 1 \ldots N_\phi, \tag{16}$$

for a set of N_ϕ test functions ϕ_j.

By using the divergence theorem on the flux summands of equation (16), we obtain

$$\begin{aligned} \int_{\Omega^{(e)}} \phi_j \frac{\partial}{\partial x_i} f_i\left(q^{(e)}\right) d\Omega^{(e)} &= \\ \int_{\delta\Omega^{(e)}} \phi_j f_i\left(q^{(-,+)}\right) \vec{n} \, d\delta\Omega^{(e)} - \int_{\Omega^{(e)}} \frac{\partial}{\partial x_i} \phi_j f_i\left(q^{(e)}\right) d\Omega^{(e)} &= \\ \sum_{k=1}^{N} \int_{\Omega_k^{(e)}} \left[\phi_j f_i\left(q^{(-,+)}\right) \vec{n}\right]_{x_k=a_k}^{x_k=b_k} d\Omega_k^{(e)} - \int_{\Omega^{(e)}} \frac{\partial}{\partial x_i} \phi_j f_i\left(q^{(e)}\right) d\Omega^{(e)}&, \end{aligned} \tag{17}$$

with $\vec{n}$ being the normal vector pointing outside $\delta\Omega^{(e)}$, and $\Omega_k^{(e)}$ being the $N-1$ dimensional face at a position x_k in the space dimension k. For example, in three dimensions we receive for $\Omega_3^{(e)}$ a rectangle spanned in the first and second dimension. As the solution q is only defined element wise, we don't know the explicit function on the boundary yet. We will denote this function by $q^{(-,+)}$ for now, and solve the problem later on.

Equation (16) becomes:

$$\begin{aligned}\int_{\Omega^{(e)}} \phi_j \left(\frac{\partial}{\partial t} q^{(e)}\right) d\Omega^{(e)} \\ + \sum_{i=1}^{N} \left(\sum_{k=1}^{N} \int_{\Omega_k^{(e)}} \left[\phi_j f_i \left(q^{(-,+)} \right) \vec{n} \right]_{x_k=a_k}^{x_k=b_k} d\Omega_k^{(e)} - \int_{\Omega^{(e)}} \frac{\partial}{\partial x_i} \phi_j f_i \left(q^{(e)} \right) d\Omega^{(e)} \right) \\ - \int_{\Omega^{(e)}} \phi_j S \left(q^{(e)} \right) d\Omega^{(e)} = 0, \forall j = 1 \ldots N_\phi\end{aligned} \tag{18}$$

The local solutions $q^{(e)}$ are now approximated in space by a set of basis functions $\psi_i^{(e)}$ with their coefficients $\hat{q}_i^{(e)}$, called degrees of freedom (DOFs),

$$q^{(e)}(\vec{x},t) \approx q_h^{(e)}(\vec{x},t) = \sum_{i=1}^{N_p} \psi_i^{(e)}(\vec{x}) \hat{q}_i^{(e)}(t). \tag{19}$$

The choice of this basis functions and the corresponding evaluation of the degrees of freedom is discussed in chapter 5. In this thesis we will use a nodal basis, which allows an simple calculation of the flux and source terms.

To keep the notation short we will denote the approximation in vector-vector notation:

$$q_h^{(e)} = \vec{\boldsymbol{\psi}}^{(e)T} \cdot \vec{\mathbf{q}}^{(e)}, \tag{20}$$

with

$$\vec{\boldsymbol{\psi}}^{(e)} = \begin{pmatrix} \psi_1^{(e)} \\ \psi_2^{(e)} \\ \cdots \\ \psi_{N_p}^{(e)} \end{pmatrix} \quad \vec{\mathbf{q}}^{(e)} = \begin{pmatrix} \hat{q}_1^{(e)} \\ \hat{q}_2^{(e)} \\ \cdots \\ \hat{q}_{N_p}^{(e)} \end{pmatrix}. \tag{21}$$

Due to the nodal basis the approximation of the flux and source terms in equation (18) can be transformed in vector-vector notation too:

$$f_i(\vec{\boldsymbol{\psi}}^{(e)T} \cdot \vec{\mathbf{q}}^{(e)}) \approx \vec{\boldsymbol{\psi}}^{(e)T} \cdot \vec{\mathbf{f}}_i^{(e)} \; S(\vec{\boldsymbol{\psi}}^{(e)T} \cdot \vec{\mathbf{q}}^{(e)}) \approx \vec{\boldsymbol{\psi}}^{(e)T} \cdot \vec{\mathbf{S}}^{(e)}, \tag{22}$$

with

$$\vec{\mathbf{f}}_i^{(e)} = \begin{pmatrix} f_i(\hat{q}_1^{(e)}) \\ \cdots \\ f_i(\hat{q}_{N_p}^{(e)}) \end{pmatrix} \vec{\mathbf{S}}^{(e)} = \begin{pmatrix} S(\hat{q}_1^{(e)}) \\ \cdots \\ S(\hat{q}_{N_p}^{(e)}) \end{pmatrix}. \tag{23}$$

Equation (18) with the vector-vector notation and the fact that the DOFs are space independent yields

$$
\begin{aligned}
&\int_{\Omega^{(e)}} \phi_j \cdot \vec{\boldsymbol{\psi}}^{(e)T} d\Omega^{(e)} \cdot \frac{\partial}{\partial t}\vec{\mathbf{q}}^{(e)} \\
&+\sum_{i=1}^{N}\left(\sum_{k=1}^{N}\int_{\Omega_k^{(e)}}\left[\phi_j f_i\left(q^{(-,+)}\right)\overrightarrow{n}\right]_{x_k=a_k}^{x_k=b_k} d\Omega_k^{(e)} - \int_{\Omega^{(e)}}\frac{\partial}{\partial x_i}\phi_j \cdot \vec{\boldsymbol{\psi}}^{(e)T} d\Omega^{(e)} \cdot \vec{\mathbf{f}}_i^{(e)}\right) \\
&- \int_{\Omega^{(e)}} \phi_j \cdot \vec{\boldsymbol{\psi}}^{(e)T} d\Omega^{(e)} \cdot \vec{\mathbf{S}}^{(e)} = 0, \\
&\forall j = 1 \dots N_\phi
\end{aligned}
\tag{24}
$$

By using the transformation theorem, and mapping the integrals of equation (24) on a reference element $\zeta = [0,1]^N$, we can simplify the calculation.

A mapping from any integration domain $\Omega^{(e)} = \bigotimes_{d\in\{1,\dots,N\}}[a_d, b_d]$ to the reference element is given by

$$
\Phi^{(e)}(\vec{x}) = \mathbf{A}^{(e)}(\vec{x} - \vec{a}^{(e)}) \tag{25}
$$

With

$$
\mathbf{A}^{(e)} = \begin{pmatrix} (b_1-a_1)^{(-1)} & 0 & \dots & 0 \\ 0 & (b_2-a_2)^{(-1)} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & (b_N-a_N)^{(-1)} \end{pmatrix} \quad \vec{a}^{(e)} = \begin{pmatrix} a_1 \\ \vdots \\ a_N \end{pmatrix} \tag{26}
$$

The transformation constant is then defined as

$$
\left|\det D_x \Phi^{(e)}\right| = \left|\det A^{(e)}\right| = \prod_{i\in\{1,\dots N\}} (b_i - a_i)^{-1} =: T^{(e)} \tag{27}
$$

In discontinuous Galerkin methods the set of basis functions are used as test functions, i.e. $\phi_j = \psi_j$. As equation (24) has to hold for all $j \in \{1,\dots,N_p\}$ we can define the mass and stiffness matrices in (28) and directly transform them on the reference element.

$$
\begin{aligned}
\frac{1}{T^{(e)}} \cdot \mathbf{M} &:= \frac{1}{T^{(e)}} \cdot \int_\zeta \psi \cdot \psi^T \, d\zeta = \int_{\Omega^{(e)}} \psi^{(e)} \cdot \psi^{(e)T} d\Omega^{(e)} \\
\mathbf{D}_i &:= \int_\zeta \psi \cdot \frac{\partial}{\partial x_i}\psi^T \, d\zeta = \frac{1}{T^{(e)}} \int_\zeta \psi \cdot \frac{\partial}{\partial x_i}\psi^T \cdot T^{(e)} \, d\zeta \\
&= \int_{\Omega^{(e)}} \psi^{(e)} \cdot \frac{\partial}{\partial x_i}\psi^{(e)T} d\Omega^{(e)}
\end{aligned}
\tag{28}
$$

Note that the basis functions now only have to be defined on the reference element.

The unknown function $q^{(-,+)}$ along a face Ω_k with $x_k = b$ is the solution to a generalized Riemann problem as defined in [7, p. 625–644]. The problem is stated by the PDE (12) and the initial condition:

$$
q^{(-,+)}(\vec{x}) = \begin{cases} q^{(-)} & \text{, for } x_k < b \\ q^{(+)} & \text{, for } x_k > b \end{cases} \tag{29}
$$

In fact the naming discontinuous Galerkin is deviated from the disconuities we're obtaining along the boundaries.

$f_i(q^{(-,+)})$, the flux along the solution of the generalized Riemann problem, is approximated by a numerical flux. We will use the local Lax-Friedrichs method, which is also called Rusanov's method [6, p. 232–234].

The flux between two elements, here denoted by $(-)$ and $(+)$, is defined by

$$\begin{gathered} f_i(q^{(-,+)}) \approx f_i^{(-,+)} = \\ \frac{1}{2}\left(\vec{\boldsymbol{\psi}}^{(-)T}\vec{\mathbf{f}}_i^{(-)} + \vec{\boldsymbol{\psi}}^{(+)T}\vec{\mathbf{f}}_i^{(+)} - \vec{n}|s_{max}|\left(\vec{\boldsymbol{\psi}}^{(+)T}\vec{\mathbf{q}}^{(+)} - \vec{\boldsymbol{\psi}}^{(-)T}\vec{\mathbf{q}}^{(-)}\right)\right) = \\ \frac{1}{2}\left(\vec{\boldsymbol{\psi}}^{(-)T}\left(\vec{\mathbf{f}}_i^{(-)} + \vec{n}\,|s_{max}|\vec{\mathbf{q}}^{(-)}\right)\right) + \frac{1}{2}\left(\vec{\boldsymbol{\psi}}^{(+)T}\left(\vec{\mathbf{f}}_i^{(+)} - \vec{n}\,|s_{max}|\vec{\mathbf{q}}^{(+)}\right)\right) \end{gathered} \tag{30}$$

where $|s_{max}|$ is the maximum wave speed of the solution to the general Riemann problem as mentioned in chapter 2.

For the boundary integral of equation (24), with $\Omega^{(l)}$ being the left neighbour, i.e $\overrightarrow{n} = -1$, and $\Omega^{(r)}$ the right neighbour, i.e $\overrightarrow{n} = 1$, of element $\Omega^{(e)}$ in direction k we obtain

$$\begin{aligned} &\int_{\Omega_k^{(e)}} \left[\phi_j f_i \left(q^{(-,+)}\right) \overrightarrow{n}\right]_{x_k=a_k}^{x_k=b_k} d\Omega_k^{(e)} \\ \approx &\frac{1}{2}\int_{\Omega_k^{(e)}} \psi_j^{(e)}|_{x_k=b_k^{(e)}} \cdot \vec{\boldsymbol{\psi}}^{(r)T}|_{x_k=a_k^{(r)}} d\Omega_k^{(e)} \left(\vec{\mathbf{f}}_i^{(r)} + |s_{max}|\vec{\mathbf{q}}^{(r)}\right) \\ &+\frac{1}{2}\int_{\Omega_k^{(e)}} \psi_j^{(e)}|_{x_k=b_k^{(e)}} \cdot \vec{\boldsymbol{\psi}}^{(e)T}|_{x_k=b_k^{(e)}} d\Omega_k^{(e)} \left(\vec{\mathbf{f}}_i^{(e)} - |s_{max}|\vec{\mathbf{q}}^{(e)}\right) \\ &-\frac{1}{2}\int_{\Omega_k^{(e)}} \psi_j^{(e)}|_{x_k=a_k^{(e)}} \cdot \vec{\boldsymbol{\psi}}^{(e)T}|_{x_k=a_k^{(e)}} d\Omega_k^{(e)} \left(\vec{\mathbf{f}}_i^{(e)} - |s_{max}|\vec{\mathbf{q}}^{(e)}\right) \\ &-\frac{1}{2}\int_{\Omega_k^{(e)}} \psi_j^{(e)}|_{x_k=a_k^{(e)}} \cdot \vec{\boldsymbol{\psi}}^{(l)T}|_{x_k=b_k^{(l)}} d\Omega_k^{(e)} \left(\vec{\mathbf{f}}_i^{(l)} + |s_{max}|\vec{\mathbf{q}}^{(l)}\right) \end{aligned} \tag{31}$$

Like the definitions in equation (28), we can transform the integrals on the reference element again. For the integrals with no contribution of face neighbours we receive:

$$\begin{aligned} \frac{1}{T_k^{(e)}} \cdot \mathbf{F}_k^+ &:= \frac{1}{2 \cdot T_k^{(e)}} \int_{\zeta_k} \vec{\boldsymbol{\psi}}|_{x_k=1} \cdot \vec{\boldsymbol{\psi}}^T|_{x_k=1} d\zeta_k \\ &= \frac{1}{2}\int_{\Omega_k^{(e)}} \vec{\boldsymbol{\psi}}^{(e)}|_{x_k=b_k^{(e)}} \cdot \vec{\boldsymbol{\psi}}^{(e),T}|_{x_k=b_k^{(e)}} d\Omega_k^{(e)} \\ \frac{1}{T_k^{(e)}} \cdot \mathbf{F}_k^- &:= \frac{1}{2 \cdot T_k^{(e)}} \int_{\zeta_k} \vec{\boldsymbol{\psi}}|_{x_k=0} \cdot \vec{\boldsymbol{\psi}}^T|_{x_k=0} d\zeta_k \\ &= \frac{1}{2}\int_{\Omega_k^{(e)}} \vec{\boldsymbol{\psi}}^{(e)}|_{x_k=a_k^{(e)}} \cdot \vec{\boldsymbol{\psi}}^{(e)T}|_{x_k=a_k^{(e)}} d\Omega_k^{(e)}. \end{aligned} \tag{32}$$

For the integrals with contribution of the face neighbours:

$$\begin{aligned}\frac{1}{T_k^{(e)}} \cdot \mathbf{F}_k^{-,+} &:= \frac{1}{2 \cdot T_k^{(e)}} \int_{\zeta_k} \vec{\boldsymbol{\psi}}|_{x_k=1} \cdot \vec{\boldsymbol{\psi}}^T|_{x_k=0} d\zeta_k \\ &= \frac{1}{2} \int_{\Omega_k^{(e)}} \vec{\boldsymbol{\psi}}^{(e)}|_{x_k=b_k^{(e)}} \cdot \vec{\boldsymbol{\psi}}^{(r)T}|_{x_k=a_k^{(r)}} d\Omega_k^{(e)} \\ \frac{1}{T_k^{(e)}} \cdot \mathbf{F}_k^{+,-} &:= \frac{1}{2 \cdot T_k^{(e)}} \int_{\zeta_k} \vec{\boldsymbol{\psi}}|_{x_k=0} \cdot \vec{\boldsymbol{\psi}}^T|_{x_k=1} d\zeta_k \\ &= \frac{1}{2} \int_{\Omega_k^{(e)}} \vec{\boldsymbol{\psi}}^{(e)}|_{x_k=a_k^{(e)}} \cdot \vec{\boldsymbol{\psi}}^{(l)T}|_{x_k=b_k^{(l)}} d\Omega_k^{(e)}.\end{aligned} \tag{33}$$

The integrals in equation (24) replaced by the definitions in equation (28), (32) and (33) yield:

$$\begin{aligned} &\frac{1}{T^{(e)}} \mathbf{M} \frac{\partial}{\partial t} \vec{\mathbf{q}}^{(e)} + \\ &+\frac{1}{2} \sum_{i=1}^{N} \sum_{k=1}^{N} \frac{1}{T_k^{(e)}} \mathbf{F}_k^{-,+} \cdot \left(\vec{\mathbf{f}}_i^{(r)} + |s_{max}| \vec{\mathbf{q}}^{(r)} \right) \\ &+\frac{1}{2} \sum_{i=1}^{N} \sum_{k=1}^{N} \frac{1}{T_k^{(e)}} \mathbf{F}_k^{+} \cdot \left(\vec{\mathbf{f}}_i^{(e)} - |s_{max}| \vec{\mathbf{q}}^{(e)} \right) \\ &-\frac{1}{2} \sum_{i=1}^{N} \sum_{k=1}^{N} \frac{1}{T_k^{(e)}} \mathbf{F}_k^{-} \cdot \left(\vec{\mathbf{f}}_i^{(e)} - |s_{max}| \vec{\mathbf{q}}^{(e)} \right) \\ &-\frac{1}{2} \sum_{i=1}^{N} \sum_{k=1}^{N} \frac{1}{T_k^{(e)}} \mathbf{F}_k^{+,-} \cdot \left(\vec{\mathbf{f}}_i^{(l)} + |s_{max}| \vec{\mathbf{q}}^{(l)} \right) \\ &-\sum_{i=1}^{N} \mathbf{D}_i \cdot \vec{\mathbf{f}}_i^{(e)} - \frac{1}{T^{(e)}} \mathbf{M} \vec{\mathbf{S}}^{(e)} = 0. \end{aligned} \tag{34}$$

We obtain, by inverting the mass matrix and putting all terms, except the q term, on the right side, the ordinary differential equation (ODE):

$$\begin{aligned} &\frac{\partial}{\partial t} \vec{\mathbf{q}}^{(e)} = \\ &-\frac{1}{2} \sum_{i=1}^{N} \sum_{k=1}^{N} \left(b_k^{(e)} - a_k^{(e)} \right)^{-1} \mathbf{M}^{-1} \mathbf{F}_k^{-,+} \cdot \left(\vec{\mathbf{f}}_i^{(r)} + |s_{max}| \vec{\mathbf{q}}^{(r)} \right) \\ &-\frac{1}{2} \sum_{i=1}^{N} \sum_{k=1}^{N} \left(b_k^{(e)} - a_k^{(e)} \right)^{-1} \mathbf{M}^{-1} \mathbf{F}_k^{+} \cdot \left(\vec{\mathbf{f}}_i^{(e)} - |s_{max}| \vec{\mathbf{q}}^{(e)} \right) \\ &+\frac{1}{2} \sum_{i=1}^{N} \sum_{k=1}^{N} \left(b_k^{(e)} - a_k^{(e)} \right)^{-1} \mathbf{M}^{-1} \mathbf{F}_k^{-} \cdot \left(\vec{\mathbf{f}}_i^{(e)} - |s_{max}| \vec{\mathbf{q}}^{(e)} \right) \\ &+\frac{1}{2} \sum_{i=1}^{N} \sum_{k=1}^{N} \left(b_k^{(e)} - a_k^{(e)} \right)^{-1} \mathbf{M}^{-1} \mathbf{F}_k^{+,-} \cdot \left(\vec{\mathbf{f}}_i^{(l)} + |s_{max}| \vec{\mathbf{q}}^{(l)} \right) \\ &+\sum_{i=1}^{N} T^{(e)} \mathbf{M}^{-1} \mathbf{D_i} \cdot \vec{\mathbf{f}}_i^{(e)} + \vec{\mathbf{S}}^{(e)} = 0 \end{aligned} \tag{35}$$

This time continuous space discrete scheme is the basis to our further discontinuous Galerkin simulations.

It shows two elementary aspects of discontinuous Galerkin methods.

- At first the only communication between elements is between direct face neighbours. For the a concrete implementation this offers, pending on the solver of the ODE, a promising opportunity as the process could easily be split in the two steps of synchronizing the DOFs between the elements and then calculating the solution to the ODE element wise in parallel.
- The second is the general definition of the local mass stiffness and flux matrices. In contrast to continuous Galerkin methods, where basis functions are defined globally, matrices can be kept short and are equal in all elements due to the transformation theorem.

3.1 From discontinuous Galerkin to finite volume method

The finite volume method can be seen as a special case of the discontinuous Galerkin method.

For the spatial order $N_p = 1$ the matrices of equation (28), (32) and (33) decompose to

$$\begin{aligned} \mathbf{M} &= 1 \\ \mathbf{D} &= 0 \\ \mathbf{F}_k^{+} = \mathbf{F}_k^{-} = \mathbf{F}_k^{+,-} = \mathbf{F}_k^{-,+} &= 1 \end{aligned} \tag{36}$$

For a PDE in one space dimension, i.e N = 1, we obtain the ODE

$$\frac{\partial}{\partial t}\vec{\mathbf{q}}^{(e)} = -\frac{1}{2}\frac{1}{\Delta x^{(e)}}\left(-f^{(l,e)} + f^{(e,r)}\right) - \vec{\mathbf{S}} \tag{37}$$

with $\Delta x^{(e)} = b^{(e)} - a^{(e)}$ being the size of element $\Omega^{(e)} = \left[a^{(e)}, b^{(e)}\right]$.

Using the Euler method, with a time step Δt yields a well known finite volume scheme, derived in [6, p. 64–66]

$$\vec{\mathbf{q}}_{n+1}^{(e)} = \vec{\mathbf{q}}_n^{(e)} - \frac{1}{2}\frac{\Delta t}{\Delta x^{(e)}}\left(f_n^{(e,r)} - f_n^{(l,e)}\right) - \vec{\mathbf{S}}(\vec{\mathbf{q}}_n^{(e)}) \tag{38}$$

3.2 Solving the Galerkin scheme with Runge-Kutta methods

For the ordinary differential equation (35) together with the initial values for $t = 0$ defined by $\mathring{q}(\vec{x})$ we receive an initial value problem.

We will abbreviate the initial value problem by

$$\begin{aligned} \frac{\partial}{\partial t}\vec{\boldsymbol{q}} &= f(\vec{\boldsymbol{q}}) \\ \vec{\boldsymbol{q}}_0 &= \mathring{q}(\vec{\mathbf{x}}) \end{aligned} \tag{39}$$

with $\vec{\boldsymbol{q}}$ being the vector of the DOFs of all elements, $\vec{\mathbf{x}}$ of the corresponding nodes and f the ODE pending on $\vec{\boldsymbol{q}}$.

The common way to solve an initial value problems is by using a Runge-Kutta method with the desired order of convergence. For example the solver we used in the previous chapter 3.1, the Euler method, is a Runge-Kutta method of order one.

A general definition for the Runge-Kutta methods with a more detailed theoretical explanation can be found here [1, p. 128 –190].

For this thesis it is sufficient to know that the orders of convergence of the Runge-Kutta methods are bound by the Butcher barrier which makes orders higher than four expensive due to the increasing number of stages.

In the further implementation we will use the classical Runge-Kutta method, also called Simpson rule which has an convergence order of four and is defined as

$$\begin{aligned} \vec{\boldsymbol{q}}_{n+1} &= \vec{\boldsymbol{q}}_n + \Delta t \left(\frac{1}{6} f_1 + \frac{1}{3} f_2 + \frac{1}{3} f_3 + \frac{1}{6} f_4 \right) \\ f_1 &= f\left(\vec{\boldsymbol{q}}_n\right) \\ f_2 &= f\left(\vec{\boldsymbol{q}}_n + \frac{1}{2} \Delta t \cdot f_1\right) \\ f_3 &= f\left(\vec{\boldsymbol{q}}_n + \frac{1}{2} \Delta t \cdot f_2\right) \\ f_4 &= f\left(\vec{\boldsymbol{q}}_n + \Delta t \cdot f_3\right) \end{aligned} \tag{40}$$

for the solution $\vec{\boldsymbol{q}}$ between time t_n and t_{n+1} with $\Delta t = t_{n+1} - t_n$ and $\vec{\boldsymbol{q}}_n$ as the functions value at time t_n.

For the parallelisation, as mentioned before, we now have to synchronize the DOFs of the solution four times, for every f_i, in each iteration step.

3.3 Simulation of the dam break Problem

In this chapter we will take a look on a example numeric solution of the dam break problem as stated in chapter 2 in equation (11).

The observed interval $[-2, 2]$ is divided in 200 elements of equal size $\Delta x = 0.02$. Each element is of order 4, the solution is approximated along 4 nodes. The time step length is set to $\Delta t = 5 \cdot 10^{-6}$ which fulfills the CFL-condition, as explained in [6, p. 65 – 71] and [5, p. 97]. For the method to be stable it is necessary to fulfill this condition.

In figure 3 is the analytic solution at time $t = 0.5$ compared to the numerical solution.

As we see the shock wave as well as the rarefaction wave evolved in the numerical solution. The middle state q_m is also apparently right.

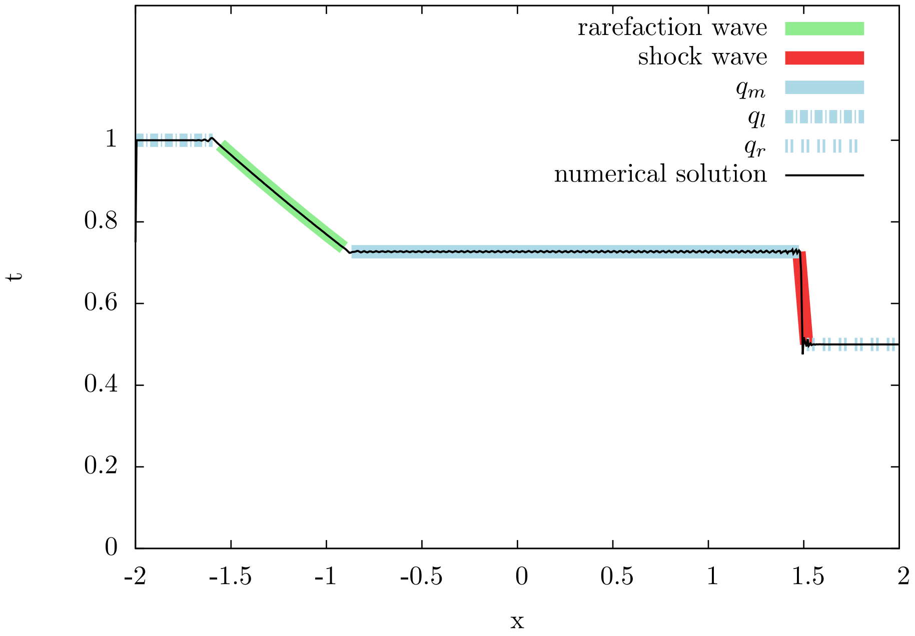


Figure 3: Comparison between the numeric solution and the analytic solution.

Along the shock wave however we observe a well known disadvantageous phenomenon in discontinuous Galerkin methods. The numerical solution oscillates along the discontinuity as shown in figure 4.

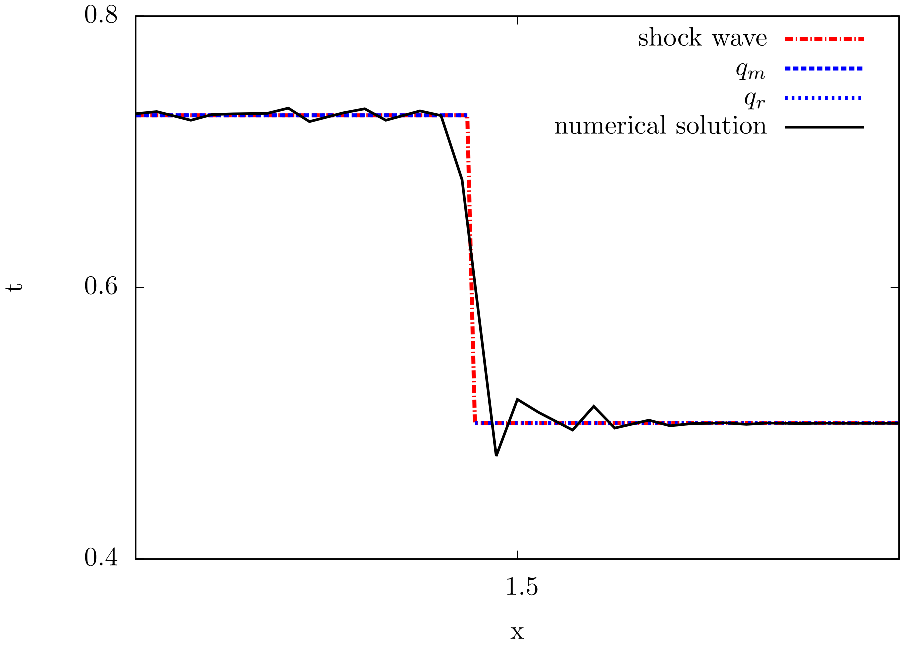


Figure 4: Comparison between the numeric solution and the analytic solution in detail. Along the discontinuity oscillations are visible.

The root to this problem is the polynomial approximation along nodes, an detailed explanation can be found in [5, p. 136 – 139]

One solution to this problem is the implementation of an limiter, a function to detect and smoothen oscillations. An example implementation of a limiter is given in [5, p. 145 – 157].

Due to time restrictions a proper implementation of the example limiter wasn't possible.

4 The ADER-DG method

The ADER-DG method, as proposed in [3], states an new approach in solving the time integration. The idea behind the method is transforming the discretisation by nodes from the spatial dimension to the time dimension additionally, to obtain a time-stepping function.

The general time iteration can than be divided in two parts:

- At first, for each element, the local DOFs in space and time are derived from the local DOFs in space of the initial time-step. These DOFs already state the prediction in time.
- Second, the prediction in space and time is used in a schema alike the discontinuous Galerkin schema (35) in section 3, to obtain the DOFs in space of the following time-step.

4.1 The ADER Time prediction

As in the formulation for discrete Galerkin methods the starting point is a common partial differential equation as defined in section 2, on a N-dimensional space:

$$\frac{\partial}{\partial t} q + \sum_{i=1}^{N} \frac{\partial}{\partial x_i} f_i = S. \tag{41}$$

In addition to the in section 3 already defined mapping from to the spatial reference element $\zeta_s = [0,1]^N$ we introduce a mapping from any time interval $[t_n, t_{n+1}]$ to a time reference element $\zeta_t = [0,1]$ and combine them to the space-time reference element $\zeta = \zeta_s \times \zeta_t$.

Equivalently to the discontinuous Galerkin formulation we receive, by multiplying the PDE with test functions Ψ_j and integrating over the space time reference element the weak form:

$$\int_{\zeta_t} \int_{\zeta_s} \Psi_j \left(\frac{\partial}{\partial t} q + \sum_{i=1}^{N} \frac{\partial}{\partial x_i} f_i - S \right) d\zeta_s \, d\zeta_t = 0. \tag{42}$$

Integrating $\Psi_j \frac{\partial}{\partial t} q$ by parts over time with $\vec{x} = (x_1, ..., x_N)$ yields:

$$\begin{gathered} \int_{\zeta_s} \Psi_j \left(\vec{x}, 1\right) q \left(\vec{x}, 1\right) d\zeta_s - \int_{\zeta_s} \Psi_j \left(\vec{x}, 0\right) q \left(\vec{x}, 0\right) d\zeta_s - \\ \int_{\zeta_s} \int_{\zeta_t} \frac{\partial}{\partial t} \Psi_j \, q \, d\zeta_t \, d\zeta_s + \int_{\zeta_s} \int_{\zeta_t} \Psi_i \left(\sum_{i=1}^{N} \frac{\partial}{\partial x_i} f_i - S \right) d\zeta_t \, d\zeta_s = 0. \end{gathered} \tag{43}$$

Again we're approximating q on the nodal basis as defined in chapter 5. With the Lagrange polynomials ψ_i on the space-time reference element and the degrees of freedom $\hat{q}_i$ we get the approximation

$$q_h = \sum_{i=1}^{N_p} \psi_i \hat{q}_i,$$

with N_p being the number of Lagrange polynomials. Or in vector-vector notation

$$\vec{\psi}^{\top}\vec{q},$$

with $\vec{\psi}$ as the vector of Lagrange polynomials and $\vec{q}$ of degrees of freedom. In the same manner we obtain the vectors $\vec{f_i}$ and $\vec{S}$.

We receive by approximating all summands despite the one at time 0:

$$\int_{\zeta_s} \Psi_j\left(\vec{x},1\right)\cdot\vec{\psi}\left(\vec{x},1\right)^{\top}\vec{q}\,d\zeta_s - \int_{\zeta_s} \Psi_j\left(\vec{x},0\right)\cdot q(\vec{x},0)\,d\zeta_s - \int_{\zeta_s}\int_{\zeta_t}\frac{\partial}{\partial t}\Psi_j\cdot\vec{\psi}^{\top}\vec{q}\,d\zeta_t\,d\zeta_s + \int_{\zeta_s}\int_{\zeta_t}\Psi_j\cdot\left(\sum_{i=1}^{N}\frac{\partial}{\partial x_i}\vec{\psi}^{\top}\vec{f_i} - \vec{\psi}^{\top}\vec{S}\right)d\zeta_t\,d\zeta_s = 0. \tag{44}$$

As we're expecting the initial DOFs at any time t_n being only defined in space, we use the spatial approximation, here denoted by $\vec{\boldsymbol{\psi}}^{T}\vec{\mathbf{q}}$, for the summand we left out in the previous formula:

$$\int_{\zeta_s} \Psi_j\left(\vec{x},0\right)\cdot q(\vec{x},0)\,d\zeta_s \approx \int_{\zeta_s}\Psi_j(\vec{x},0)\cdot\vec{\boldsymbol{\psi}}^{T}\left(\vec{x}\right)\cdot\vec{\mathbf{q}}(n)\,d\zeta_s. \tag{45}$$

Hence (44) and (45) have to be valid for all test functions Ψ_j we receive, by using the space-time basis functions as test functions and by defining the matrices

$$\begin{aligned}
\mathbf{M}^1 &= \int_{\zeta_s}\left(\vec{\psi}\cdot\vec{\psi}^{\top}\right)\left(\vec{x},1\right)d\zeta_s\\
\mathbf{M}^0 &= \int_{\zeta_s}\left(\vec{\psi}\cdot\vec{\boldsymbol{\psi}}^{T}\right)\left(\vec{x},0\right)d\zeta_s\\
\mathbf{M} &= \int_{\zeta_s}\int_{\zeta_t}\left(\vec{\psi}\cdot\vec{\psi}^{\top}\right)d\zeta_t\,d\zeta_s\\
\mathbf{K}^t &= \int_{\zeta_s}\int_{\zeta_t}\frac{\partial}{\partial t}\Psi\cdot\vec{\psi}^{\top}\,d\zeta_t\,d\zeta_s\\
\mathbf{K}^i &= \int_{\zeta_s}\int_{\zeta_t}\psi\cdot\frac{\partial}{\partial x_i}\vec{\phi}^{\top}\,d\zeta_t\,d\zeta_s,
\end{aligned} \tag{46}$$

the matrix-matrix form

$$\mathbf{M}^1\vec{q} - \mathbf{M}^0\vec{\mathbf{q}}(n) - \mathbf{K}^t\vec{q} + \sum_{i=1}^{N}\mathbf{K}^i\vec{f_i} - \mathbf{M}\vec{S} = 0. \tag{47}$$

By rearranging we receive, the non-linear system of equations for the local degrees of freedom in space in time

$$\vec{q} = \left(\mathbf{M}^1 - \mathbf{K}^t\right)^{-1}\left(\mathbf{M}^0\vec{\mathbf{q}}(\vec{x},n) - \sum_{i=1}^{N}\mathbf{K}^i\vec{f_i}(q) + \mathbf{M}\vec{S}(q)\right). \tag{48}$$

This system of equations can be solved by the iterative schema:

$$\vec{q}_{k+1} = \left(\mathbf{M}^1 - \mathbf{K}^t\right)^{-1}\left(\mathbf{M}^0\vec{\mathbf{q}}(\vec{x},n) - \sum_{i=1}^{N}\mathbf{K}^i\vec{f_i}(q_k) + \mathbf{M}\vec{S}(q_k)\right), \tag{49}$$

4.2 Fully discrete space time formulation

To obtain a fully space and time discrete formulation we restart at equation (43) now using the $N_p^{(s)}$ space basis functions $\vec{\boldsymbol{\psi}}$ as test functions.

With the set of test functions being time independent (43) becomes

$$\begin{gathered}\int_{\zeta_s} \psi_j\left(\vec{x}\right) q\left(\vec{x},1\right) d\zeta_s - \int_{\zeta_s} \psi_j\left(\vec{x}\right) q\left(\vec{x},0\right) d\zeta_s + \\ \int_{\zeta_s}\int_{\zeta_t} \psi_j \left(\sum_{i=1}^{N} \frac{\partial}{\partial x_i} f_i - S\right) d\zeta_t\, d\zeta_s = 0.\end{gathered} \tag{50}$$

As we wish to obtain a stepping function from a set of initial DOFs in space to the DOFs of a next time-step we approximate $q(\vec{x},1)$ as well as $q(\vec{x},0)$, equivalently to equation (20) in section 3, with degrees of freedom $\vec{\mathbf{q}}_1$, $\vec{\mathbf{q}}_0$ and space basis functions $\vec{\boldsymbol{\psi}}$.

Integrating the flux terms as we did in equation (17) in section 3 by parts and again approximating them in time and space as in (44) yields:

$$\begin{gathered}\int_{\zeta_s} \psi_j \cdot \vec{\boldsymbol{\psi}}^\top \vec{\mathbf{q}}_1\, d\zeta_s - \int_{\zeta_s} \psi_j \cdot \vec{\boldsymbol{\psi}}^\top \vec{\mathbf{q}}_0\, d\zeta_s + \\ \sum_{i=1}^{N}\left(\int_{\zeta_t}\int_{\delta\zeta_s} \psi_j f_i^{(-,+)}\, \vec{n}\, d\delta\zeta_s\, d\zeta_t - \int_{\zeta_s}\int_{\zeta_t} \frac{\partial}{\partial x_i}\psi_j \cdot \vec{\psi}^\top \vec{f}_i\, d\zeta_t\, d\zeta_s\right) - \\ \int_{\zeta_s}\int_{\zeta_t} \psi_j \vec{\psi}^\top \vec{S}\, d\zeta_t\, d\zeta_s = 0,\end{gathered} \tag{51}$$

with $f_i^{(-,+)}$ being the numeric flux along the boundary $\delta\zeta_s$.

The Riemann problem along the boundary $\delta\zeta_s$ is again solved by the Lax-Friedrichs flux as defined in section 3 in equation (30) now using the DOFs in space-time.

By evolving (51) and defining the matrices equivalently to the ones in section 3 we receive

$$\begin{aligned}\mathbf{M}_s &= \int_{\zeta_s} \left(\vec{\boldsymbol{\psi}} \cdot \vec{\boldsymbol{\psi}}^\top\right) d\zeta_s \\ \mathbf{M}_{st} &= \int_{\zeta_s} \left(\vec{\boldsymbol{\psi}} \cdot \vec{\psi}^\top\right) d\zeta_s \\ \mathbf{K}^i &= \int_{\zeta_s}\int_{\zeta_t} \frac{\partial}{\partial x_i}\vec{\boldsymbol{\psi}} \cdot \vec{\psi}^\top\, d\zeta_t\, d\zeta_s \\ \mathbf{F}_i^- &= \int_{\zeta_t}\int_{\zeta_s^i} \vec{\boldsymbol{\psi}} \cdot \vec{\psi}^\top|_{x_i=0}\, d\zeta_s^i\, d\zeta_t \\ \mathbf{F}_i^+ &= \int_{\zeta_t}\int_{\zeta_s^i} \vec{\boldsymbol{\psi}} \cdot \vec{\psi}^\top]_{x_i=1}\, d\zeta_s^i\, d\zeta_t \\ \mathbf{F}_i^{-,+} &= \int_{\zeta_t}\int_{\zeta_s^i} \vec{\boldsymbol{\psi}}|_{x_i=1} \cdot \vec{\psi}|_{x_i=0}\, d\zeta_s^i\, d\zeta_t \\ \mathbf{F}_i^{+,-} &= \int_{\zeta_t}\int_{\zeta_s^i} \vec{\boldsymbol{\psi}}|_{x_i=0} \cdot \vec{\psi}|_{x_i=1}\, d\zeta_s^i\, d\zeta_t.\end{aligned} \tag{52}$$

Which result in the fully discrete formulation

$$
\begin{aligned}
\vec{\mathbf{q}}_1 = \vec{\mathbf{q}}_0 + \sum_{i=1}^{N} \mathbf{M}_s^{-1}\mathbf{K}^i \vec{f}_i + \\
\frac{1}{2}\sum_{i=1}^{N}\sum_{k=1}^{N} \mathbf{M}_s^{-1}\mathbf{F}_k^{-}\left(\vec{f}_i + |s_{max}|\vec{q}_i\right) + \\
\frac{1}{2}\sum_{i=1}^{N}\sum_{k=1}^{N} \mathbf{M}_s^{-1}\mathbf{F}_k^{+}\left(\vec{f}_i + |s_{max}|\vec{q}_i\right) + \\
\frac{1}{2}\sum_{i=1}^{N}\sum_{k=1}^{N} \mathbf{M}_s^{-1}\mathbf{F}_k^{-,+}\left(\vec{f}_i^{(r)} + |s_{max}|\vec{q}_i^{(r)}\right) - \\
\frac{1}{2}\sum_{i=1}^{N}\sum_{k=1}^{N} \mathbf{M}_s^{-1}\mathbf{F}_k^{+,-}\left(\vec{f}_i^{(l)} + |s_{max}|\vec{q}_i^{(l)}\right) + \\
\mathbf{M}_s^{-1}\mathbf{M}_{st}\vec{S}.
\end{aligned}
\tag{53}
$$

Compared to the discontinuous Galerkin scheme with Runge-Kutta timestepping this scheme offers two advantages.

- Through the unbound number of nodes in time, we are able to reach a higher order of convergence than 4 in a easier way than by Runge-Kutta methods.
- And as the calculation only depends on the fluxes in the space-time interval from neighbouring elements, there's only one synchronization step required in each iteration.

However we have a higher need in memory to store the time space degrees of freedom. The exact additional amount depends on the choice of nodes as discussed in 5.

5 The nodal basis

As functional basis for the approximation of the solution of a partial differential equation we have a wide variety of options. For example the Legendre polynomials, as described in [5, p. 43 – 51], would lead to well conditioned mass matrices as used in chapters 3 and 4.

For simplicity reasons we will use the Lagrange basis in this thesis. The polynomial Lagrange basis of dimension $n+1$ on an interval $[a,b]$ consists of $n+1$ Lagrange polynomials which are distinctly defined by a set of $n+1$ nodes $\{\xi_0,...,\xi_n|\xi_i \neq \xi_j\}$ on the interval. The Lagrange polynomials themselves are defined by the term:

$$\psi_i(x) = \prod_{k\in\{0...N\}/i} \frac{(x-\xi_i)}{(\xi_k-\xi_i)}. \tag{54}$$

On the set of nodes the basis evaluates to:

$$\psi_i(\xi_j) = \begin{cases} 1 & \text{, for } i = j \\ 0 & \text{, for } i \neq j. \end{cases} \tag{55}$$

By referring to the set nodes, this is also called a nodal basis.

A solution u is approximated on this basis by its values $u(\xi_i)$ evaluated on the set of nodes

$$u \approx u_h = \sum_{i=0}^{n} \psi_i \cdot u(\xi_i). \tag{56}$$

The advantage, compared to other bases, is the simple calculation of nonlinear flux and source terms on the nodes, as already used in the chapters 3 and 4,

$$f(u_h(\xi_j)) = f(\sum_{i=0}^{n} \psi_i(\xi_j)\cdot u(\xi_j)) = f(u(\xi_j)). \tag{57}$$

In contrast, at this point the Legendre polynomials would require a L2-projection, which takes more effort than the simple evaluation on the nodal basis.

Basis functions in N dimensional spaces are defined by the tensor product of one dimensional Lagrange basis functions of order $N_p^{(j)}$ for each dimension j with variable x_j

$$\psi_i(\vec{x}) = \prod_{j=1}^{N} \psi_{i(j)}(x_j), \tag{58}$$

with the distinct index i

$$i = \sum_{j=1}^{N} i(j)(N_p^{(j-1)}+1)^{j-1}. \tag{59}$$

5.1 The choice of nodes

When choosing a set of explicit nodes we have to face three contrary aspects:

1. To reduce computational costs (as floating point operations and memory storage) we should reduce the amount of nodes to a minimum, without loosing any order of convergence.
2. The polynomial approximation should be as accurate as possible.
3. To obtain an optimal use of the underlying hardware, we want to conform the sparsity patterns of the matrices.

Hence time and space can be considered equal in integrals we will neglect the notational difference in this chapter.

5.2 The number of nodes

A polynomial in d space dimensions of degree n in each dimension is distinctly defined by

$$\frac{1}{(d+1)!}\sum_{j=1}^{d+1}(n+j)$$

nodes. This is the triangular number in d multiple dimensions. Figure 5 shows a comparison of two different sets of nodes in two dimensions, the first naive, by calculating the Cartesian product of an one dimensional set of nodes, the second triangular, by decreasing the number of nodes in the first dimension with growing second dimension.

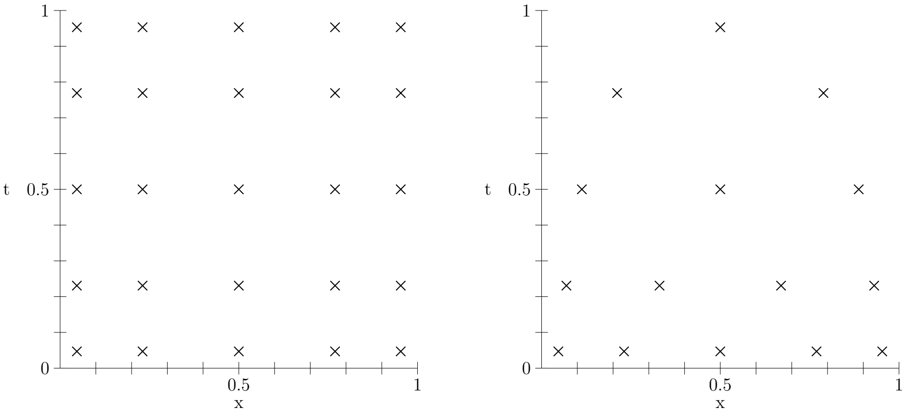


Figure 5: Legendre nodes on the space time interval $[0,1]^2$ The left set of nodes is the Cartesian product of one dimensional Legendre nodes, the right decreases the number of space nodes in the space time dimension.

Both sets of nodes define the same space of polynomials, the savings between the naive and the triangular approach of memory are

$$n^d - \frac{1}{(d+1)!}\sum_{j=1}^{d+1}(n+j) = n^d - \frac{2\cdot n+d+2}{2d!} \in O(n^d) \tag{60}$$

nodes per element.

The impact on the size of mass stiffness and flux matrices is even more significant. For the Cartesian product we obtain

$$n^{2\cdot d}$$

entries, for the triangular approach

$$\frac{4\cdot n^2 + 4n(d+2) + (d+2)^2}{4d!^2},$$

which only grows quadratic with the number of nodes.

5.3 Sparse matrices

Mass, stiffness and flux matrices will be analyzed in general definitions, as the corresponding matrices of chapter 4 and 3 differ only in index transformations and the dimension of the basis. The explicit matrices of these chapters will be mentioned as examples. The indexing of multidimensional basis functions will be equal to the one we already used in the beginning of this chapter.

At first we will look at the matrices with the Cartesian product of N one dimensional nodes as basis.

Entries of mass matrices on a domain $\zeta = [0,1]^n$ for some n, have the form,

$$\begin{aligned} M_{ij} = \int_\zeta \psi_i \cdot \psi_j \, d\zeta = \int_\zeta \prod_{d\in\{1\dots n\}} \psi_{i_d}(x_d)\cdot\psi_{j_d}(x_d)\, d\zeta = \\ \prod_{d\in\{1\dots n\}} \int_0^1 \psi_{i_d}(x)\cdot\psi_{j_d}(x) dx. \end{aligned} \tag{61}$$

With this we can analyze the integrals in one dimension. By using the definition of the Lagrange polynomials we get

$$\int_0^1 \psi_i \cdot \psi_j \, dx = \int_0^1 \prod_{k\in\{0\dots N\}/i} \frac{(x-\xi_i)}{(\xi_k-\xi_i)} \cdot \prod_{l\in\{0\dots N\}/j} \frac{(x-\xi_j)}{(\xi_l-\xi_j)}\, dx \tag{62}$$

and obtain for $i \neq j$

$$\int_0^1 \prod_{k\in\{0\dots N\}} \frac{(x-\xi_i)}{(\xi_k-\xi_i)} \cdot \prod_{k\in\{0\dots N\}/i,j} \frac{(x-\xi_j)}{(\xi_k-\xi_j)}\, dx. \tag{63}$$

The first product states a distinct polynomial of degree N with the set of nodes as roots, the second a distinct polynomial of degree $N-2$. By a set of orthogonal polynomials all entries except the diagonal would become zero. Such sets are given by the Jacobi polynomials, for which a definition can be found in [2, p. 171–178]. A special case of these orthogonal polynomials are the Legendre polynomials, which can be generated by the Gram-Schmidt algorithm with the simple monomials as initial function set, as explained in [4, p. 389–391].

We choose the roots of the Legendre polynomial of order N, as the set of nodes (which is called Legendre nodes), the first polynomial of the integral in equation (63) becomes the Legendre polynomial and is orthogonal to the second polynomial. We receive a diagonal mass matrix

$$M_{ij} = \begin{cases} \prod_{d\in\{1\dots n\}} \int_0^1 \psi_{i_d}^2\, dx & \text{, for } i = j \\ 0 & \text{, for } i \neq j. \end{cases} \tag{64}$$

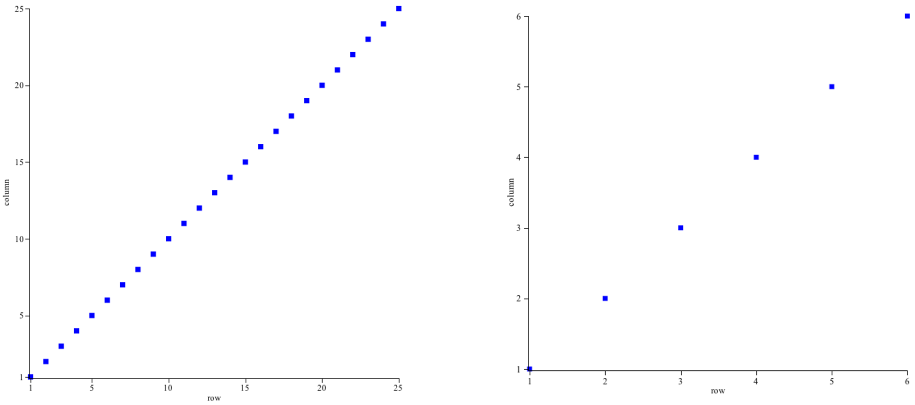


Figure 6: Sparsity pattern for the mass matrix $\mathbf{M}$ for $n = 5$ nodes in space and time, and the mass matrix $\mathbf{M}_s$ for $n = 6$ nodes in space

Analogously to (61) we can decompose a stiffness matrix derived with respect to x_k:

$$\int_\zeta \psi_i \cdot \frac{\partial}{\partial x_k}\psi_j \, d\zeta = \int_\zeta \psi_{i_k}(x_k) \cdot \frac{\partial}{\partial x_k}\psi_{j_k}(x_k) \cdot \prod_{d\in\{1\ldots n\}/k} \psi_{i_d}(x_d)\cdot\psi_{j_d}(x_d)\, d\zeta = \int_0^1 \psi_{i_k}(x)\cdot\frac{\partial}{\partial x}\psi_{j_k}(x)\, dx \cdot \prod_{d\in\{1\ldots n\}/k} \int_0^1 \psi_{i_d}(x)\cdot\psi_{j_d}(x)\, dx. \tag{65}$$

All integrals except the one containing the partial derivative are equal to the one we analyzed in equation (62) . With the definition of the Lagrange polynomials and the product rule we receive for the first integral:

$$\int_0^1 \psi_i \cdot \frac{\partial}{\partial x}\psi_j \, dx = \int_0^1 \prod_{k\in\{0\ldots N\}/i} \frac{(x-\xi_i)}{(\xi_k-\xi_i)} \cdot \sum_{l=0}^{n} \frac{1}{\xi_l-\xi_j} \prod_{m\in\{0\ldots N\}/j,l} \frac{(x-\xi_j)}{(\xi_m-\xi_j)}\, dx = \sum_{l=0}^{n} \frac{1}{\xi_l-\xi_j} \int_0^1 \prod_{k\in\{0\ldots N\}/i} \frac{(x-\xi_i)}{(\xi_k-\xi_i)} \cdot \prod_{m\in\{0\ldots N\}/j,l} \frac{(x-\xi_j)}{(\xi_m-\xi_j)}\, dx =: A_{ij}. \tag{66}$$

Unlike equation (62) only the integrals with $i \neq l$ and of $i \neq j$ can be eliminated, resulting in:

$$A_{ij} = \begin{cases} A_{ii} & \text{, for } i = j \\ \frac{1}{\xi_i-\xi_j}\int_0^1 \prod_{k\in\{0\ldots N\}/i} \frac{(x-\xi_i)}{(\xi_k-\xi_i)} \cdot \prod_{l\in\{0\ldots N\}/j,i} \frac{(x-\xi_j)}{(\xi_l-\xi_j)}\, dx & \text{, for } i \neq j. \end{cases} \tag{67}$$

In combination with equation (62) we get:

$$\int_\zeta \psi_i \cdot \frac{\partial}{\partial x_k}\psi_j \, d\zeta = \begin{cases} A_{i_k j_k} & \text{, for } i_d = j_d \, \forall \, d \in \{1,\ldots,N\}/k \\ 0 & \text{, else.} \end{cases} \tag{68}$$

Two sparsity patterns of stiffness matrices are illustrated in figure 7, the different patters are the result of the different partial derivations.

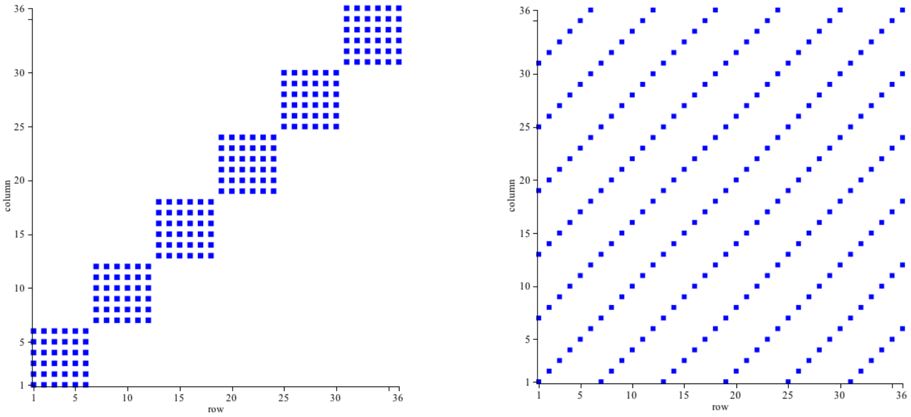


Figure 7: Sparsity patterns for the matrices $\mathbf{K}^1$ and $\mathbf{K}^t$ for the Cartesian product of Legendre nodes

However equation (68) doesn't hold for the triangular choosing of nodes of chapter 5.2. By using the Legendre nodes of order N in second dimension and of decreasing order in first dimension we still obtain a diagonal mass matrix. Equation (64) only holds for the integrals in the first dimension, if the indices in the second dimension are equal, we obtain:

$$M_{ij} = \prod_{d\in\{1,2\}} \int_0^1 \psi_{i_d} \cdot \psi_{j_d} \, dx = \begin{cases} 0 & \text{, for } i_2 \neq j_2 \\ 0 & \text{, for } i_2 = j_2 \text{ and } i_1 \neq j_1 \\ \prod_{d\in\{1,2\}} \int_0^1 \psi_{i_d}^2 \, dx & \text{, for } i_2 = j_2 \text{ and } i_1 = j_1. \end{cases} \tag{69}$$

By partially deriving in the first dimension, the varied choosing of nodes in first dimension has no impact on the structure obtained by the orthogonality of the integrals without the derivative in (65). We still get a matrix in block form as shown in figure 8 on the left.

By partially deriving in the second dimension though we lose the orthogonality of the integral without the partial derivative for indices with $i_2 \neq j_2$, the result is shown in figure 8 on the right. Only entries where the second indices are equal, are eliminated.

Independently from the number of nodes and space dimensions, only for a single stiffness matrix we obtain a sparsity pattern which is predestined for a block structured implementation. All other stiffness matrices are dense.

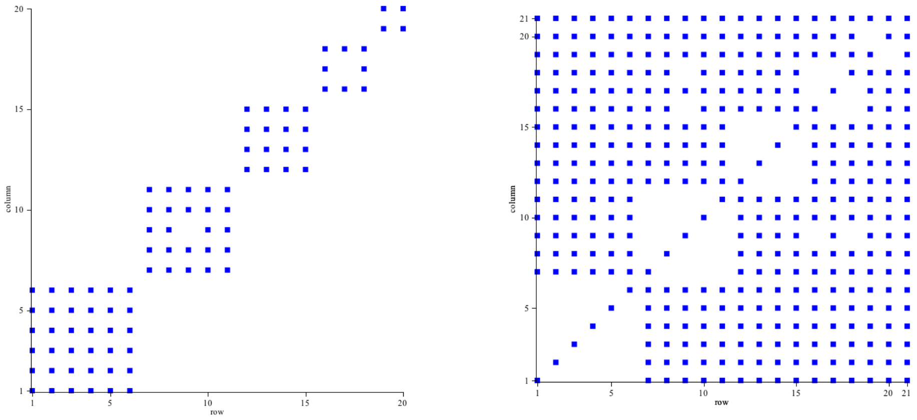


Figure 8: Sparsity patterns for the matrices $\mathbf{K}^1$ and $\mathbf{K}^t$ for the triangular set of Legendre nodes

5.4 Accuracy of the approximation

For the last aspect, to determine the accuracy of our approximation, we will use the well known Lebesgue constant

$$\Lambda_n = \left\| \sum_{i=0}^{n} |\psi_i(x)| \right\|_\infty$$

with $\|f(x)\|_\infty$ being the uniform-norm $\|f(x)\|_\infty := \sup_{x\in\Omega} |f(x)|$ and n the number of used interpolation nodes.

As shown in [2, p. 204–205] the Lebesgue constant sets an upper boundary for the error between the solution u and the approximation u_h, $E_h := \|u - u_h\|_\infty$ relatively to the error of the best polynomial approximation by n degrees of freedom E^*:

$$E_h \leq (1 + \Lambda_n)E^*.$$

By this constant we can compare the quality of the approximations of different sets of nodes. In our case we will look at the Legendre nodes from the last chapter and compare them to simple equidistant nodes, and two sets of nodes known for their small Legendre constants, the Chebyshev and the Lagrange-Gauss-Lobatto nodes, whose generation can be found here [2, p. 309–310].

In figure 9 are the evaluated Lebesgue constants, in two different scales, for the four sets of nodes for 1 up to 20 degrees of freedom. As we see in the left graph, for equidistant nodes the Legendre constant has exponential growth with n. In fact it was shown by Turetsky in 1968 [8] that the constant for equidistant nodes is approximately $\frac{2^{n+1}}{e \cdot n \log n}$. This limits the use of equidistant nodes, especially for the usage on higher orders of convergence. Instead we could use the Lagrange-Gauss-Lobatto nodes and the Chebyshev nodes, which both have proved logarithmic growth ([9]) . The Legendre nodes might make our matrices much simpler, but compared to the Lagrange-Gauss-Lobatto or Chebyshev nodes only offer linear growth.

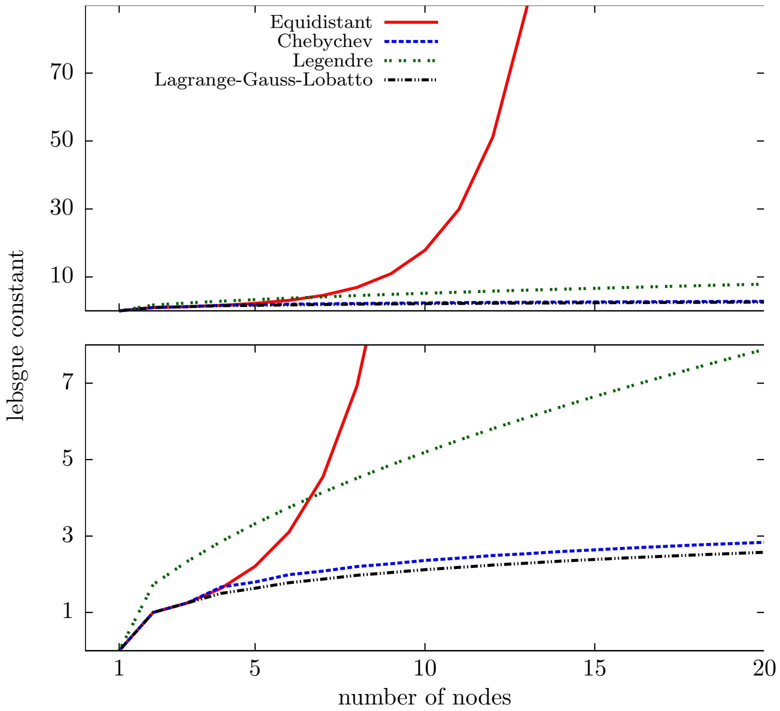


Figure 9: Lebesgue constant for equidistant, Chebyshev, Legendre and Lagrange-Gauss-Lobatto nodes.

6 Convergence Analysis

In this chapter we will compare the errors and orders of convergence of the three methods, the finite volume method, discontinuous Galerkin with Runge-Kutta time stepping and discontinuous Galerkin with the ADER DG-time predictor we introduced in the chapters 3.1 3 and 4.

To compare the assumptions we made on the order of convergence we need to define a metric for the error analysis.

In this thesis we will use the normalized L2 norm defined as

$$\left\|q^{exact} - q^{numeric}\right\|^2 = \frac{\sum_{e=1}^{\#Elements} \sum_{i=1}^{N} \left(q^{exact}(x_i^{(e)}) - q^{numeric}(x_i^{(e)})\right)^2}{\sum_{e=1}^{\#Elements} \sum_{i=1}^{N} q^{exact}(x_i^{(e)})^2} \tag{70}$$

where $\#Elements$ is the number of elements in the simulation, $x_i^{(e)}$ the i-th node in element e, and q^{exact} an analytically determined solution as we did in chapter 1.

The convergence analysis was performed for the advection equation as described in chapter 2 in equation (6). The domain was an interval of size two on which the initial values were defined by the Gaussian distribution:

$$\frac{1}{\sigma\sqrt{2\pi}} \cdot \exp\left(-\frac{1}{2} \cdot \left(\frac{x-\mu}{\sigma}\right)^2\right), \tag{71}$$

with $\mu = 1$ and $\sigma = 1$. An wave-speed of $u = 0.05$ was selected, for which the time of on wave circulation is $t = 40$. The time-step Δt was calculated pending on the element size Δx and the spatial order N by the term:

$$\Delta t = \frac{\Delta x}{u \cdot 2(N+1) \cdot 250}, \tag{72}$$

which is a combination of the CFL condition and the additional factor $\frac{1}{250}$. The intention to this factor was decreasing the temporal error, while keeping the computation time at an reasonable length.

In diagrams of the analysis we will always display the expected error graph pending on the error value e of the smallest amount of degrees of freedom d_1 as a dotted curve. For an expected convergence order of N the curve is defined by the term:

$$e \cdot \left(\frac{d_1}{x}\right)^N, \tag{73}$$

with x being the number of degrees of freedom.

6.1 The Finite Volume Method

As shown in 3.1 we can regard the finite volume method as a discontinuous Galerkin method of first order, with Runge-Kutta time stepping of first order (also called Euler method), thus we're expecting linear convergence.

In figure 6.1 we see the results of seven simulation runs with growing number of degrees of freedom.

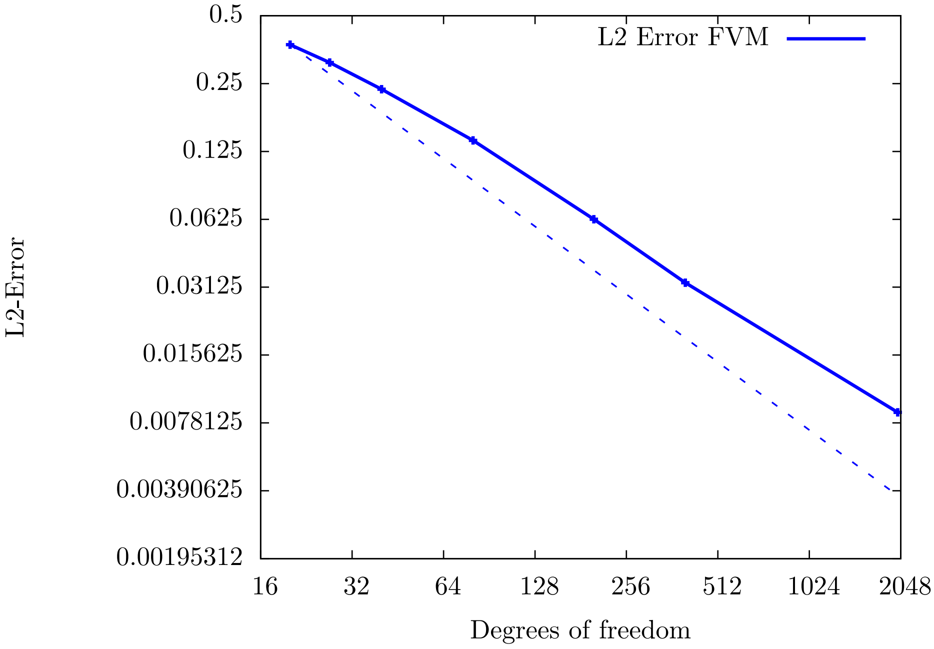


Figure 10: Error analysis for the finite volume method on a logarithmic coordinate system. The dotted graph shows the expected linear error growth due to the first measured error

The graph shows that the expected linear convergence order is almost reached. As the error is greater than the element size it won't reach machine presition for this example.

6.2 The Discontinuous Galerkin method with Runge Kutta time stepping

The order of convergence of the discontinuous Galerkin method depends on the spatial order of convergence of the elements, thus the number of nodes N in a single element and the temporal order, the order of the numerical solver of the ordinary differential equation of the semi discrete scheme we developed in section 3.

In the concrete implementation of section 3 we used the classical Runge-Kutta solver of order four, thus we're expecting an overall order of convergence of $\min(4, N)$.

In figure 11 are the results of simulations for different spatial orders N displayed.

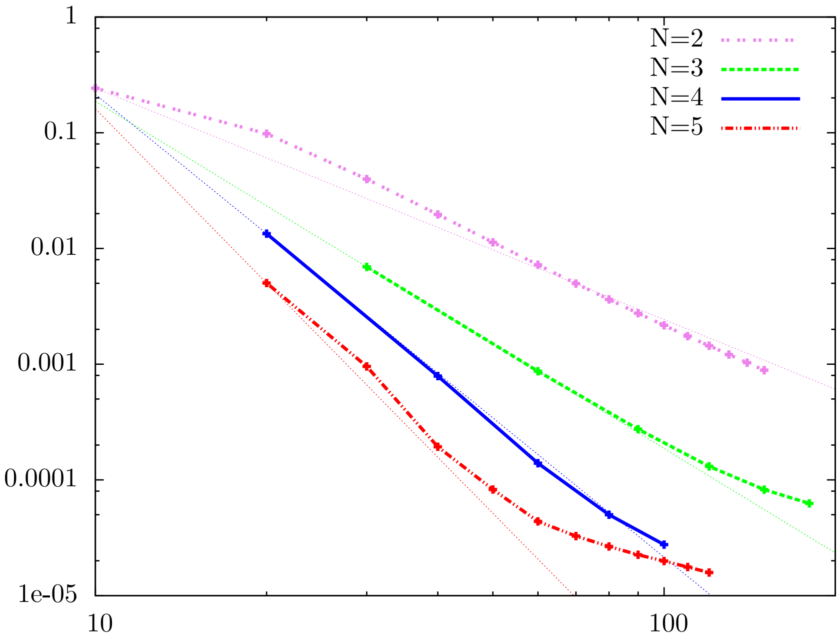


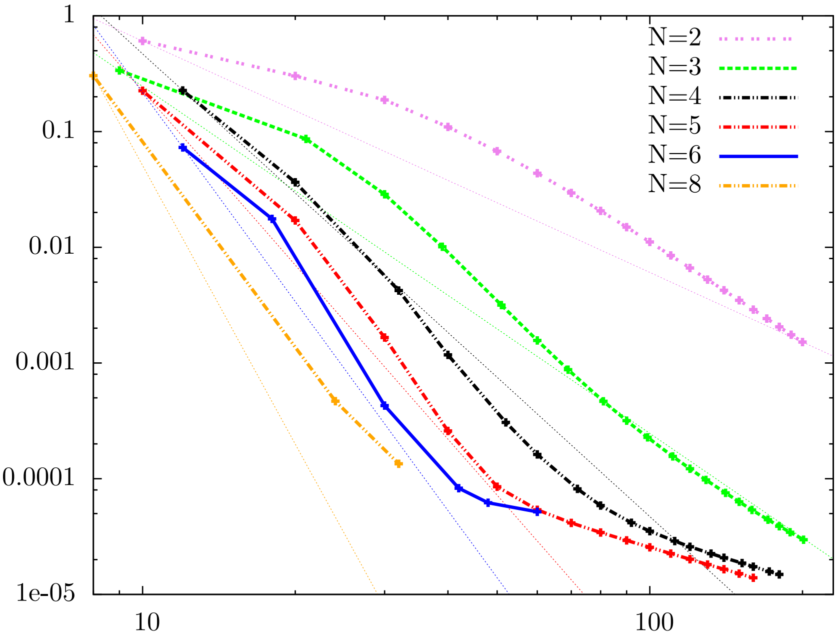
Figure 11: L2-Errors plotted for different spatial orders on a logarithmic coordinate system

In comparison to the ideal graph, spatial orders of less or equal four have the expected order of convergence.

For orders higher than four the error graph is parallel to the fourth order, and falls with growing degrees of freedom, to an even lower order of convergence. A reason why the convergence for the fifth order error graph drops on an order lower than four was not found.

6.3 The ADER DG method

For the ADER DG method we expect convergence equal to the minimum of the number of nodes in time and nodes in space according to chapter 4. In figure 6.3 is an illustration of the simulation errors.



As we see the results aren't as accurate as in the previous analysis. But the error-graphs are oriented on the optimal graphs and reach the wished order of convergence with growing degrees of freedom. One possible reason to this diffuse results could be the iterator scheme of section 4 for which no distinct statement is given on the needed number of iterations (in this analysis we used 3 steps).

7 Conclusion

In this thesis I gave an comprehensive overview of the general theory of discontinuous Galerkin methods. Additionally to solving the time integral through Runge-Kutta methods in chapter 3, we encountered the ADER time stepping approach in chapter 4.

The options we have on defining the nodal basis and the impacts they have on the accuracy of the approximation were discussed in chapter 5. Especially the size and the sparsity patterns of the mass, stiffness an flux matrices were analyzed in this chapter. As a result we acquired the nodal basis on Legendre nodes on which we gain diagonal mass matrices and a stiffness matrix in block-structure.

For other bases which can't be translated to the nodal basis, the same analysis has to be performed. Finding a set of orthogonal functions, which are also orthogonal to their derivatives, would directly lead to diagonal mass and stiffness matrices. Additionally sparse matrices can be gained trough inexact integration, as long as its error has the same magnitude as the error of the approximation of the numeric solution.

In the last chapter 6 we analyzed the errors of the methods for the simple advection equation. The orders of convergence we expected through theory could be confirmed for the finite volume method and the discontinuous Galerkin with Runge-Kutta time stepping. For the ADER approach we received less optimal convergence results. At this point further and more detailed analysis has to be performed. Furthermore the iterator function of section 4 has to be examined to give a distinct statement on the correlation between the number of iterations and the accuracy of the result.

Additionally a direct comparison between the number of operations needed with Runge-Kutta time-stepping and ADER time stepping has to be topic of future work.

If the convergence orders can be confirmed in upcoming testings, the ADER time stepping extends the toolbox to solve the time ODE with a promising alternative.

References

[1] Peter Deuflhard and Folkmar Bornemann. *Numerische Mathematik 2*. Walter de Gruyter, 2008.

[2] Peter Deuflhard and Andreas Hohmann. *Numerische Mathematik 1*. Walter de Gruyter, 2008.

[3] Michael Dumbser, Olindo Zanotti, Arturo Hidalgo, and Dinshaw S. Balsara. Ader-weno finite volume schemes with space-time adaptive mesh refinement.

[4] Gerd Fischer. *Lernbuch Lineare Algebra und Analytische Geometrie*. VIEWEG+TEUBNER, 2011.

[5] Jan S. Hesthaven and Tim Warburton. *Nodal Discontinuous Galerkin Methods*. Springer, 2008.

[6] Randall J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Number 978-0521009249. Camebridge University Press, 2002.

[7] Eleuterio F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamic*. Springer, third edition edition, 2009.

[8] A. Turetsky. Interpolation theory in problems, 1968.

[9] H Ehlich und K Zeller. Auswertung der normen von interpolationsoperatoren. *Mathematische Annalen*, (164), 1966.

[10] O.C. Zienkiewicz and R. L. Taylor. *The finite element method*, volume 1. Butterworth-Heinemann, 2000.