



TECHNISCHE
UNIVERSITÄT
WIEN

DIPLOMARBEIT

Mapped Tent Pitching Method for Hyperbolic Conservation Laws

Ausgeführt am Institut für
Analysis und Scientific Computing
der Technischen Universität Wien

unter der Anleitung von
Univ.Prof. Dipl.-Ing. Dr.techn. Joachim SCHÖBERL

durch
Christoph WINTERSTEIGER BSc
Aumühlweg 1, 4812 Pinsdorf

October 21, 2015

Abstract

This thesis introduces a novel discretization technique for solving hyperbolic conservation laws, called *Mapped Tent Pitching* (MTP) scheme. A tent pitching algorithm creates space-time domains, called tents, by vertically erecting canopies over spatial vertex patches. The structure of such a tent pitched space-time mesh is exploited by the MTP scheme to map these tents to a reference domain with a space-time tensor product structure. These domains are spatially discretized with a high order discontinuous Galerkin method, combined with a time stepping method. To obtain a robust method an entropy viscosity regularization is applied. This MTP scheme is implemented as part of this thesis based on the finite element library Netgen/NGSolve and tested with challenging problems like the Euler equations.

Acknowledgments

First of all, I want to thank Prof. Joachim Schöberl for supervising and inspiring this thesis as well as for supporting and improving this work throughout the last year with his helpful remarks.

Many thanks go to my colleagues Matthias Hochsteger, Philip Lederer, Christoph Lehrenfeld and Gerhard Kitzler for lots of time-intensive discussions and for proofreading this thesis.

Finally I want to thank my girlfriend Stefanie and my parents for supporting me during my studies.

Contents

1	Hyperbolic conservation laws	1
1.1	Description of hyperbolic conservation laws	1
1.2	Classical, weak and entropy solutions	2
2	Mapped Tent Pitching method	5
2.1	Tent pitching algorithm	5
2.2	Mapping to a space-time cylinder	10
2.2.1	Transformation of the conservation law	10
2.2.2	Transformation of the entropy admissibility condition	15
2.3	Artificial viscosity on a tent	16
3	Discontinuous Galerkin Method	18
3.1	Discretization of the Conservation Law	18
3.2	Discretization of the artificial viscosity	20
3.2.1	Entropy viscosity regularization	22
4	Euler equations	23
4.1	Description of the Euler equations	23
4.2	Mapping to a space-time cylinder	24
4.3	Entropy and entropy flux	26
4.4	Numerical results	26
4.5	Implementational aspects	27
5	Burgers equation	30
5.1	Description of the Burgers equation	30
5.2	Mapping to a space-time cylinder	30
5.3	Entropy and entropy flux	31
5.4	Numerical results	31
6	Wave equation	34
6.1	Description of the wave equation	34
6.2	Mapping to a space-time cylinder	34
6.3	Numerical results	35

Introduction

The aim of this thesis is to present a new class of methods for numerically solving hyperbolic conservation laws, called Mapped Tent Pitching (MTP) schemes. Conservation laws are systems of partial differential equations, which describe the conservation of extensive quantities, such as mass, momentum or energy. There exist various physical problems in mechanics, thermodynamics, electrodynamics and so on, which fit into this framework. Considering one of these problems, the MTP schemes can be seen as fully explicit or locally implicit schemes on unstructured space-time meshes, which are constructed by a meshing process known in the literature as tent pitching [EGSÜ05, ÜS02]. The main focus of this thesis lies on nonlinear problems (e.g. Euler equations, Burgers equation), which are solved with fully explicit MTP schemes. Locally implicit MTP schemes have their strength at linear problems, but they will not be discussed in this thesis. Based on a spatial triangulation, the meshing process creates an advancing front by vertically erecting canopies over spatial vertex patches, respecting the causality constraints of the hyperbolic problem. The resulting space-time domains are called tents. This thesis does not focus on the meshing itself, rather on a novel discretization technique that exploits tent pitched meshes.

Common discretization techniques for tent pitched meshes are space-time discontinuous Galerkin (SDG) methods. They are used in engineering applications [MH08, PHJ04, YAS⁺00] as well as in numerical analysis [FR99, MR05]. SDG schemes use a discontinuous Galerkin (DG) method to discretize the space-time variational formulation. The solution can be obtained by numerically solving the hyperbolic problem tent by tent, in an order that respects the causality. Thus the complexity is $\mathcal{O}(N_{\text{tents}})$, where N_{tents} denotes the number of tents. For higher order SDG approximations, the space-time problem on each tent gets very big.

In MTP schemes tents are mapped to a space-time reference domain with a space-time tensor product structure. This allows for a separation of the spatial and temporal discretization, in contrast to SDG schemes. On the space-time reference domain a high order DG method in space is combined with a time stepping method.

This leads to the advantages of MTP schemes:

- More efficient than SDG schemes as the tensor product structure is exploited
- High order in space and time
- Larger time step size, compared to time stepping methods on globally discretized spatial domains
- Reduced size of local problems in contrast to SDG schemes

Outline of the thesis:

In chapter 1 we introduce hyperbolic conservation laws, briefly specifying the notion of hyperbolicity and defining various notions of solutions. The concept of MTP schemes is presented in chapter 2. There we explain the construction of the space-time tent pitched mesh. A crucial part is how the MTP scheme exploits the structure of such meshes to map a tent to a space-time cylinder, which leads to a transformed conservation law on this cylinder. We discuss this transformation and formulate it for general conservation laws. This includes important aspects such as entropy viscosity regularization and its translation to MTP schemes. The transformed conservation law on the cylinder is spatially discretized with a high order DG method, which is discussed in chapter 3. The second part of chapter 3 presents details on the entropy viscosity regularization. In chapter 4 we apply the MTP scheme to the Euler equations and give details on the transformation. Further we show numerical results of the Burgers equation in chapter 5 and the wave equation in chapter 6, where we investigate the convergence rates of the MTP scheme.

Implementation:

The implementation of an fully explicit MTP scheme was done as part of this thesis based on the finite element library Netgen/NGSolve. It was used for the presented numerical results in chapters 4-6.

1 Hyperbolic conservation laws

To get a better idea of the problems discussed in this thesis, we define what hyperbolicity means for systems of partial differential equations. Further we describe various notions of solutions for such systems and the relations between them.

1.1 Description of hyperbolic conservation laws

Let $\Omega \subset \mathbb{R}^N$ be a bounded Lipschitz domain, which we denote as spatial domain. We consider the *conservation law*:

Find $u : \Omega \times (0, T] \rightarrow \mathbb{R}^n$ such that

$$\frac{\partial}{\partial t} u(x, t) + \operatorname{div}_x f(x, t, u(x, t)) = 0 \quad \forall (x, t) \in \Omega \times (0, T], \quad (1.1a)$$

$$u(x, 0) = u_0(x) \quad \forall x \in \Omega, \quad (1.1b)$$

with the given flux function

$$\begin{aligned} f : \Omega \times (0, T] \times \mathbb{R}^n &\longrightarrow \mathbb{R}^{n \times N}, \\ (x, t, u(x, t)) &\longmapsto f(x, t, u(x, t)), \end{aligned}$$

some initial data u_0 and appropriate boundary conditions on the inflow boundary. The $\operatorname{div}(\cdot)$ -operator acts row-wise in case of a matrix-valued function f . We call the system (1.1) *hyperbolic in the t -direction* [Daf10], if the matrix

$$\sum_{i=1}^N \nu_i D_u f_i \quad (1.2)$$

has real eigenvalues $\lambda_1, \dots, \lambda_n$ and corresponding linearly independent eigenvectors for all fixed x, t and all directions $\nu \in S^{N-1} = \{x \in \mathbb{R}^N : |x| = 1\}$. $D_u f_i$ denotes the $n \times n$ derivative matrix

$$D_u f_i = \begin{pmatrix} \frac{\partial f_{1i}}{\partial u_1} & \cdots & \frac{\partial f_{1i}}{\partial u_n} \\ \vdots & & \vdots \\ \frac{\partial f_{ni}}{\partial u_1} & \cdots & \frac{\partial f_{ni}}{\partial u_n} \end{pmatrix} \quad (1.3)$$

of the i -th column of the flux function. The eigenvalues λ_i are also called *characteristic speeds*. Let $c(x, t, u)$ be the maximum of these characteristic speeds in all directions.

1.2 Classical, weak and entropy solutions [Daf10, Ser99]

A *classical solution* of (1.1) is a \mathcal{C}^1 -function u , which satisfies (1.1) pointwise. An *entropy pair* (E, F) consists of an *entropy* E and an *entropy flux* F , related by

$$D_u F_i = D_u E \, D_u f_i \quad \forall i \in \{1, \dots, N\}, \quad (1.4)$$

such that for any classical solution holds the conservation law

$$\frac{\partial}{\partial t} E(u(x, t)) + \operatorname{div}_x F(u(x, t)) = 0 \quad \forall (x, t) \in \Omega \times (0, T].$$

Classical solutions can be constructed using the method of characteristics. Let us consider the one-dimensional Burgers equation

$$\frac{\partial}{\partial t} u(x, t) + \frac{\partial}{\partial x} \left(\frac{1}{2} u(x, t)^2 \right) = 0 \quad \forall (x, t) \in \Omega \times (0, T], \quad (1.5a)$$

$$u(x, 0) = u_0(x) \quad \forall x \in \Omega. \quad (1.5b)$$

For a classical solution $u \in \mathcal{C}^1$ of (1.5), we obtain the characteristic curves as the integral curves $t \mapsto (X(t), t)$ of the differential equation

$$\frac{dX}{dt} = D_u f(u(X, t)) = u(X, t).$$

For this characteristic curve, we can calculate

$$\frac{d}{dt} u(X(t), t) = \frac{dX}{dt} \frac{\partial}{\partial x} u(X, t) + \frac{\partial}{\partial t} u(X, t) = \left(\frac{\partial u}{\partial t} + D_u f \frac{\partial u}{\partial x} \right) (X, t),$$

which is equivalent to the conservation law (1.5) for a classical solution. Thus the solution u is constant on the characteristic curves $X(t)$. Due to that fact, the characteristic curves are straight lines and their slope is dependent to the initial value u_0 . For our example (1.5) the characteristics are defined by

$$X(t) = x_0 + u_0(x_0)t.$$

Now we consider two particular examples for the initial data u_0 , where the method of characteristics breaks down. For the initial data (1.6a), the characteristic lines intersect and this would lead to a multivalued solution in the gray area (cf. Figure 1.1a). Another problem occurs for the initial data (1.6b), where no classical solution exists in the gray area (cf. Figure 1.1b).

$$u_0(x) = \begin{cases} 1, & x < \frac{1}{2} \\ 0, & x \geq \frac{1}{2} \end{cases} \quad (1.6a)$$

$$u_0(x) = \begin{cases} 0, & x < \frac{1}{2} \\ 1, & x \geq \frac{1}{2} \end{cases} \quad (1.6b)$$

1 Hyperbolic conservation laws

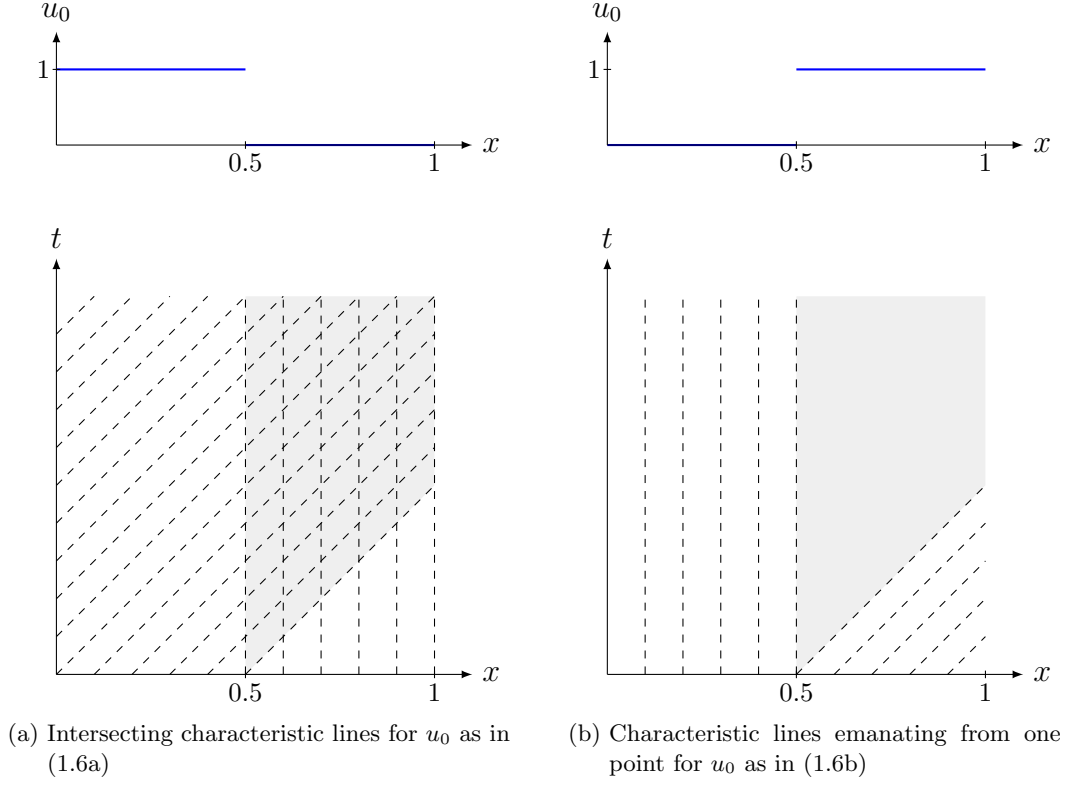


Figure 1.1: Characteristic lines of the Burgers equation with initial data u_0

This leads us to a more general class of solutions. A function u is called *weak solution* of (1.1), if it satisfies

$$\int_0^T \int_{\Omega} \left(u \frac{\partial v}{\partial t} + f \nabla v \right) dx dt + \int_{\Omega} u_0(x) v(x, 0) dx = 0 \quad (1.7)$$

for every smooth test function v with compact support on $\Omega \times [0, T)$. Every classical solution is a weak solution, which can be observed by partial integration of (1.7).

Now we can define a unique weak solution (cf. Figure 1.2a)

$$u(x, t) = \begin{cases} 1, & x < \frac{1}{2} + \frac{1}{2} t \\ 0, & x \geq \frac{1}{2} + \frac{1}{2} t \end{cases} \quad (1.8)$$

of the example (1.5) with the initial data (1.6a), by replacing the intersecting characteristics with a discontinuity. The velocity of this discontinuity is defined by the mean value of the two initial velocities.

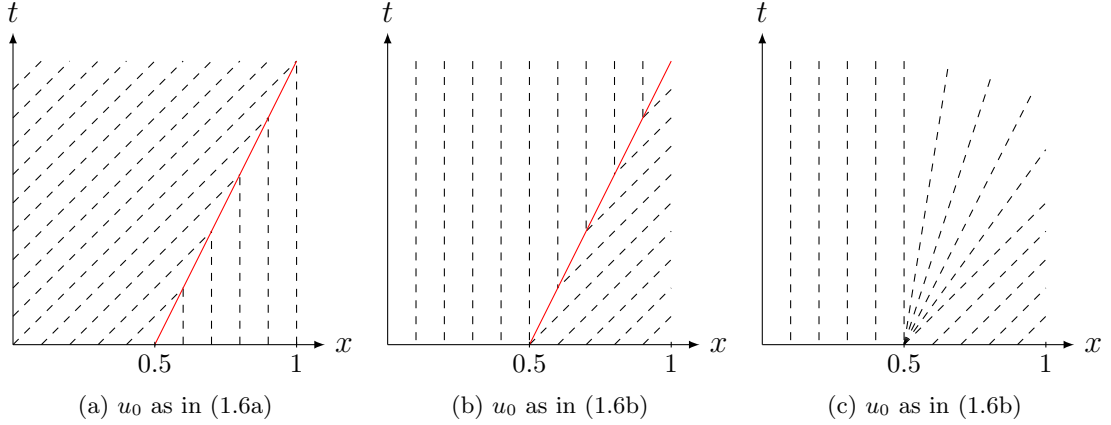


Figure 1.2: Weak solutions of the Burgers equation with initial data u_0

With the initial data (1.6b), we can easily find a discontinuous weak solution

$$u(x, t) = \begin{cases} 0, & x < \frac{1}{2} + \frac{1}{2}t \\ 1, & x \geq \frac{1}{2} + \frac{1}{2}t \end{cases} \quad (1.9a)$$

and a continuous one

$$u(x, t) = \begin{cases} 0, & x < \frac{1}{2} \\ \frac{2x-1}{2t}, & \frac{1}{2} \leq x < \frac{1}{2} + t \\ 1, & x \geq \frac{1}{2} + t \end{cases} \quad (1.9b)$$

which are shown in Figure 1.2b and 1.2c. To obtain a weak solution, which makes sense from the physical point of view, we have to find a criterion to select the continuous solution (1.9b) over discontinuous (1.9a). In other words, characteristics are just allowed to run into a discontinuity (cf. Figure 1.2a) and can not emanate from one (cf. Figure 1.2b).

Therefore we look for a weak solution u , which fulfills the *entropy admissibility condition*

$$\frac{\partial}{\partial t} E(u(x, t)) + \operatorname{div}_x F(u(x, t)) \leq 0 \quad (1.10)$$

in the sense of distributions on $\Omega \times [0, T]$. Such a solution is called *entropy solution*.

To determine the entropy solution, we consider the problem:

Find $u : \Omega \times (0, T] \rightarrow \mathbb{R}^n$ such that

$$\frac{\partial}{\partial t} u(x, t) + \operatorname{div}_x f(x, t, u(x, t)) = \nu \Delta u \quad \forall (x, t) \in \Omega \times (0, T], \quad (1.11a)$$

$$u(x, 0) = u_0(x) \quad \forall x \in \Omega, \quad (1.11b)$$

where $\nu \in \mathbb{R}$ is a small parameter. If the *viscosity solution* u_ν of (1.11) converges almost everywhere to a function u for $\nu \rightarrow 0^+$, then it can be shown that u is a weak solution which satisfies the entropy admissibility condition (1.10) [Daf10, 4.6.1 Theorem]. Thus the entropy solution can be calculated by solving (1.11) with a vanishing viscosity ν .

2 Mapped Tent Pitching method

This chapter discusses a novel discretization technique, called the Mapped Tent Pitching (MTP) method. The two main features are the use of a tent pitched (space-time) mesh and the mapping to a space-time reference domain. After motivating the MTP scheme, we explain the construction of the space-time mesh in section 2.1. Section 2.2 presents the mapping of the conservation law and the entropy admissibility condition (entropy inequality) to a space-time cylinder. Finally the viscosity operator for the artificial viscosity is introduced.

Motivation

Spatial discretization of the hyperbolic conservation law (1.1) leads to a system of ordinary differential equations. This system can then be solved with a time stepping method. Thereby global time step restrictions arise, which can be described by a Courant-Friedrichs-Levy (CFL) condition of the form

$$\Delta t \preccurlyeq \frac{h}{cp^2}.$$

The main restricting factors are the minimal mesh size h , the polynomial order p of the finite element space and the characteristic speeds c (as defined in section 1.1). To overcome these global restrictions for the time step, we introduce a MTP scheme. The crucial idea of MTP is that the structure of the tent pitched mesh is utilized to map the conservation law on each tent to a space-time cylinder. This leads to a transformed conservation law in a new variable on this cylinder, which can be spatially discretized with a suitable high order method (e.g. a discontinuous Galerkin method). The so obtained local system of ordinary differential equations can then be solved with any time stepping method. Due to the MTP scheme, we have to solve several small problems instead of one global problem. This allows for adapting the local time step size according to local quantities, such as the local mesh size h , the local polynomial order p of the spatial discretization and the locally highest characteristic speed c .

2.1 Tent pitching algorithm

This section describes how a tent pitching scheme advances the numerical solution in time. We assume that the spatial polygonal domain $\Omega \subset \mathbb{R}^N$ is meshed by a simplicial conforming finite element mesh \mathcal{T} and define $P_1(\mathcal{T})$ as set of continuous functions on $\tau : \Omega \rightarrow \mathbb{R}$, which are linear on each element $T \in \mathcal{T}$.

We denote the graph of the function $\tau_i \in P_1(\mathcal{T})$ as *advancing front* S_i at the i -th step of

2 Mapped Tent Pitching method

the tent pitching scheme. The numerical solution is then available for all $x \in \Omega$ and all $t \in (0, \tau_i(x))$. The function τ_i is fully defined by its values at the vertices v of the spatial mesh \mathcal{T} . Therefore just these values have to be updated to advance in time, which is done according to the following algorithm.

Initialize: Set $\tau_0 \equiv 0$, $S_0 = \Omega$ and the solution on S_0 to the initial data u_0 on Ω .

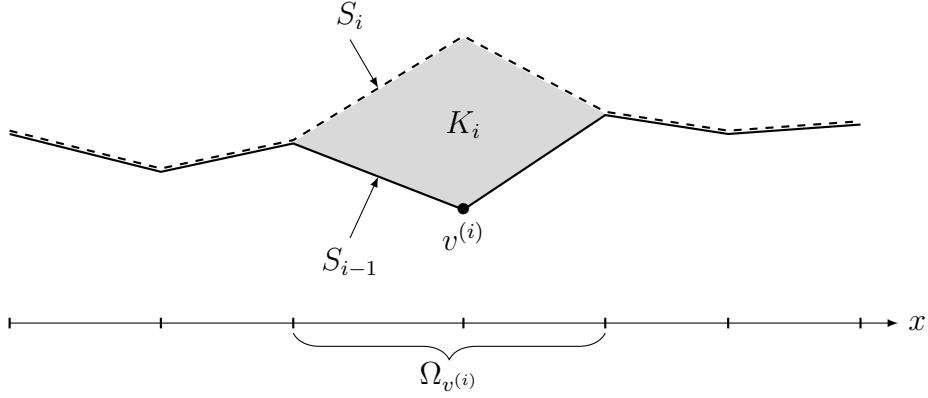
for $i = 1, 2, \dots$ **do:**

- Find a vertex $v^{(i)}$ where “good relative” progress in time can be made and calculate the *tent height* k_i , which denotes the time step at the vertex $v^{(i)}$. There are many ways to define the term “good relative” progress. One reasonable choice will be discussed in (2.7).
- Pitch a “space-time tent” K_i at the point $(v^{(i)}, \tau_{i-1}(v^{(i)}))$ on the advancing front S_{i-1} , where the solution is given. Set

$$\tau_i := \tau_{i-1} + k_i \eta_i, \quad (2.1)$$

with the function $\eta_i \in P_1(\mathcal{T})$, which is one at the vertex $v^{(i)}$ and zero at all other vertices of \mathcal{T} . We define the *vertex patch* $\Omega_{v^{(i)}}$ as the interior of the union of all simplices in \mathcal{T} connected to $v^{(i)}$. Then we obtain the tent

$$K_i := \{(x, t) : x \in \Omega_{v^{(i)}}, \tau_{i-1}(x) < t < \tau_i(x)\}.$$



- Solve the conservation law (1.1) on K_i with the solution on S_{i-1} as initial data. Use global boundary conditions on $\partial\Omega \times (0, t_{\max})$ as boundary condition on K_i for the case $v^{(i)} \in \partial\Omega$.
- If $\tau_i(v) \geq t_{\max}$ for all vertices v , then exit.

Algorithm 2.1: Advancing in time with a tent pitching algorithm

2 Mapped Tent Pitching method

Using this algorithm, we have to find $u(x, t)$ such that

$$\frac{\partial}{\partial t} u(x, t) + \operatorname{div}_x f(x, t, u(x, t)) = 0 \quad \forall (x, t) \in K_i, \quad (2.2a)$$

$$u(x, \tau_{i-1}(x)) = u_{i-1}(x, \tau_{i-1}(x)) \quad \forall x \in \Omega_{v^{(i)}}, \quad (2.2b)$$

where u_{i-1} denotes the solution on the advancing front S_{i-1} . To ensure that (2.2) is numerically solvable, the tent height k_i in each step should be determined in consideration of a local CFL condition. For the choice of the vertex $v^{(i)}$ we should also take the height of the neighboring vertices into account, to avoid that some vertices advance much faster in time than others. There exist various advancing front meshing strategies, which have been studied by other authors [EGSÜ05, ÜS02]. We want to use the following simple strategy [GSW].

First of all, we need an approximation of the maximal characteristic speed at a point $(x, \tau_{i-1}(x))$ on the advancing front S_{i-1} . We denote this approximation as $\bar{c}(x, \tau_{i-1}(x), u(x, \tau_{i-1}(x)))$, which could be dependent on the numerical solution u itself. For simplicity, we assume that $\bar{c} = \bar{c}(x)$ is just dependent on the spatial variable x . Now we want to ensure that the CFL condition

$$|\nabla \tau_i(x)| \leq \frac{1}{\bar{c}(x)} \quad (2.3)$$

holds for all $x \in \Omega$ and every step i of the scheme, where $|\cdot|$ denotes the Euclidean norm in \mathbb{R}^N . Since \bar{c} is independent of time, we can replace (2.3) with a more stringent CFL condition

$$|\nabla \tau_i|_T \leq \frac{1}{c_T} \quad \forall T \in \mathcal{T}, \quad (2.4)$$

with $c_T = \max_{x \in T} \bar{c}(x)$. For the implementation, we use the following sufficient condition

$$\frac{|\tau_i(e_1) - \tau_i(e_2)|}{|e|} \leq \frac{C_{\mathcal{T}}}{c_e} \quad \text{for all mesh edges } e \text{ in } \mathcal{T}, \quad (2.5)$$

where $C_{\mathcal{T}}$ is a constant that depends on the shape regularity of the spatial mesh \mathcal{T} . Further we denote the endpoints of the edge e with e_1 and e_2 , its length as $|e|$ and c_e as the maximal c_T over all elements which have e as edge. The left hand side of (2.5) is the gradient of τ_i restricted to the edge e . For a shape regular mesh, it is possible to find a suitable constant $C_{\mathcal{T}}$, such that (2.5) implies (2.4).

Now we can maintain a list of the possible time advance $\tilde{k}_l^{(i)}$ for each vertex v_l at the i -th step of the tent pitching scheme, which would not violate (2.5). The set of all mesh edges, which are connected to the vertex v_l , is denoted as \mathcal{E}_l and we assume that the endpoint $e_1 = v_l$ for all $e \in \mathcal{E}_l$. If our CFL condition (2.5) holds for τ_i , then it stays valid for τ_{i+1} , if we ensure that

$$\frac{|\tau_i(e_1) + \tilde{k}_l^{(i)} - \tau_i(e_2)|}{|e|} \leq \frac{C_{\mathcal{T}}}{c_e} \quad (2.6)$$

2 Mapped Tent Pitching method

is satisfied for all $e \in \mathcal{E}_l$, while pitching a tent at $(v_l, \tau_i(v_l))$. Since we are just interested in a positive upper bound of $\tilde{k}_l^{(i)}$, we can omit the absolute value in (2.6). This leads to

$$\tilde{k}_l^{(i)} \leq \min_{e \in \mathcal{E}_l} \left(\tau_i(e_2) - \tau_i(v_l) + |e| \frac{C_{\mathcal{T}}}{c_e} \right).$$

For our algorithm, we define the reference height

$$r_l := \min_{e \in \mathcal{E}_l} |e| \frac{C_{\mathcal{T}}}{c_e}$$

for each vertex v_l of the mesh \mathcal{T} , which corresponds to the maximal tent height on a flat advancing front $\tau_i \equiv \text{const}$. Now we can define a set of vertex indices

$$J_i := \left\{ l : \tilde{k}_l^{(i)} \geq \gamma r_l \right\}, \quad (2.7)$$

where good relative progress can be made at the i -th step of our scheme. The constant $\gamma \in (0, 1)$ can be chosen small to obtain moderate progress in time at many vertices. This means that a vertex v_l is already marked to be ready, when the potential tent height is a relatively low percentage of the optimal height r_l . A high γ leads to a more aggressive progress in time at fewer vertices. Usually γ is set to $\frac{1}{2}$. Algorithm 2.2 describes this maintaining process and can be used to precompute the tents for the time interval $(0, t_{\max})$. Since the spacial discretization does not change during this process, we can reuse these tents to propagate the solution from t_{\max} to $2t_{\max}$. Meaning that we can discretize the time domain with equidistant time slabs and just have to compute the tents for such a slab once. Figures 2.1 and 2.2 show such time slabs.

Initialize: Set $\tau_0 \equiv 0$, $\tilde{k}_l^{(0)} = r_l$ and $J_0 = \{1, 2, \dots, N_{\mathcal{T}}\}$, where $N_{\mathcal{T}}$ is the number of vertices in the mesh \mathcal{T} .

for $i = 1, 2, \dots$ **do:**

- Pick any vertex index $l_* \in J_{i-1}$.
- Set $v^{(i)} = v_{l_*}$ and $k_i = \tilde{k}_{l_*}^{(i-1)}$.
- Update τ_i by (2.1).
- Set $\tilde{k}_l^{(i)} = \tilde{k}_l^{(i-1)}$ for all vertices v_l and update the values for the vertices v_l adjacent to $v^{(i)}$ by

$$\tilde{k}_l^{(i)} = \min \left(t_{\max} - \tau_i(v_l), \min_{e \in \mathcal{E}_l} \left(\tau_i(e_2) - \tau_i(v_l) + |e| \frac{C_{\mathcal{T}}}{c_e} \right) \right).$$

- Set J_i according to (2.7).

Algorithm 2.2: Updating potential tent heights

At the beginning of the tent pitching algorithm, all vertices are marked as ready vertices (see Algorithm 2.2), since we could pitch a tent of optimal height at every vertex. An

2 Mapped Tent Pitching method

important part is now, how to pick a vertex from the list J_i of ready vertices. The simplest choice would be to take always the first vertex index in the list. This would lead to the tents shown in Figure 2.1, where the solution on tent T_{i-1} is needed for solving the problem on tent T_i . Therefore the calculation is poorly parallelizable.

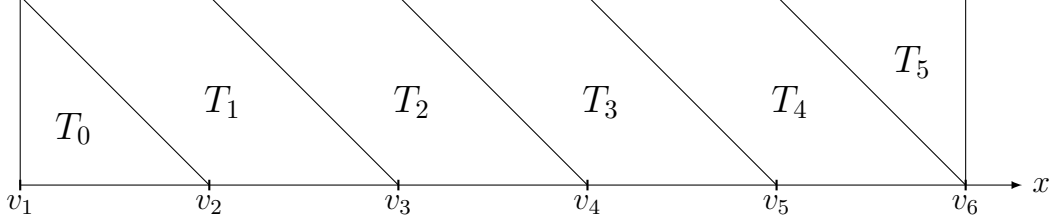


Figure 2.1: Poorly parallelizable tents in 1D

To solve this issue, we define the level of a tent pitched at vertex $v^{(i)}$ as

$$L_i := 1 + \max_{j \in N(v^{(i)})} L_j, \quad (2.8)$$

where $N(v^{(i)})$ is the set of vertex indices, which are connected to $v^{(i)}$ with an edge e . With this definition, the levels of the tents in Figure 2.1 are $L_i = i$. Now we can state more precisely how to choose the next vertex for pitching a tent. At the i -th step, we choose the index $l_* \in J_i$ such that the level of the resulting tent at $v^{(i)} = v_{l_*}$ is minimal. This would lead to the tents shown in Figure 2.2, with the level zero tents $\{T_0, T_1, T_2\}$ and level one tents $\{T_3, T_4, T_5\}$, where all tents of one level can be solved independently.

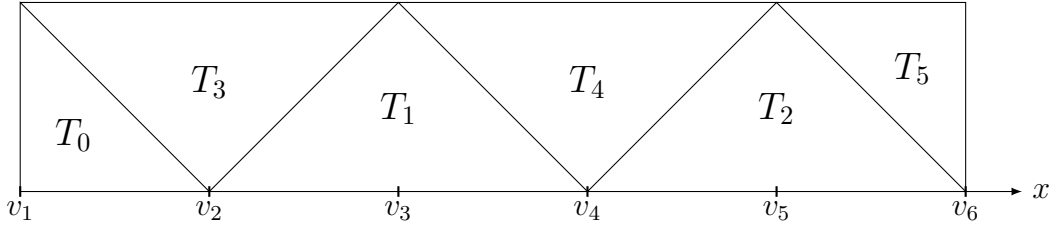


Figure 2.2: Parallelizable tents 1D

Figure 2.3 shows the levels of the tents in the two-dimensional case, where the tents in light gray have the level stated in the caption.

2 Mapped Tent Pitching method

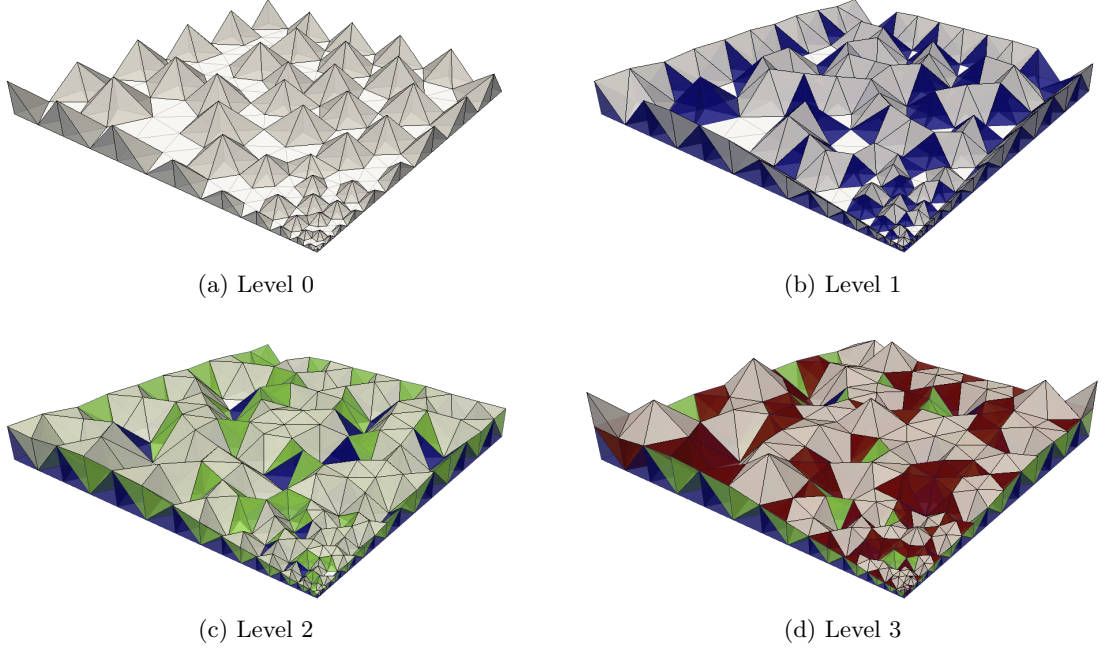


Figure 2.3: Level of tents over a two-dimensional mesh

2.2 Mapping to a space-time cylinder

Due to the tent pitching algorithm, we reduced the global problem (1.1) to several small problems of the following type. Find $u : K_i \rightarrow \mathbb{R}^n$ such that

$$\frac{\partial}{\partial t} u(x, t) + \operatorname{div}_x f(x, t, u(x, t)) = 0 \quad \forall (x, t) \in K_i, \quad (2.9a)$$

$$u(x, \tau_{i-1}(x)) = u_{i-1}(x, \tau_{i-1}(x)) \quad \forall x \in \Omega_{v(i)}. \quad (2.9b)$$

2.2.1 Transformation of the conservation law

For the actual calculation we want to map each tent K_i to a space-time cylinder $\hat{K}_i := \Omega_{v(i)} \times (0, 1)$ over the vertex patch $\Omega_{v(i)}$. Therefore we use a Duffy-like transformation

$$\begin{aligned} \Phi : \hat{K}_i &\longrightarrow K_i, \\ (x, \hat{t}) &\longmapsto (x, \varphi(x, \hat{t})), \end{aligned}$$

with

$$\varphi(x, \hat{t}) := (1 - \hat{t}) \tau_{i-1}(x) + \hat{t} \tau_i(x). \quad (2.10)$$

In the one-dimensional case, the constant \hat{t}^* is mapped to a piecewise linear continuous function $\varphi(x, \hat{t}^*)$ (cf. Figure 2.4). For higher space dimensions, the constant quasi-time \hat{t}^* is mapped to a non-smooth space-time manifold.

2 Mapped Tent Pitching method

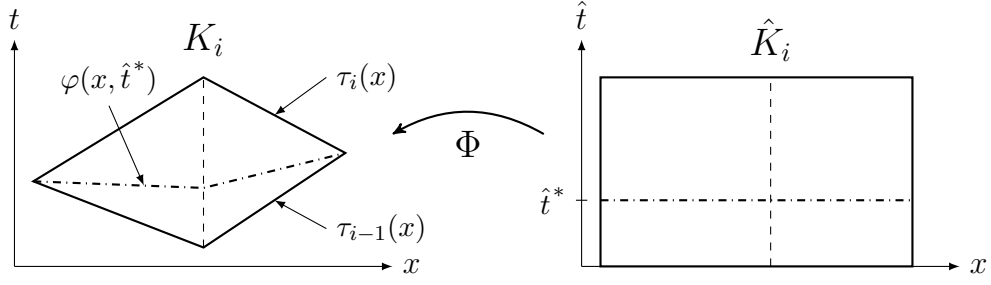


Figure 2.4: Mapping to space-time cylinder in 1D

First of all, we rewrite (2.9a) as

$$\operatorname{div}_{(x,t)} F(x, t, u(x, t)) = 0 \quad \forall (x, t) \in K_i, \quad (2.11)$$

where F is defined as

$$F(x, t, u(x, t)) := (f(x, t, u(x, t)), u(x, t)) \in \mathbb{R}^{n \times (N+1)}.$$

To preserve the structure of our conservation equation, we will use a transformation, which maps the divergence with respect to (x, t) to a divergence with respect to our reference variables (x, \hat{t}) .

Lemma 2.2.1. *Let $\Phi : \hat{K} \rightarrow K$ be a continuous and piecewise continuously differentiable, invertible and surjective mapping. For a given vector function \hat{w} the Piola transformation is defined as*

$$w \circ \Phi := \frac{1}{\det(\hat{D}\Phi)} \hat{D}\Phi \hat{w} \quad (2.12)$$

and there holds

$$(\operatorname{div}_x w) \circ \Phi = \frac{1}{\det(\hat{D}\Phi)} \operatorname{div}_{\hat{x}} \hat{w}. \quad (2.13)$$

Proof. The following proof is taken from [Mon03, Lemma 3.59].

Let $v \in \mathbb{C}_0^\infty(K)$ and $\hat{v} = v \circ \Phi$, then there holds

$$\nabla v \circ \Phi = [\hat{D}\Phi]^{-\top} \hat{\nabla} \hat{v}. \quad (2.14)$$

To show (2.13), we transform the following integral, using the Piola transformation (2.12) and the transformed gradient (2.14).

$$\begin{aligned} \int_K \operatorname{div}_x w \, v \, dx &= - \int_K w \cdot \nabla v \, dx \\ &= - \int_{\hat{K}} \frac{1}{\det(\hat{D}\Phi)} \left(\hat{D}\Phi \hat{w} \right) \cdot \left([\hat{D}\Phi]^{-\top} \hat{\nabla} \hat{v} \right) |\det(\hat{D}\Phi)| \, d\hat{x} \\ &= - \int_{\hat{K}} \frac{|\det(\hat{D}\Phi)|}{\det(\hat{D}\Phi)} \hat{w} \cdot \hat{\nabla} \hat{v} \, d\hat{x} \\ &= \int_{\hat{K}} \frac{|\det(\hat{D}\Phi)|}{\det(\hat{D}\Phi)} \operatorname{div}_{\hat{x}} \hat{w} \, \hat{v} \, d\hat{x} \end{aligned}$$

2 Mapped Tent Pitching method

By treating $\operatorname{div}_{\hat{x}} \hat{w}$ as scalar function, we can transform the integral back to K and we obtain

$$\int_K \operatorname{div}_x w v \, dx = \int_{\hat{K}} \frac{1}{\det(\hat{D}\Phi)} (\operatorname{div}_{\hat{x}} \hat{w}) \circ \Phi^{-1} v \, dx \quad (2.15)$$

Since (2.15) holds for all $v \in \mathbb{C}_0^\infty(K)$, the statement (2.13) is proven. \square

Now we want to transform (2.11) by using Lemma 2.2.1. According to (2.12), we define the function \hat{F}_l on \hat{K}_i as

$$\hat{F}_l := \det(\hat{D}\Phi) [\hat{D}\Phi]^{-1} F_l \circ \Phi,$$

where $F_l = (f_l, u_l)^\top$ is the transposed l -th row of the function F . Thus we obtain the equality

$$\frac{1}{\det(\hat{D}\Phi)} \operatorname{div}_{(x,\hat{t})} \hat{F}_l(x, \hat{t}, u(\Phi(x, \hat{t}))) = [\operatorname{div}_{(x,t)} F_l(x, t, u(x, t))] \circ \Phi = 0,$$

which leads to the conservation equation

$$\operatorname{div}_{(x,\hat{t})} \hat{F}_l(x, \hat{t}, u(\Phi(x, \hat{t}))) = 0 \quad \forall (x, \hat{t}) \in \hat{K}_i. \quad (2.16)$$

Now we take a closer look at the explicit form of \hat{F}_l . Therefore we need the Jacobian matrix $\hat{D}\Phi(x, \hat{t})$ and its inverse

$$\hat{D}\Phi(x, \hat{t}) = \begin{pmatrix} I & 0 \\ [\nabla \varphi(x, \hat{t})]^\top & \delta(x) \end{pmatrix}, \quad [\hat{D}\Phi(x, \hat{t})]^{-1} = \frac{1}{\delta(x)} \begin{pmatrix} \delta(x) I & 0 \\ -[\nabla \varphi(x, \hat{t})]^\top & 1 \end{pmatrix},$$

with the identity matrix $I \in \mathbb{R}^{N \times N}$ and

$$\delta(x) := \frac{\partial}{\partial \hat{t}} \varphi(x, \hat{t}) \stackrel{(2.10)}{=} \tau_i(x) - \tau_{i-1}(x). \quad (2.17)$$

The function δ describes the “height” of the tent and is zero on the boundary of the vertex patch $\partial\Omega_{v(i)}$ (cf. Figure 2.5).

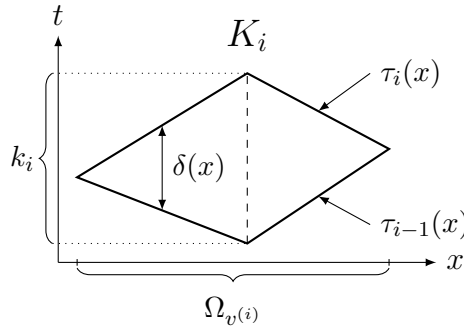


Figure 2.5: Tent in 1D

2 Mapped Tent Pitching method

Together with $\det(\hat{D}\Phi) = \delta$, we obtain

$$\begin{aligned}\hat{F}_l &= \det(\hat{D}\Phi)[\hat{D}\Phi]^{-1}(F_l \circ \Phi) \\ &= \delta \frac{1}{\delta} \begin{pmatrix} \delta I & 0 \\ -\hat{\nabla}\varphi^\top & 1 \end{pmatrix} \begin{pmatrix} f_l \circ \Phi \\ u_l \circ \Phi \end{pmatrix} \\ &= \begin{pmatrix} \delta f_l \\ u_l - \nabla\varphi^\top f_l \end{pmatrix} \circ \Phi.\end{aligned}$$

For simplicity we write $f_l(x, \hat{t})$ and $u_l(x, \hat{t})$ instead of $f_l(x, t) \circ \Phi$ and $u_l(x, t) \circ \Phi$. The above equation and (2.16) leads to

$$\frac{\partial}{\partial \hat{t}} [u_l - \nabla\varphi^\top f_l] + \operatorname{div}_x [\delta f_l] = 0 \quad \forall l \in \{1, \dots, n\} \forall (x, t) \in \hat{K}_i, \quad (2.18)$$

which is equivalent to

$$\frac{\partial}{\partial \hat{t}} [u(x, \hat{t}) - f(x, \hat{t}, u(x, \hat{t})) \nabla\varphi(x, \hat{t})] + \operatorname{div}_x [\delta(x) f(x, \hat{t}, u(x, \hat{t}))] = 0 \quad \forall (x, \hat{t}) \in \hat{K}_i.$$

With the definition

$$\hat{u}(x, \hat{t}) := u(x, \hat{t}) - f(x, \hat{t}, u(x, \hat{t})) \nabla\varphi(x, \hat{t}), \quad (2.19)$$

we obtain

$$\frac{\partial}{\partial \hat{t}} \hat{u}(x, \hat{t}) + \operatorname{div}_x [\delta(x) f(x, \hat{t}, u(x, \hat{t}))] = 0,$$

which is a conservation equation in the new variable \hat{u} . The flux $f(x, \hat{t}, u)$ is still dependent on the old variable u and therefore we have to find a function $u(\hat{u})$. This means we have to solve (2.19) for u . Depending on the flux function $f(x, \hat{t}, u)$, it may not be possible to find an analytical solution, but Theorem 2.2.2 shows the existence of a unique solution.

Finally we end up with the equation

$$\frac{\partial}{\partial \hat{t}} \hat{u}(x, \hat{t}) + \operatorname{div}_x [\delta(x) f(x, \hat{t}, u(\hat{u}(x, \hat{t})))] = 0 \quad \forall (x, \hat{t}) \in \hat{K}_i. \quad (2.20)$$

To get a better idea of the function $u(\hat{u})$, we have a look at some examples.

Linear convection equation

As first example we consider a scalar convection equation for $u(x, t)$ with the flux function

$$f(x, t, u(x, t)) := b(x, t)u(x, t),$$

where $b(x, t) \in \mathbb{R}^N$ is a given velocity field.

To find the transformation of \hat{u} to u , we have to solve

$$\begin{aligned}\hat{u}(x, \hat{t}) &= u(x, \hat{t}) - f(x, \hat{t}, u(x, \hat{t})) \cdot \nabla\varphi(x, \hat{t}) \\ &= u(x, \hat{t}) - b(x, \hat{t})u(x, \hat{t}) \cdot \nabla\varphi(x, \hat{t}) \\ &= (1 - b(x, \hat{t}) \cdot \nabla\varphi(x, \hat{t})) u(x, \hat{t}).\end{aligned} \quad (2.21)$$

2 Mapped Tent Pitching method

At this point we can see that the choice of the tent height is crucial. To guarantee solvability of (2.21), we have to choose the slope of the tent small enough, such that

$$|\nabla\varphi(x, \hat{t})| < \frac{1}{|b(x, \hat{t})|} \quad \forall(x, \hat{t}) \in \hat{K}_i. \quad (2.22)$$

Here we have to note that (2.22) is again the known CFL condition (2.3). With this choice holds $b(x, \hat{t}) \cdot \nabla\varphi(x, \hat{t}) < 1$ and the function $u(\hat{u})$ is now defined by

$$u(x, \hat{t}, \hat{u}(x, \hat{t})) = \frac{\hat{u}(x, \hat{t})}{1 - b(x, \hat{t}) \cdot \nabla\varphi(x, \hat{t})}. \quad (2.23)$$

For $\nabla\varphi \rightarrow 0$, the functions \hat{u} and u should be identical, according to the definition (2.19). Therefore the function $u(x, \hat{t}, \hat{u}(x, \hat{t}))$ has to converge to $\hat{u}(x, \hat{t})$ for $\nabla\varphi \rightarrow 0$, which is fulfilled by our solution (2.23).

Burgers equation

An example with a nonlinear flux function f , where the function $u(\hat{u})$ can be found quite easily is the Burgers equation. It is also a scalar conservation law with f defined as

$$f(x, t, u(x, t)) := \frac{1}{2} [u(x, t)]^2 \mathbf{1}, \quad \mathbf{1} := \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Solving (2.19) with the flux function defined above, leads to

$$u(x, t, \hat{u}(x, t)) = \frac{2\hat{u}(x, t)}{1 + \sqrt{1 - 2(\mathbf{1} \cdot \nabla\varphi(x, t))\hat{u}(x, t)}}.$$

The derivation of the equation above is done in detail in section 5.2.

Arbitrary flux function

For some flux functions f it may not be possible to solve (2.19) explicitly. Therefore we will show in the following theorem that (2.19) has a unique solution u , if the slope of the tent is not too steep. This solution can be found using a fixed point iteration or the Newton's method.

Theorem 2.2.2. *Let be*

$$c := \sup_{\substack{\nu \in \mathbb{R}^N \\ |\nu|=1}} \rho \left(\sum_{i=1}^N \nu_i D_u f_i \right),$$

which corresponds to the maximal characteristic wave speed in all directions ν (see section 1.1). If there holds

$$|\nabla\varphi| < \frac{1}{c} \quad (2.24)$$

then (2.19) has a unique solution.

2 Mapped Tent Pitching method

Proof. First of all, we rewrite (2.19) as

$$u = \hat{u} - f(u) \nabla \varphi =: \Psi(u)$$

and define the function $\Psi(u)$. Solving (2.19) for u is now equivalent to finding a fixed point $u = \Psi(u)$ of Ψ . To be able to apply the Banach fixed point theorem, we have to show that Ψ is Lipschitz continuous with a constant $0 \leq L < 1$. Obviously there holds

$$L < \sup_{u \in \mathbb{R}^n} \|\mathrm{D}_u \Psi\|,$$

for a smooth flux function f . Now we have to bound $\mathrm{D}_u \Psi$ in a suitable norm. There holds

$$\mathrm{D}_u \Psi = \sum_{i=1}^N (\nabla \varphi)_i \mathrm{D}_u f_i \in \mathbb{R}^{N \times N}, \quad (2.25)$$

which is the scaled matrix in (1.2) with $\nu_i = (\nabla \varphi)_i$. Since we are considering hyperbolic systems, the matrix in (2.25) is diagonalizable and thus there exists a norm such that $\|\mathrm{D}_u \Psi\| = \rho(\mathrm{D}_u \Psi)$, where $\rho(\cdot)$ denotes the spectral radius. Using this norm, we obtain

$$\|\mathrm{D}_u \Psi\| = |\nabla \varphi| \rho \left(\sum_{i=1}^N \frac{(\nabla \varphi)_i}{|\nabla \varphi|} \mathrm{D}_u f_i \right) \leq |\nabla \varphi| c.$$

The bound (2.24) for $|\nabla \varphi|$ implies $\|\mathrm{D}_u \Psi\| < 1$ and a Lipschitz constant $L < 1$. Thus the function Ψ is a contraction and has, according to the Banach fixed point theorem, a unique fixed point u . □

2.2.2 Transformation of the entropy admissibility condition

As explained in section 1.1, the entropy solution of a conservation law has to fulfill the entropy admissibility condition

$$\frac{\partial}{\partial t} E(u(x, t)) + \operatorname{div}_x F(u(x, t)) \leq 0 \quad \forall (x, t) \in K_i. \quad (2.26)$$

To be able to do all the calculations on the reference space-time cylinder \hat{K}_i , we apply Lemma 2.2.1 to (2.26) and obtain

$$\frac{\partial}{\partial \hat{t}} [E(x, \hat{t}) - F(x, \hat{t}, u(x, \hat{t})) \nabla \varphi(x, \hat{t})] + \operatorname{div}_x [\delta(x) F(x, \hat{t}, u(x, \hat{t}))] \leq 0 \quad \forall (x, \hat{t}) \in \hat{K}_i,$$

where we use that $\det(\hat{\mathbf{D}}\Phi) = \delta(x)$ is positive. This means we found an entropy admissibility condition on the reference domain \hat{K}_i for the entropy pair

$$(\hat{E}, \hat{F}) := (E - F \nabla \varphi, \delta F). \quad (2.27)$$

The entropy pair (E, F) of the original conservation law in u is related by

$$\mathrm{D}_u F_i = \mathrm{D}_u E \mathrm{D}_u f_i \quad \forall i \in \{1, \dots, N\}, \quad (1.4)$$

2 Mapped Tent Pitching method

where f is the flux function and $D_u f_i$ denotes the $n \times n$ derivative matrix as defined in (1.3). Since we are interested in solving the conservation law (2.20) for \hat{u} on \hat{K}_i , we have to verify that (\hat{E}, \hat{F}) is an entropy pair for \hat{u} . Therefore we have to show

$$D_{\hat{u}} \hat{F}_i = D_{\hat{u}} \hat{E} D_{\hat{u}} \hat{f}_i \quad \forall i \in \{1, \dots, N\}, \quad (2.28)$$

with the flux function $\hat{f} = \delta f$ of the transformed equation (2.20) on \hat{K}_i . By using the definition (2.19) of \hat{u} , we obtain

$$I = D_{\hat{u}} \hat{u} = D_{\hat{u}} u - \sum_{i=1}^N (\nabla \varphi)_i D_{\hat{u}} f_i = \left[I - \sum_{i=1}^N (\nabla \varphi)_i D_u f_i \right] D_{\hat{u}} u, \quad (2.29)$$

with the identity matrix $I \in \mathbb{R}^{n \times n}$. Further we calculate the occurring derivatives in (2.28), using (2.27) and $\hat{f} = \delta f$:

$$\begin{aligned} D_{\hat{u}} \hat{E} &= \left[D_u E - \sum_{i=1}^N (\nabla \varphi)_i D_u F_i \right] D_{\hat{u}} u \\ D_{\hat{u}} \hat{f}_i &= \delta D_u f_i D_{\hat{u}} u \\ D_{\hat{u}} \hat{F}_i &= \delta D_u F_i D_{\hat{u}} u \end{aligned}$$

These derivatives lead to

$$\begin{aligned} D_{\hat{u}} \hat{E} D_{\hat{u}} \hat{f}_i &= \left[D_u E - \sum_{i=1}^N (\nabla \varphi)_i D_u F_i \right] D_{\hat{u}} u \delta D_u f_i D_{\hat{u}} u \\ &\stackrel{(1.4)}{=} D_u E \left[I - \sum_{i=1}^N (\nabla \varphi)_i D_u f_i \right] D_{\hat{u}} u \delta D_u f_i D_{\hat{u}} u \\ &\stackrel{(2.29)}{=} D_u E \delta D_u f_i D_{\hat{u}} u \\ &\stackrel{(1.4)}{=} \delta D_u F_i D_{\hat{u}} u = D_{\hat{u}} \hat{F}_i. \end{aligned}$$

Thus the entropy \hat{E} and the entropy flux \hat{F} , occurring in the transformed entropy admissibility condition (2.26), are indeed an entropy pair for the transformed variable \hat{u} .

2.3 Artificial viscosity on a tent

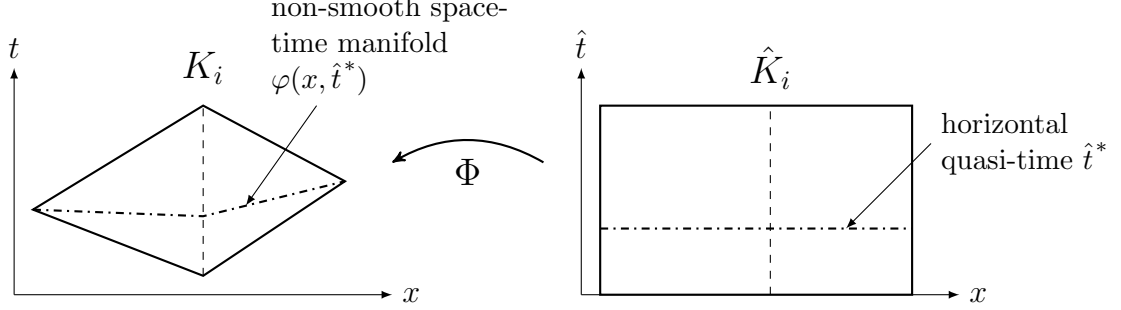
For some conservation laws, we have to add artificial viscosity to obtain the unique entropy solution. As described in section 1.1, we have to consider (1.11) instead of (1.1). In our case that means we have to find $\hat{u} : \hat{K}_i \rightarrow \mathbb{R}^n$ such that

$$\begin{aligned} \frac{\partial}{\partial \hat{t}} \hat{u}(x, \hat{t}) + \operatorname{div}_x [\delta(x) f(x, \hat{t}, u(\hat{u}(x, \hat{t})))] &= \\ \frac{\nu}{k_i} \operatorname{div}_x [\delta(x) \nabla u(\hat{u}(x, \hat{t}))] &\quad \forall (x, \hat{t}) \in \hat{K}_i, \quad (2.30a) \end{aligned}$$

$$\hat{u}(x, 0) = \hat{u}_{i-1}(x, 1) \quad \forall x \in \Omega_{v(i)}. \quad (2.30b)$$

2 Mapped Tent Pitching method

We add a Laplace operator on the horizontal quasi-time \hat{t}^* , which is then mapped to the non-smooth space-time manifold $\varphi(x, \hat{t}^*)$. The viscosity coefficient $\nu \in \mathbb{R}$ is a positive constant on one tent and divided by the tent height $k_i = \delta(v^{(i)})$.



Since we consider a conservation law in \hat{u} , a natural choice would be to apply the Laplace operator to the conserved quantity \hat{u} . The gradient of the non-smooth function $\varphi(x, \hat{t}^*)$ is discontinuous. When we now recall the definition

$$\hat{u} := u - f(u) \nabla \varphi, \quad (2.19)$$

it is easy to see that \hat{u} can be discontinuous, even for a smooth solution u . Thus we have to add a Laplace operator acting on $u(\hat{u})$ instead of \hat{u} to obtain the desired properties.

3 Discontinuous Galerkin Method

This chapter gives a brief overview of the spatial discretization that we apply on each space-time cylinder, a discontinuous Galerkin (DG) method. Due to the conservation property and the ability to handle discontinuities, these methods are commonly used for numerically solving hyperbolic problems [HW07, MR05, ZGMP13]. The locally defined polynomial basis leads to a block diagonal mass matrix, which makes the method highly parallelizable. When using higher polynomial degrees, these methods often produce spurious oscillations at discontinuities. This issue can be solved by adding artificial viscosity. To avoid getting a overly dissipative method, we add a nonlinear viscosity, which is based on the entropy production [GNPY14, GN14, ZGMP13, GPP11]. This leads to a large amount of viscosity at discontinuities and almost no viscosity in smooth regions. The robustness of this approach is indicated by numerical results.

3.1 Discretization of the Conservation Law

Let \mathcal{T} be the triangulation of the spatial domain Ω and \mathcal{E} the edges in \mathcal{T} . The *DG function space* is defined as

$$\mathcal{V} := \{v \in L_2(\Omega) : v|_T \in H^1(T) \ \forall T \in \mathcal{T}\} ,$$

which means that a function $v \in \mathcal{V}$ can be discontinuous across an edge $E \in \mathcal{E}$. For an element $T \in \mathcal{T}$, we define the *jump* across an edge $E = \partial T \cap \partial T'$

$$[[v]] := v|_T - v|_{T'} ,$$

where $T' \in \mathcal{T}$ is the neighbouring element.

Now, let us recall the conservation law on the reference domain \hat{K}_i

$$\frac{\partial}{\partial \hat{t}} \hat{u}(x, \hat{t}) + \operatorname{div}_x [\delta(x) f(x, \hat{t}, u(\hat{u}(x, \hat{t})))] = 0 \quad \forall (x, \hat{t}) \in \hat{K}_i . \quad (2.20)$$

Since (2.20) is only defined on the vertex patch $\Omega_{v(i)}$, we define the triangulation of the vertex patch as

$$\mathcal{T}_i := \{T \in \mathcal{T} : T \cap \Omega_{v(i)} \neq \emptyset\}$$

and the corresponding local DG space

$$\mathcal{V}_i := \{v \in L_2(\Omega_{v(i)}) : v|_T \in H^1(T) \ \forall T \in \mathcal{T}_i\} .$$

To obtain the weak formulation, we multiply (2.20) with a test function $v \in \mathcal{V}_i$ and integrate over the spatial domain

$$\sum_{T \in \mathcal{T}_i} \left(\int_T \frac{\partial}{\partial \hat{t}} \hat{u}(x, \hat{t}) v(x) + \int_T \operatorname{div}_x [\delta(x) f(x, \hat{t}, u(\hat{u}(x, \hat{t})))] v(x) \right) dx = 0 .$$

3 Discontinuous Galerkin Method

From now on, let \hat{t} denote an arbitrary, but fixed time $\hat{t} \in [0, 1]$ in our reference domain \hat{K}_i . For simplicity, we write \hat{u} instead of $\hat{u}(\cdot, \hat{t})$. Integration by parts in each component leads to

$$\sum_{T \in \mathcal{T}_i} \int_T \frac{\partial}{\partial \hat{t}} \hat{u} v + \sum_{T \in \mathcal{T}_i} \left(- \int_T \delta f(u(\hat{u})) \nabla v + \int_{\partial T} \delta f_\eta(u(\hat{u})) v \right) = 0,$$

where $f_\eta(u(\hat{u})) := f(u(\hat{u})) \eta$, with the outer normal vector η . We denote f_η as *numerical flux* and define $B : [\mathcal{V}_i]^n \times \mathcal{V}_i \rightarrow \mathbb{R}^n$ as

$$B(\hat{u}, v) := \sum_{T \in \mathcal{T}_i} \left(- \int_T \delta f(u(\hat{u})) \nabla v + \int_{\partial T} \delta f_\eta(u(\hat{u})) v \right). \quad (3.1)$$

Thus the weak formulation is: Find $\hat{u}(\cdot, \hat{t}) \in [\mathcal{V}_i]^n$, such that

$$\sum_{T \in \mathcal{T}_i} \int_T \frac{\partial}{\partial \hat{t}} \hat{u} v + B(\hat{u}, v) = 0 \quad \forall v \in \mathcal{V}_i. \quad (3.2)$$

Now we want to discretize (3.2). Therefore we define the *finite element space*

$$\mathcal{V}_{ih} := \{v \in L_2(\Omega_{v(i)}) : v|_T \in P^p(T) \ \forall T \in \mathcal{T}_i\} \subset \mathcal{V}_i$$

of order $p \in \mathbb{N}$. Let $\{\varphi_1, \dots, \varphi_m\}$ be a basis of \mathcal{V}_{ih} . Note that for each $k \in \{1, \dots, m\}$ exists just one element T , where the function φ_k is non-zero. Using this basis, we can find a coefficient matrix $\hat{c}(\hat{t}) \in \mathbb{R}^{m \times n}$, such that

$$\hat{u}_j(x, \hat{t}) \approx \hat{u}_{h,j}(x, \hat{t}) = \sum_{l=1}^m \hat{c}_{lj}(\hat{t}) \varphi_l(x) \quad \forall j \in \{1, \dots, n\}. \quad (3.3)$$

Each column of \hat{c} corresponds to a component of our solution vector \hat{u}_h . With this approximation, we can evaluate the integral in (3.2)

$$\begin{aligned} \sum_{T \in \mathcal{T}_i} \int_T \frac{\partial}{\partial \hat{t}} \hat{u}_{h,j}(x, \hat{t}) \varphi_k(x) &= \sum_{T \in \mathcal{T}_i} \int_T \frac{\partial}{\partial \hat{t}} \left(\sum_{l=1}^m \hat{c}_{lj}(\hat{t}) \varphi_l(x) \right) \varphi_k(x) \\ &= \sum_{l=1}^m \frac{\partial}{\partial \hat{t}} \hat{c}_{lj}(\hat{t}) \underbrace{\sum_{T \in \mathcal{T}_i} \int_T \varphi_l(x) \varphi_k(x)}_{=: M_{kl}} \end{aligned} \quad (3.4)$$

for all $k \in \{1, \dots, m\}$. This leads to the definition of the mass matrix $M \in \mathbb{R}^{m \times m}$

$$M_{kl} := \sum_{T \in \mathcal{T}_i} \int_T \varphi_k(x) \varphi_l(x) \quad \forall k, l \in \{1, \dots, m\}. \quad (3.5)$$

Each φ_k is defined as a polynomial up to order p on one element $T = T(k)$ and zero on all other elements in \mathcal{T}_i . Thus just one integral in (3.5) is non-zero. If the basis functions are ordered element-wise, we obtain a block diagonal mass matrix, which simplifies solving the resulting linear system.

3 Discontinuous Galerkin Method

Since the flux function f in (3.1) can be nonlinear, we cannot separate in spatial variables and time. Therefore we have to evaluate the function $B(\hat{u}_h(\cdot, \hat{t}), \varphi_k)$ for each time \hat{t} and we obtain the matrix

$$B(\hat{c}(\hat{t})) := \begin{pmatrix} B(\hat{u}_h(\cdot, \hat{t}), \varphi_1)^\top \\ \vdots \\ B(\hat{u}_h(\cdot, \hat{t}), \varphi_m)^\top \end{pmatrix} \in \mathbb{R}^{m \times n}. \quad (3.6)$$

Using the linear structure of (3.4) and (3.6) leads to the system of ordinary differential equations

$$M \frac{d\hat{c}}{d\hat{t}} + B(\hat{c}) = 0 \quad (3.7)$$

for the coefficient matrix $\hat{c}(\hat{t})$.

3.2 Discretization of the artificial viscosity

As explained in section 1.2, we have to consider the viscous conservation law

$$\frac{\partial}{\partial \hat{t}} \hat{u}(x, \hat{t}) + \operatorname{div}_x [\delta(x) f(x, \hat{t}, u(\hat{u}(x, \hat{t})))] = \frac{\nu}{k_i} \operatorname{div}_x [\delta(x) \nabla u(\hat{u}(x, \hat{t}))] \quad \forall (x, \hat{t}) \in \hat{K}_i, \quad (2.30)$$

instead of the conservation law (2.20) to obtain the entropy solution. The viscosity coefficient $\nu \in \mathbb{R}$ is a yet to be defined positive constant on \hat{K}_i and $k_i \in \mathbb{R}$ denotes the height of the tent. Since we already derived the weak formulation of the left hand side of (2.30) (see (3.2) in section 3.1), we now just consider the right hand side. Again, we multiply with a test function $v \in \mathcal{V}_i$ and integrate over the vertex patch $\Omega_{v(i)}$. Integration by parts leads to

$$\begin{aligned} \frac{\nu}{k_i} \sum_{T \in \mathcal{T}_i} \int_T \operatorname{div}_x [\delta \nabla u_j(\hat{u})] v = \\ \frac{\nu}{k_i} \sum_{T \in \mathcal{T}_i} \left(- \int_T \delta \nabla u_j(\hat{u}) \cdot \nabla v + \int_{\partial T} \delta \frac{\partial u_j(\hat{u})}{\partial n} v \right) \quad \forall j \in \{1, \dots, n\}, \end{aligned} \quad (3.8)$$

where \hat{u} again denotes $\hat{u}(\cdot, \hat{t})$, for a fixed time $\hat{t} \in [0, 1]$. For the further derivation, we assume that u is a smooth solution of the conservation law. Thus the normal derivative is continuous and

$$\frac{\nu}{k_i} \sum_{T \in \mathcal{T}_i} \left(- \int_T \delta \nabla u_j(\hat{u}) \cdot \nabla v + \frac{1}{2} \int_{\partial T} \delta \frac{\partial u_j(\hat{u})}{\partial n} \llbracket v \rrbracket \right) \quad \forall j \in \{1, \dots, n\} \quad (3.9)$$

is equivalent to the right hand side of (3.8). To show solvability of the resulting discrete problem, we need that (3.9) defines a symmetric and coercive bilinear form in $u_j(\hat{u})$ and

3 Discontinuous Galerkin Method

v for each j . Therefore we add two consistent terms ($\llbracket u_j(\hat{u}) \rrbracket = 0$ for a smooth solution) to (3.9) and define the bilinear form

$$a(u_j(\hat{u}), v) := \frac{\nu}{k_i} \sum_{T \in \mathcal{T}_i} \left(\int_T \delta \nabla u_j(\hat{u}) \cdot \nabla v - \frac{1}{2} \int_{\partial T} \delta \frac{\partial u_j(\hat{u})}{\partial n} \llbracket v \rrbracket - \frac{1}{2} \int_{\partial T} \delta \frac{\partial v}{\partial n} \llbracket u_j(\hat{u}) \rrbracket + \frac{\alpha}{2h} \int_{\partial T} \delta \llbracket u_j(\hat{u}) \rrbracket \llbracket v \rrbracket \right),$$

where $\alpha \in \mathbb{R}$ has to be chosen sufficiently large, to obtain coercivity on \mathcal{V}_i for each j . Using these bilinear forms, we define $A : [\mathcal{V}_i]^n \times \mathcal{V}_i \rightarrow \mathbb{R}^n$ as

$$A(\hat{u}, v) := \begin{pmatrix} a(u_1(\hat{u}(\cdot, \hat{t})), v) \\ \vdots \\ a(u_n(\hat{u}(\cdot, \hat{t})), v) \end{pmatrix} \in \mathbb{R}^n. \quad (3.10)$$

Together with (3.2), we obtain the weak formulation: Find $\hat{u}(\cdot, \hat{t}) \in [\mathcal{V}_i]^n$, such that

$$\sum_{T \in \mathcal{T}_i} \int_T \frac{\partial}{\partial \hat{t}} \hat{u} v + B(\hat{u}, v) + A(\hat{u}, v) = 0 \quad \forall v \in \mathcal{V}_i. \quad (3.11)$$

For the spatial discretization of (3.11), we again choose a basis $\{\varphi_i, \dots, \varphi_m\}$ of \mathcal{V}_{ih} . Similar to the approximation of \hat{u} in (3.3), we define the coefficient matrix $c(\hat{t}) \in \mathbb{R}^{m \times n}$ such that

$$u_j(x, \hat{t}) \approx u_{h,j}(x, \hat{t}) = \sum_{l=1}^m c_{lj}(\hat{t}) \varphi_l(x) \quad \forall j \in \{1, \dots, n\}.$$

These coefficients can be calculated by solving (2.19) (which gives a relation between \hat{u} and u) in the sense of an L^2 -projection. Using the approximation u_h , we can evaluate $A(\hat{u}_h, \varphi_k)$. Due to the linearity of $a(\cdot, \cdot)$, there holds

$$(A(\hat{u}_h, \varphi_k))_j = a(u_{h,j}(\hat{u}_h), \varphi_k) = \sum_{l=1}^m c_{lj}(\hat{t}) a(\varphi_l(x), \varphi_k(x)) \quad \forall j \in \{1, \dots, n\}.$$

This leads to the definition of the stiffness matrix $A \in \mathbb{R}^{m \times m}$

$$A_{kl} := a(\varphi_l(x), \varphi_k(x)) \quad \forall k, l \in \{1, \dots, m\}.$$

Testing with all basis functions $\{\varphi_1, \dots, \varphi_m\}$ gives

$$\begin{pmatrix} a(u_{h,1}(\hat{u}_h), \varphi_1) & \cdots & a(u_{h,n}(\hat{u}_h), \varphi_1) \\ \vdots & & \vdots \\ a(u_{h,1}(\hat{u}_h), \varphi_m) & \cdots & a(u_{h,n}(\hat{u}_h), \varphi_m) \end{pmatrix} = A c(\hat{t}).$$

Together with (3.7), we obtain the system of ordinary differential equation

$$M \frac{d\hat{c}}{d\hat{t}} + B(\hat{c}) + A c(\hat{c}) = 0 \quad (3.12)$$

for the coefficient matrix $\hat{c}(\hat{t})$.

3.2.1 Entropy viscosity regularization

In this section, we focus on the calculation of the viscosity coefficient $\nu \in \mathbb{R}$. The method presented hereafter was proposed by Guermond, Pasquetti and Popov [GPP11].

We assume that (\hat{E}, \hat{F}) is an entropy pair for the conservation law

$$\frac{\partial}{\partial t} \hat{u}(x, \hat{t}) + \operatorname{div}_x [\delta(x) f(x, \hat{t}, u(\hat{u}(x, \hat{t})))] = 0 \quad \forall (x, \hat{t}) \in \hat{K}_i, \quad (2.20)$$

which satisfies the entropy inequality

$$\frac{\partial}{\partial t} \hat{E}(u(\hat{u})) + \operatorname{div}_x \hat{F}(u(\hat{u})) \leq 0. \quad (3.13)$$

The entropy \hat{E} is a conserved quantity in regions where the solution $u(\hat{u})$ is smooth. This means that (3.13) holds with an equality in smooth regions of $u(\hat{u})$ and as inequality in non-smooth regions (e.g. in shocks). We denote the amount of violation of the entropy conservation as *entropy production*. Choosing the viscosity coefficient ν proportional to the entropy production, leads to a large artificial viscosity at discontinuities and almost no viscosity in smooth regions.

Let $\hat{u}_h(x, \hat{t})$ be the numerical approximation of the exact solution $\hat{u}(x, \hat{t})$ at time \hat{t} . Then the *entropy residual* is defined as

$$r_E(x, \hat{t}) := \frac{\partial}{\partial t} \hat{E}(u_h(\hat{u}_h(x, \hat{t}))) + \operatorname{div}_x \hat{F}(u_h(\hat{u}_h(x, \hat{t}))) \quad \forall (x, \hat{t}) \in \hat{K}_i. \quad (2.20)$$

Using this residual, we obtain a viscosity coefficient on one element $T \in \mathcal{T}_i$

$$\nu_E(\hat{t})|_T := \max_{x \in T} \frac{c_E h_T^2 R(r_E(x, \hat{t}))}{|\bar{E}(\hat{u}_h)|},$$

where $h_T := \operatorname{diam}(T)/p$ is the effective local mesh size of the element T , \bar{E} is the average entropy over the element T and c_E is constant on \mathcal{T}_i . To avoid negative viscosity coefficients ν_E , we have to choose a positive functional R . The simplest choice is $R(\cdot) = |\cdot|$. Further we construct an upper bound of the viscosity coefficient

$$\nu_{\max}(\hat{t})|_T := c_{\max} h_T \max_{y \in T} |D_u f(u_h(\hat{u}_h(y, \hat{t})))|,$$

where $|D_u f(u_h(\hat{u}_h))|$ corresponds to the local characteristic speed (see section 1.1) and c_{\max} is constant on \mathcal{T}_i . For one element T , we obtain

$$\nu|_T := \min(\nu_{\max}|_T, \nu_E|_T).$$

The viscosity coefficient on the whole tent \hat{K}_i is then defined as

$$\nu := \max_{T \in \mathcal{T}_i} (\nu|_T).$$

4 Euler equations

4.1 Description of the Euler equations

For this example we write (2.2) using a more general notation

$$\frac{\partial}{\partial t} g(w) + \operatorname{div} f(w) = 0,$$

where $g(w)$ is the state vector, $f(w)$ the flux function and w a set of variables. The transformation from a tent to the space-time cylinder over the vertex patch leads to the equivalent equation

$$\frac{\partial}{\partial \hat{t}} [g(w) - f(w) \nabla \varphi] + \operatorname{div} [\delta f(w)] = 0,$$

which can be written as

$$\frac{\partial}{\partial \hat{t}} \hat{w} + \operatorname{div} [\delta f(w(\hat{w}))] = 0, \quad (4.1)$$

with the transformed state vector

$$\hat{w} := g(w) - f(w) \nabla \varphi. \quad (4.2)$$

Thus we have to find a transformation $w(\hat{w})$ to be able to solve (4.1) with a MTP scheme. See section 2.2 for further explanation of this transformation. To obtain the Euler equations, we define

$$g(\rho, u, E) := \begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix} \in \mathbb{R}^{N+2}$$

as function of the density ρ , the velocity u and the total energy E . The flux function is defined by

$$f(\rho, u, E) := \begin{pmatrix} \rho u \\ \rho u \otimes u + pI \\ u(E + p) \end{pmatrix} \in \mathbb{R}^{(N+2) \times N}$$

with the pressure $p(\rho, u, E)$. Additionally we define the internal energy e , the temperature T and the pressure p as

$$\begin{aligned} e &:= \frac{E}{\rho} - \frac{1}{2} |u|^2, \\ T &:= \frac{4}{d} e, \\ p &:= \frac{1}{2} \rho T, \end{aligned} \quad (4.3)$$

where d stands for the degrees of freedom of the gas particles, which is set to $d = 5$ for an ideal gas.

4.2 Mapping to a space-time cylinder

To find the above mentioned transformation $w(\hat{w})$ for the Euler equations, we have to solve (4.2) for w , which denotes the set of variables (ρ, u, E) . Thus we have to solve

$$\begin{aligned}\hat{w} = \begin{pmatrix} \hat{\rho} \\ \widehat{\rho u} \\ \hat{E} \end{pmatrix} &= \begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix} - \begin{pmatrix} \rho u \\ \rho u \otimes u + pI \\ u(E + p) \end{pmatrix} \nabla \varphi \\ &= \begin{pmatrix} \rho(1 - \nabla \varphi \cdot u) \\ \rho u(1 - \nabla \varphi \cdot u) - \frac{2}{d}\rho e \nabla \varphi \\ E(1 - \nabla \varphi \cdot u) - \frac{2}{d}\rho e \nabla \varphi \cdot u \end{pmatrix},\end{aligned}\quad (4.4)$$

which is equivalent to

$$\hat{\rho} = \rho(1 - \nabla \varphi \cdot u), \quad (4.5)$$

$$\widehat{\rho u} = \rho u(1 - \nabla \varphi \cdot u) - \frac{2}{d}\rho e \nabla \varphi, \quad (4.6)$$

$$\hat{E} = \rho \left(e + \frac{1}{2}|u|^2 \right) (1 - \nabla \varphi \cdot u) - \frac{2}{d}\rho e \nabla \varphi \cdot u. \quad (4.7)$$

By using equation (4.5) and its equivalent form $\rho \nabla \varphi \cdot u = \rho - \hat{\rho}$ we obtain

$$\widehat{\rho u} = u \hat{\rho} - \frac{2}{d}\rho e \nabla \varphi, \quad (4.8)$$

$$\hat{E} = \left(e + \frac{1}{2}|u|^2 \right) \hat{\rho} - \frac{2}{d}e(\rho - \hat{\rho}). \quad (4.9)$$

The inner product of (4.8) with $\nabla \varphi$ and (4.5) leads to

$$\widehat{\rho u} \cdot \nabla \varphi = \hat{\rho} \left(1 - \frac{\hat{\rho}}{\rho} \right) - \frac{2}{d}\rho e |\nabla \varphi|^2. \quad (4.10)$$

Multiplying (4.9) with $\hat{\rho}$ gives

$$\hat{E} \hat{\rho} = \frac{1}{2}|u|^2 \hat{\rho}^2 - \frac{2}{d}\hat{\rho} e \rho + \left(1 + \frac{2}{d} \right) \hat{\rho}^2 e. \quad (4.11)$$

To eliminate $|u|$ in (4.11), we use (4.8) in the following form

$$|u \hat{\rho}|^2 = |\widehat{\rho u} + \frac{2}{d}\rho e \nabla \varphi|^2 = |\widehat{\rho u}|^2 + \frac{4}{d}\rho e \widehat{\rho u} \cdot \nabla \varphi + \frac{4}{d^2}\rho^2 e^2 |\nabla \varphi|^2.$$

This leads to

$$\begin{aligned}\hat{E} \hat{\rho} &= \frac{1}{2} \left(|\widehat{\rho u}|^2 + \frac{4}{d}\rho e \widehat{\rho u} \cdot \nabla \varphi + \frac{4}{d^2}\rho^2 e^2 |\nabla \varphi|^2 \right) - \frac{2}{d}\hat{\rho} e \rho + \left(1 + \frac{2}{d} \right) \hat{\rho}^2 e \\ &\stackrel{(4.10)}{=} \frac{1}{2} |\widehat{\rho u}|^2 - \frac{2}{d^2}\rho^2 e^2 |\nabla \varphi|^2 + \hat{\rho}^2 e\end{aligned}\quad (4.12)$$

4 Euler equations

When we multiply (4.10) by ρe and take (4.12), we obtain

$$\begin{aligned} 0 &= \widehat{\rho u} \cdot \nabla \varphi \rho e - \hat{\rho} \rho e + \frac{2}{d} \rho^2 e^2 |\nabla \varphi|^2 + \hat{\rho}^2 e, \\ 0 &= \hat{E} \hat{\rho} - \frac{1}{2} |\widehat{\rho u}|^2 + \frac{2}{d^2} \rho^2 e^2 |\nabla \varphi|^2 - \hat{\rho}^2 e. \end{aligned}$$

By summing up these two equation, we end up with the quadratic equation in ρe

$$0 = \left(\frac{2}{d^2} + \frac{2}{d} \right) |\nabla \varphi|^2 (\rho e)^2 + (\widehat{\rho u} \cdot \nabla \varphi - \hat{\rho}) \rho e + \hat{E} \hat{\rho} - \frac{1}{2} |\widehat{\rho u}|^2. \quad (4.13)$$

With the definitions

$$a := \frac{2}{d^2} (d+1) |\nabla \varphi|^2, \quad b := \widehat{\rho u} \cdot \nabla \varphi - \hat{\rho} \quad \text{and} \quad c := \hat{E} \hat{\rho} - \frac{1}{2} |\widehat{\rho u}|^2,$$

we get the following two possible solutions of (4.13).

$$\begin{aligned} \rho e &= \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \\ &= \frac{(-b \pm \sqrt{b^2 - 4ac})}{2a} \frac{(-b \mp \sqrt{b^2 - 4ac})}{(-b \mp \sqrt{b^2 - 4ac})} \\ &= \frac{b^2 - (b^2 - 4ac)}{2a (-b \mp \sqrt{b^2 - 4ac})} \\ &= \frac{2c}{-b \mp \sqrt{b^2 - 4ac}} \end{aligned}$$

Due to (4.4) holds

$$\begin{pmatrix} \hat{\rho} \\ \widehat{\rho u} \\ \hat{E} \end{pmatrix} \xrightarrow{\nabla \varphi \rightarrow 0} \begin{pmatrix} \rho \\ \rho u \\ E \end{pmatrix}.$$

and $\nabla \varphi \rightarrow 0$ also implies

$$a \rightarrow 0, \quad b \rightarrow -\rho \quad \text{and} \quad c \rightarrow E\rho - \frac{1}{2} |\rho u|^2.$$

Thus we obtain that

$$\rho e = \frac{2c}{-b \mp \sqrt{b^2 - 4ac}} \xrightarrow{\nabla \varphi \rightarrow 0} \frac{2(E\rho - \frac{1}{2} |\rho u|^2)}{-\rho \mp \sqrt{\rho^2}}.$$

The product ρe has to stay bounded for all $\nabla \varphi$, which means we have to choose the solution

$$\rho e = \frac{2c}{-b - \sqrt{b^2 - 4ac}} \xrightarrow{\nabla \varphi \rightarrow 0} \frac{E\rho - \frac{1}{2} |\rho u|^2}{\rho}.$$

4 Euler equations

For the limit $\nabla\varphi \rightarrow 0$, this solution is also consistent with the definitions in (4.3). By using ρe and (4.8), we obtain the velocity

$$u = \frac{1}{\hat{\rho}} \left(\widehat{\rho u} + \frac{2}{d} \rho e \nabla \varphi \right) \xrightarrow{\nabla \varphi \rightarrow 0} \frac{\rho u}{\rho}.$$

Together with (4.5), we can calculate the density

$$\rho = \frac{\hat{\rho}}{1 - \nabla \varphi \cdot u} \xrightarrow{\nabla \varphi \rightarrow 0} \rho$$

and according to (4.3), the energy E can be obtained by

$$E = \rho e + \frac{1}{2} \rho |u|^2.$$

4.3 Entropy and entropy flux

The Euler equations can be derived from the Boltzmann equation by using a Maxwell distribution as ansatz. In the same fashion, we can derive conservation of Entropy

$$\frac{\partial}{\partial t} S + \operatorname{div} (u S) = 0$$

for smooth solutions. The arising entropy and entropy flux functions are

$$S := \rho \left(\ln \rho - \frac{d}{2} \ln T \right), \quad (4.14a)$$

$$F := u S. \quad (4.14b)$$

4.4 Numerical results

We consider the wind tunnel with a forward facing step shown in Figure 4.1. The initial conditions are set to

$$\rho = 1.4, \quad u = \begin{pmatrix} 3 \\ 0 \end{pmatrix}, \quad p = 1, \quad (4.15)$$

such that the Mach number is equal to 3. The boundary at $x_1 = 0$ is an inflow boundary and the values are set to the initial conditions (4.15) for all $t > 0$. The outflow boundary at $x_1 = 3$ is handled as free boundary, which has no effect on the flow. All other boundaries are solid walls.

The obtained solution at $t = 4$ (cf. Figure 4.3) corresponds to reference solutions that can be found in the literature (e.g. [ZGMP13]). As expected, the entropy residual is large at discontinuities, which then leads to locally higher viscosity coefficients in these regions. Further a Kelvin-Helmholtz instability develops at the contact discontinuity emerging from the three-shock interaction point (cf. Figure 4.2).

4 Euler equations

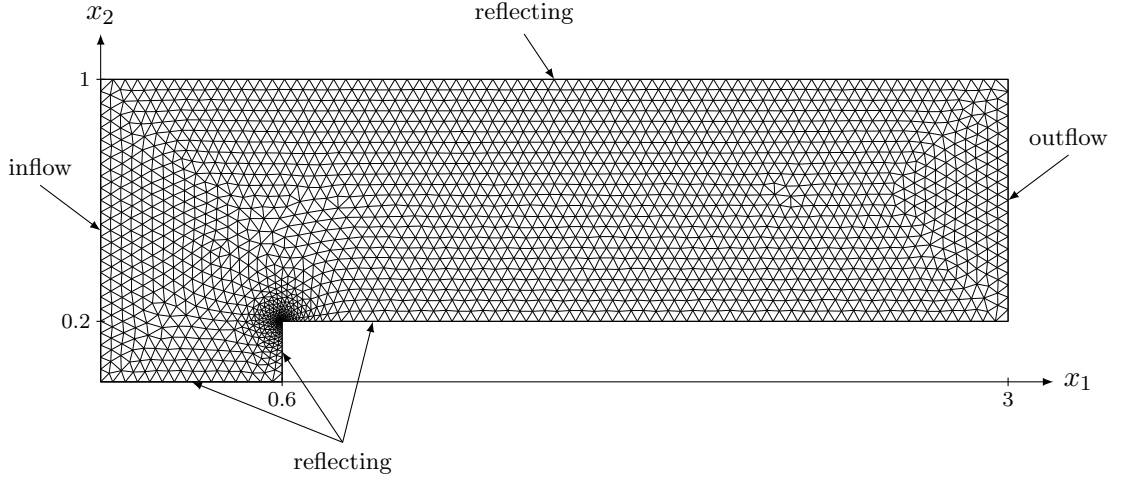


Figure 4.1: Geometry and mesh of wind tunnel, 3951 Triangles

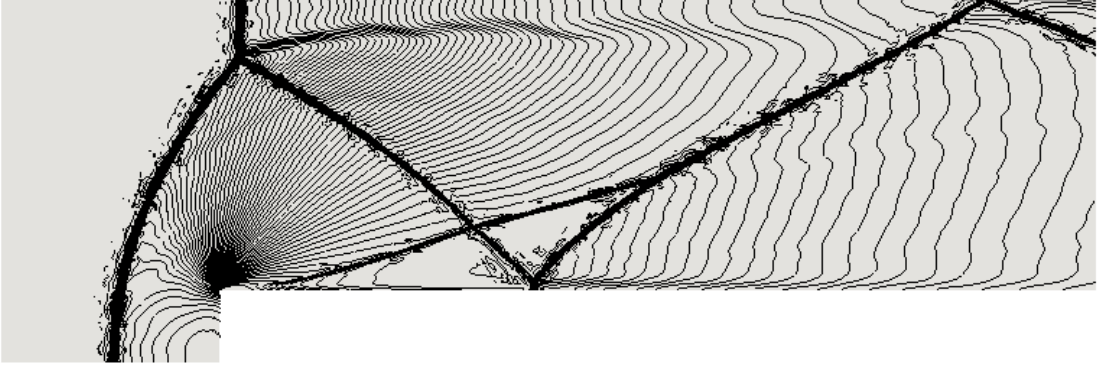


Figure 4.2: Wind tunnel at $t = 3$

4.5 Implementational aspects

Due to the physical meaning of the Euler equations, the implementation of the entropy viscosity regularization slightly differs from the description in section 3.2.1 and is done as follows:

- Evaluate the entropy residual on \hat{K}_i :

$$r_E := \frac{\partial}{\partial t} \hat{S} + \operatorname{div}_x \hat{F} \quad (4.16)$$

- Compute the local viscosity coefficient on one element $T \in \mathcal{T}_i$:

$$\nu_E|_T := c_E h_T^2 \|r_E\|_{L^\infty(T)},$$

with a constant $c_E \in \mathbb{R}$ and the effective local mesh size h_T .

4 Euler equations

- Calculate an upper bound of the local viscosity coefficient:

$$\nu_{\max}|_T := c_{\max} h_T \|\rho(|u| + \sqrt{\gamma T})\|_{L^\infty(T)},$$

where $c_{\max} \in \mathbb{R}$ is constant and $\gamma = \frac{d+2}{d} = 1.4$ for an ideal gas.

- Bound the local viscosity coefficient:

$$\nu|_T := \min(\nu_{\max}|_T, \nu_E|_T).$$

- Define the viscosity coefficient on the whole domain \hat{K}_i as

$$\nu := \max_{T \in \hat{T}_i} (\nu|_T).$$

The calculation of the entropy residual in (4.16) involves the derivative of \hat{S} with respect to \hat{t} . When we recall the definition (4.14) of the entropy pair (S, F) , we get the transformed entropy

$$\hat{S}(w, \hat{t}) = S(w) - \nabla \varphi(\hat{t}) \cdot F(w), \quad (4.17)$$

where w denotes the set of variables (ρ, u, E) . By using the chain rule, we obtain

$$\frac{\partial \hat{S}(w(\hat{t}), \hat{t})}{\partial \hat{t}} = \frac{\partial \hat{S}}{\partial w} \frac{\partial w}{\partial \hat{t}} + \frac{\partial \hat{S}}{\partial \hat{t}}.$$

Since we are solving the mapped conservation law, we do not have any derivative of w . Therefore we have to use the transformation $w(\hat{w})$ (see section 4.2) and the chain rule to calculate

$$\frac{\partial w}{\partial \hat{t}} = D_{\hat{w}} w \frac{\partial \hat{w}}{\partial \hat{t}},$$

where $D_{\hat{w}} w$ denotes the derivative of w with respect to \hat{w} . For solving the mapped conservation law

$$\frac{\partial \hat{w}}{\partial \hat{t}} + \operatorname{div} \hat{f} = 0, \quad (4.1)$$

the L^2 -projection of $\operatorname{div} \hat{f}$ has to be computed in each time step. Thus the derivative of \hat{w} with respect to \hat{t} is known and we can calculate the derivative of \hat{S} with respect to \hat{t} by

$$\frac{\partial \hat{S}(w(\hat{t}), \hat{t})}{\partial \hat{t}} = \frac{\partial \hat{S}}{\partial w} (D_{\hat{w}} w) \frac{\partial \hat{w}}{\partial \hat{t}} + \frac{\partial \hat{S}}{\partial \hat{t}},$$

which is a composition of known quantities.

4 Euler equations

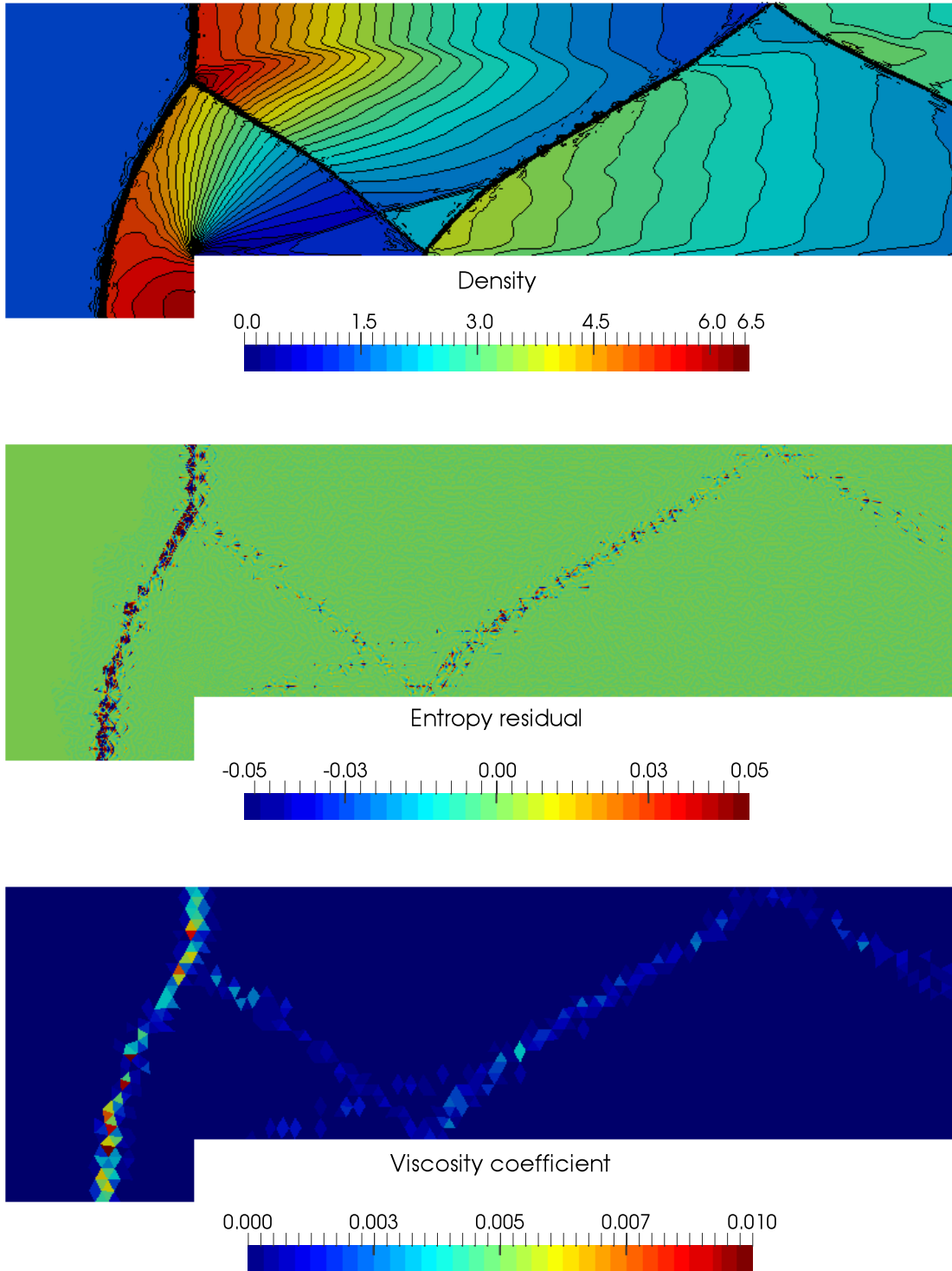


Figure 4.3: Solution of Mach 3 wind tunnel at $t = 4$, P^4 discontinuous finite elements on the mesh shown in Figure 4.1

5 Burgers equation

5.1 Description of the Burgers equation

A well known example for a scalar conservation law is the Burgers equation. We consider the two-dimensional domain $\Omega = [0, 1]^2$, where the flux function f is defined as

$$f(x, t, u(x, t)) := \frac{1}{2} [u(x, t)]^2 \mathbf{1}, \quad \mathbf{1} := \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad (5.1)$$

which leads to

$$\frac{\partial}{\partial t} u(x, t) + \operatorname{div}_x \left[\frac{1}{2} [u(x, t)]^2 \mathbf{1} \right] = 0 \quad \forall (x, t) \in \Omega \times (0, T], \quad (5.2a)$$

$$u(x, 0) = u_0(x) \quad \forall x \in \Omega. \quad (5.2b)$$

5.2 Mapping to a space-time cylinder

To regain u out of \hat{u} , we have to solve (2.19), which means solving

$$\hat{u} = u - \frac{1}{2} (\mathbf{1} \cdot \nabla \varphi) u^2. \quad (5.3)$$

We can write (5.3) as quadratic equation

$$\frac{1}{2} (\mathbf{1} \cdot \nabla \varphi) u^2 - u + \hat{u} = 0,$$

which has the solutions

$$\begin{aligned} u &= \frac{1 \pm \sqrt{1 - 2(\mathbf{1} \cdot \nabla \varphi) \hat{u}}}{(\mathbf{1} \cdot \nabla \varphi)} \\ &= \frac{1 \pm \sqrt{1 - 2(\mathbf{1} \cdot \nabla \varphi) \hat{u}}}{(\mathbf{1} \cdot \nabla \varphi)} \frac{(1 \mp \sqrt{1 - 2(\mathbf{1} \cdot \nabla \varphi) \hat{u}})}{(1 \mp \sqrt{1 - 2(\mathbf{1} \cdot \nabla \varphi) \hat{u}})} \\ &= \frac{1 - (1 - 2(\mathbf{1} \cdot \nabla \varphi) \hat{u})}{(\mathbf{1} \cdot \nabla \varphi)(1 \mp \sqrt{1 - 2(\mathbf{1} \cdot \nabla \varphi) \hat{u}})} \\ &= \frac{2\hat{u}}{1 \mp \sqrt{1 - 2(\mathbf{1} \cdot \nabla \varphi) \hat{u}}}. \end{aligned}$$

According to (5.3), the solution has to fulfill

$$u(\hat{u}) \xrightarrow{\nabla \varphi \rightarrow 0} \hat{u}.$$

5 Burgers equation

This results in the unique solution

$$u(x, \hat{t}, \hat{u}(x, \hat{t})) = \frac{2\hat{u}(x, \hat{t})}{1 + \sqrt{1 - 2(\mathbf{1} \cdot \nabla \varphi(x, \hat{t}))\hat{u}(x, \hat{t})}}.$$

5.3 Entropy and entropy flux

A well known convex entropy function for the Burgers equation (5.2) is

$$E(u) := \frac{1}{2}u^2.$$

The associated entropy flux F has to fulfill

$$D_u f_i D_u E = D_u F_i \quad \forall i \in \{1, 2\}. \quad (1.4)$$

For the above stated entropy and the flux function (5.1) holds

$$D_u f_i D_u E = u^2 \quad \forall i \in \{1, 2\},$$

which leads to the entropy flux

$$F(u) := \frac{1}{3}u^3 \mathbf{1}.$$

5.4 Numerical results

As numerical example, we consider (5.2) with the initial data

$$u_0(x_1, x_2) = \begin{cases} -0.2 & \text{if } x_1 < 0.5 \text{ and } x_2 > 0.5, \\ -1 & \text{if } x_1 > 0.5 \text{ and } x_2 > 0.5, \\ 0.5 & \text{if } x_1 < 0.5 \text{ and } x_2 < 0.5, \\ 0.8 & \text{if } x_1 > 0.5 \text{ and } x_2 < 0.5. \end{cases}$$

This problem has the exact solution

$$u_{ex}(x_1, x_2, t) = \begin{cases} \begin{cases} -0.2 & \text{if } x_1 < \frac{1}{2} - \frac{3t}{5} \text{ and } \begin{cases} x_2 > \frac{1}{2} + \frac{3t}{20}, \\ \text{otherwise,} \end{cases} \\ 0.5 & \text{otherwise,} \end{cases} \\ \begin{cases} -1 & \text{if } \frac{1}{2} - \frac{3t}{5} < x_1 < \frac{1}{2} - \frac{t}{4} \text{ and } \begin{cases} x_2 > -\frac{8x_1}{7} + \frac{15}{14} - \frac{15t}{28}, \\ \text{otherwise,} \end{cases} \\ 0.5 & \text{otherwise,} \end{cases} \\ \begin{cases} -1 & \text{if } \frac{1}{2} - \frac{t}{4} < x_1 < \frac{1}{2} + \frac{t}{2} \text{ and } \begin{cases} x_2 > \frac{x_1}{6} + \frac{5}{12} - \frac{5t}{24}, \\ \text{otherwise,} \end{cases} \\ 0.5 & \text{otherwise,} \end{cases} \\ \begin{cases} -1 & \text{if } \frac{1}{2} + \frac{t}{2} < x_1 < \frac{1}{2} + \frac{4t}{5} \text{ and } \begin{cases} x_2 > x_1 - \frac{5}{18t} \left(x_1 + t - \frac{1}{2}\right)^2, \\ \text{otherwise,} \end{cases} \\ \frac{2x_1-1}{2t} & \text{otherwise,} \end{cases} \\ \begin{cases} -1 & \text{if } x_1 > 0.5 \text{ and } \begin{cases} x_2 > \frac{1}{2} - \frac{t}{10}, \\ \text{otherwise,} \end{cases} \\ 0.8 & \text{otherwise,} \end{cases} \end{cases}$$

5 Burgers equation

Triangles	ndofs	L^1 -error	rate	L^2 -error	rate
852	12780	2.64e-01	-	5.22e-01	-
3408	51120	7.64e-03	5.1	6.70e-02	2.96
13632	202920	3.93e-03	0.96	4.86e-02	0.46
54528	817920	1.96e-03	1.01	3.46e-02	0.49

Table 5.1: Convergence rates for the two-dimensional Burgers equation using P^4 discontinuous finite elements

as presented in [GPP11]. The numerical solution shown in Figure 5.1 was calculated using fourth order polynomials for the spatial approximation and an improved Euler method for the time stepping. As expected, the viscosity coefficient is localized at the discontinuities (cf. Figure 5.1b). Further we observed convergence rates 1 and $\frac{1}{2}$ for the error measured in the L^1 - and L^2 -norm (cf. Table 5.1)

5 Burgers equation

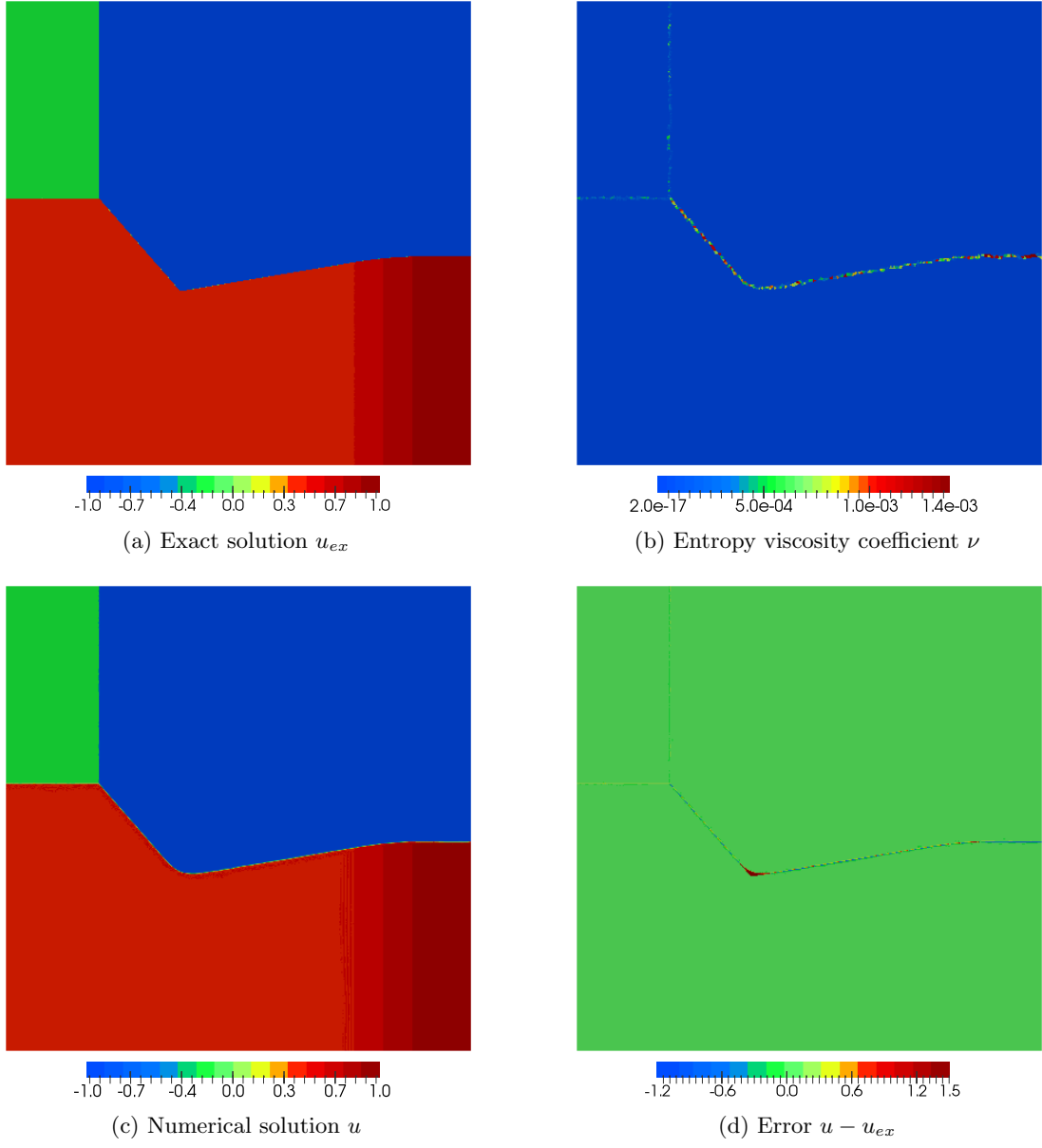


Figure 5.1: Solution of Burgers equation at $t = 0.5$, P^4 discontinuous finite elements and 54528 triangles

6 Wave equation

6.1 Description of the wave equation

The wave equation on a two-dimensional domain $\Omega \subset \mathbb{R}^2$ is given by

$$\frac{\partial^2 \psi}{\partial t^2} = c^2 \Delta \psi \quad \text{in } \Omega \times (0, T],$$

supplemented with appropriate boundary and initial conditions. For simplicity we set $c = 1$. By introducing the variable $v \in \mathbb{R}^2$, which fulfills the partial differential equation

$$\frac{\partial \psi}{\partial t} = \operatorname{div} v \quad \text{in } \Omega \times (0, T],$$

we obtain the system of conservation laws

$$\frac{\partial}{\partial t} \begin{pmatrix} v \\ \psi \end{pmatrix} + \operatorname{div} \begin{pmatrix} \psi I \\ v^\top \end{pmatrix} = 0 \quad \text{in } \Omega \times (0, T]. \quad (6.1)$$

The conserved variables are

$$u := \begin{pmatrix} v \\ \psi \end{pmatrix} \quad (6.2)$$

and the flux function is defined by

$$f(u) := \begin{pmatrix} \psi I \\ v^\top \end{pmatrix}, \quad (6.3)$$

where $I \in \mathbb{R}^{2 \times 2}$ is the identity matrix.

6.2 Mapping to a space-time cylinder

To obtain the required mapping, we have to solve (2.19) with the definitions (6.2) and (6.3). This means solving

$$\begin{pmatrix} \hat{v} \\ \hat{\psi} \end{pmatrix} = \begin{pmatrix} v \\ \psi \end{pmatrix} - \begin{pmatrix} \psi I \\ v^\top \end{pmatrix} \nabla \varphi$$

for v and ψ , which can be split into a vector valued equation

$$\hat{v} = v - \psi \nabla \varphi \quad (6.4)$$

6 Wave equation

and a scalar one

$$\hat{\psi} = \psi - v \cdot \nabla \varphi. \quad (6.5)$$

The inner product of (6.4) with $\nabla \varphi$ gives

$$\hat{v} \cdot \nabla \varphi = v \cdot \nabla \varphi - \psi |\nabla \varphi|^2. \quad (6.6)$$

We can eliminate the unknown v in (6.6) by using (6.5), which leads to

$$\hat{v} \cdot \nabla \varphi = \psi(1 - |\nabla \varphi|^2) - \hat{\psi}. \quad (6.7)$$

By rearranging (6.7), we obtain the solution

$$\psi = \frac{\hat{\psi} + \hat{v} \cdot \nabla \varphi}{1 - |\nabla \varphi|^2}$$

and (6.4) gives

$$v = \hat{v} + \psi \nabla \varphi.$$

6.3 Numerical results

To investigate the convergence rates of the MTP method, we consider (6.1) on $\Omega = [0, \pi]^2$ with the initial condition

$$u_0(x_1, x_2) = \begin{pmatrix} \psi_0 \\ v_0 \end{pmatrix} = \begin{pmatrix} \sin(x_1) \sin(x_2) \\ 0 \\ 0 \end{pmatrix}$$

for which we have the exact solution

$$u_{ex}(x_1, x_2, t) = \frac{1}{\sqrt{2}} \begin{pmatrix} \sqrt{2} \sin(x_1) \sin(x_2) \cos(\sqrt{2}t) \\ -\cos(x_1) \sin(x_2) \sin(\sqrt{2}t) \\ -\sin(x_1) \cos(x_2) \sin(\sqrt{2}t) \end{pmatrix}.$$

The calculations were done using a Runge-Kutta method of fourth order for the time stepping. Table 6.1 shows the L^2 -errors at $t = \sqrt{2}\pi$ and the convergence rates for P^2 , P^3 and P^5 finite elements, which are close to the optimal rates.

triangles	$p = 2$			$p = 3$			$p = 4$		
	ndofs	L^2 -error	rate	ndofs	L^2 -error	rate	ndofs	L^2 -error	rate
96	576	9.033e-03	-	960	7.270e-04	-	1440	2.834e-05	-
384	2304	1.358e-03	2.73	3840	6.392e-05	3.51	5760	1.283e-06	4.47
1536	9216	2.649e-04	2.36	15360	6.250e-06	3.35	23040	6.043e-08	4.41
6144	36864	4.516e-05	2.55	61440	5.585e-07	3.48	92160	3.207e-09	4.24

Table 6.1: Convergence rates for the two-dimensional wave equation, P^2 , P^3 and P^4 discontinuous finite elements

List of Figures

1.1	Characteristic lines of the Burgers equation with initial data u_0	3
1.2	Weak solutions of the Burgers equation with initial data u_0	4
2.1	Poorly parallelizable tents in 1D	9
2.2	Parallelizable tents 1D	9
2.3	Level of tents over a two-dimensional mesh	10
2.4	Mapping to space-time cylinder in 1D	11
2.5	Tent in 1D	12
4.1	Geometry and mesh of wind tunnel, 3951 Triangles	27
4.2	Wind tunnel at $t = 3$	27
4.3	Solution of Mach 3 wind tunnel at $t = 4$, P^4 discontinuous finite elements on the mesh shown in Figure 4.1	29
5.1	Solution of Burgers equation at $t = 0.5$, P^4 discontinuous finite elements and 54528 triangles	33

List of Tables

5.1	Convergence rates for the two-dimensional Burgers equation using P^4 discontinuous finite elements	32
6.1	Convergence rates for the two-dimensional wave equation, P^2 , P^3 and P^4 discontinuous finite elements	35

Bibliography

- [Daf10] Constantine M. Dafermos. *Hyperbolic Conservation Laws in Continuum Physics*, volume 325 of *Grundlehren der mathematischen Wissenschaften*. Springer-Verlag, Berlin, 2010.
- [EGSÜ05] Jeff Erickson, Damrong Guoy, John M. Sullivan, and Alper Üngör. Building spacetime meshes over arbitrary spatial domains. *Engineering with Computers*, 20(4):342–353, 2005.
- [FR99] Richard S. Falk and Gerard R. Richter. Explicit Finite Element Methods for Symmetric Hyperbolic Equations. *SIAM Journal on Numerical Analysis*, 36(3):935–952, 1999.
- [GN14] Jean-Luc Guermond and Murtazo Nazarov. A maximum-principle preserving finite element method for scalar conservation equations. *Computer Methods in Applied Mechanics and Engineering*, 272:198 – 213, 2014.
- [GNPY14] Jean-Luc Guermond, Murtazo Nazarov, Bojan Popov, and Yong Yang. A Second-Order Maximum Principle Preserving Lagrange Finite Element Technique for Nonlinear Scalar Conservation Equations. *SIAM Journal on Numerical Analysis*, 52(4):2163–2182, 2014.
- [GPP11] Jean-Luc Guermond, Richard Pasquetti, and Bojan Popov. Entropy viscosity method for nonlinear conservation laws. *Journal of Computational Physics*, 230(11):4248 – 4267, 2011. Special issue High Order Methods for {CFD} Problems.
- [GSW] Jay Gopalakrishnan, Joachim Schöberl, and Christoph Wintersteiger. Mapped Tent Pitching schemes for hyperbolic systems. in preparation.
- [HW07] Jan S. Hesthaven and Tim Warburton. *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*. Springer Science & Business Media, 2007.
- [MH08] Scott T. Miller and Robert B. Haber. A spacetime discontinuous Galerkin method for hyperbolic heat conduction. *Computer Methods in Applied Mechanics and Engineering*, 198(2):194 – 209, 2008.
- [Mon03] Peter Monk. *Finite Element Methods for Maxwell’s Equations*. Oxford University Press, 2003.
- [MR05] Peter Monk and Gerard R. Richter. A Discontinuous Galerkin Method for Linear Symmetric Hyperbolic Systems in Inhomogeneous Media. *Journal of Scientific Computing*, 22-23(1-3):443–477, 2005.

Bibliography

- [PHJ04] Jayandran Palaniappan, Robert B. Haber, and Robert L. Jerrard. A space-time discontinuous Galerkin method for scalar conservation laws. *Computer Methods in Applied Mechanics and Engineering*, 193(33–35):3607–3631, 2004.
- [Ser99] Denis Serre. *Systems of Conservation Laws 1: Hyperbolicity, Entropies, Shock Waves*. Cambridge University Press, 1999.
- [ÜS02] Alper Üngör and Alla Sheffer. Pitching Tents in Space-Time: Mesh Generation for Discontinuous Galerkin Method. *International Journal of Foundations of Computer Science*, 13(02):201–221, 2002.
- [YAS⁺00] Lin Yin, Amit Acharya, Nahil Sobh, Robert B. Haber, and Daniel A. Tortorelli. A Space-Time Discontinuous Galerkin Method for Elastodynamic Analysis. In Bernardo Cockburn, George E. Karniadakis, and Chi-Wang Shu, editors, *Discontinuous Galerkin Methods*, volume 11 of *Lecture Notes in Computational Science and Engineering*, pages 459–464. Springer Berlin Heidelberg, 2000.
- [ZGMP13] Valentin Zingan, Jean-Luc Guermond, Jim Morel, and Bojan Popov. Implementation of the entropy viscosity method with the discontinuous Galerkin method. *Computer Methods in Applied Mechanics and Engineering*, 253:479–490, 2013.