SBP-SAT schemes for hyperbolic problems

by
Anita Gjesteland

Master of Science Thesis in
Applied and Computational Mathematics

Department of Mathematics
University of Bergen

June 2019

Acknowledgements

First and foremost, I want to express my sincere gratitude towards my supervisor, professor Magnus Svärd, for his guidance and encouragement during this process. I have greatly enjoyed the work that has resulted in this thesis. Thank you for always taking the time to help me. I am particularly grateful for all the excellent answers and explanations you have provided me with. Lastly, I want to thank you for all the interesting conversations we have had (even though I had to turn to Google Translate sometimes afterwards).

I would also like to thank my fellow students for five wonderful years. Thank you for all the useful discussions, walks around the fourth floor and the late evenings in the office that usually ended in fits of laughter.

Last, but not least, a big thank you to my parents, for always providing me with advice and support when I need it, and for answering all my stressed calls during the many exam periods over the years. Your encouragement has been more important to me than you know.

Abstract

Numerical methods for solving partial differential equations is an important field of study, as it helps us to describe many different processes in the world. An important property of a numerical method, is that it should be a stable approximation of the governing differential equation. For numerical approximations that satisfy a summation-by-parts rule, and that are combined with the simultaneous approximation term technique at the boundaries, energy estimates can be derived to prove stability. The Summation-By-Parts Simultaneous Approximation Term (SBP-SAT) technique was first developed in the context of the finite difference method. More recently, it has been shown that other numerical methods, such as the finite volume method, also can be formulated in the SBP framework.

The finite volume method is a popular numerical method, as it can be formulated on unstructured grids. However, Svärd et al. showed in [SGN07] that some approximations of the second derivative are in fact inconsistent on such grids. Consistency is another key feature of a numerical method. The method should be consistent in order for us to know that we are solving the correct equation.

In this thesis, we study the extension of the SBP-SAT technique to the finite volume method. We introduce a methodology for implementing a second derivative approximation on general unstructured grids by including a transformation to a computational domain, where accuracy is expected to be recovered. The numerical experiments demonstrate that full accuracy is not obtained when including the transformation. There are still nodes along and near the boundary that are inconsistent. However, numerical experiments indicate that we have convergence.

Contents

Notation

The following notation will be used throughout the thesis.

Capital letters (A): Matrices (if not otherwise stated)

Bold letters ($\boldsymbol{u}$): Vectors

$u_p^{(n)}$: nth derivative of u with respect to variable p

$\mathcal{O}$: Big O notation

Abbreviations

PDE - Partial Differential Equation

CFD - Computational Fluid Dynamics

IBVP - Initial-Boundary-Value Problem

FDM - Finite Difference Method

FVM - Finite Volume Method

SBP - Summation-By-Parts

SAT - Simultaneous Approximation Term

Chapter 1

Introduction

1.1 Introduction

Partial Differential Equations (PDEs) are of great importance in several fields of research, among which Computational Fluid Dynamics (CFD) is one. Since a large class of these equations cannot be solved analytically, numerical methods are required to obtain solutions to the PDEs. The foundation of a numerical method is the discretisation of the given domain, which can be either structured or unstructured. Thereafter, a numerical method is formulated on the discrete grid. One such numerical method that will be used in this thesis, is the Finite Difference Method (FDM). This method starts with the given equation in differential form, and the derivatives in the governing equations are approximated by finite differences, usually obtained using Taylor series expansions. Its simplicity, and the ease at which it is to obtain higher-order approximations, are advantages of this method. The finite difference method is often used in the CFD community, and for problems that are to be solved over long time intervals or that require small errors in the solutions, high-order approximations are favoured. However, the treatment of the boundaries for such approximations can be complicated, and they must be handled in a way that leads to stable schemes. (For more information about the FDM, see e.g. [Bla07, FP99, KCDT16, GKO95, Gus08]). Stability in the numerical analysis is the analogous concept to well-posedness in the continuous setting. If the problem

is well-posed, and the numerical method is a consistent and stable approximation of it, then the Lax-Richtmyer Equivalence Theorem ([LR56]) guarantees that the numerical solution will converge to the true solution. However, demonstrating that the numerical scheme is in fact stable, need not be a trivial task, especially for high-order finite difference methods. This changed with the introduction of the Summation-By-Parts Simultaneous Approximation Term (SBP-SAT) schemes, which are finite difference schemes combined with a weak enforcement of the boundary conditions. The SBP operators were first derived by Kreiss and Scherer in [KS74]. These operators mimic the continuous integration-by-parts rule, which plays a central role in the derivation of energy estimates, and are constructed in a way that resembles the energy loss at outflow boundaries for the equation (see [Gus08]). Nevertheless, the operators alone only allow for stability proofs for simple problems, but with the establishment of the SAT technique near the boundaries, one can now prove stability for more complicated problems. The SAT procedure was first developed in [CGA94] by Carpenter, Gottlieb and Abarbanel. The SAT enforces the boundary conditions in a weak way, by introducing a penalty term to the scheme. For a more comprehensive summary of the history of the SBP-SAT technique, see [SN14] and [DRFHZ14].

Even though the SBP operators were constructed in the framework of the finite difference method, these operators have more recently been formulated in the context of other numerical methods as well. Examples here are the Finite Volume Method (FVM) (see e.g. [NB01, NFAE03, SN04]) and the spectral collocation methods (see e.g. [Gas13, DRFBZ14, CFNF14]). In this thesis, our main focus will be the SBP-SAT technique formulated using the finite volume method. An advantage of this method, is that it, in contrast to the finite difference method, can be formulated on unstructured grids. Another distinction between the finite difference and the finite volume method, is that the latter is based on the integral form of the given PDE. After the domain is discretised into a set of non-overlapping sub-domains, called dual volumes (or grid cells), the equation is integrated over each such volume. From here, we derive discrete approximations of the average value of the solution in each grid cell. Some of the integrals are converted into line (or surface) integrals (using for example Green's theorem or the divergence theorem). These integrals represents fluxes that can be approximated as the sum of fluxes over each edge in

a dual volume. The fluxes are often assumed to be constant along a grid face, and evaluated at the midpoint of the edge (for a more complete introduction to the FVM, see for example [DB16, FP99, Lev02, Bla07, KCDT16]). As it can be formulated on unstructured grids, the finite volume method is also a popular method in the CFD community. Nevertheless, it has been shown in [SGN07] that care must be taken when approximating the second derivative using this method. In this article, it was demonstrated that two commonly used approximations of the Laplacian are inconsistent on general meshes.

As a demonstration of the SBP-SAT technique, we will in this thesis first discretise the second-order wave equation (in one space dimension) using a high order SBP finite difference operator, where the boundaries are treated using the SAT procedure. Next, the SBP-SAT technique is formulated in the context of the finite volume method. We consider a first-derivative approximation from [NFAE03], and investigate if second-order accuracy is obtained by transforming the physical domain into equilateral triangles. We propose stable schemes for the advection equation on both single-block and multi-block domains. Lastly, we discretise the second-order wave equation in two space dimensions using the first-derivative approximation twice, and investigate if the transformation of the physical domain leads to a consistent scheme. One immediate advantage of including the transformation, is that the implementation makes mesh refinement an easy task, as a number of grid points along the boundary is specified for refinement within the elements. In the interest of solving CFD problems using higher (than one) order approximations on complex geometries, the overall goal of the project is to derive a methodology for implementing the considered second-derivative approximation on general unstructured grids. Then we can supplement high-order finite difference approximations with the finite volume method to handle the complex geometries of the mesh. In order to obtain a more structured derivation, this thesis considers simpler problems than those often solved in CFD.

1.2 Thesis outline

In the next chapter, some preliminary theory regarding well-posedness of initial-boundary-value problems, and the basic idea of the SBP-SAT technique, are introduced. In Chapter 3, we analyse the second-order wave equation in one space dimension and discretise it using a high-order finite difference SBP operator combined with the SAT procedure at the boundaries. We investigate stability of the scheme and discuss convergence. Chapter 4 presents an extension of the SBP-SAT theory in the framework of the finite volume method. We verify an approximation for the first derivative, and introduce stable schemes for domains with and without interfaces. Furthermore, we investigate if the first-derivative approximation applied twice yields a consistent second-derivative approximation, and propose a stable scheme for the implementation of the second-order wave equation in two space dimension. We also discuss the expected covergence rates, and present the results obtained from the numerical experiments. Lastly, Chapter 5 provides conclusions of this work and some possible directions of further work.

All numerical schemes that are proposed in this thesis were coded from scratch in MATLAB (except the SBP operator used in Chapter 3).

Chapter 2

Preliminaries

Before proceeding to the main topics, we introduce some definitions that will be used throughout the thesis. Since all parts of this project have been twofold, one part concerning the continuous problem and the other the semi-discrete problem, we will divide this chapter in the same way.

2.1 Preliminaries for the continuous analysis

The starting point of every section in this project will be to demonstrate that the given problem is well-posed. We now introduce some theory concerning this property.

Consider the following general Initial-Boundary-Value Problem (IBVP)

$$\begin{aligned} u_t &= P(x, \partial_x, t)u + F(x,t), \;\; 0 \leq x \leq 1, \; t \geq 0, \\ L_0(\partial_x, t)u(0,t) &= g_0(t), \\ L_1(\partial_x, t)u(1,t) &= g_1(t), \\ u(x,0) &= f(x), \end{aligned} \tag{2.1}$$

where P is a differential operator; $F(x,t)$ is a forcing function; L_0 and L_1 are

differential operators acting on the boundary, and $g_0(t)$, $g_1(t)$ and $f(x)$ are the boundary and initial data ([SN14]). We introduce the following definition.

Definition 2.1 ([Gus08]). *The IBVP (2.1) is well-posed if for $F = 0$, $g_0 = 0$ and $g_1 = 0$, there is a unique solution that satisfies the estimate*

$$\|u(\cdot, t)\| \leq Ke^{\alpha t} \|f(\cdot)\|,$$

where K and α are constants independent of f. ⌟

The norm appearing in Definition 2.1 is the L^2-norm induced by the L^2-inner product, defined as

$$\langle u, v \rangle = \int_0^1 uv \, dx, \qquad \|u\|^2 = \langle u, u \rangle = \int_0^1 u^2 \, dx.$$

For problems with nonzero forcing function, $F \neq 0$, the following definition from [KL89] applies.

Definition 2.2. *The IBVP (2.1) is well-posed if for $g_0 = 0$ and $g_1 = 0$, there is a unique smooth solution that satisfies the estimate*

$$\|u(\cdot, t)\|^2 \leq K(t) \left(\|f(\cdot)\|^2 + \int_0^t \|F(\cdot, \tau)\|^2 \, d\tau \right),$$

where K is a function of t, but does not depend on the problem data. ⌟

This means that the forcing function can be neglected to simplify the analysis, since both the problem with and without this term is well-posed ([Gus08]).

Both the above definitions require zero boundary data, but we would like to consider problems with inhomogeneous boundary data as well. This is possible if we make a

transformation $\tilde{u} = u - \psi$ that yields homogeneous boundary data (see e.g. [SN14]). The forthcoming proposition demonstrates that the problem with inhomogeneous data is indeed well-posed.

Proposition 2.3. *The IBVP (2.1) is well-posed for $F(x,t) \neq 0$, $g_0(t) \neq 0$ and $g_1(t) \neq 0$, with g_0 and g_1 differentiable, if the corresponding problem with homogeneous data is well-posed.*

Proof. We make the transformation mentioned above, $\tilde{u} = u - \psi$, where ψ is sufficiently smooth and bounded, and is chosen such that it satisfies

$$\begin{aligned} \psi(x,0) &= f(x), \\ L_0(\partial_x, t)\psi(0,t) &= g_0(t), \\ L_1(\partial_x, t)\psi(1,t) &= g_1(t). \end{aligned} \tag{2.2}$$

By inserting $u = \tilde{u} + \psi$ in Equation (2.1), we obtain

$$\begin{aligned} u_t &= (\tilde{u} + \psi)_t = \tilde{u}_t + \psi_t, \\ Pu + F(x,t) &= P(\tilde{u} + \psi) + F(x,t) = P\tilde{u} + P\psi + F(x,t), \\ \tilde{u}_t &= P\tilde{u} + P\psi + F(x,t) - \psi_t = P\tilde{u} + F_1(x,t), \end{aligned}$$

where $F_1(x,t) = F(x,t) + P\psi - \psi_t$. However, we know from Definition 2.2 that the forcing function can be disregarded in the analysis, so for simplicity we neglect F_1. We then obtain the following problem

$$\begin{aligned} \tilde{u}_t &= P\tilde{u}, \\ \tilde{u}(x,0) &= 0, \\ L_0(\partial_x, t)\tilde{u}(0,t) &= 0, \\ L_1(\partial_x, t)\tilde{u}(1,t) &= 0. \end{aligned} \tag{2.3}$$

If this problem is well-posed, it satisfies the estimate in Definition 2.1, i.e.,

$$\|\tilde{u}(\cdot,t)\| \leq Ke^{\alpha t} \|f(\cdot)\|,$$

and since $f(x) = 0$, it follows that $\|\tilde{u}(\cdot,t)\| = 0$. We now make use of the fact that $\tilde{u} = u - \psi$ to obtain an estimate for u. Using the triangle inequality yields

$$\|u(\cdot,t)\| = \|\tilde{u}(\cdot,t) + \psi(\cdot,t)\| \leq \|\tilde{u}(\cdot,t)\| + \|\psi(\cdot,t)\|,$$

which implies

$$\|u(\cdot,t)\| \leq \|\psi(\cdot,t)\|.$$

This estimate holds as long as the data $g_0(t)$ and $g_1(t)$ is sufficiently differentiable in time such that the conditions (2.2) hold. Hence, well-posedness of the problem (2.3) with homogeneous data implies well-posedness for the problem (2.1) with inhomogeneous data. □

The above proposition together with the fact that any forcing function can be neglected, allows us to simplify the analysis of problems with inhomogeneous data, by setting boundary and forcing data to zero.

For some problems, it is possible to obtain a stronger estimate for the solution. This is the case when the estimate also involves the boundary data. Problems for which the following estimate holds, are called *strongly well-posed* in [Gus08].

Definition 2.4 ([Gus08]). *The IBVP (2.1) is strongly well-posed if there is a unique solution that satisfies the estimate*

$$\|u(\cdot,t)\|^2 \leq Ke^{\alpha t} \left(\|f(\cdot)\|^2 + \int_0^t \|F(\cdot,\tau)\|^2 + |g_0(\tau)|^2 + |g_1(\tau)|^2 \, d\tau \right),$$

where K and α are constants independent of f, F, g_0 and g_1. ⌟

Throughout this thesis, we will derive estimates such as the ones above for demonstrating well-posedness for every problem considered. These estimates can be obtained by using the energy method. We will demonstrate this method in Example 2.9 below.

Remark. *The above definitions involve the condition that there exists a solution. In [GKO95] the authors provide an explanation of how to prove existence of solutions given that we can obtain an energy estimate. We will not discuss this any further, since it is well-known that every problem considered in this project has a solution.*

2.2 Preliminaries for the discrete analysis

We now turn to the case of semi-discrete approximations of the general initial-boundary-value problem (2.1). Throughout this project, we will not consider fully-discrete approximations in the analyses of the problems. This is of course needed for the implementation of the schemes. However, Kreiss and Wu demonstrated in [KW93] that if the semi-discrete approximation is stable, then, given that certain conditions are met, the fully-discrete approximation is stable if we discretise time using an appropriate Runge-Kutta method. We therefore focus our attention on the analyses of the semi-discrete approximations.

To introduce the approximation of the general problem (2.1), we first divide the spatial domain into $n+1$ grid points with equal distance h. The ith grid point is denoted $x_i = ih$, where $i = 0, 1, ..., n$. Then the semi-discretisation of the IBVP (2.1) can be written

$$\begin{aligned} \boldsymbol{u}_t &= D(\boldsymbol{x},t)\boldsymbol{u} + \boldsymbol{F}(\boldsymbol{x},t), \\ B_0(t)u_0(t) &= g_0(t), \\ B_1(t)u_n(t) &= g_1(t), \\ \boldsymbol{u}(0) &= \boldsymbol{f}, \end{aligned} \tag{2.4}$$

where D is an approximation of the differential operator P; B_0 and B_1 are approximations of the differential operators L_0 and L_1, and $\boldsymbol{F}(\boldsymbol{x}, t)$, g_0, g_1 and $\boldsymbol{f}$ are the forcing, boundary and initial data, respectively. $\boldsymbol{x} = (x_0, x_1, ..., x_n)$ is a vector with the grid points as its elements.

The corresponding concept of well-posedness in the semi-discrete analysis is stability. We now introduce the analogous definitions to the continuous case.

Definition 2.5 ([Gus08]). *The approximation (2.4) of the IBVP (2.1) is stable if for $F = 0$, $g_0 = 0$ and $g_1 = 0$, the solution satisfies the estimate*

$$\|\boldsymbol{u}(t)\|_h \leq K e^{\alpha t} \|\boldsymbol{f}\|_h ,$$

where K and α are constants independent of f and h. ⌟

The constants K and α appearing in Definition 2.5 are generally different from the respective constants in the continuous case. The norm $\|\cdot\|_h$ is a discrete L^2-equivalent norm. As shown in Proposition 2.3 for the continuous case, we can extend the stability definition to problems with inhomogeneous data by a transformation of the problem into one with homogeneous data ([SN14]). However, for some approximations, it is possible to obtain a stronger estimate that includes the forcing and boundary data. Such approximations are called *strongly stable*, which is defined in [Gus08] as follows.

Definition 2.6 ([Gus08]). *The approximation (2.4) of the IBVP (2.1) is strongly stable if there is a unique solution that satisfies the estimate*

$$\|\boldsymbol{u}(t)\|_h^2 \leq K e^{\alpha t} \left(\|\boldsymbol{f}\|_h^2 + \int_0^t \|\boldsymbol{F}(\tau)\|_h^2 + |g_0(\tau)|^2 + |g_1(\tau)|^2 \, d\tau \right),$$

where K and α are constants independent of $\boldsymbol{f}$, $\boldsymbol{F}$, g_0, g_1 and h. ⌟

In a similar manner as the continuous case, such estimates can be obtained by using the discrete energy method, as will be demonstrated in Example 2.9 below.

SBP operators

The approximations of the differential operators used in this thesis are so-called Summation-By-Parts (SBP) operators. These are operators that mimic the continuous integration-by-parts rule, which is an essential part in the derivation of energy estimates. In this chapter, some general definitions of these operators are introduced, while their specific form will be explained in more detail in Chapter 4 and Appendix B. We use the definitions found in [SN17] (for similar definitions, see for example [SN14]).

Definition 2.7 ([SN17]). *An SBP-operator for the first derivative is defined by*

$$D_1 \boldsymbol{u} = P^{-1} Q \boldsymbol{u},$$

where $Q + Q^T = B = \text{diag}(-1, 0, \ldots, 0, 1)$. *$P$ is a symmetric positive-definite matrix with elements of size $\mathcal{O}(h)$, where h is the grid spacing. P also defines an inner product $\langle \boldsymbol{u}, \boldsymbol{v} \rangle = \boldsymbol{u}^T P \boldsymbol{v}$, and an L^2-equivalent norm $\|\boldsymbol{u}\|^2 = \langle \boldsymbol{u}, \boldsymbol{u} \rangle$. P is diagonal in the interior, but can have blocks of elements near the boundary.* ⌟

Definition 2.8 ([SN17]). *An SBP-operator for the second derivative is defined by*

$$D_2 \boldsymbol{u} = P^{-1} \left(-A + BS \right) \boldsymbol{u},$$

where A is a symmetric positive semi-definite matrix and $S\boldsymbol{u}$ is a first derivative approximation. ⌟

For the finite volume method in two space dimensions, the relation $Q + Q^T = \text{diag}(-1, 0, \ldots, 0, 1)$ will not hold. However, a similar property applies in this case, as will be demonstrated in Chapter 4.

To illustrate how the analyses of the continuous and semi-discrete problems are handled in a similar manner, we consider an example with a first derivative SBP-

operator for the advection equation in one space dimension, as is often done in the literature. Stability directly from the use of SBP-operators is only possible to obtain for simple problems, and therefore they are often coupled with the Simultaneous Approximation Term (SAT). These are penalty terms that impose the boundary conditions weakly (see [SN14] and [DRFHZ14] for further discussion). The example below will also include a demonstration of the SAT procedure.

Example 2.9. Consider the advection equation in one space dimension

$$\begin{aligned} u_t + au_x &= 0, \qquad 0 \leq x \leq 1, \;\; t \geq 0, \\ u(x,0) &= f(x). \end{aligned} \tag{2.5}$$

If $a > 0$, we have the boundary condition $u(0,t) = g(t)$, and if $a < 0$, we have the boundary condition $u(1,t) = g(t)$.

We now demonstrate that the problem is well-posed. Using the energy method, multiply Equation (2.5) by u, and integrate over the domain.

$$\int_0^1 uu_t \, dx = -\int_0^1 auu_x \, dx.$$

The integrand on the left-hand side can be written $\frac{1}{2}\frac{d}{dt}\left\|u(\cdot,t)\right\|^2$, and by splitting the integral on the right-hand side into two equal parts and applying the integration-by-parts rule on one of them, we obtain

$$\begin{aligned} \frac{1}{2}\frac{d}{dt}\left\|u(\cdot,t)\right\|^2 &= -\frac{1}{2}au^2(x,t)|_{x=0}^{x=1}, \\ \frac{d}{dt}\left\|u(\cdot,t)\right\|^2 &= -a(u^2(1,t) - u^2(0,t)). \end{aligned}$$

Depending on the sign of a, either $-au^2(1,t) \geq 0$ or $au^2(0,t) \geq 0$. However, the term that will contribute to a growth in the norm is the one with the boundary condition. We let $a > 0$ for the rest of the derivation (the case with $a < 0$ could be treated in the same way). From Proposition 2.3, we can set boundary data to zero

without loss of well-posedness. By doing this, the estimate reads

$$\frac{d}{dt}\left\|u(\cdot,t)\right\|^2 = -au^2(1,t).$$

Since $a > 0$, $-au^2(1,t) \leq 0$, and we get $\frac{d}{dt}\left\|u(\cdot,t)\right\| \leq 0$. Integration in time yields the final estimate

$$\left\|u(\cdot,t)\right\| \leq \left\|f(\cdot)\right\|,$$

which proves well-posedness of problem (2.5) in the sense of Definition 2.1.

We now consider the following approximation of problem (2.5)

$$\begin{aligned} \boldsymbol{u}_t + aP^{-1}Q\boldsymbol{u} &= \tau P^{-1}(u_0 - g_0(t))\boldsymbol{v}, \qquad a > 0,\ t \geq 0, \\ \boldsymbol{u}(0) &= \boldsymbol{f}, \end{aligned} \tag{2.6}$$

where $P^{-1}Q$ is an SBP operator with diagonal P. The term on the right-hand side of the equation is the SAT, where τ is a parameter to be determined for stability reasons, and we have defined $\boldsymbol{v} = (1, 0, \ldots, 0)^T$.

Using the discrete energy method, multiply Equation (2.6) by $\boldsymbol{u}^T P$ and add the transpose to obtain

$$\begin{aligned} \boldsymbol{u}^T P\boldsymbol{u}_t + \boldsymbol{u}_t^T P\boldsymbol{u} &= -a\boldsymbol{u}^T Q\boldsymbol{u} - a\boldsymbol{u}^T Q^T\boldsymbol{u} + \tau\boldsymbol{u}^T(u_0 - g_0(t))\boldsymbol{v} + \tau\boldsymbol{v}^T(u_0 - g_0(t))\boldsymbol{u}, \\ \frac{d}{dt}\left\|\boldsymbol{u}(t)\right\|^2 &= -a\boldsymbol{u}^T(Q + Q^T)\boldsymbol{u} + 2\tau\boldsymbol{u}^T(u_0 - g_0(t))\boldsymbol{v}. \end{aligned}$$

Recall from Definition 2.7 that $Q + Q^T = B = \text{diag}(-1, 0, ..., 0, 1)$. Using this yields

$$\frac{d}{dt}\left\|\boldsymbol{u}(t)\right\|^2 = -a(u_N^2 - u_0^2) + 2\tau u_0^2 - 2\tau u_0 g_0(t).$$

We see that the SBP operator produces boundary terms analogous to the continuous integration-by-parts rule.

In the same way as for the continuous case, we can set the boundary data to zero without loosing stability. The estimate then becomes $\frac{d}{dt}\left\|\boldsymbol{u}\right\|^2 = -au_N^2 + (a+2\tau)u_0^2$. The parameter τ must be chosen such that $\tau \leq -\frac{a}{2}$ holds, in order for the scheme to be stable. We then have $\frac{d}{dt}\left\|\boldsymbol{u}\right\|^2 \leq 0$. Integration in time yields the final estimate

$$\left\|\boldsymbol{u}(t)\right\| \leq \left\|\boldsymbol{u}(0)\right\| = \left\|\boldsymbol{f}\right\|.$$

⌟

Remark. *In the above derivation of stability of the numerical scheme, we could have shown strong stability by adding the terms* $\frac{\tau^2}{(a+2\tau)}g^2(t) - \frac{\tau^2}{(a+2\tau)}g^2(t) = 0$*, which would have resulted in the estimate* $\left\|\boldsymbol{u}\right\|^2 \leq \left\|\boldsymbol{f}\right\|^2 - \frac{\tau^2}{a+2\tau}\int_0^t g^2(T)\,dT$*. This requires a stronger restriction on* τ*, namely that it should satisfy* $\tau < -\frac{a}{2}$*.*

The example concludes this chapter. In the following two chapters, numerical schemes such as the one above will be further introduced.

Chapter 3

Finite differences and the SBP-SAT technique

As an introduction for the main investigations to be carried out in this project, we consider a finite difference SBP operator to discretise the wave equation in one space dimension.

3.1 Continuous analysis for the wave equation

Consider the second-order wave equation in one space dimension with homogeneous Dirichlet boundary conditions.

$$\begin{aligned} u_{tt} &= u_{xx}, \quad 0 \leq x \leq 1, \quad t \geq 0, \\ u(0,t) &= 0, \\ u(1,t) &= 0, \\ u(x,0) &= f(x), \\ u_t(x,0) &= g(x). \end{aligned} \tag{3.1}$$

Using the energy method, multiply the equation in problem (3.1) by u_t and integrate

over the domain to obtain

$$\int_0^1 u_t u_{tt}\, dx = \int_0^1 u_t u_{xx}\, dx.$$

We recognize the left-hand side as a time derivative of the norm $\|u_t\|$. We can therefore write the above equation as

$$\frac{1}{2}\frac{d}{dt}\|u_t\|^2 = \int_0^1 u_t u_{xx}\, dx.$$

If we now apply the integration-by-parts rule on the right-hand side of the equation, we have

$$\begin{aligned}
\frac{1}{2}\frac{d}{dt}\|u_t\|^2 &= u_t u_x|_{x=0}^{x=1} - \int_0^1 u_{tx} u_x\, dx, \\
&= -\int_0^1 \frac{1}{2}\frac{\partial}{\partial t}(u_x)^2\, dx, \\
\frac{d}{dt}\left(\|u_t\|^2 + \|u_x\|^2\right) &= 0.
\end{aligned}$$

Integration in time results in

$$\|u_t(\cdot,t)\|^2 + \|u_x(\cdot,t)\|^2 = \|g(\cdot)\|^2 + \|f_x(\cdot)\|^2 .$$

To obtain a bound on u itself, we use a technique proposed in [WK17].

$$\begin{aligned}
&\frac{d}{dt}\|u\|^2 = 2\|u\|\frac{d}{dt}\|u\|, \\
&\frac{d}{dt}\|u\|^2 = \int_0^1 u^2\, dx = \int_0^1 2uu_t\, dx = 2\langle u, u_t\rangle \leq 2\|u\|\,\|u_t\| .
\end{aligned} \tag{3.2}$$

Consequently, we have

$$\frac{d}{dt}\|u\| \leq \|u_t\| \lesssim \|u_t\|^2 + \|u_x\|^2,$$

which leads to

$$\frac{d}{dt}\|u\| \leq \|g(\cdot)\|^2 + \|f_x(\cdot)\|^2.$$

Integration in time yields the final result

$$\|u(\cdot,t)\|^2 \leq \|f(\cdot)\|^2 + \int_0^t \|g(\cdot)\|^2 + \|f_x(\cdot)\|^2 \, d\tau,$$

which demonstrates well-posedness of the problem (3.1) in the sense of Definition 2.4.

3.2 Discrete analysis for the wave equation

We consider the following semi-discretisation of problem (3.1) from [SN17] (also found in [WK17]), where we also include the right boundary.

$$\boldsymbol{u}_{tt} = D_2\boldsymbol{u} + P^{-1}\left(-S^T E_0 - \frac{\tau_1}{h}E_0\right)\boldsymbol{u} + P^{-1}\left(S^T E_N - \frac{\tau_2}{h}E_N\right)\boldsymbol{u}. \tag{3.3}$$

Here, E_0 and E_N are matrices with zero elements everywhere except in the upper left and lower right corner, respectively; P is a diagonal matrix and τ_1 and τ_2 are parameters to be determined for stability reasons. Recall that the second-derivative operator has the form $D_2 = P^{-1}(-A + BS)$.

Before demonstrating stability of the above scheme, we introduce a lemma from [MHI08] that will be used in the analysis. This lemma applies to our problem, since according to the paper, our operator D_2 is a narrow-diagonal second derivative SBP operator, which is defined as follows.

Definition 3.1 ([MHI08]). *An explicit pth-order accurate finite difference scheme with minimal stencil width of a Cauchy problem, is called a pth-order accurate narrow stencil.* ⌟

Lemma 3.2 ([MHI08]). *The dissipative part A of a narrow-diagonal second derivative SBP operator has the property*

$$\boldsymbol{u}^T A \boldsymbol{u} = h\alpha(BS\boldsymbol{u})_0^2 + h\alpha(BS\boldsymbol{u})_N^2 + \boldsymbol{u}^T \tilde{A} \boldsymbol{u},$$

where $\tilde{A}$ is a symmetric and positive semi-definite matrix, and α is a positive constant independent of h.

For the proof, see [MHI08], where also the different values of α are listed for the second, fourth and sixth-order accurate second derivative SBP operators.

We now turn to the demonstration of stability. We begin by multiplying Equation (3.3) by $\boldsymbol{u}_t^T P$ and adding the transpose. We then have

$$\begin{aligned} \boldsymbol{u}_t^T P \boldsymbol{u}_{tt} + \boldsymbol{u}_{tt}^T P \boldsymbol{u}_t = {} & \boldsymbol{u}_t^T(-A + BS)\boldsymbol{u} + \boldsymbol{u}^T(-A + BS)^T \boldsymbol{u}_t \\ & + \boldsymbol{u}_t^T \left(-S^T E_0 - \frac{\tau_1}{h} E_0\right) \boldsymbol{u} + \boldsymbol{u}^T \left(-S^T E_0 - \frac{\tau_1}{h} E_0\right)^T \boldsymbol{u}_t \\ & + \boldsymbol{u}_t^T \left(S^T E_N - \frac{\tau_2}{h} E_N\right) \boldsymbol{u} + \boldsymbol{u}^T \left(S^T E_N - \frac{\tau_2}{h} E_N\right)^T \boldsymbol{u}_t. \end{aligned} \tag{3.4}$$

The left-hand side can be recognized as $\frac{d}{dt} \|\boldsymbol{u}_t\|^2$. For convenience, we split the right-hand side into three components and consider them separately.

Component 1: The terms from the second derivative approximation

These are the first two terms on the right-hand side in the above equation. By rearranging terms, we obtain

$$\begin{aligned}\boldsymbol{u}_t^T(-A+BS)\boldsymbol{u}+\boldsymbol{u}^T(-A+BS)^T\boldsymbol{u}_t &= -\boldsymbol{u}_t^TA\boldsymbol{u}-\boldsymbol{u}^TA^T\boldsymbol{u}_t+\boldsymbol{u}_t^TBS\boldsymbol{u}+\boldsymbol{u}^T(BS)^T\boldsymbol{u}_t,\\ &= -\frac{d}{dt}\left(\boldsymbol{u}^TA\boldsymbol{u}-\boldsymbol{u}^T(BS\boldsymbol{u})\right).\end{aligned}$$

We now apply Lemma 3.2, which yields

$$\begin{aligned}&\boldsymbol{u}_t^T(-A+BS)\boldsymbol{u}+\boldsymbol{u}^T(-A+BS)^T\boldsymbol{u}_t =\\ &-\frac{d}{dt}\left(\boldsymbol{u}^T\tilde{A}\boldsymbol{u}+h\alpha(BS\boldsymbol{u})_0^2+h\alpha(BS\boldsymbol{u})_N^2-\boldsymbol{u}^T(BS\boldsymbol{u})\right).\end{aligned}\tag{3.5}$$

Component 2: The left SAT

These are the terms $\boldsymbol{u}_t^T\left(-S^TE_0-\frac{\tau_1}{h}E_0\right)\boldsymbol{u}+\boldsymbol{u}^T\left(-S^TE_0-\frac{\tau_1}{h}E_0\right)^T\boldsymbol{u}_t$ in Equation (3.4). Writing them out, we obtain after some manipulations

$$\begin{aligned}\boldsymbol{u}_t^T\left(-S^TE_0-\frac{\tau_1}{h}E_0\right)\boldsymbol{u}+\boldsymbol{u}^T\left(-S^TE_0-\frac{\tau_1}{h}E_0\right)^T\boldsymbol{u}_t &=\\ -\boldsymbol{u}_t^TS^TE_0\boldsymbol{u}-\frac{\tau_1}{h}\boldsymbol{u}_t^TE_0\boldsymbol{u}-\boldsymbol{u}^T(S^TE_0)^T\boldsymbol{u_t}-\frac{\tau_1}{h}\boldsymbol{u}^TE_0^T\boldsymbol{u}_t &=\\ -2(S\boldsymbol{u}_t)_0\boldsymbol{u}_0-2\frac{\tau_1}{h}(\boldsymbol{u}_t)_0\boldsymbol{u}_0.&\end{aligned}\tag{3.6}$$

The resulting terms can be recognized as time derivatives, and we have

$$\begin{aligned}\boldsymbol{u}_t^T\left(-S^TE_0-\frac{\tau_1}{h}E_0\right)\boldsymbol{u}+\boldsymbol{u}^T\left(-S^TE_0-\frac{\tau_1}{h}E_0\right)^T\boldsymbol{u}_t &=\\ -\frac{d}{dt}\left((S\boldsymbol{u})_0\boldsymbol{u}_0+\frac{\tau_1}{h}\boldsymbol{u}_0^2\right).&\end{aligned}\tag{3.7}$$

Component 3: The right SAT

We consider now the two last terms of Equation (3.4). In the same fashion as for the left SAT, we obtain the following.

$$\begin{aligned}\boldsymbol{u}_t^T\left(S^TE_N-\frac{\tau_2}{h}E_N\right)\boldsymbol{u}+\boldsymbol{u}^T\left(S^TE_N-\frac{\tau_2}{h}E_N\right)^T\boldsymbol{u}_t &= 2(S\boldsymbol{u}_t)_N\boldsymbol{u}_N-2\frac{\tau_2}{h}(\boldsymbol{u}_t)_N\boldsymbol{u}_N,\\ &= \frac{d}{dt}\left((S\boldsymbol{u})_N\boldsymbol{u}_N-\frac{\tau_2}{h}\boldsymbol{u}_N^2\right).\end{aligned}$$

Combining the three components (3.5)-(3.7), Equation (3.4) reads

$$\begin{aligned}&\frac{d}{dt}\left\|\boldsymbol{u}_t\right\|^2 =\\ &-\frac{d}{dt}\left(\boldsymbol{u}^T\tilde{A}\boldsymbol{u}+h\alpha(S\boldsymbol{u})_0^2+h\alpha(S\boldsymbol{u})_N^2+2\boldsymbol{u}_0(S\boldsymbol{u})_0-2\boldsymbol{u}_N(S\boldsymbol{u})_N+\frac{\tau_1}{h}\boldsymbol{u}_0^2+\frac{\tau_2}{h}\boldsymbol{u}_N^2\right).\end{aligned}$$

By rearranging terms and writing the resulting right-hand side on matrix form, we obtain

$$\begin{aligned}\frac{d}{dt}\left(\left\|\boldsymbol{u}_t\right\|^2+\left\|\boldsymbol{u}\right\|_{\tilde{A}}^2\right) = \frac{d}{dt}\Bigg(&\begin{pmatrix}(S\boldsymbol{u})_0\\ \boldsymbol{u}_0\end{pmatrix}^T\begin{pmatrix}h\alpha & 1\\ 1 & \frac{\tau_1}{h}\end{pmatrix}\begin{pmatrix}(S\boldsymbol{u}_0)\\ \boldsymbol{u}_0\end{pmatrix}\\ &+\begin{pmatrix}(S\boldsymbol{u})_N\\ \boldsymbol{u}_N\end{pmatrix}^T\begin{pmatrix}h\alpha & -1\\ -1 & \frac{\tau_2}{h}\end{pmatrix}\begin{pmatrix}(S\boldsymbol{u})_N\\ \boldsymbol{u}_N\end{pmatrix}\Bigg)\end{aligned}$$

We define $\boldsymbol{v}_0=\begin{pmatrix}(S\boldsymbol{u})_0\\ \boldsymbol{u}_0\end{pmatrix}$, $M_0=\begin{pmatrix}h\alpha & 1\\ 1 & \frac{\tau_1}{h}\end{pmatrix}$, $\boldsymbol{v}_N=\begin{pmatrix}(S\boldsymbol{u})_N\\ \boldsymbol{u}_N\end{pmatrix}$ and $M_N=\begin{pmatrix}h\alpha & -1\\ -1 & \frac{\tau_2}{h}\end{pmatrix}$.
Consequently, the above equation can be written as

$$\frac{d}{dt}\left(\left\|\boldsymbol{u}_t\right\|^2+\left\|\boldsymbol{u}\right\|_{\tilde{A}}^2\right) = -\frac{d}{dt}\left(\boldsymbol{v}_0^TM_0\boldsymbol{v}_0+\boldsymbol{v}_N^TM_N\boldsymbol{v}_N\right).$$

Integration in time leads to

$$\begin{aligned}\|\boldsymbol{u}_t(t)\|^2 + \|\boldsymbol{u}(t)\|_{\tilde{A}}^2 &= -\boldsymbol{v}_0^T M_0 \boldsymbol{v}_0|_0^t - \boldsymbol{v}_N^T M_N \boldsymbol{v}_N|_0^t + \|\boldsymbol{u}_t(0)\|^2 + \|\boldsymbol{u}(0)\|_{\tilde{A}}^2\,, \\ &= \boldsymbol{v}_0^T(0) M_0 \boldsymbol{v}_0(0) + \boldsymbol{v}_N^T(0) M_N \boldsymbol{v}_N(0) - \boldsymbol{v}_0^T(t) M_0 \boldsymbol{v}_0(t) \\ &\quad - \boldsymbol{v}_N^T(t) M_N \boldsymbol{v}_N(t) + \|\boldsymbol{g}\|^2 + \|\boldsymbol{f}\|_{\tilde{A}}^2\,.\end{aligned}$$

For the scheme to be stable, we require that M_0 and M_N are positive semi-definite. This yields the conditions $\tau_1,\ \tau_2 \geq \frac{1}{\alpha}$. We then have

$$\|\boldsymbol{u}_t(t)\|^2 + \|\boldsymbol{u}(t)\|_{\tilde{A}}^2 \leq \boldsymbol{v}_0^T(0) M_0 \boldsymbol{v}_0(0) + \boldsymbol{v}_N^T(0) M_N \boldsymbol{v}_N(0) + \|\boldsymbol{g}\|^2 + \|\boldsymbol{f}\|_{\tilde{A}}^2\,.$$

The two first terms on the right-hand side above is some known constant C obtained from the initial data, hence we can write

$$\|\boldsymbol{u}_t(t)\|^2 + \|\boldsymbol{u}(t)\|_{\tilde{A}}^2 \leq C + \|\boldsymbol{g}\|^2 + \|\boldsymbol{f}\|_{\tilde{A}}^2\,.$$

Using the same relations (3.2) as for the continuous case, we can obtain a bound on $\|\boldsymbol{u}\|$, and we therefore conclude that the scheme is stable.

3.2.1 Accuracy and convergence rates

Even though it is quite straightforward to obtain high-order approximations in the interior of the domain, more care must be taken near the boundaries. To obtain optimal convergence rates, boundary conditions must be approximated to at most one order less than the interior points (see e.g. [Gus08, Gus75, Gus81]). In [Str94], Strand investigated SBP operators approximating the first derivative. Among them, operators using a diagonal norm, with accuracy in the interior of the domain $2p$, and accuracy $p \leq 4$ at the boundary. This will according to [Gus75, Gus81] yield a convergence rate of $p + 1$. For second derivative approximations satisfying a

summation-by-parts rule, Svärd and Nordström proved in [SN06] that for schemes with boundary accuracy p, the convergence rate is raised to $p + 2$, i.e., two orders are gained. In fact, for PDEs with a nth-order spatial derivative, the accuracy at and near the boundary can be lowered n orders to obtain the convergence rate of the inner scheme.

When analysing numerical schemes such as the one introduced in this chapter, it is possible to obtain an a priori estimate of the convergence rate by deriving an energy estimate for the error between the exact and the numerical solution. To see how this procedure is carried out, see for instance [Gus08]. Here, it is explained that the energy method sometimes demonstrates that the convergence rate is one order higher than the accuracy at the boundary. For other cases, however, only a factor of $\mathcal{O}(h^{1/2})$ is gained by the use of the energy method. From the theory discussed above, it is clear that the observed convergence rate is often higher than what is expected from the analysis based on the energy method.

3.3 Numerical results

In this section, we present the results obtained when implementing the above numerical scheme with an SBP operator that is 6th-order accurate in the interior and 3rd-order accurate near the boundary (see Appendix B for its specific form). The analytical solution is $u(x,t) = \sin(2\pi x)\cos(2\pi t)$, which yields no forcing function, and the scheme is run until $t = 1$.

Table 3.1: Table showing the L^2 errors and convergence using the $(6,3)$ scheme

Grid points	L^2-error	L^2-convergence
100	1.39153e-08	-
200	3.22325e-10	5.39
300	3.72387e-11	5.30
400	7.52297e-12	5.54
500	2.16224e-12	5.58
600	7.99859e-13	5.44

The results listed in Table 3.1 indicate that the convergence rate of the scheme is around 5.5. In Figure 3.1, we see the numerical solution when the number of grid points is 600. The exact solution is not included here, since it looks identical to the numerical solution in the plot. However, Figure 3.2 shows the error between the calculated and the exact solution (notice the scale of the axes). We see from this figure that the biggest errors are along the boundaries, which agrees with the fact that the scheme is three orders less accurate here.

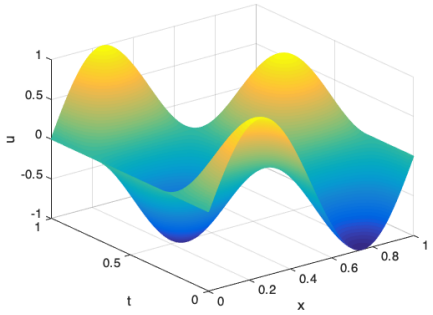


Figure 3.1: Plot of the numerical solution using 600 grid points.

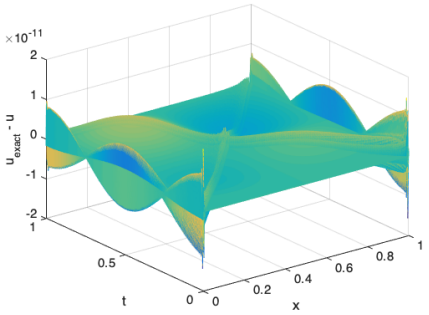


Figure 3.2: Plot of the error between the exact and the numerical solution using 600 grid points.

Chapter 4

Finite volumes and the SBP-SAT technique

For the second part of this project, we consider the finite volume method in two space dimensions. In [SGN07], Svärd et al., showed that two common approximations of the second derivative are inconsistent on unstructured grids. However, if the grid is constructed by equilateral polygons, both approximations are consistent in the interior. The idea of this project is to take an unstructured triangular grid, and transform every triangle to a standard equilateral triangle (see Figure 4.1). Afterwards, we refine the mesh by adding grid points in such a manner that the standard triangle is consisting of only equilateral triangles. In this way, we might recover the accuracy of the second-derivative approximation.

In this chapter, we consider one of the approximations discussed in the paper [SGN07], namely the application of the first-derivative approximation twice. We derive a finite volume method with operators that satisfy a summation-by-parts rule. We begin by discretising the advection equation using these operators and verify by numerical experiments what convergence rates we obtain by including the transformation. Next, we discretise the second-order wave equation using the same operators, and investigate consistency and convergence rates.

4.1 The transformation

Before proceeding to the finite volume method, we start this chapter by introducing the transformation from the physical domain to the standard triangle. The transformation we have used is linear and of the form

$$x = a_1 + a_2\xi + a_3\eta,$$
$$y = b_1 + b_2\xi + b_3\eta.$$

Here, (x, y) are the coordinates in the physical domain, while (ξ, η) are the coordinates in the computational domain (the standard triangle). Each triangle in the unstructured mesh will be transformed into a standard triangle with fixed vertices (see Figure 4.1).

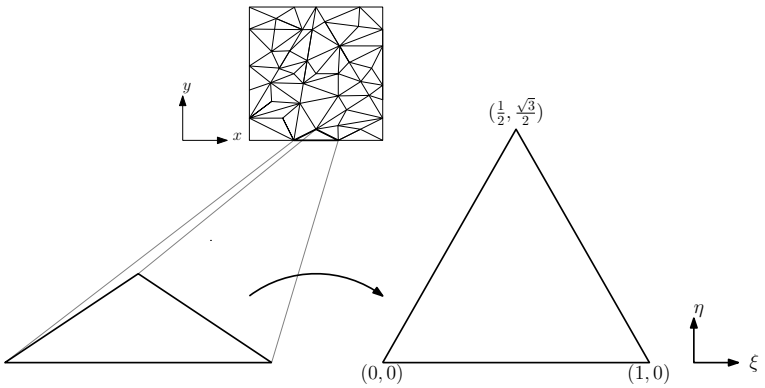


Figure 4.1: Every triangle in the unstructured mesh is transformed into a standard triangle with vertices at $(0,0)$, $(1,0)$ and $(\frac{1}{2}, \frac{\sqrt{3}}{2})$.

Later in this chapter, we will see how the transformation influences the problems we are investigating.

4.2 The finite volume method

In [NFAE03], Nordström et al. analysed the unstructured node-centred finite volume method, and showed that it can be regarded in an SBP framework. The following

derivation was originally found in this article. For the purpose of this thesis, it was verified and we will present it here for the reader's convenience.

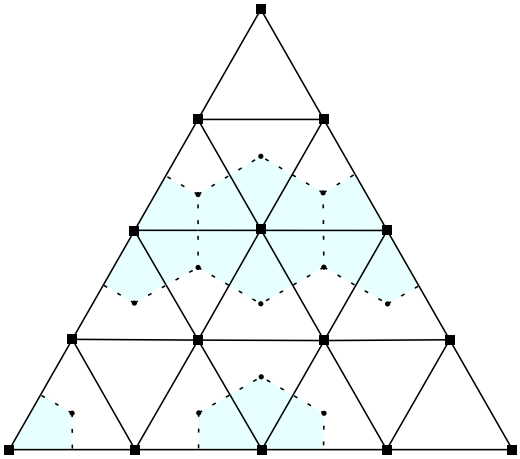


Figure 4.2: Example of a grid on the standard triangle. The dashed lines are the boundaries of the dual volumes, the dots are the centroids and the squares are the nodes.

Consider the advection equation in two space dimensions

$$u_t + au_x + bu_y = 0, \quad (x, y) \in \Omega \tag{4.1}$$

where Ω is the standard triangle, and the domain is divided into equilateral triangles. Unlike the finite difference method, the finite volume method is based on the integral form of the given PDE. We start by dividing the spatial domain into a number of non-overlapping dual volumes. Then we integrate Equation (4.1) over each such volume, which in this case is defined as as the area inside the polygon with vertices at the centroids of the triangles surrounding node i (see Figure 4.2).

$$\iint_{V_i} u_t \, dxdy + \iint_{V_i} au_x + bu_y \, dxdy = 0.$$

By using Green's theorem, we obtain

$$\iint_{V_i} u_t + \oint_{\partial V_i} -bu\,dx + au\,dy = 0. \tag{4.2}$$

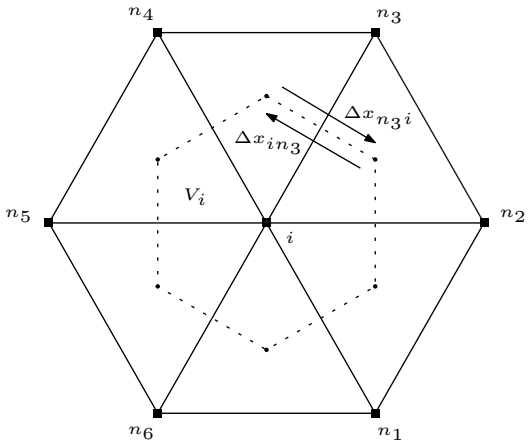


Figure 4.3: An interior point i of the grid with its dual volume V_i and surrounding neighbours $n_1 - n_6$. The differences in the coordinates of the centroids are the Δx and Δy in the approximations of the derivatives.

We want to approximate the volume average of u, which can be expressed as $\bar{u} = \frac{1}{V_i}\iint_{V_i} u\,dxdy$. This means that the first term on the left-hand side of Equation (4.2) can be written $V_i(\bar{u}_i)_t$. Hence, the equation now reads

$$V_i(\bar{u}_i)_t + \oint_{\partial V_i} -bu\,dx + \oint_{\partial V_i} au\,dy = 0. \tag{4.3}$$

The line integrals (fluxes) above are equal to the sum of the line integrals over each edge in the dual volume. We approximate these line integrals by the mean value of the solution at node i and the neighbouring node n, times the difference between the coordinates in the two centroids constituting the corresponding volume side. The orientation of the line integrals is in the counter-clockwise direction. Let N_i be the set of all neighbouring nodes to node i. Then the line integrals in Equation (4.3) can be written as

$$\oint_{\partial V_i} -bu\,dx = -b\sum_{n\in N_i}\frac{u_i+u_n}{2}\Delta x_{in} \qquad \oint_{\partial V_i} au\,dy = a\sum_{n\in N_i}\frac{u_i+u_n}{2}\Delta y_{in}$$

If we now divide Equation (4.3) by V_i, we obtain

$$(u_i)_t - \frac{b}{V_i}\sum_{n\in N_i}\frac{u_i+u_n}{2}\Delta x_{in} + \frac{a}{V_i}\sum_{n\in N_i}\frac{u_i+u_n}{2}\Delta y_{in} = 0, \tag{4.4}$$

which is a discretised version of the original equation (4.1). This means that the x- and y-derivatives are approximated as

$$u_x|_{x_i,y_i} \approx \frac{1}{V_i}\sum_{n\in N_i}\frac{u_i+u_n}{2}\Delta y_{in}, \qquad u_y|_{x_i,y_i} \approx -\frac{1}{V_i}\sum_{n\in N_i}\frac{u_i+u_n}{2}\Delta x_{in}.$$

Hence, the following is a semi-discrete version of Equation (4.1)

$$\boldsymbol{u} + aP^{-1}Q_x\boldsymbol{u} + bP^{-1}Q_y\boldsymbol{u} = 0 \tag{4.5}$$

Here, P^{-1} is a matrix with $\frac{1}{V_i}$ on the diagonal, and the specific form of Q_x and Q_y will be described below.

First, we rewrite the sums in Equation (4.4) as

$$\begin{aligned}\sum_{n\in N_i}\frac{u_i+u_n}{2}\Delta y_{in} &= \sum_{n\in N_i} u_i\frac{\Delta y_{in}}{2} + \sum_{n\in N_i} u_n\frac{\Delta y_{in}}{2}\\ -\sum_{n\in N_i}\frac{u_i+u_n}{2}\Delta x_{in} &= -\sum_{n\in N_i} u_i\frac{\Delta x_{in}}{2} - \sum_{n\in N_i} u_n\frac{\Delta x_{in}}{2}\end{aligned} \tag{4.6}$$

Next, consider all interior points of the grid.

We have that

$$Q_{x_{ii}} = \sum_{n \in N_i} \frac{\Delta y_{in}}{2} = 0 \qquad\qquad Q_{y_{ii}} = -\sum_{n \in N_i} \frac{\Delta x_{in}}{2} = 0,$$

since we are summing over a closed loop. This means that in the interior, Q_x and Q_y will have zeros along the diagonals. Further, we have that the contribution from a neighbouring node n to node i has equal size but opposite sign of the contribution from node i to the neighbouring node n (see Figure 4.3), i.e.,

$$Q_{x_{in}} = \frac{\Delta y_{in}}{2} = -Q_{x_{ni}} \qquad\qquad Q_{y_{in}} = -\frac{\Delta x_{in}}{2} = -Q_{y_{ni}}.$$

This means that Q_x and Q_y are skew-symmetric in the interior.

Let us now consider the nodes along the boundaries. We denote these nodes by b instead of i to clarify that they are indeed boundary nodes.

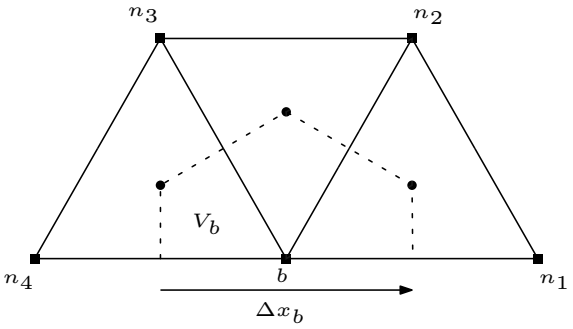


Figure 4.4: The grid around a boundary node b with neighbouring nodes n_1 - n_4. The dashed lines together with the boundary defines the dual volume V_b.

The flux through the boundary is approximated by the value of u at node b times the Δx_b or Δy_b (depending on which integral we are considering) along the boundary (see Figure 4.4). This means that for boundary nodes, we have

$$\text{flux} = \sum_{n \in N_b} \frac{u_b + u_n}{2} \Delta y_{bn} + u_b \Delta y_b - \left(\sum_{n \in N_b} \frac{u_b + u_n}{2} \Delta x_{bn} + u_b \Delta x_b \right).$$

From Figure 4.4, we have that

$$\sum_{n \in N_b} \Delta x_{bn} = -\Delta x_b, \qquad \sum_{n \in N_b} \Delta y_{bn} = -\Delta y_b,$$

which implies that the flux at a boundary node is given by

$$\text{flux} = \sum_{n \in N_b} u_n \frac{\Delta y_{bn}}{2} + u_b \frac{\Delta y_b}{2} - \left(\sum_{n \in N_b} u_n \frac{\Delta x_{bn}}{2} + u_b \frac{\Delta x_b}{2} \right).$$

This means that

$$Q_{x_{bb}} = \frac{\Delta y_b}{2}, \qquad Q_{y_{bb}} = -\frac{\Delta x_b}{2}.$$

As for the interior points, the contribution from a neighbouring node n to the boundary node b has the same size but opposite sign of the contribution from the boundary node b to the neighbouring node n. Thus,

$$Q_{x_{bn}} = \frac{\Delta y_{bn}}{2} = -Q_{x_{nb}}, \qquad Q_{y_{bn}} = -\frac{\Delta x_{bn}}{2} = -Q_{y_{nb}}.$$

Remark. *The property that Q_x and Q_y are almost skew-symmetric means that the sums $Q_x + Q_x^T$ and $Q_y + Q_y^T$ satisfy*

$$Q_x + Q_x^T = B_x, \qquad Q_y + Q_y^T = B_y,$$

where B_x and B_y are diagonal matrices with the boundary elements of Q_x and Q_y, respectively. That is, B_x contains the elements Δy_b, and B_y the elements $-\Delta x_b$. This means that the SBP operators in two space dimensions have the same property as the ones in one space dimension, and the above result corresponds to the SBP property $Q + Q^T = \text{diag}(-1, 0, ..., 0, 1)$ presented in Chapter 2.

We conclude this section by summing up the main results: the matrix P is a diagonal matrix with elements V_i, and Q_x and Q_y are almost skew-symmetric matrices.

4.3 The advection equation

In this section we analyse the advection equation in two space dimensions. To reduce notation, we first consider the equation on a single domain. Thereafter, we show that the problem is well-posed also if we consider blocks that are coupled together by an interface.

4.3.1 The advection equation in the computational domain

The equation must be transformed so that it can be solved in the computational domain. Inserting the transformation presented in Section 4.1, yields

$$\begin{aligned} u_t(x(\xi,\eta), y(\xi,\eta), t) + a u_x(x(\xi,\eta), y(\xi,\eta), t) + b u_y(x(\xi,\eta), y(\xi,\eta), t) = 0, \\ u_t(\xi,\eta,t) + (a\xi_x + b\xi_y) u_\xi(\xi,\eta,t) + (a\eta_x + b\eta_y) u_\eta(\xi,\eta,t) = 0. \end{aligned}$$

As is seen from the above equation, the linear transformation results in a constant coefficient problem in the computational domain that is analogous to the one in the physical domain. The constants $a\xi_x + b\xi_y$ and $a\eta_x + b\eta_y$ corresponds to the a

and b in the original problem, respectively. Hence, proving well-posedness in the computational domain will be equivalent to proving well-posedness in the physical domain. For this reason, we only consider the analysis in the computational domain. Furthermore, for a cleaner presentation in the forthcoming sections, we denote the coordinates in the computational domain (x, y) instead of (ξ, η). We also let a denote $a\xi_x + b\xi_y$ and b denote $a\eta_x + b\eta_y$.

4.3.2 Analysis of the continuous problem without interfaces

We first analyse the continuous problem on a single triangle and prescribe boundary conditions so that the problem is well-posed.

Consider again the advection equation in two space dimensions.

$$u_t + au_x + bu_y = 0, \quad (x, y) \in \Omega. \tag{4.7}$$

To demonstrate well-posedness using the energy method, multiply Equation (4.7) by u and integrate over the domain Ω.

$$\iint_\Omega uu_t \, dxdy = -\iint_\Omega auu_x + buu_y \, dxdy$$

We define $\boldsymbol{v} = (au, bu)$, such that the right-hand side of the above equation can be written

$$\iint_\Omega uu_t \, dxdy = -\iint_\Omega \boldsymbol{\nabla} u \cdot \boldsymbol{v} \, dxdy.$$

Here, we take the nabla operator to mean $\boldsymbol{\nabla} = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y})$. We now split the integral on the right-hand side into two equal parts, and use the integration-by-parts rule

on one of them.

$$\begin{aligned}\frac{1}{2}\frac{d}{dt}\|u\|^2 &= -\iint_\Omega \boldsymbol{\nabla} u \cdot \boldsymbol{v}\, dxdy \\ &= -\frac{1}{2}\iint_\Omega \boldsymbol{\nabla} u \cdot \boldsymbol{v}\, dxdy - \frac{1}{2}\oint_{\partial\Omega} u(\boldsymbol{v}\cdot\boldsymbol{n})\, ds + \frac{1}{2}\iint_\Omega u\boldsymbol{\nabla}\cdot\boldsymbol{v}\, dxdy.\end{aligned}$$

The first and last term of the right-hand side cancel, and we obtain

$$\begin{aligned}\frac{1}{2}\frac{d}{dt}\|u\|^2 &= -\frac{1}{2}\oint_{\partial\Omega} u(\boldsymbol{v}\cdot\boldsymbol{n})\, ds \\ &= -\frac{1}{2}\oint_{\partial\Omega} u^2(a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n}\, ds\end{aligned} \tag{4.8}$$

where $\boldsymbol{e}_x$ and $\boldsymbol{e}_y$ are the unit vectors, $\boldsymbol{n}ds = (dy, -dx)$ and $|\boldsymbol{n}| = 1$. Let $((a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n})^-$ denote the part of the boundary where $(a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n} < 0$, and $((a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n})^+$ the part where $(a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n} \geq 0$. Then Equation (4.8) can be written

$$\frac{1}{2}\frac{d}{dt}\|u\|^2 = -\frac{1}{2}\oint_{\partial\Omega} u^2((a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n})^-\, ds - \frac{1}{2}\oint_{\partial\Omega} u^2((a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n})^+\, ds.$$

The term $(a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n} \geq 0$, does not contribute to any growth in the norm of the solution, hence this part of the boundary can be disregarded, and we obtain

$$\frac{d}{dt}\|u\|^2 \leq -\oint_{\partial\Omega} u^2((a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n})^-\, ds.$$

Following the procedure done in [SN04], we add the penalty term $\oint_{\partial\Omega} u(u-g)(a\boldsymbol{e}_x + b\boldsymbol{e}_y\cdot\boldsymbol{n})^-\, ds = 0$ (where g is the boundary data), which yields

$$\begin{aligned}\frac{d}{dt}\|u\|^2 &= -\oint_{\partial\Omega} u^2((a\boldsymbol{e_x}+b\boldsymbol{e_y})\cdot\boldsymbol{n})^- \, ds + \oint_{\partial\Omega} u(u-g)(a\boldsymbol{e_x}+b\boldsymbol{e_y}\cdot\boldsymbol{n})^- \, ds \\ &= -\oint_{\partial\Omega} \big(u^2((a\boldsymbol{e_x}+b\boldsymbol{e_y})\cdot\boldsymbol{n})^- - u^2((a\boldsymbol{e_x}+b\boldsymbol{e_y})\cdot\boldsymbol{n})^- \\ &\qquad + ug((a\boldsymbol{e_x}+b\boldsymbol{e_y})\cdot\boldsymbol{n})^-\big) \, ds \\ &= -\oint_{\partial\Omega} ug((a\boldsymbol{e_x}+b\boldsymbol{e_y})\cdot\boldsymbol{n})^- \, ds\end{aligned}$$

By setting the boundary data to zero, and integrating in time, we obtain the following estimate

$$\|u(\cdot,\cdot,t)\|^2 = \|u(\cdot,\cdot,0)\|^2 = \|f(\cdot,\cdot)\|^2,$$

which proves well-posedness of problem (4.7) in the sense of Definition 2.1, with $u = g$ along the boundary where $(a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n} < 0$.

4.3.3 Analysis of the discrete problem without interfaces

We now introduce the scheme for the advection equation on a single triangle. Based on the operators derived in Section 4.2, we propose the following semi-discrete scheme.

Proposition 4.1. *The scheme*

$$\boldsymbol{u}_t + aP^{-1}Q_x\boldsymbol{u} + bP^{-1}Q_y\boldsymbol{u} = P^{-1}\mathbb{S}\mathbb{A}\mathbb{T}, \tag{4.9}$$

with

$$\mathbb{SAT} = \begin{cases} (a\Delta y_b - b\Delta x_b)(u_b - g_b), & \text{for boundary nodes } b \text{ with boundary conditions} \\ 0, & \text{otherwise} \end{cases}$$

is a stable semi-discretisation of Equation (4.7) *with* $u = g$ *on the boundaries with boundary conditions.*

Proof. The goal is to derive an energy estimate. A similar derivation can be found in [NFAE03]. Using the discrete energy method, multiply Equation (4.9) by $\boldsymbol{u}^T P$ and add the transpose.

$$\begin{aligned} \boldsymbol{u}^T P \boldsymbol{u}_t + \boldsymbol{u}_t^T P \boldsymbol{u} &= -a\boldsymbol{u}^T Q_x \boldsymbol{u} - a\boldsymbol{u}^T Q_x^T \boldsymbol{u} - b\boldsymbol{u}^T Q_y \boldsymbol{u} - b\boldsymbol{u}^T Q_y^T \boldsymbol{u} + 2\boldsymbol{u}^T \mathbb{SAT}, \\ &= -a\boldsymbol{u}^T (Q_x + Q_x^T)\boldsymbol{u} - b\boldsymbol{u}^T (Q_y + Q_y^T)\boldsymbol{u} + 2\boldsymbol{u}^T \mathbb{SAT} \end{aligned}$$

We recognize the left-hand side as a time derivative, and utilize the fact that $Q_x + Q_x^T = B_x$, $Q_y + Q_y^T = B_y$.

$$\begin{aligned} \frac{d}{dt} \|\boldsymbol{u}\|^2 &= -2a \sum_{i \in B} \frac{u_i^2}{2} \Delta y_i + 2b \sum_{i \in B} \frac{u_i^2}{2} \Delta x_i + 2 \sum_{i \in B} u_i \mathbb{SAT}_i, \\ &= -\sum_{i \in B} u_i^2 (a\Delta y_i - b\Delta x_i) + 2 \sum_{i \in B} u_i \mathbb{SAT}_i. \end{aligned}$$

Here, B denotes the set of all boundary nodes. In the same fashion as for the continuous case, we divide the sum $-\sum_{i \in B} u_i^2 (a\Delta y_i - b\Delta x_i)$ into two parts.

$$\frac{d}{dt} \|\boldsymbol{u}\|^2 = -\sum_{\substack{i \in B \text{ s.t.} \\ (a\Delta y_i - b\Delta x_i) \geq 0}} u_i^2 (a\Delta y_i - b\Delta x_i) - \sum_{\substack{i \in B \text{ s.t.} \\ (a\Delta y_i - b\Delta x_i) < 0}} u_i^2 (a\Delta y_i - b\Delta x_i) + 2 \sum_{i \in B} u_i \mathbb{SAT}_i.$$

The first term on the right-hand side is less than or equal to zero, and will therefore

not contribute to any growth. The estimate can therefore be written

$$\frac{d}{dt}\|\boldsymbol{u}\|^2 \leq -\sum_{\substack{i\in B \text{ s.t.} \\ (a\Delta y_i - b\Delta x_i)<0}} u_i^2(a\Delta y_i - b\Delta x_i) + 2\sum_{i\in B} u_i \mathbb{SAT}_i.$$

Next, we insert the specific SAT-term from the proposition to prove that the scheme is stable.

$$\begin{aligned}\frac{d}{dt}\|\boldsymbol{u}\|^2 &\leq -\sum_{\substack{i\in B \text{ s.t.} \\ (a\Delta y_i - b\Delta x_i)<0}} u_i^2(a\Delta y_i - b\Delta x_i) + 2\sum_{i\in B} u_i(a\Delta y_i - b\Delta x_i)(u_i - g_i), \\ &= \sum_{\substack{i\in B \text{ s.t.} \\ (a\Delta y_i - b\Delta x_i)<0}} -u_i^2(a\Delta y_i - b\Delta x_i) + 2u_i^2(a\Delta y_i - b\Delta x_i) - 2u_i g_i(a\Delta y_i - b\Delta x_i), \\ &= \sum_{\substack{i\in B \text{ s.t.} \\ (a\Delta y_i - b\Delta x_i)<0}} u_i^2(a\Delta y_i - b\Delta x_i) - 2u_i g_i(a\Delta y_i - b\Delta x_i).\end{aligned}$$

The first term on the right-hand side is less than or equal to zero, which means we have

$$\frac{d}{dt}\|\boldsymbol{u}\|^2 \leq -2u_i g_i(a\Delta y_i - b\Delta x_i). \tag{4.10}$$

From Chapter 2, we can set boundary data to zero in the stability analysis, without loss of stability, hence the estimate reads

$$\frac{d}{dt}\|\boldsymbol{u}\|^2 \leq 0.$$

Integration in time yields the final estimate

$$\|\boldsymbol{u}(t)\|^2 \leq \|\boldsymbol{u}(0)\|^2 ,$$

which proves stability of the scheme in the sense of Definition 2.5. □

4.3.4 Analysis of the continuous problem with interfaces

The theory about the interface treatment can be found in [LN14], [SN14] and [CNG99].

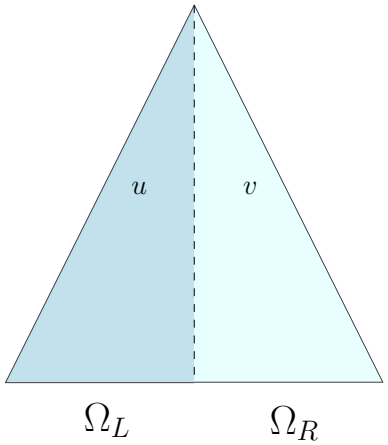


Figure 4.5: Example of a grid with an interface (dashed line).

For simplicity, we consider a physical domain with only one interface, like the one in Figure 4.5. The extension of the analysis to several interfaces is straightforward as they are handled in the same way. We let u denote the solution in the left sub-domain Ω_L, and v the solution in the right sub-domain Ω_R. We have to show that the problem is well-posed, even with the coupling of the two blocks along the interface. We still let (x, y) denote the coordinates in the computational domain. We let also from now Ω_L and Ω_R denote the respective computational domains of the triangles in Figure 4.5.

First, we split the equation into two parts

$$u_t + au_x + bu_y = 0, \quad (x, y) \in \Omega_L$$
$$v_t + av_x + bv_y = 0, \quad (x, y) \in \Omega_R$$

Next, we apply the energy method to both parts, and then add the two equations. We refer the reader to Section 4.3.2 for the derivation of the energy estimate for the advection equation. We skip to the part of the derivation where the intergration-by-parts rule has been applied. We then have

$$\frac{d}{dt} \|u(\cdot, \cdot, t)\|^2_{\Omega_L} = -\int_{\partial\Omega_{L_B}} u^2(a\boldsymbol{e}_x + b\boldsymbol{e}_y) \cdot \boldsymbol{n}_L \, ds - \int_{\partial\Omega_{L_I}} u^2(a\boldsymbol{e}_x + b\boldsymbol{e}_y) \cdot \boldsymbol{n}_L \, ds,$$
$$\frac{d}{dt} \|v(\cdot, \cdot, t)\|^2_{\Omega_R} = -\int_{\partial\Omega_{R_B}} v^2(a\boldsymbol{e}_x + b\boldsymbol{e}_y) \cdot \boldsymbol{n}_R \, ds - \int_{\partial\Omega_{R_I}} v^2(a\boldsymbol{e}_x + b\boldsymbol{e}_y) \cdot \boldsymbol{n}_R \, ds.$$

In the above equations, the subscripts L and R denote the left and right part of the domain, respectively, B denotes the parts of the sub-domains that are outer boundaries, while I denotes the parts of the sub-domains that are interfaces. We have already seen that the problem with only outer boundaries is well-posed, so we will disregard this part and focus only on the interface. We now add the two equations and use the short-hand notation $\frac{d}{dt} \|w(\cdot, \cdot, t)\|^2 = \frac{d}{dt} \|u(\cdot, \cdot, t)\|^2_{\Omega_L} + \frac{d}{dt} \|v(\cdot, \cdot, t)\|^2_{\Omega_R}$ to obtain

$$\frac{d}{dt} \|w(\cdot, \cdot, t)\|^2 = -\int_{\partial\Omega_{L_I}} u^2(a\boldsymbol{e}_x + b\boldsymbol{e}_y) \cdot \boldsymbol{n}_L \, ds - \int_{\partial\Omega_{R_I}} v^2(a\boldsymbol{e}_x + b\boldsymbol{e}_y) \cdot \boldsymbol{n}_R \, ds.$$

We have that $\partial\Omega_{L_I} = \partial\Omega_{R_I}$ and $\boldsymbol{n}_R = -\boldsymbol{n}_L$. Using this, yields

$$\frac{d}{dt} \|w(\cdot, \cdot, t)\|^2 = -\int_{\partial\Omega_I} (u^2 - v^2)(a\boldsymbol{e}_x + b\boldsymbol{e}_y) \cdot \boldsymbol{n}_L \, ds.$$

We follow again the procedure in [SN04] and add the term $\int_{\partial\Omega_I} \tau_1 u(u-v)(a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n}_L - \tau_2 v(v-u)(a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n}_L\, ds = 0$ to obtain

$$\begin{aligned}\frac{d}{dt}\left\|w(\cdot,\cdot,t)\right\|^2 = &-\int_{\partial\Omega_I}(u^2-v^2)(a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n}_L\, ds \\ &+\int_{\partial\Omega_I}(\tau_1 u(u-v)(a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n}_L \\ &\qquad -\tau_2 v(v-u)(a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n}_L)\; ds, \\ = &\int_{\partial\Omega_I}\left((\tau_1-1)u^2 - \tau_1 uv + (1-\tau_2)v^2 + \tau_2 uv\right)(a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n}_L\, ds.\end{aligned}$$

Depending on the sign of $(a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n}_L$, we get different criteria for the parameters τ_1 and τ_2. If $(a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n}_L < 0$, then $\tau_1 \geq 1$, $\tau_1 + \tau_2 = 2$ is required for well-posedness. If $(a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n}_L > 0$, then we require that $\tau_1 \leq 1$ and $\tau_1 + \tau_2 = 2$. When $(a\boldsymbol{e}_x + b\boldsymbol{e}_y)\cdot\boldsymbol{n}_L = 0$, the integral over the interface will vanish, and hence not contribute to any growth in the solution.

4.3.5 Analysis of the discrete problem with interfaces

We now turn to the semi-discretisation of the advection equation on a grid consisting of two blocks with an interface. Also here, we can extend the theory to include several interfaces, but for a cleaner presentation, we consider only one.

We proceed in the same way as for the continuous case. We divide the equation into two parts - one for each block, then we impose the interface conditions weakly using the SAT technique in a similar manner as for the boundary conditions. Let now $\boldsymbol{u}$ and $\boldsymbol{v}$ denote the solutions in the left and right sub-domains, respectively. We propose the following scheme.

Proposition 4.2. *The scheme*

$$\boldsymbol{u}_t + a_L P_L^{-1} Q_{x_L} \boldsymbol{u} + b_L P_L^{-1} Q_{y_L} \boldsymbol{u} = P_L^{-1} \mathbb{SAT}_B + P_L^{-1} \mathbb{SAT}_I \tag{4.11a}$$

$$\boldsymbol{v}_t + a_R P_R^{-1} Q_{x_R} \boldsymbol{v} + b_R P_R^{-1} Q_{y_R} \boldsymbol{v} = P_R^{-1} \mathbb{SAT}_B + P_R^{-1} \mathbb{SAT}_I \tag{4.11b}$$

with $\mathbb{SAT}_B$ *as in Proposition 4.1, and*

$$\mathbb{SAT}_I = \begin{cases} \omega_1(u_i - v_i)\left((a_L \boldsymbol{e}_{xL} + b_L \boldsymbol{e}_{yL}) \cdot \boldsymbol{n}_L\right), & i \in \partial\Omega_{L_I} \\ \omega_2(v_i - u_i)\left((a_R \boldsymbol{e}_{xR} + b_R \boldsymbol{e}_{yR}) \cdot \boldsymbol{n}_R\right), & i \in \partial\Omega_{R_I} \\ 0, & \text{otherwise} \end{cases}$$

is stable.

Proof. Using the discrete energy method, multiply Equation (4.11a) by $\boldsymbol{u}^T P_L$ and Equation (4.11b) by $\boldsymbol{v}^T P_R$, and add the transposes to obtain

$$\begin{aligned} \frac{d}{dt}\|\boldsymbol{u}\|_{\Omega_L}^2 &= -a_L \boldsymbol{u}^T (Q_{x_L} + Q_{x_L}^T)\boldsymbol{u} - b_L \boldsymbol{u}^T (Q_{y_L} + Q_{y_L}^T)\boldsymbol{u} + 2\boldsymbol{u}^T \mathbb{SAT}_B + 2\boldsymbol{u}^T \mathbb{SAT}_{I_L} \\ &= -2a_L \left(\sum_{i \in B} \frac{u_i^2}{2} \Delta y_i + \sum_{i \in I_L} \frac{u_i^2}{2} \Delta y_i \right) + 2b_L \left(\sum_{i \in B} \frac{u_i^2}{2} \Delta x_i + \sum_{i \in I_L} \frac{u_i^2}{2} \Delta x_i \right) \\ &\quad + 2 \sum_{i \in B} u_i \mathbb{SAT}_i + 2 \sum_{i \in I_L} u_i \mathbb{SAT}_i \\ \frac{d}{dt}\|\boldsymbol{v}\|_{\Omega_R}^2 &= -2a_R \left(\sum_{i \in B} \frac{v_i^2}{2} \Delta y_i + \sum_{i \in I_R} \frac{v_i^2}{2} \Delta y_i \right) + 2b_R \left(\sum_{i \in B} \frac{v_i^2}{2} \Delta x_i + \sum_{i \in I_R} \frac{v_i^2}{2} \Delta x_i \right) \\ &\quad + 2 \sum_{i \in B} v_i \mathbb{SAT}_i + 2 \sum_{i \in I_R} v_i \mathbb{SAT}_i. \end{aligned}$$

From the proof of Proposition 4.1, we know that the boundary nodes are not causing any instabilities, hence we disregard these nodes in the rest of the analysis, and focus only on the interface nodes.

We now add the two equations and use the short-hand notation $\frac{d}{dt}\|\boldsymbol{w}\|^2 = \frac{d}{dt}\|\boldsymbol{u}\|^2_{\Omega_L} + \frac{d}{dt}\|\boldsymbol{v}\|^2_{\Omega_R}$. This yields

$$\begin{aligned}\frac{d}{dt}\|\boldsymbol{w}\|^2 = &-2a_L\sum_{i\in I_L}\frac{u_i^2}{2}\Delta y_i + 2b_L\sum_{i\in I_L}\frac{u_i^2}{2}\Delta x_i + 2\sum_{i\in I_L}u_i\mathbb{SAT}_i\\ &-2a_R\sum_{i\in I_R}\frac{v_i^2}{2}\Delta y_i + 2b_R\sum_{i\in I_R}\frac{v_i^2}{2}\Delta x_i + 2\sum_{i\in I_R}v_i\mathbb{SAT}_i.\end{aligned}$$

For simplicity, we consider now only one interface node k. Since this node is arbitrary, stability of the whole scheme follows if we are able to prove stability for this node. For the rest of the proof, we write only the right-hand side of the above equation.

$$\begin{aligned}&-a_Lu_k^2\Delta y_{k_L} + b_Lu_k^2\Delta x_{k_L} + 2u_k\mathbb{SAT}_{I_{kL}} - a_Rv_k^2\Delta y_{k_R} + b_Rv_k^2\Delta x_{k_R} + 2v_k\mathbb{SAT}_{I_{kR}}\\ &= -u_k^2(a_L\boldsymbol{e}_{xL} + b_L\boldsymbol{e}_{yL})\cdot\boldsymbol{n}_L + 2u_k\mathbb{SAT}_{I_{kL}} - v_k^2(a_R\boldsymbol{e}_{xR} + b_R\boldsymbol{e}_{yR})\cdot\boldsymbol{n}_R + 2v_k\mathbb{SAT}_{I_{kR}},\end{aligned}$$

where $\boldsymbol{n}_L = (\Delta y_{k_L}, -\Delta x_{k_L})$ and $\boldsymbol{n}_R = (\Delta y_{k_R}, -\Delta x_{k_R})$. We insert the respective SAT-terms to obtain

$$\begin{aligned}&-u_k^2(a_L\boldsymbol{e}_{xL} + b_L\boldsymbol{e}_{yL})\cdot\boldsymbol{n}_L + 2\omega_1u_k(u_k - v_k)\left((a_L\boldsymbol{e}_{xL} + b_L\boldsymbol{e}_{yL})\cdot\boldsymbol{n}_L\right)\\ &-v_k^2(a_R\boldsymbol{e}_{xR} + b_R\boldsymbol{e}_{yR})\cdot\boldsymbol{n}_R + 2\omega_2v_k(v_k - u_k)\left((a_R\boldsymbol{e}_{xR} + b_R\boldsymbol{e}_{yR})\cdot\boldsymbol{n}_R\right)\\ &= -u_k^2(a_L\boldsymbol{e}_{xL} + b_L\boldsymbol{e}_{yL})\cdot\boldsymbol{n}_L + 2\omega_1u_k^2(a_L\boldsymbol{e}_{xL} + b_L\boldsymbol{e}_{yL})\cdot\boldsymbol{n}_L - 2\omega_1u_kv_k(a_L\boldsymbol{e}_{xL} + b_L\boldsymbol{e}_{yL})\cdot\boldsymbol{n}_L\\ &-v_k^2(a_R\boldsymbol{e}_{xR} + b_R\boldsymbol{e}_{yR})\cdot\boldsymbol{n}_R + 2\omega_2v_k^2(a_R\boldsymbol{e}_{xR} + b_R\boldsymbol{e}_{yR})\cdot\boldsymbol{n}_R - 2\omega_2v_ku_k(a_R\boldsymbol{e}_{xR} + b_R\boldsymbol{e}_{yR})\cdot\boldsymbol{n}_R.\end{aligned}$$

We now utilize the fact that $(a_R\boldsymbol{e}_{xR} + b_R\boldsymbol{e}_{yR})\cdot\boldsymbol{n}_R = -(a_L\boldsymbol{e}_{xL} + b_L\boldsymbol{e}_{yL})\cdot\boldsymbol{n}_L$, which yields

$$\left((2\omega_1 - 1)u_k^2 - 2(\omega_1 - \omega_2)u_kv_k + (1 - 2\omega_2)v_k^2\right)(a_L\boldsymbol{e}_{xL} + b_L\boldsymbol{e}_{yL})\cdot\boldsymbol{n}_L.$$

As for the continuous case, we get different criteria on the parameters ω_1 and ω_2 depending on the sign of $(a_L\boldsymbol{e}_{xL} + b_L\boldsymbol{e}_{yL}) \cdot \boldsymbol{n}_L$. If $(a_L\boldsymbol{e}_{xL} + b_L\boldsymbol{e}_{yL}) \cdot \boldsymbol{n}_L < 0$, then we require $\omega_1 \geq \frac{1}{2}$ and $\omega_1 + \omega_2 = 1$, and if $(a_L\boldsymbol{e}_{xL} + b_L\boldsymbol{e}_{yL}) \cdot \boldsymbol{n}_L > 0$, then $\omega_1 \leq \frac{1}{2}$ and $\omega_1 + \omega_2 = 1$. The case when $(a_L\boldsymbol{e}_{xL} + b_L\boldsymbol{e}_{yL}) \cdot \boldsymbol{n}_L = 0$ yields no restrictions on the parameter as the above terms vanish. □

The restrictions on ω_1 and ω_2 given in the proposition above, proves stability for the numerical scheme. In addition to the scheme being stable, we also want it to be conservative, since the governing equations are conservation laws. The following theory applied to our problem can be found in for example [EAN11], [SN14], [LN14] and [CNG99].

The weak form of the advection equation (4.7) can be obtained by multiplying by a smooth test function Φ with compact support (which in this case means that it vanishes at the boundaries) and integrating over the spatial domain and in time.

$$\int_\Omega \Phi u|_0^t \, d\Omega - \int_\Omega \int_0^t \Phi_t u + a\Phi_x u + b\Phi_y u \, dt \, d\Omega = 0.$$

Here, we have used the integration-by-parts rule to move the spatial derivatives from the solution u to the test function Φ. We want the numerical scheme to mimic the above equation. To demonstrate that the conservation property indeed applies to the numerical scheme, multiply equation (4.11a) by $\boldsymbol{\phi}_L^T P_L$ and equation (4.11b) by $\boldsymbol{\phi}_R^T P_R$. Here, $(\boldsymbol{\phi}_{L,R})_i(t) = \Phi(x_i, y_i, t)$. Since Φ has compact support, all outer boundary terms will vanish, and we therefore neglect the boundary SAT in the derivation.

$$\begin{aligned}
\boldsymbol{\phi}_L^T P_L \boldsymbol{u}_t + a_L \boldsymbol{\phi}_L^T Q_{x_L} \boldsymbol{u} + b_L \boldsymbol{\phi}_L^T Q_{y_L} \boldsymbol{u} &= \boldsymbol{\phi}_L^T \mathbb{SAT}_{I_L}, \\
\boldsymbol{\phi}_R^T P_R \boldsymbol{v}_t + a_R \boldsymbol{\phi}_R^T Q_{x_R} \boldsymbol{v} + b_R \boldsymbol{\phi}_R^T Q_{y_R} \boldsymbol{v} &= \boldsymbol{\phi}_R^T \mathbb{SAT}_{I_R}.
\end{aligned}$$

We now add the two equations, and utilize the fact that $\boldsymbol{\phi}^T P \boldsymbol{w}_t = \frac{d}{dt}(\boldsymbol{\phi}^T P \boldsymbol{w}) - \boldsymbol{\phi}_t^T P \boldsymbol{w}$ and $Q_x + Q_x^T = B_x$, $Q_y + Q_y^T = B_y$. We then obtain

$$\begin{aligned}\boldsymbol{\phi}_L^T P_L \boldsymbol{u}|_0^t + \boldsymbol{\phi}_R^T P_R \boldsymbol{v}|_0^t - \int_0^t \Bigg(& (\boldsymbol{\phi})_t^T P_L \boldsymbol{u} + (\boldsymbol{\phi}_R)_t^T P_R \boldsymbol{v} \\ & + a_L (D_{x_L} \boldsymbol{\phi}_L)^T P_L \boldsymbol{u} + b_L (D_{y_L} \boldsymbol{\phi}_L)^T P_L \boldsymbol{u} \\ & + a_R (D_{x_R} \boldsymbol{\phi}_R)^T P_R \boldsymbol{v} + b_R (D_{y_R} \boldsymbol{\phi}_R)^T P_R \boldsymbol{v} \\ & + a_L \sum_{i \in I} \phi_i u_i \Delta y_{i_L} + a_R \sum_{i \in I} \phi_i v_i \Delta y_{i_R} \\ & - b_L \sum_{i \in I} \phi_i u_i \Delta x_{i_L} - b_R \sum_{i \in I} \phi_i v_i \Delta x_{i_R} \\ & - \omega_1 \sum_{i \in I} \phi_i (u_i - v_i) \left((a\boldsymbol{e}_x + b\boldsymbol{e}_y) \cdot \boldsymbol{n}_L \right) \\ & - \omega_2 \sum_{i \in I} \phi_i (v_i - u_i) \left((a\boldsymbol{e}_x + b\boldsymbol{e}_y) \cdot \boldsymbol{n}_R \right) \Bigg) \, dt = 0.\end{aligned}$$

For the semi-discretization to be conservative, we need the last four lines in the integral above to cancel. We consider now only one interface node k, and rearrange the terms in question. This gives us

$$\phi_k (a\boldsymbol{e}_x + b\boldsymbol{e}_y) \cdot \boldsymbol{n}_L (u_k - v_k - \omega_1 (u_k - v_k) - \omega_2 (u_k - v_k)),$$

which cancels due to the stability condition $\omega_1 + \omega_2 = 1$. Hence, the numerical scheme is conservative.

Before presenting the numerical results, we briefly investigate what convergence rates the scheme will generate, theoretically. For finite difference methods, better rates are often observed in the numerical experiments, and there is a chance we have the same case for the finite volume method. However, the analysis provides an idea of what rates we could at least expect.

4.3.6 Convergence analysis

According to [SGN07], it is possible to show that the approximations for the first derivatives are first-order accurate in the interior of the domain on unstructured grids. We therefore expect at least first-order accuracy on the standard triangle as well. In fact, it can be shown (see Appendix A) that the x-derivative is second-order accurate in the interior of this domain. Numerical experiments corroborate this result, and show that we have a similar case for the y-derivative. The truncation errors can be summed up as follows for the first derivatives. For interior nodes, we have $T = \mathcal{O}(h^2)$, for boundary nodes $T = \mathcal{O}(h)$ and for corner nodes $T = \mathcal{O}(1)$.

The theory for convergence rates for the finite difference methods does not apply for the finite volume method formulated on unstructured grids. However, we can estimate the convergence rate by determining an energy estimate for the error of the solution. See for example [Gus08] or [GKO95] for more about the following procedure.

To distinguish the true solution from the numerical solution, we denote them by u and v, respectively. The error at a point (x_i, y_i, t) is then expressed as $e_i = u(x_i, y_i, t) - v_i(t)$. The error will satisfy the scheme

$$\boldsymbol{e}_t + aP^{-1}Q_x\boldsymbol{e} + bP^{-1}Q_x\boldsymbol{e} + \boldsymbol{T} = P_b^{-1}(a\Delta y_b - b\Delta x_b)e_b. \tag{4.12}$$

Where $\boldsymbol{T}$ is the vector containing the truncation errors. In the above equation, we have allowed for a slight abuse of notation. The boundary term $P_b^{-1}(a\Delta y_b - b\Delta x_b)e_b$ should naturally be written in vector form as well, but since it does not play a big role in the derivation of the convergence rate, we let it represent its corresponding term in vector form. We now derive an energy estimate for Equation (4.12) to obtain a bound for the error.

$$\begin{aligned}\boldsymbol{e}^T P\boldsymbol{e}_t + \boldsymbol{e}_t^T P\boldsymbol{e} + a\boldsymbol{e}^T(Q_x + Q_x^T)\boldsymbol{e} + b\boldsymbol{e}^T(Q_y + Q_y^T)\boldsymbol{e} \\ -2e_b^T(a\Delta y_b - b\Delta x_b)e_b + \boldsymbol{e}^T P\boldsymbol{T} + \boldsymbol{T}^T P\boldsymbol{e} = 0.\end{aligned}$$

From the earlier analysis of the advection equation, we obtain

$$\begin{aligned} \frac{d}{dt}\|\boldsymbol{e}\|^2 &\leq 2\langle \boldsymbol{e}, \boldsymbol{T}\rangle \leq 2\|\boldsymbol{e}\|\,\|\boldsymbol{T}\|\,, \\ \frac{d}{dt}\|\boldsymbol{e}\|^2 = 2\|\boldsymbol{e}\|\frac{d}{dt}\|\boldsymbol{e}\| &\leq 2\|\boldsymbol{e}\|\,\|\boldsymbol{T}\|\,, \\ \frac{d}{dt}\|\boldsymbol{e}\| &\leq \|\boldsymbol{T}\|\,. \end{aligned}$$

We now integrate in time and utilize the fact that $\|\boldsymbol{e}(0)\| = 0$. Let N denote the number of grid points along each boundary. Then we have $\mathcal{O}(N) = \mathcal{O}(1/h)$. The dual volume is of order h^2, i.e., $V = Ch^2$, where C is some constant. This yields

$$\begin{aligned} \|\boldsymbol{e}\| &\leq \int_0^t \sqrt{Ch^2 \cdot \mathcal{O}(h^2)^2 \cdot \mathcal{O}(N^2) + Ch^2 \cdot \mathcal{O}(h)^2 \cdot \mathcal{O}(N) + Ch^2 \cdot \mathcal{O}(1)^2 \cdot \mathcal{O}(1)}\, dt, \\ &\leq \int_0^t \mathcal{O}(h^2) + \mathcal{O}(h^{3/2}) + \mathcal{O}(h)\, dt. \end{aligned}$$

This means that we expect a convergence rate of at least one for the numerical schemes proposed in this section.

4.3.7 Numerical results

The Advection Equation without Interfaces

In this section we look at the results obtained when implementing the scheme proposed in Proposition 4.1. We show two different cases.

Case 1:

We consider the problem on the physical domain showed in Figure 4.6 with the problem data $a = 2.0$, $b = 0.5$ and the analytical solution $u(x, y, t) = e^{-3(x-at)^2-3(y-bt)^2}$, which results in a zero forcing function. The scheme is run until $t = 1$. The results are presented in Table 4.1. The results listed here, show that the convergence rate

is higher than what was expected from the analysis. However, we notice that it is deteriorating, which indicates that we cannot draw the conclusion that a full order is gained.

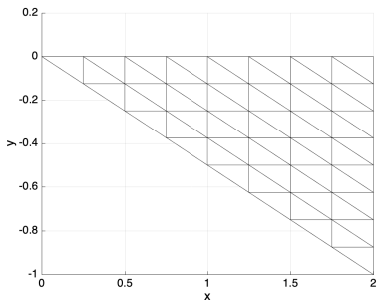


Figure 4.6: The physical domain for the implementation of the advection equation on a single block with the data $a = 2.0$, $b = 0.5$, $u(x, y, t) = e^{-3(x-at)^2-3(y-bt)^2}$. Here displayed with a refinement number of 9.

Table 4.1: Table showing the L^2 errors and convergence for the advection equation on a single block with the data $a = 2.0$, $b = 0.5$, $u(x, y, t) = e^{-3(x-at)^2-3(y-bt)^2}$, on the grid displayed in Figure 4.6.

Grid points along each boundary	L^2-error	L^2-convergence
9	0.01913	-
17	0.00427	2.16
33	9.84074e-04	2.12
65	2.43164e-04	2.02
129	6.19436e-05	1.97
257	1.61835e-05	1.94

Case 2:

We now consider the problem on the physical domain displayed in Figure 4.7 with the problem data $a = 2.0$, $b = -1.0$ and the analytical solution $u(x, y, t) = e^{-3(x-at)^2-3(y-bt)^2}$. The scheme is run until $t = 1$. The results are presented in Table 4.2. The results listed here, also show that the convergence rate is higher than what was expected from the analysis. However, they are not as high as for

case 1, which gives an even stronger indication that we do not gain one order of convergence.

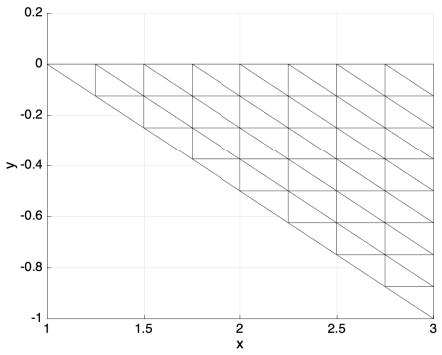


Figure 4.7: The physical domain for the implementation of the advection equation with data $a = 2.0$, $b = -1.0$, $u(x, y, t) = e^{-3(x-at)^2-3(y-bt)^2}$. Here displayed with a refinement number of 9.

Table 4.2: Table showing the L^2 errors and convergence for the advection equation on a single block with data $a = 2.0$, $b = -1.0$, $u(x, y, t) = e^{-3(x-at)^2-3(y-bt)^2}$, on the grid displayed in Figure 4.7.

Grid points along each boundary	L^2-error	L^2-convergence
9	0.04026	-
17	0.01180	1.77
33	0.00355	1.73
65	0.00115	1.63
129	3.82936e-04	1.58
257	1.31285e-04	1.54

The Advection Equation with Interfaces

We implemented the scheme proposed in Proposition 4.2 on a mesh consisting of six triangles (see Figure 4.8 and Figure 4.9). Due to long run times, the highest refinement number for the grids in these cases is 129. For both cases below, the parameters ω_1 and ω_2 was chosen such that if $(a\boldsymbol{e}_x + b\boldsymbol{e}_y) \cdot \boldsymbol{n} < 0$, then $\omega_1 = 1$ and $\omega_2 = 0$, where $\boldsymbol{n}$ is the outward pointing unit vector of each triangle. This means

that if triangle I and II are neighbours, and $(a\boldsymbol{e}_x + b\boldsymbol{e}_y) \cdot \boldsymbol{n}_I < 0$, then the interface SAT is applied to triangle I, and not to triangle II.

Case 1:

We consider the problem on the physical domain displayed in Figure 4.8 with the problem data $a = 2$, $b = 0.5$, and the analytical solution $u(x, y, t) = e^{-2(x-at)^2-2(y-bt)^2}$. The code is run until $t = 1$. The L^2 errors and convergence rates are listed in Table 4.3.

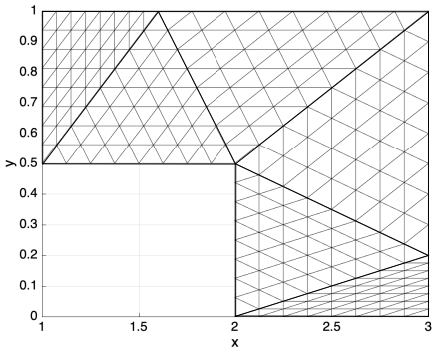


Figure 4.8: The physical domain for the implementation of the advection equation on a grid with multiple blocks, with problem data $a = 2.0$, $b = 0.5$, $u(x, y, t) = e^{-2(x-at)^2-2(y-bt)^2}$. Here displayed with a refinement number of 9.

Table 4.3: Table showing the L^2 errors and convergence for the advection equation with the data $a = 2.0$ and $b = 0.5$, $u(x, y, t) = e^{-2(x-at)^2-2(y-bt)^2}$, on the mesh displayed in Figure 4.8.

Grid points along each boundary	L^2-error	L^2-convergence
9	0.02247	-
17	0.00557	2.01
33	0.00141	1.99
65	3.612e-04	1.96
129	9.499e-05	1.93

Also for this case, the results demonstrates a better convergence rate than predicted

by the analysis. However, we see a similar deterioration as in the cases on a single-block domain.

Case 2:

We now consider the problem on the physical domain shown in Figure 4.9 with the problem data $a = 1.0$, $b = -1.0$, and the analytical solution $u(x, y, t) = e^{-3(x-at)^2-3(y-bt)^2}$. The code is run until $t = 1$. The L^2 errors and convergence rated are listed in Table 4.4.

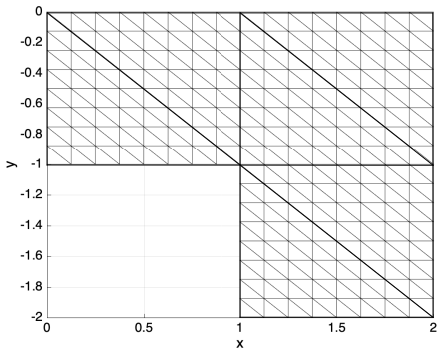


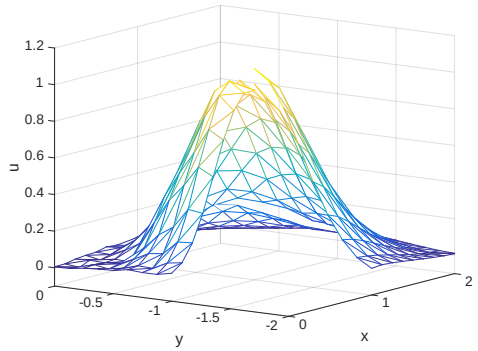
Figure 4.9: The physical domain for the implementation of the advection equation on a grid with multiple blocks, with problem data $a = 1.0$, $b = -1.0$, $u(x, y, t) = e^{-3(x-at)^2-3(y-bt)^2}$. Here displayed with a refinement number of 9.

Table 4.4: Table showing the L^2 errors and convergence for the advection equation with the data $a = 1.0$, $b = -1.0$, $u(x, y, t) = e^{-3(x-at)^2-3(y-bt)^2}$, on the mesh displayed in Figure 4.9.

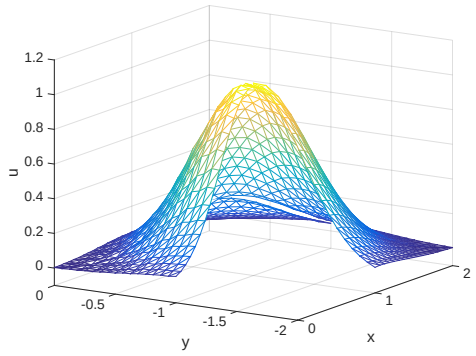
Grid points along each boundary	L^2-error	L^2-convergence
9	0.06175	-
17	0.01547	2.00
33	0.00393	1.98
65	0.00104	1.92
129	2.8709e-04	1.85

We see that the convergence rate in this case is dropping faster than for the previous case. In addition, as is seen in the figures 4.10a-4.10d, there is a jump in the solution

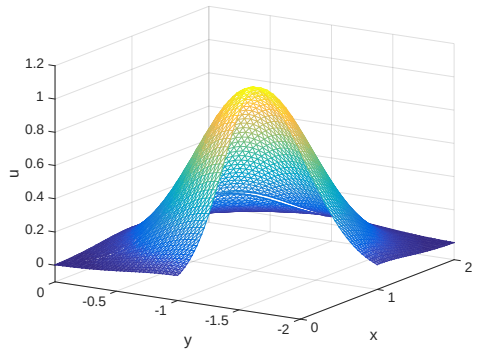
along an edge between two of the triangles in the mesh. The reason for this is that the direction of the wave is parallel to the interface, and therefore, there is no exchange in data here. However, the plots indicate that the solutions from the two triangles that share this interface, converge towards each other. Table 4.5 shows the obtained convergence rates for each triangle to the true solution and also the rate at which they are converging towards each other. The results demonstrates that the numerical solutions along this interface is converging towards the true solution at the rate 1, which is expected since the truncation error along the boundary is of $\mathcal{O}(h)$.



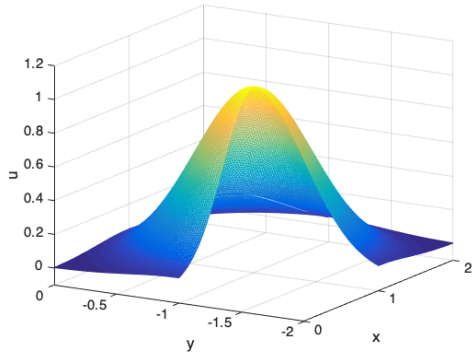
4.10a: Plot of the numerical solution with refinement number 9.



4.10b: Plot of the numerical solution with refinement number 17.



4.10c: Plot of the numerical solution with refinement number 33.



4.10d: Plot of the numerical solution with refinement number 65.

Table 4.5: Table showing the L^2 errors and convergence for the interface in the upper right corner. Subscript L denotes the data from the leftmost triangle, and subscript R the data from the rightmost triangle. u_{ex} denotes the exact solution.

Grid points along each boundary	L^2-convergence $(u_{ex} - u_L)$	L^2-convergence $(u_{ex} - u_R)$	L^2-convergence $(u_R - u_L)$
9	-	-	-
17	0.98	1.07	0.96
33	0.97	1.04	0.99
65	0.98	1.03	1.00
129	0.99	1.01	1.00

Summary

The results presented in this section for the advection equation clearly demonstrates that the convergence rates are higher than what was predicted by the theoretical convergence analysis. However, different tests show different rates, and it is therefore unclear what the actual convergence rate is. For both the single-block and multi-block domains, we ran a test where the direction of the wave is parallel to a boundary or an interface. The obtained convergence rates in these cases are lower than for the cases where the wave is not parallel to any boundary or interface.

4.4 The wave equation

In this section, we analyse the second-order wave equation in two space dimensions with Neumann boundary conditions on a single-block domain. Due to time limits of the project, we do not consider meshes with interfaces.

The analysis for this equation differs from the one of the advection equation. We explained in Section 4.3 that the transformation of the advection equation to the standard triangle results in a problem analogous to the one in the physical domain. This is not the case for the wave equation, and we therefore deal with the transfor-

mation of this equation in a slightly different way. In this section, the coordinates in the physical and computational domain is denoted (x, y) and (ξ, η), respectively.

4.4.1 Analysis for the continuous problem in the physical domain

Since the resulting problem when transforming the wave equation to the standard triangle is not analogous to the one in physical space (we obtain cross derivatives), we first analyse the problem in the physical domain, and afterwards show that the analysis in the computational domain corresponds to the one in the physical domain.

Consider the second-order wave equation in two space dimensions

$$u_{tt} = u_{xx} + u_{yy} = \boldsymbol{\nabla}^2 u, \qquad (x, y) \in \Omega_{\boldsymbol{x}}, \tag{4.13}$$

where $\boldsymbol{\nabla}^2 = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}\right)^2$ is the Laplacian operator and $\Omega_{\boldsymbol{x}}$ is an arbitrary triangle. To obtain an energy estimate for this equation, we proceed as usual by multiplying the equation by u_t and integrating over the domain $\Omega_{\boldsymbol{x}}$.

$$\begin{aligned} \int_{\Omega_{\boldsymbol{x}}} u_t u_{tt}\, dxdy &= \int_{\Omega_{\boldsymbol{x}}} u_t \boldsymbol{\nabla}^2 u\, dxdy, \\ \frac{1}{2}\frac{d}{dt}\left\|u_t\right\|^2_{\Omega_{\boldsymbol{x}}} &= \oint_{\partial\Omega_{\boldsymbol{x}}} u_t \boldsymbol{\nabla} u \cdot \boldsymbol{n}\, ds - \int_{\Omega_{\boldsymbol{x}}} \boldsymbol{\nabla} u_t \cdot \boldsymbol{\nabla} u\, dxdy. \end{aligned}$$

Here, we have applied the integration-by-parts rule on the right-hand side of the equation. Rewriting the last integral yields

$$\begin{aligned} \frac{1}{2}\frac{d}{dt}\left\|u_t\right\|^2_{\Omega_{\boldsymbol{x}}} &= \oint_{\partial\Omega_{\boldsymbol{x}}} u_t \boldsymbol{\nabla} u \cdot \boldsymbol{n}\, ds - \int_{\Omega_{\boldsymbol{x}}} \frac{1}{2}\frac{\partial}{\partial t}(\boldsymbol{\nabla} u)^2\, dxdy, \\ \frac{1}{2}\frac{d}{dt}\left(\left\|u_t\right\|^2_{\Omega_{\boldsymbol{x}}} + \left\|u_x\right\|^2_{\Omega_{\boldsymbol{x}}} + \left\|u_y\right\|^2_{\Omega_{\boldsymbol{x}}}\right) &= \oint_{\partial\Omega} u_t \boldsymbol{\nabla} u \cdot \boldsymbol{n}\, ds. \end{aligned}$$

Now, if we add to the right-hand side the penalty term $-\oint_{\partial\Omega_{\boldsymbol{x}}} u_t(\boldsymbol{\nabla} u \cdot \boldsymbol{n} - g)\, ds = 0$ in accordance with the procedure in [SN04], we obtain

$$\frac{1}{2}\frac{d}{dt}\left(\|u_t\|^2_{\Omega_{\boldsymbol{x}}} + \|u_x\|^2_{\Omega_{\boldsymbol{x}}} + \|u_y\|^2_{\Omega_{\boldsymbol{x}}}\right) = \oint_{\partial\Omega_{\boldsymbol{x}}} u_t g\, ds,$$

which demonstrates well-posedness if we set $g = 0$.

4.4.2 Transformation to the standard triangle

Next, we introduce some general theory that will be applied in the demonstration of well-posedness of the problem. This theory can be found in [NS05].

The transformation from the physical domain to the standard triangle is on the form

$$\xi = \xi(x, y), \qquad \eta = \eta(x, y),$$

and is given by the inverse of the transformation introduced in Section 4.1

$$\begin{aligned} \xi &= \frac{b_3 x - a_3 y - a_1 b_3 + b_1 a_3}{a_2 b_3 - b_2 a_3}, \\ \eta &= \frac{a_2 y - b_2 x + a_1 b_2}{a_2 b_3 - b_2 a_3}. \end{aligned} \tag{4.14}$$

Denote $\boldsymbol{\xi} = (\xi, \eta)$ and $\boldsymbol{x} = (x, y)$. The Jacobian of $\boldsymbol{\xi}$ is defined as

$$\mathbb{J} = \begin{pmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{pmatrix}. \tag{4.15}$$

Let $\boldsymbol{\nabla}_{\boldsymbol{\xi}} = (\frac{\partial}{\partial\xi}, \frac{\partial}{\partial\eta})$, such that $\mathbb{J} = (\boldsymbol{\nabla}_{\boldsymbol{\xi}}\boldsymbol{x})^T$. We have that the identity matrix, $I = (\boldsymbol{\nabla}_{\boldsymbol{\xi}}\boldsymbol{x})^T(\boldsymbol{\nabla}_{\boldsymbol{x}}\boldsymbol{\xi})^T$. This means that the inverse of the Jacobian can be expressed

as

$$\mathbb{J}^{-1} = (\boldsymbol{\nabla}_{\boldsymbol{x}}\boldsymbol{\xi})^T = \begin{pmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{pmatrix}.$$

$\mathbb{J}^{-1}$ can also be found by inverting $\mathbb{J}$. By doing so, we obtain

$$\mathbb{J}^{-1} = \frac{1}{\det(\mathbb{J})} \begin{pmatrix} y_\eta & -x_\eta \\ -y_\xi & x_\xi \end{pmatrix}.$$

The two different expressions must be equal, and so must their derivatives, hence we obtain the following relations

$$\begin{aligned} (J\xi_x)_\xi + (J\eta_x)_\eta &= (y_\eta)_\xi - (y_\xi)_\eta = 0, \\ (J\xi_y)_\xi + (J\eta_y)_\eta &= -(x_\eta)_\xi + (x_\xi)_\eta = 0, \end{aligned} \tag{4.16}$$

where, $J = \det(\mathbb{J})$.

4.4.3 Analysis for the continuous problem in the computational domain

We now turn to the analysis of the transformed problem. We refer the readers to the papers [NS05], [ÅN19] and [NC01] for the theory applied in this section.

Consider again the second-order wave equation in two space dimensions

$$u_{tt} = u_{xx} + u_{yy} = k_x + l_y, \qquad (x, y) \in \Omega_{\boldsymbol{x}} \tag{4.17}$$

Here, $\Omega_{\boldsymbol{x}}$ is the physical domain which is an arbitrary triangle, and $k = u_x$ and $l = u_y$. Since $u(x, y, t) = u(x(\xi, \eta), y(\xi, \eta), t)$, we have that $u_x = u_\xi \xi_x + u_\eta \eta_x$ and $u_y = u_\xi \xi_y + u_\eta \eta_y$. By multiplying Equation (4.17) by J and using these relations, we obtain

$$Ju_{tt} = J(k_x + l_x) = Jk_\xi \xi_x + Jk_\eta \eta_x + Jl_\xi \xi_y + Jl_\eta \eta_y.$$

We now recognize that each of the terms on the right-hand side can be written as $Jk_\xi \xi_x = (J\xi_x k)_\xi - (J\xi_x)_\xi k$, due to the chain rule. Then the above equation can be written

$$\begin{aligned} Ju_{tt} &= (J\xi_x k)_\xi - (J\xi_x)_\xi k + (J\eta_x k)_\eta - (J\eta_x)_\eta k + (J\xi_y l)_\xi - (J\xi_y)_\xi l + (J\eta_y l)_\eta - (J\eta_y)_\eta l, \\ &= (J\xi_x k + J\xi_y l)_\xi + (J\eta_x k + J\eta_y l)_\eta - R_1 - R_2, \end{aligned}$$

where $R_1 = (J\xi_x)_\xi k + (J\xi_y)_\xi l$ and $R_2 = (J\eta_x)_\eta k + (J\eta_y)_\eta l$. By using (4.16), we get $R_1 + R_2 = 0$. Hence, the above equation now reads

$$Ju_{tt} = (J\xi_x k + J\xi_y l)_\xi + (J\eta_x k + J\eta_y l)_\eta = K_\xi + L_\eta, \tag{4.18}$$

where $K = (J\xi_x k + J\xi_y l)$ and $L = (J\eta_x k + J\eta_y l)$.

We now turn to the derivation of the energy estimate. As usual, multiply Equation (4.18) by u_t and integrate over the domain (which is now the standard triangle, as we have transformed the equation).

$$\begin{aligned} \int_{\Omega_{\boldsymbol{\xi}}} u_t Ju_{tt}\, d\xi d\eta &= \int_{\Omega_{\boldsymbol{\xi}}} u_t (K_\xi + L_\eta)\, d\xi d\eta, \\ &= \int_{\Omega_{\boldsymbol{\xi}}} (u_t K)_\xi - u_{t\xi} K + (u_t L)_\eta - u_{t\eta} L\, d\xi d\eta. \end{aligned} \tag{4.19}$$

Here, we have used the chain rule once again to obtain the last equality. We first look at the left-hand side of the above equation. We have the following

$$\begin{aligned}\int_{\Omega_{\boldsymbol{\xi}}} u_t J u_{tt}\, d\xi d\eta &= \int_{\Omega_{\boldsymbol{\xi}}} \frac{1}{2}\frac{\partial}{\partial t}(u_t)^2\, J d\xi d\eta, \\ &= \int_{\Omega_{\boldsymbol{x}}} \frac{1}{2}\frac{\partial}{\partial t}(u_t)^2\, dxdy = \frac{1}{2}\frac{d}{dt}\left\|u_t\right\|^2_{\Omega_{\boldsymbol{x}}}.\end{aligned} \tag{4.20}$$

We now divide the integral on the right-hand side of Equation (4.19) into two parts. Let

$$\begin{aligned}I_1 &= -\int_{\Omega_{\boldsymbol{\xi}}} u_{t\xi}K + u_{t\eta}L\, d\xi d\eta, \\ I_2 &= \int_{\Omega_{\boldsymbol{\xi}}} (u_t K)_\xi + (u_t L)_\eta\, d\xi d\eta.\end{aligned}$$

We consider first the integral I_1. Inserting K and L and rearranging terms yields

$$I_1 = -\int_{\Omega_{\boldsymbol{\xi}}} J\left((u_{t\xi}\xi_x + u_{t\eta}\eta_x)u_x\right) + J\left((u_{t\xi}\xi_y + u_{t\eta}\eta_y)u_y\right)\, d\xi d\eta,$$

where $(u_{t\xi}\xi_x + u_{t\eta}\eta_x) = u_{tx}$ and $(u_{t\xi}\xi_y + u_{t\eta}\eta_y) = u_{ty}$, which means we have

$$\begin{aligned}I_1 &= -\int_{\Omega_{\boldsymbol{\xi}}} J u_{tx}u_x + J u_{ty}u_y\, d\xi d\eta, \\ &= -\int_{\Omega_{\boldsymbol{\xi}}} \frac{1}{2}\frac{\partial}{\partial t}(u_x^2 + u_y^2)\, J d\xi d\eta, \\ &= -\frac{1}{2}\frac{d}{dt}\int_{\Omega_{\boldsymbol{x}}} u_x^2 + u_y^2\, dxdy, \\ &= -\frac{1}{2}\frac{d}{dt}\left(\left\|u_x\right\|^2_{\Omega_{\boldsymbol{x}}} + \left\|u_y\right\|^2_{\Omega_{\boldsymbol{x}}}\right).\end{aligned} \tag{4.21}$$

Next, we turn to the integral I_2. By applying Green's theorem, the integral can be written

$$\begin{aligned} I_2 &= \oint_{\partial\Omega_{\boldsymbol{\xi}}} -u_t L \, d\xi + u_t K \, d\eta, \\ &= \oint_{\partial\Omega_{\boldsymbol{\xi}}} u_t(K, L) \cdot \boldsymbol{n}_{\boldsymbol{\xi}} \, ds_{\boldsymbol{\xi}}, \end{aligned}$$

where $\boldsymbol{n}_{\boldsymbol{\xi}} ds_{\boldsymbol{\xi}} = (d\eta, -d\xi)$, and $\boldsymbol{n}_{\boldsymbol{\xi}}$ is the outward pointing unit normal vector in the transformed space. Since the boundaries of the standard triangle are piecewise linear, we have that $d\xi = \xi_2 - \xi_1$ and $d\eta = \eta_2 - \eta_1$, where (ξ_1, η_1) and (ξ_2, η_2) are two points along the boundary in question. These normal vector components can be expressed in terms of the corresponding x- and y-coordinates because of the inverse transformation (4.14).

$$\begin{aligned} d\xi &= \frac{b_3}{c}(x_2 - x_1) - \frac{a_3}{c}(y_2 - y_1), \\ &= \frac{b_2}{c} dx - \frac{a_3}{c} dy, \\ d\eta &= \frac{a_2}{c}(y_2 - y_1) - \frac{b_2}{c}(x_2 - x_1), \\ &= \frac{a_2}{c} dy - \frac{b_2}{c} dx. \end{aligned}$$

Here we have defined $c = a_2 b_3 - b_2 a_3$ to reduce notation. The constants appearing in the above expressions can be recognized as derivatives of ξ and η. Substitution of these constants gives the following expressions

$$\begin{aligned} d\xi &= \xi_x dx + \xi_y dy, \\ d\eta &= \eta_x dx + \eta_y dy. \end{aligned} \tag{4.22}$$

We now turn back to the integral I_2. After inserting the above expressions, we have

$$I_2 = \oint_{\partial\Omega_{\boldsymbol{\xi}}} Ju_t(\xi_x k + \xi_y l, \eta_x k + \eta_y l) \cdot (\eta_x dx + \eta_y dy, -\xi_x dx - \xi_y dy).$$

Writing the above integrand out, we obtain after some manipulations

$$I_2 = \oint_{\partial\Omega_{\boldsymbol{x}}} Ju_t(\xi_x\eta_y - \xi_y\eta_x)\left((u_x, u_y) \cdot (dy, -dx)\right),$$

where we recognize that $\xi_x\eta_y - \xi_y\eta_x = \det(\mathbb{J}^{-1}) = \frac{1}{\det(\mathbb{J})}$. The resulting integral therefore reads

$$I_2 = \oint_{\partial\Omega_{\boldsymbol{x}}} u_t(u_x, u_y) \cdot \boldsymbol{n}_{\boldsymbol{x}} ds_{\boldsymbol{x}}. \tag{4.23}$$

Combining the three parts (4.20), (4.21) and (4.23), we obtain

$$\frac{1}{2}\frac{d}{dt}\left(\|u_t\|^2_{\Omega_{\boldsymbol{x}}} + \|u_x\|^2_{\Omega_{\boldsymbol{x}}} + \|u_y\|^2_{\Omega_{\boldsymbol{x}}}\right) = \oint_{\partial\Omega_{\boldsymbol{x}}} u_t \boldsymbol{\nabla} u \cdot \boldsymbol{n}_{\boldsymbol{x}} ds_{\boldsymbol{x}},$$

which we know from the analysis in the physical domain, proves well-posedness.

4.4.4 Analysis for the discrete problem

The scheme for the wave equation was derived by mimicking the continuous case, by applying the theory found in for example [ÅN19] or [NC01]. We refer the reader to these articles for additional information about the following concept.

Proposition 4.3. *The approximation*

$$J\boldsymbol{u}_{tt} = JD_\xi \tilde{K} + JD_\eta \tilde{L} + JP^{-1}_{\boldsymbol{\xi}} \mathbb{S}\mathbb{A}\mathbb{T},$$

of the problem (4.13) *with Neumann boundary conditions and*

$$\mathbb{SAT} = \begin{cases} -\Big((\tilde{K},\tilde{L}) - (G_1,G_2)\Big)\cdot \boldsymbol{n}_{\boldsymbol{\xi}}, & \text{for boundary nodes} \\ 0, & \text{otherwise,} \end{cases}$$

and

$$\begin{aligned} \tilde{K} &= \xi_x(\xi_x D_\xi \boldsymbol{u} + \eta_x D_\eta \boldsymbol{u}) + \xi_y(\xi_y D_\xi \boldsymbol{u} + \eta_y D_\eta \boldsymbol{u}), \\ \tilde{L} &= \eta_x(\xi_x D_\xi \boldsymbol{u} + \eta_x D_\eta \boldsymbol{u}) + \eta_y(\xi_y D_\xi \boldsymbol{u} + \eta_y D_\eta \boldsymbol{u}), \\ G_1 &= \xi_x(\xi_x D_\xi \boldsymbol{g} + \eta_x D_\eta \boldsymbol{g}) + \xi_y(\xi_y D_\xi \boldsymbol{g} + \eta_y D_\eta \boldsymbol{g}), \\ G_2 &= \eta_x(\xi_x D_\xi \boldsymbol{g} + \eta_x D_\eta \boldsymbol{g}) + \eta_y(\xi_y D_\xi \boldsymbol{g} + \eta_y D_\eta \boldsymbol{g}), \end{aligned}$$

is stable. Here, $\boldsymbol{g}$ *is the boundary data in the physical domain, and* $D_\xi = P_{\boldsymbol{\xi}}^{-1} Q_\xi$, $D_\eta = P_{\boldsymbol{\xi}}^{-1} Q_\eta$.

Proof. The goal is to derive an energy estimate. Following the usual procedure, multiply the above equation by $\boldsymbol{u}_t^T P_{\boldsymbol{\xi}}$ and add the transpose.

$$\begin{aligned} \boldsymbol{u}_t^T P_{\boldsymbol{\xi}} J \boldsymbol{u}_{tt} + \boldsymbol{u}_{tt}^T P_{\boldsymbol{\xi}} J \boldsymbol{u}_t &= J\boldsymbol{u}_t^T P_{\boldsymbol{\xi}} D_\xi \tilde{K} + J\boldsymbol{u}_t^T P_{\boldsymbol{\xi}} D_\eta \tilde{L} + \tilde{K}^T D_\xi^T P_{\boldsymbol{\xi}} J \boldsymbol{u}_t + \tilde{L} D_\eta^T P_{\boldsymbol{\xi}} J \boldsymbol{u}_t \\ &\quad + 2J\boldsymbol{u}_t^T \mathbb{SAT}, \\ \frac{d}{dt}\left\|\boldsymbol{u}_t\right\|_{\Omega_{\boldsymbol{x}}} &= \boldsymbol{u}_t^T J B_\xi \tilde{K} - \boldsymbol{u}_t^T J Q_\xi^T \tilde{K} + \tilde{K}^T B_\xi J \boldsymbol{u}_t - \tilde{K}^T Q_\xi J \boldsymbol{u}_t + \boldsymbol{u}_t^T J B_\eta \tilde{L} \\ &\quad - \boldsymbol{u}_t^T J Q_\eta^T \tilde{L} + \tilde{L}^T B_\eta J \boldsymbol{u}_t - \tilde{L} Q_\eta J \boldsymbol{u}_t + 2J\boldsymbol{u}_t^T \mathbb{SAT}. \end{aligned}$$

Here, we have defined $P_{\boldsymbol{x}} = JP_{\boldsymbol{\xi}}$. Now, consider first all interior nodes of $\Omega_{\boldsymbol{\xi}}$.

$$-J\left(\boldsymbol{u}_t^T Q_\xi^T \tilde{K} + \tilde{K}^T Q_\xi \boldsymbol{u}_t + \boldsymbol{u}_t^T Q_\eta \tilde{L} + \tilde{L}^T Q_\eta \boldsymbol{u}_t\right).$$

By inserting the specific form of $\tilde{K}$ and $\tilde{L}$, we obtain after some tedious manipulations

$$
\begin{aligned}
&- J\overbrace{(\xi_x(D_\xi \boldsymbol{u})_t^T + \eta_x(D_\eta \boldsymbol{u})_t^T)}^{(D_x\boldsymbol{u})_t^T} P_{\boldsymbol{\xi}} D_x \boldsymbol{u} - J(D_x \boldsymbol{u})^T P_{\boldsymbol{\xi}} \overbrace{(\xi_x(D_\xi \boldsymbol{u})_t^T + \eta_x(D_\eta \boldsymbol{u})_t^T)}^{(D_x\boldsymbol{u})_t} \\
&- J\overbrace{(\xi_y(D_\xi \boldsymbol{u})_t^T + \eta_y(D_\eta \boldsymbol{u})_t^T)}^{(D_y\boldsymbol{u})_t^T} P_{\boldsymbol{\xi}} D_y \boldsymbol{u} - J(D_y \boldsymbol{u})^T P_{\boldsymbol{\xi}} \overbrace{(\xi_y(D_\xi \boldsymbol{u})_t^T + \eta_y(D_\eta \boldsymbol{u})_t^T)}^{(D_y\boldsymbol{u})_t}, \\
&= -J(D_x \boldsymbol{u})_t^T P_{\boldsymbol{\xi}} D_x \boldsymbol{u} - J(D_x \boldsymbol{u})^T P_{\boldsymbol{\xi}} (D_x \boldsymbol{u})_t - J(D_y \boldsymbol{u})_t^T P_{\boldsymbol{\xi}} (D_y \boldsymbol{u}) - J(D_y \boldsymbol{u})^T P_{\boldsymbol{\xi}} (D_y \boldsymbol{u})_t, \\
&= -\frac{d}{dt} \left(\|\boldsymbol{u}_x\|^2_{\Omega_{\boldsymbol{x}}} + \|\boldsymbol{u}_y\|^2_{\Omega_{\boldsymbol{x}}} \right).
\end{aligned}
$$

Next, we turn to the boundary nodes.

$$2J(\boldsymbol{u}_t^T B_\xi \tilde{K} + \boldsymbol{u}_t^T B_\eta \tilde{L} + \boldsymbol{u}_t^T \mathbb{SAT}).$$

We now insert the specific form of $\tilde{K}$ and $\tilde{L}$ and consider only one boundary node b for simplicity. We then have

$$2J(\boldsymbol{u}_t)_b(\xi_x(D_x \boldsymbol{u})_b + \xi_y(D_y \boldsymbol{u})_b, \eta_x(D_x \boldsymbol{u})_b + \eta_y(D_y \boldsymbol{u})_b) \cdot (\Delta\eta, -\Delta\xi) + 2J(\boldsymbol{u}_t)_b \mathbb{SAT}_b.$$

Relations analogous to 4.22 hold for $\Delta\xi$ and $\Delta\eta$. Using these and inserting the SAT term, yields

$$
\begin{aligned}
&2(\boldsymbol{u}_t)_b((D_x \boldsymbol{u})_b, (D_y \boldsymbol{u})_b) \cdot (\Delta y, -\Delta x) - 2(\boldsymbol{u}_t)_b(((D_x \boldsymbol{u})_b, (D_y \boldsymbol{u})_b) \cdot (\Delta y, -\Delta x) - \boldsymbol{g}_b), \\
&= 2(\boldsymbol{u}_t)_b \boldsymbol{g}_b,
\end{aligned}
$$

i.e., the scheme is stable. □

We have proved that the scheme for the problem in the computational space is

stable, and we will now investigate at which rate the scheme is at least expected to converge at.

4.4.5 Convergence analysis

According to [SGN07] the application of the first derivative approximation twice yields a truncation error of $\mathcal{O}(h)$ in the interior of the domain on grids where the first derivative approximation has an error of $\mathcal{O}(h^2)$. However, it was observed by numerical experiments that the truncation error at these points is $\mathcal{O}(h^2)$. In Appendix A, we provide an explanation for this. Corner- and boundary points are not discussed in the article, but it is shown that if the first derivative approximation contains an $\mathcal{O}(h)$ error, then the second derivative approximation will possibly have an error of $\mathcal{O}(1)$. Boundary nodes for the first-derivative approximation have such an error, and therefore, we would expect a truncation error of $\mathcal{O}(1)$ along the boundary and along the second outer "layer" for the second-derivative approximation. Extending this argumentation to corner points that have an error of $\mathcal{O}(1)$ (for the first-derivative approximation), we would expect an error of $\mathcal{O}(1/h)$ in these points for the second derivative. Indeed, this is what is observed in the numerical experiments. See Figure 4.11 for description of the truncation errors for the second derivatives. This figure shows the worst case scenario.

Remark. *In the numerical experiments, it was observed that the truncation error for the second derivative with respect to x along the boundary parallel to the x-axis (in the computational domain) was of $\mathcal{O}(h)$.*

Also for the wave equation, we derive an estimate for the convergence rate. We use the truncation errors displayed in Figure 4.11. In the same fashion as for the advection equation, we let u and v denote the true and numerical solutions, respectively. The error is then defined as $e_i = u(x_i, y_i, t) - v_i(t)$, and will satisfy the scheme

$$J\boldsymbol{e}_{tt} = JD_\xi E_1 + JD_\eta E_2 - P^{-1}J\left((K_1, L_1) - (K_2, L_2)\right) \cdot \boldsymbol{n_\xi} + J\boldsymbol{T},$$

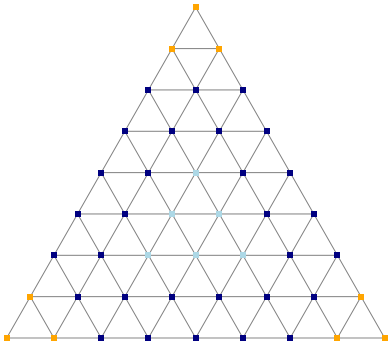


Figure 4.11: Figure showing the pattern of the truncation errors. The orange nodes have $T = \mathcal{O}(1/h)$, the dark blue nodes have $T = \mathcal{O}(1)$, and the light blue nodes have $T = \mathcal{O}(h^2)$.

where $E_1 = K_1 - K_2$, $E_2 = L_1 - L_2$, and $K_{1,2}$, $L_{1,2}$ corresponds to $\tilde{K}$ and $\tilde{L}$ in the original scheme but with the exact $(K_1,\, L_1)$ and numerical solution $(K_2,\, L_2)$. $\boldsymbol{T}$ is a vector containing the truncation errors.

Following the same procedure as usual for deriving an energy estimate and using the results of the earlier discrete analysis for the wave equation, yields

$$\frac{d}{dt}\left(\|\boldsymbol{e}_t\|^2_{\Omega_{\boldsymbol{x}}} + \|\boldsymbol{e}_x\|^2_{\Omega_{\boldsymbol{x}}} + \|\boldsymbol{e}_y\|^2_{\Omega_{\boldsymbol{x}}}\right) \leq 2\,\|\boldsymbol{e}_t\|_{\Omega_{\boldsymbol{x}}}\,\|\boldsymbol{T}\|_{\Omega_{\boldsymbol{x}}}\,.$$

Define now $E^2 = \|\boldsymbol{e}_t\|^2_{\Omega_{\boldsymbol{x}}} + \|\boldsymbol{e}_x\|^2_{\Omega_{\boldsymbol{x}}} + \|\boldsymbol{e}_y\|^2_{\Omega_{\boldsymbol{x}}}$. Then we have

$$\begin{aligned}
&\frac{d}{dt}E^2 = 2E\frac{d}{dt}E \leq 2\,\|\boldsymbol{e}_t\|_{\Omega_{\boldsymbol{x}}}\,\|T\|_{\Omega_{\boldsymbol{x}}} \leq 2\sqrt{\|\boldsymbol{e}_t\|^2_{\Omega_{\boldsymbol{x}}} + \|\boldsymbol{e}_x\|^2_{\Omega_{\boldsymbol{x}}} + \|\boldsymbol{e}_y\|^2_{\Omega_{\boldsymbol{x}}}}\,\|\boldsymbol{T}\|_{\Omega_{\boldsymbol{x}}}\,,\\
&\frac{d}{dt}E \leq \|\boldsymbol{T}\|_{\Omega_{\boldsymbol{x}}}\,.
\end{aligned}$$

Let again N denote the number of nodes along each boundary, such that $\mathcal{O}(N) = \mathcal{O}(1/h)$. Let now $\boldsymbol{\nabla} = (\frac{\partial}{\partial t}, \frac{\partial}{\partial x}, \frac{\partial}{\partial y})$. By making use of the fact that $\|\boldsymbol{e}_t\|^2 + \|\boldsymbol{e}_x\|^2 + \|\boldsymbol{e}_y\|^2 = \|\boldsymbol{\nabla e}\|^2$, we can, due to conservation, apply the Poincaré inequality to obtain the following estimate

$$\begin{aligned}\|\boldsymbol{e}(t)\|_{\Omega_{\boldsymbol{x}}} &\leq \int_0^t \|\boldsymbol{T}\|_{\Omega_{\boldsymbol{x}}}\, dt, \\ &\leq \int_0^t \sqrt{\mathcal{O}(1/h)^2 \cdot Ch^2 \cdot \mathcal{O}(1) + \mathcal{O}(1)^2 \cdot Ch^2 \cdot \mathcal{O}(N) + \mathcal{O}(h^2)^2 \cdot Ch^2 \cdot \mathcal{O}(N^2)}\, dt, \\ &\leq \mathcal{O}(1).\end{aligned}$$

This suggest that as a worst case scenario, the proposed scheme will not converge.

4.4.6 Numerical results

We ran the proposed scheme on the physical domain displayed in Figure 4.12.

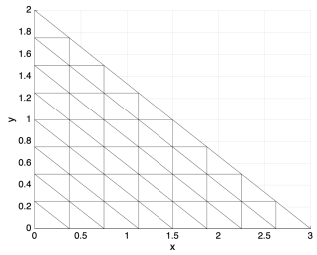


Figure 4.12: Figure showing the triangle used as the physical domain. Here, it is displayed with refinement number 9.

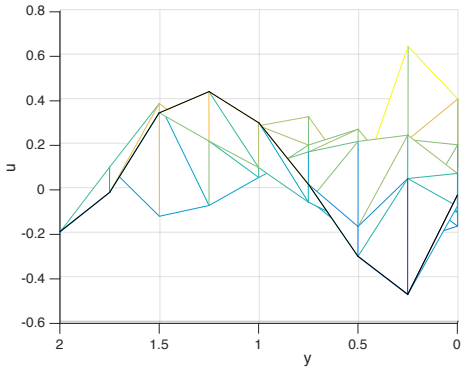
The exact solution is $u(x, y, t) = \sin(\pi x)\cos(\pi y)\cos(\sqrt{2}\pi t)$, which yields no forcing function. The code was run until $t = 0.5$. Table 4.6 shows the obtained L^2 errors and convergence rates.

Table 4.6: Table showing the L^2 errors and convergence using the first derivative approximation twice.

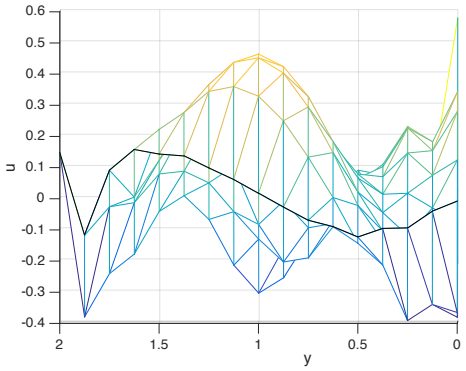
Grid points along each boundary	L^2-error	L^2-convergence
9	0.28798	-
17	0.11476	1.33
33	0.04831	1.25
65	0.02203	1.13
129	0.01052	1.07
257	0.00514	1.03

As is seen from this table, even though there are many inconsistent nodes compared to the total number of nodes (especially for lower refinement numbers), the scheme seems to converge with first order. Hence, we obtain better convergence rates than expected from the convergence analysis.

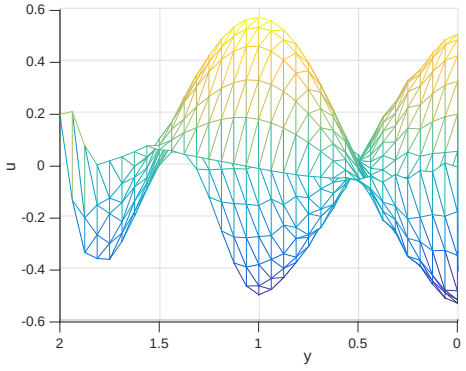
Investigations of the resulting plots indicate that the numerical solution along the boundary $x = 0$ (where the solution is $u = 0$) is converging to the true solution (see the figures 4.13a-4.13g). Figure 4.14 corroborates this indication. We also investigated the boundary $y = 0$, to see if we have the same case here. Figure 4.15 demonstrates that we have convergence along this boundary as well. This indicates that the boundaries are converging, even though they are inconsistent.



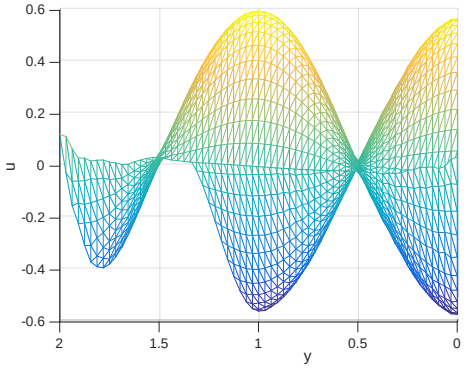
4.13a: Plot of the numerical solution with refinement number 9. The black line represents the boundary along $x = 0$.



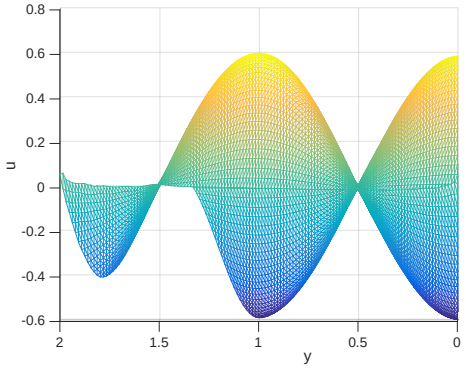
4.13b: Plot of the numerical solution with refinement number 17.The black line represents the boundary along $x = 0$.



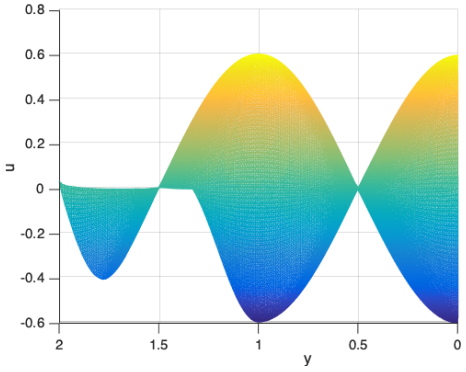
4.13c: Plot of the numerical solution with refinement number 33.



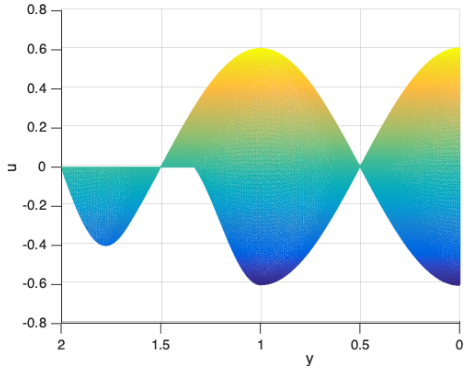
4.13d: Plot of the numerical solution with refinement number 65.



4.13e: Plot of the numerical solution with refinement number 129.



4.13f: Plot of the numerical solution with refinement number 257.



4.13g: Plot of the exact solution with 257 as the number of grid points along each boundary.

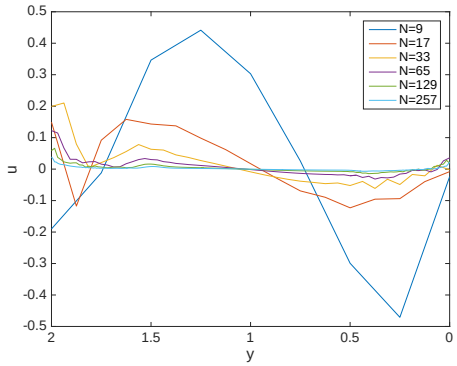


Figure 4.14: Plot of the numerical solutions along the boundary $x = 0$. N denotes the number of grid points along the boundary.

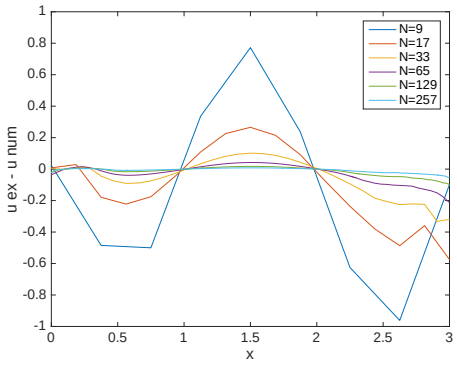


Figure 4.15: Plot of the difference between the exact and numerical solution along the boundary $y = 0$. N denotes the number of grid points along the boundary.

Chapter 5

Conclusions and further work

In this thesis, we have studied the extension of the SBP-SAT technique to the finite volume method. The goal of this project was to introduce a methodology for implementing both first and second derivatives on general unstructured grids, where the higher accuracy of the approximations on structured grids is utilized by introducing a transformation to a computational domain.

The results presented in Chapter 4, demonstrate that the introduction of this transformation indeed raises the accuracy of the approximations. However, none of them are fully consistent, and the case for the second derivative is especially unfavourable. At least for lower refinement numbers, the number of inconsistent points is too high to conclude that it is a good approximation. However, if such an approximation is to be used on an unstructured grid, this methodology can be used to recover some accuracy. For the first derivative approximation, the procedure introduced in this thesis can be utilized to obtain higher convergence rates if the desired mesh is unstructured.

Although full accuracy is not recovered, the results presented in this work clearly demonstrates that the numerical schemes are convergent. We also noticed that the observed convergence rates were higher than what was predicted by the theoretical analysis. However, as discussed in Section 4.3.7 and 4.4.6, it is not clear what convergence rates the schemes are producing, and we are missing the theory for determining sharp estimates for the convergence rates.

Based on the results of this thesis, possible future work could include further numerical experiments to investigate the convergence rates of the schemes. Another desirable matter is the derivation of a consistent second derivative approximation formulated by the finite volume method that satisfy a summation-by-parts rule. Lastly, it would be satisfactory to derive a methodology for transforming curved boundaries, in order to allow for even more general grids.

Bibliography

[ÅN19] O. Ålund and J. Nordström. Encapsulated high order difference operators on curvilinear non-conforming grids. *Journal of Computational Physics*, 385:209–224, 2019.

[Bla07] J. Blazek. *Computational Fluid Dynamics: Principles and Applications*. Elsevier, Amsterdam, 2 edition, 2007.

[CFNF14] M. H. Carpenter, T. C. Fisher, E. J. Nielsen, and S. H. Frankel. Entropy stable spectral collocation schemes for the navier-stokes equations: discontinuous interfaces. *SIAM Journal on Scientific Computing*, 36(5):B835–B867, 2014.

[CGA94] M. H. Carpenter, D. Gottlieb, and S. Abarbanel. Time-stable boundary conditions for finite-difference schemes solving hyperbolic systems: Methodology and application to high-order compact schemes. *Journal of Computational Physics*, 111:220–236, 1994.

[CNG99] M.H. Carpenter, J. Nordström, and D. Gottlieb. A stable and conservative interface treatment of arbitrary spatial accuracy. *Journal of Computational Physics*, 148:341–365, 1999.

[DB16] H.L. Dret and Lucquin B. *Partial Differential Equations: Modeling, Analysis and Numerical Approximation*. International Series of Numerical Mathematics. Birkhäuser, Cham, 2016.

[DRFBZ14] D. C. Del Rey Fernández, P. D. Boom, and D. W. Zingg. A generalized framework for nodal first derivative summation-by-parts operators. *Journal of Computational Physics*, 266:214–239, 2014.

[DRFHZ14] D.C Del Rey Fernández, J.E. Hicken, and D.W. Zingg. Review of summation-by-parts operators with simultaneous approximation terms for the numerical solution of partial differential equations. *Computers & Fluids*, 95:171–196, 2014.

[EAN11] S. Eriksson, Q. Abbas, and J. Nordström. A stable and conservative method for locally adapting the design order of finite difference schemes. *Journal of Computational Physics*, 230:4216–4231, 2011.

[FP99] J.H. Ferziger and M. Perić. *Computational Methods for Fluid Dynamics*. Springer, Berlin, 2 edition, 1999.

[Gas13] G. J. Gassner. A skew-symmetric discontinuous galerkin spectral element discretization and its relation to sbp-sat finite difference methods. *SIAM Journal on Scientific Computing*, 35(3):A1233–A1253, 2013.

[GKO95] B. Gustafsson, H.O. Kreiss, and J. Oliger. *Time dependent problems and difference methods*. Pure and Applied Mathematics. John Wiley & Sons, New York, 1995.

[Gus75] B. Gustafsson. The convergence rate for difference approximations to mixed initial boundary value problems. *Mathematics of Computation*, 29(130):396–406, 1975.

[Gus81] B. Gustafsson. The convergence rate for difference approximations to general mixed initial boundary value problems. *SIAM Journal on Numerical Analysis*, 18(2):179–190, 1981.

[Gus08] B. Gustafsson. *High Order Difference Methods for Time Dependent PDE*. Springer Series in Computational Mathematics. Springer, Berlin, Heidelberg, 2008.

[KCDT16] P.K. Kundu, I.M. Cohen, D.R. Dowling, and with contributions by G. Tryggvason. Computational fluid dynamics. In *Fluid Mechanics*, chapter 6, pages 228–291. Elsevier/AP, Amsterdam, 6 edition, 2016.

[KL89] H.O. Kreiss and J. Lorenz. *Initial-Boundary Value Problems and the Navier-Stokes Equations*, volume 136 of *Pure and applied mathematics*. Academic Press, Boston, 1989.

[KS74] H.O. Kreiss and G. Scherer. Finite element and finite difference methods for hyperbolic partial differential equations. 1974. In: Mathematical Aspects of Finite Elements in Partial Differential Equations.

[KS77] H.O. Kreiss and G. Scherer. On the existence of energy estimates for difference approximations for hyperbolic systems. Technical report, Department of Scientific Computing, Uppsala University, 1977.

[KW93] H.O. Kreiss and L. Wu. On the stability definition of difference approximations for the initial boundary value problem. *Applied Numerical Mathematics*, 12:213–227, 1993.

[Lev02] R. Leveque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge texts in applied mathematics. Cambridge University Press, Cambridge, United Kingdom, 2002.

[LN14] T. Lundquist and J. Nordström. The sbp-sat technique for initial value problems. *Journal of Computational Physics*, 270:86–104, 2014.

[LR56] P.D. Lax and R.D. Richtmyer. Survey of the stability of linear finite difference equations. *Communications on Pure and Applied Mathematics*, IX:267–293, 1956.

[MHI08] K. Mattson, F. Ham, and G. Iaccarino. Stable and accurate wave-propagation in discontinuous media. *Journal of Computational Physics*, 227:8753–8767, 2008. doi: 10.1016/j.jcp.2008.06.023.

[NB01] J. Nordström and M. Björck. Finite volume approximations and strict stability for hyperbolic problems. *Applied Numerical Mathematics*, 38:237–255, 2001.

[NC01] J. Nordström and M. H. Carpenter. High-order finite difference methods, multidimensional linear problems, and curvilinear coordinates. *Journal of Computational Physics*, 173:149–174, 2001. doi: 10.1006/jcph.2001.6864.

[NFAE03] J. Nordström, K. Forsberg, C. Adamsson, and P. Eliasson. Finite volume methods, unstructured meshes and strict stability for hyperbolic problems. *Applied Numerical Mathematics*, 45:453–473, 2003.

[NS05] J. Nordström and M. Svärd. Well-posed boundary conditions for the navier-stokes equations. *SIAM Journal on Numerical Analysis*, 43(3):1231–1255, 2005.

[SGN07] M. Svärd, J. Gong, and J. Nordström. An accuracy evaluation of unstructured node-centred finite volume methods. *Applied Numerical Mathematics*, 58:1142–1158, 2007. doi: 10.1016/j.apnum.2007.05.002.

[SN04] M. Svärd and J. Nordström. Stability of finite volume approximations for the laplacian operator on quadrilateral and triangular grids. *Applied Numerical Mathematics 51*, 51:101–125, 2004.

[SN06] M. Svärd and J. Nordström. On the order of accuracy for difference approximations of initial-boundary value problems. *Journal of Computational Physics*, 218:333–352, 2006.

[SN14] M. Svärd and J. Norström. Review of summation-by-parts schemes for initial-boundary-value problems. *Journal of Computational Physics*, 268:17–38, 2014.

[SN17] M. Svärd and J. Nordström. On the convergence rates of energy-stable finite-difference schemes. October 2017. On review in Journal of Computational Physics.

[Str94] B. Strand. Summation by parts for finite difference approximations for d/dx. *Journal of Computational Physics*, 110:47–67, 1994.

[WK17] S. Wang and G. Kreiss. Convergence of summation-by-parts finite difference methods for the wave equation. *Journal of Scientific Computing*, 71:219–245, 2017. doi: 10.1007/s10915-016-0297-3.

Appendices

Appendix A

Truncation errors for the first derivative

A.1 Interior nodes

We begin by considering interior nodes of the domain (see Figure A.1).

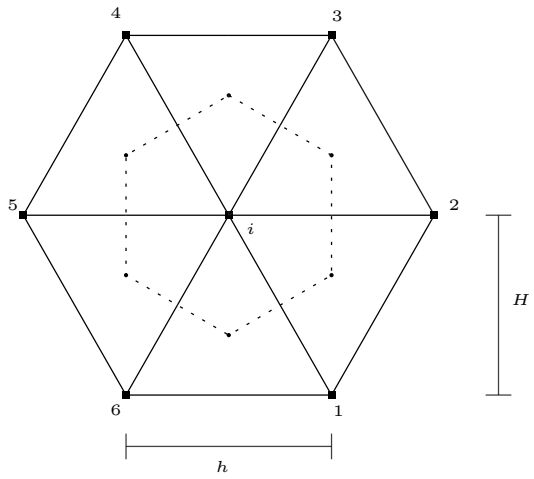


Figure A.1: Figure showing an interior node i of the domain and its neighbours. H is the height of a triangle, and h is the step size in space.

We denote the difference between the y-coordinates of the centroids in the triangles consisting of nodes $(6, 1, i)$ and $(1, 2, i)$ as Δy_1. The rest of the edges of the dual

volume is denoted in the same manner.

First, we recognize that the height of a triangle is given by $H = \frac{\sqrt{3}}{2}h$, where h distance between two grid points.

Second, the Δys are as listed below.

$$\Delta y_1 = \frac{\sqrt{3}}{6}h, \qquad \Delta y_2 = \frac{\sqrt{3}}{3}h, \qquad \Delta y_3 = \frac{\sqrt{3}}{6}h,$$

$$\Delta y_4 = -\frac{\sqrt{3}}{6}h, \qquad \Delta y_5 = -\frac{\sqrt{3}}{3}h, \qquad \Delta y_6 = -\frac{\sqrt{3}}{6}h,$$

The goal is to find the order of the truncation error for an interior node. Recall that the approximation of the x-derivative at node i is given by

$$\frac{1}{V_i}\sum_{n\in N_i}\frac{u_i+u_n}{2}\Delta y_{in},$$

where N_i is the set of all neighbouring nodes to node i. Writing out the sum in the approximation, yields

$$\left(\frac{u_i+u_1}{2}\Delta y_1 + \frac{u_i+u_2}{2}\Delta y_2 + \frac{u_i+u_3}{2}\Delta y_3 + \frac{u_i+u_4}{2}\Delta y_4 + \frac{u_i+u_6}{2}\Delta y_5 + \frac{u_i+u_6}{2}\Delta y_6\right). \tag{A.1}$$

From Figure A.1, we have that

$$
\begin{aligned}
u_1 &= u(x + \frac{1}{2}h, y - H),\\
u_2 &= u(x + h, y),\\
u_3 &= u(x + \frac{1}{2}h, y + H),\\
u_4 &= u(x - \frac{1}{2}h, y + H),\\
u_5 &= u(x - h, y),\\
u_6 &= u(x - \frac{1}{2}h, y - H).
\end{aligned}
$$

The Taylor expansions are given by

$$
\begin{aligned}
u(x + \frac{1}{2}h, y - H) &= u + u_x\left(\frac{1}{2}h\right) - u_y H + \frac{1}{2!}\left(u_{xx}\left(\frac{1}{2}h\right)^2 - 2u_{xy}\left(\frac{1}{2}h\right)H + u_{yy}H^2\right)\\
&\quad + \frac{1}{3!}\left(u_x^{(3)}\left(\frac{1}{2}h\right)^3 - 3u_{xxy}\left(\frac{1}{2}h\right)^2 H + 3_{yyx}\left(\frac{1}{2}h\right)H^2 + u_y^{(3)}H^3\right)\\
&\quad + \mathcal{O}(h^4),\\
u(x + h, y) &= u + u_x h + \frac{1}{2!}u_{xx}h^2 + \frac{1}{3}u_x^{(3)}h^3\\
&\quad + \mathcal{O}(h^4),\\
u(x + \frac{1}{2}h, y + H) &= u + u_x\left(\frac{1}{2}h\right) + u_y H + \frac{1}{2!}\left(u_{xx}\left(\frac{1}{2}h\right)^2 + 2u_{xy}\left(\frac{1}{2}h\right)H + u_{yy}H^2\right)\\
&\quad + \frac{1}{3!}\left(u_x^{(3)}\left(\frac{1}{2}h\right)^3 + 3u_{xxy}\left(\frac{1}{2}h\right)^2 H + 3u_{yyx}\left(\frac{1}{2}h\right)H^2 + u_y^{(3)}\right)\\
&\quad + \mathcal{O}(h^4),
\end{aligned}
$$

$$
\begin{aligned}
u(x-\frac{1}{2}h, y+H) &= u - u_x\left(\frac{1}{2}h\right) + u_y H + \frac{1}{2!}\left(u_{xx}\left(\frac{1}{2}h\right)^2 - 2u_{xy}\left(\frac{1}{2}h\right)H + u_{yy}H^2\right) \\
&\quad + \frac{1}{3!}\left(-u_x^{(3)}\left(\frac{1}{2}h\right)^3 + 3u_{xxy}\left(\frac{1}{2}h\right)^2 H - 3u_{yyx}\left(\frac{1}{2}h\right)H^2 + u_y^{(3)}H^3\right), \\
&\quad + \mathcal{O}(h^4) \\
u(x-h, y) &= u - u_x h + \frac{1}{2!}u_{xx}h^2 - \frac{1}{3!}u_x^{(3)}h^3 + \mathcal{O}(h^4), \\
u(x-\frac{1}{2}h, y-H) &= u - u_x\left(\frac{1}{2}h\right) - u_y H + \frac{1}{2!}\left(u_{xx}\left(\frac{1}{2}h\right)^2 + 2u_{xy}\left(\frac{1}{2}h\right)H + u_{yy}H^2\right) \\
&\quad + \frac{1}{3!}\left(-u_x^{(3)}\left(\frac{1}{2}h\right)^2 - 3u_{xxy}\left(\frac{1}{2}h\right)^2 H - 3u_{yyx}\left(\frac{1}{2}h\right)H^2 - u_y^{(3)}H^3\right) \\
&\quad + \mathcal{O}(h^4).
\end{aligned}
$$

By using the above expansions and the relations between the Δys, we obtain after some tedious calculations

$$
\begin{aligned}
&\frac{1}{2}\Delta y_1(2u_x h + \frac{1}{2\cdot 3!}u_x^{(3)}h^3 + \frac{6}{3!}u_{yyx}hH^2) + \frac{1}{2}\Delta y_2(2u_x h + \frac{2}{3!}u_x^{(3)}h^3), \\
&= \frac{\sqrt{3}}{2}u_x h^2 + \frac{3\sqrt{3}}{8\cdot 3!}u_x^{(3)}h^4 + \frac{3\sqrt{3}}{8\cdot 3!}u_{yyx}h^4 + \mathcal{O}(h^5).
\end{aligned}
$$

Division by $V_i = \frac{\sqrt{3}}{2}h^2$, gives the final result

$$
\frac{1}{V_i}\sum_{n\in N_i}\frac{u_i+u_n}{2}\Delta y_{in} = u_x + \frac{3}{4\cdot 3!}u_x^{(3)}h^2 + \frac{3}{4\cdot 3!}u_{yyx}h^2 + \mathcal{O}(h^3).
$$

This means that the truncation errors at interior points for the x-derivative are of $\mathcal{O}(h^2)$.

A.2 Boundary nodes

We now consider all boundary nodes except those located at the corners of the domain. We have three different edges in our domain, and we will therefore consider them separately. We name the edges by $(1, 2, 3)$, going in an counter-clockwise direction, where edge 1 is the horizontal one.

Recall that the approximation of the x-derivative at a boundary node b is given by

$$\frac{1}{V_b}\sum_{n\in B}\frac{u_b+u_n}{n}\Delta y_{bn}+u_b\Delta y_b,$$

where B is the set of all neighbouring nodes to node b. The dual volume for such a point is $V_i=\frac{\sqrt{3}}{4}h^2$.

Boundary nodes along boundary 1

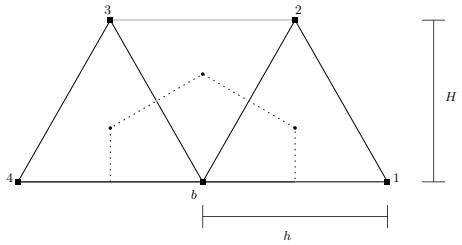


Figure A.2: Figure of edge number 1 with a boundary node b and its neighbouring nodes.

Consider now a node along boundary 1. In this case, we have

$$\Delta y_b=0,\quad \Delta y_1=\frac{\sqrt{3}}{6}h,\quad \Delta y_2=\frac{\sqrt{3}}{6}h,\quad \Delta y_3=-\frac{\sqrt{3}}{6}h,\quad \Delta y_4=-\frac{\sqrt{3}}{6}h,$$

and

$$
\begin{aligned}
u_1 &= u(x+h, y), \\
u_2 &= u(x+\frac{1}{2}h, y+H), \\
u_3 &= u(x-\frac{1}{2}h, y+H), \\
u_4 &= u(x-h, y).
\end{aligned}
$$

By inserting the Taylor expansions, we obtain

$$
\begin{aligned}
\frac{1}{V_b}\sum_{n\in B}\frac{u_b+u_n}{n}\Delta y_{bn} + u_b\Delta y_b &= \frac{1}{V_b}\left(\frac{1}{2}\Delta y_1\left(3u_x h + \frac{\sqrt{3}}{2!}u_{xy}h^3 + \frac{9}{4\cdot 3!}u_x^{(3)}h^3 + \mathcal{O}(h^2)\right)\right), \\
&= u_x + \frac{\sqrt{3}}{6}u_{xy}h + \frac{9}{4\cdot 3!}u_x^{(3)}h^2 + \mathcal{O}(h^2).
\end{aligned}
$$

Which means that we have a truncation error of $\mathcal{O}(h)$ along boundary 1.

Boundary nodes along edge 2

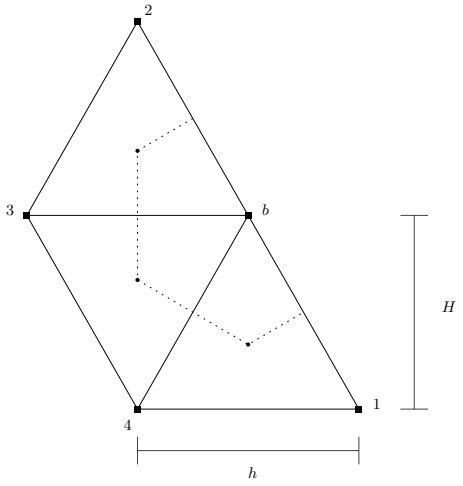


Figure A.3: Figure of edge number 2 with a boundary node b and its neighbouring nodes.

Consider now a node along boundary 2. In this case, we have

$$\Delta y_b = \frac{\sqrt{3}}{2}h, \quad \Delta y_1 = \frac{\sqrt{3}}{12}h, \quad \Delta y_2 = -\frac{\sqrt{3}}{12}h, \quad \Delta y_3 = -\frac{\sqrt{3}}{3}h, \quad \Delta y_4 = -\frac{\sqrt{3}}{6}h,$$

and

$$\begin{aligned}
u_1 &= u(x + \frac{1}{2}h, y - H), \\
u_2 &= u(x - \frac{1}{2}h, y + H), \\
u_3 &= u(x - h, y), \\
u_4 &= u(x - \frac{1}{2}h, y - H).
\end{aligned}$$

Inserting the Taylor expansions, we obtain after some manipulations

$$\begin{aligned}
\frac{1}{V_b}\sum_{n\in B}\frac{u_b + u_n}{n}\Delta y_{bn} + u_b\Delta y_b &= \frac{1}{V_b}\left(\frac{1}{2}\Delta y_1\left(6u_x h - \frac{9}{4}u_{xx}h^2 - \frac{\sqrt{3}}{2}u_{xy}h^2 - \frac{3}{4}u_{yy}h^2 + \mathcal{O}(h^3)\right)\right), \\
&= u_x - \frac{3}{8}u_{xx}h - \frac{1}{4\sqrt{3}}u_{xy}h - \frac{1}{8}u_{yy}h + \mathcal{O}(h^2).
\end{aligned}$$

Hence, we have a truncation error of $\mathcal{O}(h)$ in the nodes along boundary 2.

Boundary nodes along edge 3

Consider now a node along boundary 3. In this case, we have

$$\Delta y_b = -\frac{\sqrt{3}}{2}h, \quad \Delta y_1 = -\frac{\sqrt{3}}{12}h, \quad \Delta y_2 = \frac{\sqrt{3}}{6}h, \quad \Delta y_3 = \frac{\sqrt{3}}{3}h, \quad \Delta y_4 = \frac{\sqrt{3}}{12}h,$$

and

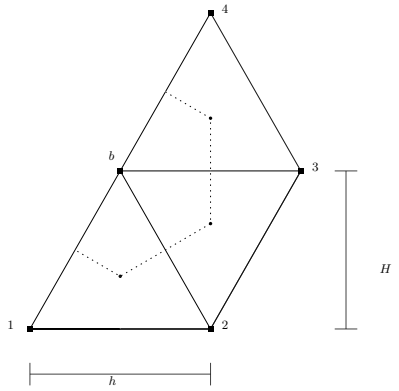


Figure A.4: Figure of edge number 3 with a boundary node b and its neighbouring nodes.

$$
\begin{aligned}
u_1 &= u(x - \frac{1}{2}h, y - H),\\
u_2 &= u(x + \frac{1}{2}h, y - H),\\
u_3 &= u(x + h, y),\\
u_4 &= u(x + \frac{1}{2}h, y + H).
\end{aligned}
$$

Inserting the above information into the approximation yields

$$
\begin{aligned}
\frac{1}{V_b}\sum_{n\in B}\frac{u_b + u_n}{n}\Delta y_{bn} + u_b\Delta y_b &= \frac{1}{V_b}\left(\frac{1}{2}\Delta y_1\left(-6u_x h - \frac{9}{4}u_{xx}h^2 + \frac{\sqrt{3}}{2}u_{xy}h^2 - \frac{3}{4}u_{yy}h^2 + \mathcal{O}(h^3)\right)\right),\\
&= u_x + \frac{3}{8}u_{xx}h - \frac{1}{4\sqrt{3}}u_{xy}h + \frac{1}{8}u_{yy}h + \mathcal{O}(h^2).
\end{aligned}
$$

That is, we have an error of $\mathcal{O}(h)$ in the points along boundary 3.

A.3 Corner nodes

Lastly, we consider the three corner nodes that appears in the domain. We denote these corner 1, corner 2 and corner 3, starting from the leftmost corner and

continuing in a counter-clockwise direction.

The approximation of the x-derivative at a corner node is given by

$$\frac{1}{V_c}\sum_{n\in C}\frac{u_c+u_n}{n}\Delta y_{cn}+u_c\Delta y_{c1}+u_c\Delta y_{c2},$$

where C is the set of all neighbouring nodes to c, and Δy_{c1} and Δy_{c2} are the two Δys along the boundaries. Here, the dual volume is $V_c=\frac{\sqrt{3}}{12}h^2$.

Corner 1

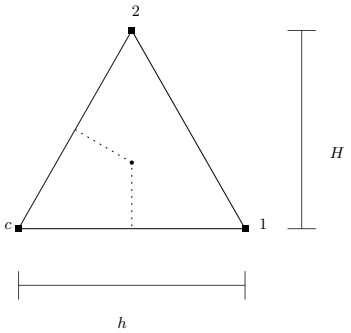


Figure A.5: Figure of corner node 1 and its neighbouring nodes.

Consider now corner 1. In this case, we have

$$\Delta y_{c1}=0,\qquad \Delta y_{c2}=-\frac{\sqrt{3}}{4}h,\qquad \Delta y_1=\frac{\sqrt{3}}{6}h,\qquad \Delta y_2=\frac{\sqrt{3}}{12}h,$$

and

$$u_1=u(x+h,y),$$
$$u_2=u(x+\frac{1}{2}h,y+H).$$

Inserting the Taylor expansions, yields

$$\frac{1}{V_c}\sum_{n\in C}\frac{u_c+u_n}{n}\Delta y_{cn}+u_c\Delta y_{c1}+u_c\Delta y_{c2}=$$
$$\frac{1}{V_c}\left(\frac{1}{2}\Delta y_1\left(3u+\frac{5}{4}u_xh+\frac{1}{2}u_yH+\mathcal{O}(h^2)\right)+u\Delta y_{c2}\right)=\frac{5}{4}u_x+\frac{\sqrt{3}}{4}u_y+\mathcal{O}(h).$$

This means that we have an error of $\mathcal{O}(1)$ in corner 1.

Corner 2

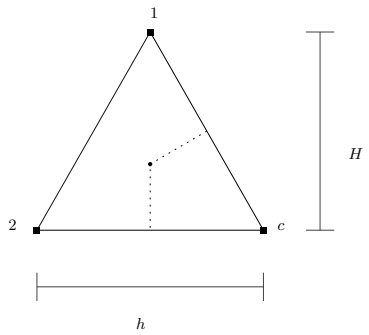


Figure A.6: Figure of corner node 2 and its neighbouring nodes.

Consider now corner 2. In this case, we have

$$\Delta y_{c1}=\frac{\sqrt{3}}{4}h,\qquad \Delta y_{c2}=0,\qquad \Delta y_1=-\frac{\sqrt{3}}{12}h,\qquad \Delta y_2=-\frac{\sqrt{3}}{6}h,$$

and

$$u_1=u(x-\frac{1}{2}h,y+H),$$
$$u_2=u(x-h,y).$$

Inserting the Taylor expansions, yields

$$\frac{1}{V_c}\sum_{n\in C}\frac{u_c+u_n}{n}\Delta y_{cn}+u_c\Delta y_{c1}+u_c\Delta y_{c2}=$$
$$\frac{1}{V_c}\left(\frac{1}{2}\Delta y_1\left(6u-\frac{5}{2}u_xh+u_yH+\mathcal{O}(h^2)\right)+u\Delta y_{c2}\right)=\frac{5}{4}u_x-\frac{\sqrt{3}}{4}u_y+\mathcal{O}(h).$$

Hence, we also have an error of $\mathcal{O}(1)$ in corner 2.

Corner 3

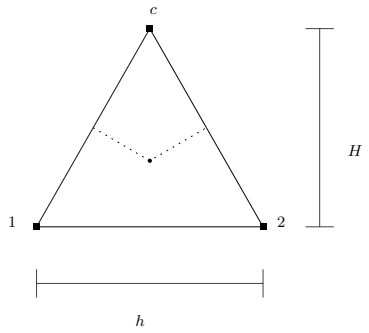


Figure A.7: Figure of corner node 3 and its neighbouring nodes.

Lastly, consider corner 3. In this case, we have

$$\Delta y_{c1}=-\frac{\sqrt{3}}{4}h,\qquad \Delta y_{c2}=\frac{\sqrt{3}}{4}h,\qquad \Delta y_1=-\frac{\sqrt{3}}{12}h,\qquad \Delta y_2=\frac{\sqrt{3}}{12}h,$$

and

$$u_1=u(x-\frac{1}{2}h,y-H),$$
$$u_2=u(x+\frac{1}{2}h,y-H).$$

Inserting the Taylor expansions, yields

$$\frac{1}{V_c}\sum_{n\in C}\frac{u_c+u_n}{n}\Delta y_{cn}+u_c\Delta y_{c1}+u_c\Delta y_{c2}=$$
$$\frac{1}{V_c}\left(\frac{1}{2}\Delta y_1\left(-u_x h+\frac{\sqrt{3}}{2}u_{xy}h^2+\mathcal{O}(h^4)\right)\right)=\frac{1}{2}u_x+\frac{\sqrt{3}}{4}u_{xy}h+\mathcal{O}(h^2).$$

Which shows that we have an error of $\mathcal{O}(1)$ also for corner 3.

A similar derivation could also be carried out for the y-derivative to obtain similar results.

A.4 The second derivative approximation

In [SGN07], it was shown that applying the first derivative approximation used in this thesis, twice, yields an error of $\mathcal{O}(h)$ for the approximation of the second derivative (in the interior). However, it was observed when implementing the approximation, that the error of the approximation for the second derivative was actually of $\mathcal{O}(h^2)$ in the interior. We now try to give an explanation for this.

From the above sections, we know that the error of an interior point is $\frac{1}{3!}u_x^{(3)}h^2$. Applying the first derivative approximation twice, yields

$$(u_{xx})_i=\frac{1}{V_i}\sum_{n\in N_i}\frac{(u_x)_i+\frac{3}{4\cdot 3!}(u_x^{(3)})_i h^2+\frac{3}{4\cdot 3!}(u_{yyx})_i h^2+\mathcal{O}(h^3)}{2}\Delta y_{in}$$
$$+\frac{(u_x)_n+\frac{3}{4\cdot 3!}(u_x^{(3)})_n h^2+\frac{3}{4\cdot 3!}(u_{yyx})_n h^2+\mathcal{O}(h^3)}{2}\Delta y_{in}.$$

Since $(u_x)_i^{(3)}=(u_x)_n^{(3)}$ and $(u_{yyx})_i=(u_{yyx})_n$ for polynomials of degree three or less, the error in each node will be identical. This means that these errors will cancel each other because of the relations between the Δys. We know that the application of the first derivative approximation in the interior yields an error of $\mathcal{O}(h^2)$, hence, the error for the second derivative in the interior is also of $\mathcal{O}(h^2)$.

The above argumentation does not necessarily hold if the error at some neighbouring nodes are of lower degrees. This explains why also the second outer "layer" of nodes have an error of $\mathcal{O}(1)$ for the second derivative.

Appendix B

SBP finite difference operator

The SBP finite difference operator used in Chapter 3 is a 6th order operator with diagonal norm matrix P. The following matrices that constitutes this operator can be found in for example [Gus08].

$$P = h \begin{bmatrix} \frac{13649}{43200} & & & & & & & & & \\ & \frac{12013}{8640} & & & & & & & & \\ & & \frac{2711}{4320} & & & & & & & \\ & & & \frac{5359}{4320} & & & & & & \\ & & & & \frac{7877}{8640} & & & & & \\ & & & & & \frac{43801}{43200} & & & & \\ & & & & & & 1 & & & \\ & & & & & & & \ddots & & \\ & & & & & & & & & \end{bmatrix}.$$

$$S = \frac{1}{h}\begin{bmatrix} -\frac{25}{12} & 4 & -3 & \frac{4}{3} & -\frac{1}{4} & & & \\ & 0 & & & & & & \\ & & 0 & & & & & \\ & & & \ddots & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & \frac{1}{4} & -\frac{4}{3} & 3 & -4 & \frac{25}{12} \end{bmatrix},$$

$$D_2 = \frac{1}{h^2}\begin{bmatrix} \frac{114170}{40947} & -\frac{438107}{54596} & \frac{336409}{40947} & -\frac{276997}{81894} & \frac{3747}{13649} & \frac{21035}{163788} & & & & & & & \\ \frac{6173}{5860} & -\frac{2066}{879} & \frac{3283}{1758} & -\frac{303}{293} & \frac{2111}{3516} & -\frac{601}{4395} & & & & & & & \\ -\frac{52391}{81330} & \frac{134603}{32532} & -\frac{21982}{2711} & \frac{112915}{16266} & -\frac{46969}{16266} & \frac{30409}{54220} & & & & & & & \\ \frac{68603}{321540} & -\frac{12423}{10718} & \frac{112915}{32154} & -\frac{75934}{16077} & \frac{53369}{21436} & -\frac{54899}{160770} & \frac{48}{5359} & & & & & & \\ -\frac{7053}{39385} & \frac{86551}{94524} & -\frac{46969}{23631} & \frac{53369}{15754} & -\frac{87904}{23631} & \frac{820271}{472620} & -\frac{1296}{7877} & \frac{96}{7877} & & & & & \\ \frac{21035}{525612} & -\frac{24641}{131403} & \frac{30409}{87602} & -\frac{54899}{131403} & \frac{820271}{525612} & -\frac{117600}{43801} & \frac{64800}{43801} & -\frac{6480}{43801} & \frac{480}{43801} & & & & \\ & & & \frac{1}{90} & -\frac{3}{20} & \frac{3}{2} & -\frac{49}{18} & \frac{3}{2} & -\frac{3}{20} & \frac{1}{90} & & \\ & & & & \frac{1}{90} & -\frac{3}{20} & \frac{3}{2} & -\frac{49}{18} & \frac{3}{2} & -\frac{3}{20} & \frac{1}{90} & \\ & & & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ & & & & & & & & & & & \end{bmatrix}.$$