Numerical methods for hyperbolic systems

Eric Sonnendrücker

Max-Planck-Institut for Plasma Physics

und

Department of Mathematics, TU Munich

Lecture notes

Summer semester 2020

July 4, 2020
## Contents

1 Introduction 3

2 1D linear advection 7
   2.1 Finite Difference schemes for the advection equation .......................... 7
       2.1.1 Obtaining a Finite Difference scheme ........................................ 7
       2.1.2 The first order explicit upwind scheme ...................................... 7
       2.1.3 The first order upwind implicit scheme ....................................... 8
       2.1.4 The method of lines ................................................................. 9
       2.1.5 Convergence of linear finite difference schemes ............................ 9
       2.1.6 von Neumann stability analysis .................................................. 11
       2.1.7 High-order time schemes ........................................................... 16
       2.1.8 The Cauchy-Kowalevski procedure ............................................. 20
       2.1.9 Modified equation analysis ....................................................... 22
   2.2 The Finite Volume method ............................................................... 27
       2.2.1 The first order Finite Volume schemes ......................................... 27
       2.2.2 Higher order schemes ............................................................... 28
   2.3 The Finite Element method ............................................................... 30
   2.4 The Discontinuous Galerkin (DG) method ............................................ 34

3 Linear hyperbolic systems 38
   3.1 The Riemann problem ....................................................................... 38
   3.2 The discontinuous Galerkin method for linear systems .......................... 40
       3.2.1 General formulation ....................................................................... 40
       3.2.2 Derivation of DG scheme for Maxwell’s equations ......................... 41
   3.3 The Discontinuous Galerkin method for linear systems in 2D ................ 42
       3.3.1 Derivation of the discrete equations ............................................. 43
       3.3.2 Matrix formulation of the discrete problem ................................... 45
       3.3.3 Computation of the fluxes ............................................................. 50
       3.3.4 Semi-discrete energy conservation .............................................. 52
       3.3.5 A leap-frog time scheme ............................................................. 53

4 Non linear conservation laws 54
   4.1 Characteristics .................................................................................... 54
   4.2 Weak solutions .................................................................................... 55
       4.2.1 Definition ....................................................................................... 55
       4.2.2 The Rankine-Hugoniot condition .................................................. 56
       4.2.3 Entropy solution ............................................................................ 58
   4.3 The Riemann problem ........................................................................... 60
   4.4 Numerical methods ............................................................................... 61
       4.4.1 The Godunov method .................................................................... 61
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.4.2</td>
<td>Approximate Riemann solvers</td>
<td>62</td>
</tr>
<tr>
<td>4.4.3</td>
<td>Higher order methods</td>
<td>63</td>
</tr>
<tr>
<td>4.4.4</td>
<td>Strong stability preserving (SSP) Runge-Kutta methods</td>
<td>63</td>
</tr>
<tr>
<td>4.5</td>
<td>Nonlinear systems of conservation laws</td>
<td>64</td>
</tr>
<tr>
<td>4.5.1</td>
<td>The Rusanov flux</td>
<td>64</td>
</tr>
<tr>
<td>4.5.2</td>
<td>The Roe flux</td>
<td>64</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

Hyperbolic systems arise naturally from the conservation laws of physics, when dissipation effects are not included. Writing down the conservation of mass, momentum and energy yields a system of equations that needs to be solved in order to describe the evolution of the system. In contrast to parabolic equations which have a smoothing effect, hyperbolic equations transport discontinuities and nonlinear hyperbolic equations can even generate discontinuities, called shocks, from smooth initial data. Note that the absence of diffusion is an idealized problem. In real physics there is always some dissipation, but in some cases it is very small so that the hyperbolic system is the correct model. However, in cases where hyperbolic systems have multiple weak solutions, the correct solution is obtained from the limit when the dissipation goes to zero.

In this lecture we will introduce the classical methods for numerically solving such systems. Because local conservation is essential to capture correctly the shocks in numerical methods, Finite Volume discretizations that are based on the cell average of the unknowns rather than the point values and where the interaction of neighboring cells is implemented through the definition of numerical flux between cell are more natural. Finite Volume methods, unless relying on data from neighboring grid cells, are of order one. This is why a new class of very efficient and flexible method has emerged as a generalization of Finite Volume Methods the Discontinuous Galerkin method, which is based on the numerical flux coming from Finite Volumes and the piecewise polynomial description on a cell coming from Finite Elements.

Another generalization of the Finite Volume Method can be useful for some equations, when the important conservation properties are not the local conservation of cell averages, but of edge or face averages. This leads to a class of less known methods, which are called generalized Finite Volumes.

Even though local conservation properties are essential for capturing shocks, often other conservation properties of the initial system of equations also play an important role and should be kept at the discrete level.

Examples of scalar hyperbolic equations Classical time dependent partial differential fall into two classes:

- parabolic equations, which have second order space derivatives, the standard example of which is the heat equation

  \[
  \frac{\partial u}{\partial t} + \Delta u = 0, \quad (1.1)
  \]

  which has to be supplemented with boundary conditions for example \( u = g \) on the boundary of the considered domain.

- hyperbolic equations, which have first order time derivatives and for which a typical simple
example is the linear advection equation

\[ \frac{\partial u}{\partial t} + \mathbf{a} \cdot \nabla u = 0 \]  

(1.2)

This only needs incoming boundary conditions, \textit{i.e.} \(u = g\) if \(\mathbf{a} \cdot \mathbf{\nu} < 0\), where \(\mathbf{\nu}\) is the outward normal. If \(\nabla \cdot \mathbf{a} = 0\), the advection equation can also be written in conservative form

\[ \frac{\partial u}{\partial t} + \nabla \cdot (\mathbf{a} u) = 0. \]  

(1.3)

- A classical nonlinear hyperbolic conservation law is the Burgers equation

\[ \frac{\partial u}{\partial t} + \frac{1}{2} \frac{\partial u^2}{\partial x} = 0, \]  

(1.4)

which can also be written

\[ \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = 0. \]  

(1.5)

- The Vlasov equation is an advection equation in phase space:

\[ \frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla_x f + \frac{q}{m} (\mathbf{E} + \mathbf{v} \times \mathbf{B}) \cdot \nabla_v f = 0. \]  

(1.6)

**Linear hyperbolic systems**

- Maxwell’s equations

\[ \frac{\partial \mathbf{E}}{\partial t} + \nabla \times \mathbf{B} = \mathbf{J}, \]  

(1.7)

\[ \frac{\partial \mathbf{B}}{\partial t} + \nabla \times \mathbf{E} = 0, \]  

(1.8)

\[ \nabla \cdot \mathbf{E} = \rho, \]  

(1.9)

\[ \nabla \cdot \mathbf{B} = 0, \]  

(1.10)

where the charge density \(\rho\) and the current density \(\mathbf{J}\) are the sources that are related by the continuity equation \(\frac{\partial \rho}{\partial t} + \nabla \cdot \mathbf{J} = 0\), and all the constants are put to one.

The hyperbolic system consists of the first two equations (Ampère’s and Faraday’s laws). The two last equations are constraints, which are satisfied automatically at continuous level. There enforcement by a numerical scheme is an important issue.

- The linearized acoustics equations

\[ \frac{\partial p}{\partial t} + \nabla \cdot \mathbf{u} = 0, \]  

(1.11)

\[ \frac{\partial \mathbf{u}}{\partial t} + \nabla p = 0, \]  

(1.12)

(1.13)

where \(p\) is the pressure and \(\mathbf{u}\) the velocity and all constants are put to one.
Non linear systems of conservation laws

- The compressible Euler equations in conservative form read
  \[ \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \]
  \[ \frac{\partial (\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) + \nabla p = 0, \]
  \[ \frac{\partial E}{\partial t} + \nabla \cdot (\mathbf{u}(E + p)) = 0, \]
  where \( E \) is the total energy per unit volume. For an ideal gas it is related to the pressure by
  \( E = \rho \mathbf{u}^2 + \frac{p}{\gamma - 1}. \)

- The compressible ideal MHD equations read
  \[ \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \]
  \[ \rho \left( \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) + \nabla p - (\nabla \times \mathbf{B}) \times \mathbf{B} = 0, \]
  \[ \frac{\partial \mathbf{B}}{\partial t} - \nabla \times (\mathbf{u} \times \mathbf{B}) = 0, \]
  \[ \frac{\partial p}{\partial t} + \nabla \cdot (p \mathbf{u}) + (\gamma - 1)p \nabla \cdot \mathbf{u} = 0 \]

General form of 1D systems of conservation laws  In 1D the systems of conservation laws we consider have the form
  \[ \frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{u})}{\partial x} = 0, \]
  where \( \mathbf{u} \) is a vector of unknown values. This can be written also
  \[ \frac{\partial \mathbf{u}}{\partial t} + A(\mathbf{u}) \frac{\partial \mathbf{u}}{\partial x} = 0, \]
  where \( A(\mathbf{u}) \) is the Jacobian matrix with components \( \left( \frac{\partial F_i}{\partial u_j} \right)_{i,j} \).

Definition 1 (Hyperbolic system).

- The system (1.21) is called hyperbolic if for all \( \mathbf{u} \) the matrix \( A \) has only real eigenvalues and is diagonalizable.

- It is called strictly hyperbolic if all eigenvalues are distinct.

Examples:

- 1D Maxwell’s equation
  \[ \frac{\partial E}{\partial t} + \frac{\partial B}{\partial x} = J \]
  \[ \frac{\partial B}{\partial t} + \frac{\partial E}{\partial x} = 0 \]

- 1D Euler equations
  \[ \frac{\partial}{\partial t} \begin{pmatrix} \rho \\ \rho \mathbf{u} \\ E \end{pmatrix} + \frac{\partial}{\partial x} \begin{pmatrix} \rho \mathbf{u} \\ \rho \mathbf{u}^2 + p \\ E \mathbf{u} + p \mathbf{u} \end{pmatrix} = 0, \]
where $\rho$, $u$ and $E$ are the density, velocity and energy density of the gas and $p$ is the pressure which is a known function of $\rho$.

The idea behind all numerical methods for hyperbolic systems is to use the fact that the system is locally diagonalizable and thus can be reduced to a set of scalar equations. For this reason, before going to systems it will be useful to first understand the scalar case and then see how it can be extended to systems by local diagonalization.

The first part of the lecture will be devoted to the linear case, starting with the scalar case which boils down to linear advection for which the core methods will be first introduced. This is fairly straightforward.

Many additional problems arise in the nonlinear case. Indeed in this case even starting from a smooth initial condition discontinuities can appear in the solution. In this case the concept of weak solutions need to be introduced and there can be several solutions only one of which is physical. We thus need a criterion to find the physical solution and numerical schemes that capture the physical solution. The notion of conservativity plays an essential role there. These will be addressed in the second part.
Chapter 2

1D linear advection

2.1 Finite Difference schemes for the advection equation

We consider first the linear 1D advection equation

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0 \quad \text{for} \ x \in [0, L], \ t \geq 0. \quad (2.1)$$

Let us assume for simplicity that the boundary conditions are periodic. This means that $u$ and all its derivatives are periodic of period $L$. We have in particular $u(0) = u(L)$. The constant $a$ is given. As the problem is time dependent we also need an initial condition $u(x, 0) = u_0(x)$. For this problem, the exact solution is given by

$$u(t, x) = u_0(x - at).$$

2.1.1 Obtaining a Finite Difference scheme

We first consider a uniform mesh of the 1D computational domain, i.e. of the interval $[0, L]$ where we want to compute the solution, see Figure 2.1. The cell size or space step is defined by $\Delta x = \frac{L}{N}$.

![Uniform mesh of [0, L]](image)

Figure 2.1: Uniform mesh of [0, L]

where $N$ is the number of cells in the mesh. The coordinates of the grid points are then defined by $x_j = j \Delta x$. We then need a time step $\Delta t$ and we will compute approximations of the solution at discrete times $t_n = n \Delta t$, $n \in \mathbb{N}$. As we assume the solution to be periodic of period $L$ it will be defined by its values at $x_j$ for $0 \leq i \leq N - 1$ and we shall have $u(t_n, x_N) = u(t_n, x_0)$.

We shall denote by $u^n_j = u(t_n, x_j)$.

2.1.2 The first order explicit upwind scheme

A Finite Difference scheme is classically obtained by approximating the derivatives appearing in the partial differential equation by a Taylor expansion up to some given order which will give the order of the scheme. As we know only the values of the unknown function at the grid points, we
use Taylor expansion at different grid points $x_j$ and linearly combine them so as to eliminate all 
derivatives up to the needed order. For example, considering only space derivatives

$$u(x_{j+1}) = u(x_j) + \Delta x \frac{\partial u}{\partial x} (x_j) + \frac{\Delta x^2 \partial^2 u}{2} (x_j) + O(\Delta x^3), \quad (2.2)$$

$$u(x_{j-1}) = u(x_j) - \Delta x \frac{\partial u}{\partial x} (x_j) + \frac{\Delta x^2 \partial^2 u}{2} (x_j) + O(\Delta x^3). \quad (2.3)$$

The same can be done for the time discretization. For an approximation of order 1 in space and 
time, we can simply write

$$\frac{\partial u}{\partial t}(t_n,x_j) = \frac{u(t_{n+1},x_j) - u(t_n,x_j)}{\Delta t} + O(\Delta t), \quad (2.4)$$

$$\frac{\partial u}{\partial x}(t_n,x_j) = \frac{u(t_n,x_j) - u(t_n,x_{j-1})}{\Delta x} + O(\Delta x). \quad (2.5)$$

Denoting by $u^n_j$ the approximation of the solution at point $x_j$ and time $t_n$ and using the above 
formulas for the approximation of the partial derivatives we get the following approximation (2.1) 
at point $x_j$ and time $t_n$:

$$\frac{u^{n+1}_j - u^n_j}{\Delta t} + a \frac{u^n_j - u^n_{j-1}}{\Delta x} = 0. \quad (2.6)$$

We thus obtain the following explicit formula which enables to compute $u^{n+1}_j$ in function of the 
values of $u$ at time $t_n$ and points $x_j$ and $x_{j-1}$:

$$u^{n+1}_j = u^n_j - a \frac{\Delta t}{\Delta x} (u^n_j - u^n_{j-1}). \quad (2.7)$$

Denoting by $U^n$ the vector of $\mathbb{R}^N$ whose components are $u^n_1, \ldots, u^n_{N-1}$ and

$$A = \begin{pmatrix}
(1 - a \frac{\Delta t}{\Delta x}) & 0 & a \frac{\Delta t}{\Delta x} \\
0 & \ddots & \ddots \\
0 & \ddots & 0 & \ddots & 0 \\
0 & \ddots & \ddots & 0 & \ddots & 0 \\
0 & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\
(1 - a \frac{\Delta t}{\Delta x}) & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0
\end{pmatrix}. $$

The terms at the end of the first line comes from the periodic boundary conditions. We use that 
$u^{n-1}_1 = u^n_{N-1}$ and $u^n_N = u^n_0$. Except on the two diagonals all the terms vanish.

In this case the scheme (2.7) can be written in matrix form

$$U^{n+1} = AU^n.$$ 

2.1.3 The first order upwind implicit scheme

When using an uncentered difference scheme in the other direction for the time derivative, we get

$$\frac{\partial u}{\partial t}(x_j,t_{n+1}) = \frac{u(x_j,t_{n+1}) - u(x_j,t_n)}{\Delta t} + O(\Delta t), \quad (2.8)$$

We use the same finite difference approximation for the space derivative. We then get the following 
formula

$$u^{n+1}_j + a \frac{\Delta t}{\Delta x} (u^{n+1}_j - u^{n+1}_{j-1}) = u^n_j. \quad (2.9)$$
In this case the $u_{j}^{n+1}$ are defined implicitly from the $u_{j}^{n}$ as solutions of a linear system. This is why this scheme is called implicit.

Denote by $B$ the matrix of the linear system:

$$
B = \begin{pmatrix}
(1 + \frac{a\Delta t}{\Delta x}) & 0 & -\frac{a\Delta t}{\Delta x} \\
-\frac{a\Delta t}{\Delta x} & \ddots & \ddots \\
\ddots & \ddots & 0 \\
0 & \ddots & (1 + \frac{a\Delta t}{\Delta x})
\end{pmatrix}.
$$

The term at the end of the first line comes from the periodic boundary conditions. We use that $u_{n-1}^{n} = u_{N-1}^{n}$ and $u_{N}^{n} = u_{0}^{n}$. The terms not on the two diagonals vanish.

Going now from time step $n$ to $n+1$ the implicit scheme in matrix form becomes

$$
BU^{n+1} = U^{n}.
$$

2.1.4 The method of lines

As we saw, the time discretization can be performed by finite differences as for the space discretization. However it is generally more convenient to separate the space and time discretization for a better understanding. The method of lines consists in applying only a discretization scheme in space first (this can be Finite Differences or any other scheme). Then one obtains a system of ordinary differential equations of the form

$$
\frac{dU}{dt} = \mathcal{L}(U),
$$

where $U(t)$ is the vector whose components are $u_{i}(t)$ the unknown values at the grid point at any time $t$. Then one can use any Ordinary Differential Equation (ODE) solver for the time discretization. For example using an explicit Euler method with the upwind method in space yields the previous explicit upwind scheme and when we use an implicit Euler method we get the implicit upwind scheme.

2.1.5 Convergence of linear finite difference schemes

Let us consider Finite Difference schemes in the following generic matrix form

$$
U^{n+1} = AU^{n},
$$

(2.10)

where $A$ is a matrix which does not depend on $U$.

Let us denote by

$$
\bar{U}(t_n) = \begin{pmatrix}
u(t_n, x_0) \\
\vdots \\
u(t_n, x_{N-1})
\end{pmatrix},
$$

where $u$ is the exact solution of the Partial Differential Equation (PDE) that is approximated by the scheme (2.10).

Definition 2. A scheme of the form (2.10) is called consistent of order $(q, p)$ if

$$
\bar{U}(t_{n+1}) - A\bar{U}(t_n) = \Delta t O(\Delta t^q + \Delta x^p).
$$
Definition 3. A scheme of the form (2.10) is called stable for some given norm $\|\|$ if there exist constants $\kappa$ and $\tau$ independent of $\Delta t$ such that its solution $U^n$ at time $t_n = n\Delta t \leq T$ verifies

$$\|U^n\| \leq \kappa\|U^0\| \quad \forall \Delta t \text{ such that } 0 < \Delta t < \tau.$$ 

Theorem 1 (Lax). The scheme (2.10) is convergent if it is stable and consistent.

Proof. Let $\tilde{U}(t_n)$ be the vector whose components are the exact solution at the grid points at time $t_n$. Then, as the scheme is consistent, we have

$$\tilde{U}(t_{n+1}) = A\tilde{U}(t_n) + \Delta t e_n, \quad \text{with } e_n = O(\Delta t^q + \Delta x^p), \quad \text{i.e. } \|e_n\| \leq \bar{\kappa}(\Delta t^q + \Delta x^p),$$

for some constant $\bar{\kappa}$. Denote by $E^n = \tilde{U}(t_n) - U^n$ the vector containing the errors at each grid point at time $t_n$, then as $U^{n+1} = AU^n$, we have $E^{n+1} = AE^n + \Delta t e_n$. Hence

$$E^{n+1} = AE^n + \Delta t e_n,$$

Then for $E^n = \tilde{U}(t_n) - U^n$, we have $E^{n+1} = AE^n + \Delta t e_n$. Hence

$$\|E^{n+1}\| \leq \|A\|E^n\| + \Delta t(\|A\|e_0\| + \|A\|^{-1}e_1\| + \cdots + \|e_n\|).$$

The stability condition implies that $\|A\|e_n\| \leq \kappa\|e_n\|$ and the consistency condition implies that $\|e_m\| \leq \bar{\kappa}(\Delta t^q + \Delta x^p)$ for all $m \leq n$. Plugging this into (2.11), we obtain:

$$\|E^{n+1}\| \leq \kappa\|E^n\| + \kappa\bar{\kappa}\Delta t(\Delta t^q + \Delta x^p).$$

Then, as on the one hand $E^0$ taking as an initial condition in the scheme $U^0 = \tilde{U}(0)$, and on the other hand $\Delta t \leq T$ the maximal considered time, we have

$$\|E^{n+1}\| \leq \kappa\bar{\kappa}T(\Delta t^q + \Delta x^p),$$

which proves the convergence of the scheme. \qed

Application to our first order schemes

Proposition 1. If the exact solution is of class $C^3$, the upwind explicit scheme (2.7) and the upwind implicit scheme (2.9) are consistent of order 1 in $t$ and in $x$.

Proof. Let’s start with (2.7). Using the formulas (2.4) and (2.5), we obtain for the $i$th line of $\tilde{U}(t_{n+1}) - A\tilde{U}(t_n)$,

$$(\tilde{U}(t_{n+1}) - A\tilde{U}(t_n))_i = u(x_i, t_{n+1}) - u(x_i, t_n) + \frac{a\Delta t}{\Delta x}(u(x_i, t_n) - u(x_{i-1}, t_n))$$

$$= \Delta t \left( \frac{\partial u}{\partial t}(x_i, t_n) + O(\Delta t) + a\frac{\partial u}{\partial x}(x_i, t_n) + O(\Delta x) \right).$$

The result follows as $u$ is a solution of our equation. We thus get the (1,1) consistency of this scheme.

Then for (2.9), we use (2.8) and (2.5). The $i$th line of $B\tilde{U}(t_n) - \tilde{U}(t_{n-1})$,

$$(B\tilde{U}(t_n) - \tilde{U}(t_{n-1}))_i = u(x_i, t_n) - u(x_i, t_{n-1}) + \frac{a\Delta t}{\Delta x}(u(x_i, t_n) - u(x_{i-1}, t_n))$$

$$= \Delta t \left( \frac{\partial u}{\partial t}(x_i, t_n) + O(\Delta t) + a\frac{\partial u}{\partial x}(x_i, t_n) + O(\Delta x) \right).$$

The result follows as $u$ is a solution of the equation. We thus get the (1,1) consistency of this scheme. \qed
Proposition 2. Assuming \( a > 0 \), the upwind scheme (2.7) is stable for the \( L^\infty \) norm provided

\[
\frac{a \Delta t}{\Delta x} \leq 1.
\]

This condition is called the **Courant-Friedrichs-Lewy** or **CFL condition**.

**Proof.** Consider the scheme (2.7). Grouping the terms in \( u_{j-1}^n \) and \( u_j^n \), this scheme (2.7) becomes

\[
u_{j}^{n+1} = \frac{a \Delta t}{\Delta x} u_{j-1}^n + (1 - \frac{a \Delta t}{\Delta x}) u_j^n.
\]

As, \( a, \Delta t \) and \( \Delta x \) are positive, if \( \frac{a \Delta t}{\Delta x} \leq 1 \) the two factors of \( u_{j-1}^n, u_j^n \) are positive and equal to their absolute value. Hence

\[
|u_{j}^{n+1}| \leq \frac{a \Delta t}{\Delta x}|u_{j-1}^n| + (1 - \frac{a \Delta t}{\Delta x})|u_j^n| \leq \left( \frac{a \Delta t}{\Delta x} + (1 - \frac{a \Delta t}{\Delta x}) \right) \max_j |u_j^n|,
\]

and so

\[
\max_j |u_{j}^{n+1}| \leq \max_j |u_j^n|,
\]

from which it follows that \( \max_j |u_j^n| \leq \max_j |u_j^0| \) for all \( n \), which yields the \( L^\infty \) stability. \( \square \)

### 2.1.6 von Neumann stability analysis

Let us start with an example. Consider the implicit centered Crank-Nicolson scheme. First the space discretization with the centered scheme yields the system of differential equations:

\[
\frac{d u_j}{dt} = -a \frac{u_{j+1} - u_{j-1}}{2 \Delta x}.
\]

Then integrating between \( t_n \) and \( t_{n+1} \) and taking the trapezoidal quadrature rule yields

\[
u_{j}^{n+1} = u_j^n - \frac{a \Delta t}{4 \Delta x} (u_{j+1}^n - u_{j-1}^n + u_{j+1}^{n+1} - u_{j-1}^{n+1}).
\]

Writing on the left the variables at \( t_{n+1} \) and on the right the variables at \( t_n \) yields

\[
u_{j}^{n+1} + \frac{a \Delta t}{4 \Delta x} (u_{j+1}^{n+1} - u_{j-1}^{n+1}) = u_j^n - \frac{a \Delta t}{4 \Delta x} (u_{j+1}^n - u_{j-1}^n).
\] (2.12)

The idea of the von Neumann stability analysis is to evaluate the action of the scheme on the grid Fourier modes. For \( k \in \mathbb{N} \) the continuous \( k^{th} \) Fourier mode on a L-periodic domain is \( e^{2i \pi k x_L} \). Its values at the grid points \( x_j = j \Delta x = jL/N \) of our L-periodic grid with \( N \) cells are \( e^{2i \pi k j / N} \). So we assume that \( u_j^n = \hat{u}_k^n e^{2i \pi k j / N} \) and \( u_{j}^{n+1} = \hat{u}_k^{n+1} e^{2i \pi k j / N} \). Plugging these expressions into (2.12) yields

\[
e^{\frac{2i \pi k j}{2 \Delta x}} \left(1 + \frac{a \Delta t}{4 \Delta x} (e^{\frac{2i \pi k}{N}} - e^{\frac{-2i \pi k}{N}})\right) \hat{u}_k^{n+1} = e^{\frac{2i \pi k j}{2 \Delta x}} \left(1 - \frac{a \Delta t}{4 \Delta x} (e^{\frac{2i \pi k}{N}} - e^{\frac{-2i \pi k}{N}})\right) \hat{u}_k^n.
\] (2.13)

It follows that

\[
\hat{u}_k^{n+1} = \frac{1 + i \frac{a \Delta t}{2 \Delta x} \sin \frac{2 \pi k}{N}}{1 - i \frac{a \Delta t}{2 \Delta x} \sin \frac{2 \pi k}{N}} \hat{u}_k^n,
\]

so that \( |\hat{u}_k^{n+1}| = |\hat{u}_k^n| \). The Fourier modes are neither amplified nor attenuated by this scheme. It is therefore stable.
Let us now consider an arbitrary linear scheme for updating $u^n_j$ the approximation at $t_n$ to $u^{n+1}_j$ at $t_{n+1}$ of the form

$$\sum_{l=-l_1}^{l_2} \alpha_l u^{n+1}_{j+l} = \sum_{m=-m_1}^{m_2} \beta_m u^n_{j+m},$$  \hspace{1cm} (2.14)

with $(\alpha_l)_{-l_1 \leq l \leq l_2}$ and $(\beta_m)_{-m_1 \leq m \leq m_2}$ given constants. We also consider that the sequence $(u_j)_j$ is periodic of period $N$ the number of grid cells, which means that $u_{j+qN} = u_j$ for any $q \in \mathbb{Z}$.

So if we take a grid function $U^n$ such that $u^n_j = \hat{u}^n_k e^{2\pi jk/N}$, then

$$\sum_{m=-m_1}^{m_2} \beta_m u^n_{j+m} = \sum_{m=-m_1}^{m_2} \beta_m \hat{u}^n_k e^{2\pi m(j+m)} = \hat{u}^n_k e^{2\pi jk} \sum_{m=-m_1}^{m_2} \beta_m e^{2\pi km}. \hspace{1cm} (2.15)$$

In the same way the left-hand-side of (2.14) becomes

$$\sum_{l=-l_1}^{l_2} \alpha_l u^{n+1}_{j+l} = \sum_{l=-l_1}^{l_2} \alpha_l \hat{u}^{n+1}_k e^{2\pi l(j+l)} = \hat{u}^{n+1}_k e^{2\pi jk} \sum_{l=-l_1}^{l_2} \alpha_l e^{2\pi kl}. \hspace{1cm} (2.16)$$

So if we denote by

$$\lambda_{\alpha,k} = \sum_{l=-l_1}^{l_2} \alpha_l e^{2\pi kl}, \quad \text{and} \quad \lambda_{\beta,k} = \sum_{m=-m_1}^{m_2} \beta_m e^{2\pi km}, \hspace{1cm} (2.17)$$

we see that our scheme (2.14) becomes, assuming that $\lambda_{\alpha,k} \neq 0$

$$\hat{u}^{n+1}_k = \frac{\lambda_{\beta,k}}{\lambda_{\alpha,k}} \hat{u}^n_k. \hspace{1cm} (2.18)$$

This means that the amplification factor of the scheme for the $k^{th}$ Fourier mode is $|\frac{\lambda_{\beta,k}}{\lambda_{\alpha,k}}|$, which means that the scheme is stable if this term directly related to the coefficients of the scheme is less than one.

In fact, the von Neumann analysis consists in applying the discrete Fourier transform to the discretized equation. To make this more precise we first recall the definition and main properties of the discrete Fourier transform.

Let us denote by

$$E_k = \begin{pmatrix} 1 \\ \vdots \\ e^{2\pi ijk/N} \\ \vdots \\ e^{2\pi i(N-1)jk/N} \end{pmatrix} \hspace{1cm} (2.19)$$

the values at the grid points of the $k^{th}$ Fourier mode $x \mapsto e^{2\pi kx/L}$.

Let $P$ be the symmetric matrix whose columns are the $\frac{1}{\sqrt{N}} E_k$. They are normalized by $\frac{1}{\sqrt{N}}$ to get a similar form for the inverse. The coefficients of $P$ are $P_{jk} = \frac{1}{\sqrt{N}} e^{2\pi ijk/N}$. We denote by $\bar{P}$ the complex conjugate of $P$ and by $P^* = \bar{P}^\top$ its conjugate transpose.

**Lemma 1.** The matrix $P$ is unitary and symmetric, i.e. $P^{-1} = P^* = \bar{P}$.
Proof. We clearly have $P^T = P$, so $P^* = \hat{P}$. There remains to prove that $P\hat{P} = I_N$. We compute

$$(P\hat{P})_{jk} = \frac{1}{N} \sum_{l=0}^{N-1} e^{\frac{2\pi i l j}{N}} e^{-\frac{2\pi i l k}{N}} = \frac{1}{N} \sum_{l=0}^{N-1} e^{\frac{2\pi i (j-k) l}{N}}.$$ 

Then if $j = k$ it follows that $(P\hat{P})_{jk} = 1$ and if $j \neq k$ we have that

$$\sum_{l=0}^{N-1} \left( e^{\frac{2\pi i (j-k) l}{N}} \right)^l = \frac{1 - e^{\frac{2\pi i N(j-k)}{N}}}{1 - e^{\frac{2\pi i (j-k)}{N}}} = 0,$$

so that indeed $P\hat{P} = I_N$. \qed

Definition 4. Discrete Fourier Transform.

- The **discrete Fourier transform** of a vector $U \in \mathbb{C}^N$ is the vector $V = P^*U$.
- The **inverse discrete Fourier transform** of a vector $V \in \mathbb{C}^N$ is the vector $U = P^{*-1}V = PV$.

Corollary 1. Let $F,G \in \mathbb{C}^n$ and denote by $\hat{F} = P^*F$ and $\hat{G} = P^*G$, their discrete Fourier transforms. Then we have

- the discrete Parseval identity:
  $$(F,G) = F^\top \bar{G} = \hat{F}^\top \bar{\hat{G}} = (\hat{F},\hat{G}), \quad (2.20)$$

- The discrete Plancherel identity:
  $$\|F\| = \|\hat{F}\|, \quad (2.21)$$

where $(.,.)$ and $\|\|$ denote the usual euclidean dot product and norm in $\mathbb{C}^n$.

Proof. The dot product in $\mathbb{C}^N$ of $F = (f_1, \ldots, f_N)^\top$ and $G = (g_1, \ldots, g_N)^\top$ is defined by

$$(F,G) = \sum_{i=1}^{N} f_i \bar{g}_i = F^\top G.$$ 

Then using the definition of the inverse discrete Fourier transform, we have $F = P\hat{F}$, $G = P\hat{G}$, we get

$$F^\top \bar{G} = (P\hat{F})^\top \bar{P} \hat{G} = \hat{F}^\top P^\top \bar{P} \hat{G} = \hat{F}^\top \hat{G},$$

as $P^\top = P$ and $\bar{P} = P^{-1}$. The Plancherel identity follows from the Parseval identity by taking $G = F$. \qed

Remark 1. The discrete Fourier transform is defined as a matrix-vector multiplication. Its computation hence requires a priori $N^2$ multiplications and additions. But because of the specific structure of the matrix there exists a very fast algorithm, called Fast Fourier Transform (FFT) for performing it in $O(N \log_2 N)$ operations. This makes it particularly interesting for many applications, and many fast PDE solvers make use of it.
Circulant matrices

Let us now consider linear Finite Difference schemes on a periodic uniform grid. They can be expressed as in (2.14), which can be written in matrix form

\[ AU^{N+1} = BU^N, \]  

(2.22)

where each line of \( A \) consists of the coefficients \( \alpha_l \) and each line of \( B \) of the coefficients \( \beta_m \). Due to the periodicity, going to the next line consists in shifting the coefficients by one index. So these matrices are of the form

\[
C = \begin{pmatrix}
c_0 & c_1 & c_2 & \ldots & c_{N-1} \\
c_{N-1} & c_0 & c_1 & c_{N-2} \\
c_{N-2} & c_{N-1} & c_0 & c_{N-3} \\
\vdots & \ddots & \ddots & \ddots \\
c_1 & c_2 & c_3 & \ldots & c_0
\end{pmatrix}
\]  

(2.23)

with \( c_0, c_1, \ldots, c_{N-1} \in \mathbb{R} \) are called circulant.

Example 1. The explicit upwind scheme reads for \( a > 0 \)

\[ u_j^{n+1} = u_j^n - \frac{a \Delta t}{\Delta x} (u_j^n - u_{j-1}^n) = \left(1 - \frac{a \Delta t}{\Delta x}\right) u_j^n + \frac{a \Delta t}{\Delta x} u_{j-1}^n \]

This yields the matrix form \( U^{n+1} = CU^n \) with

\[ c_0 = 1 - \frac{a \Delta t}{\Delta x}, \quad c_{N-1} = \frac{a \Delta t}{\Delta x} \]

and \( c_j = 0 \) for \( 1 \leq j \leq N - 2 \).

Proposition 3. The eigenvalues of the circulant matrix \( C \) are given by

\[ \lambda_k = \sum_{j=0}^{N-1} c_j e^{2i\pi jk/N}. \]  

(2.24)

Proof. Let \( J \) be the circulant matrix of the form (2.23) with \( c_1 = 1 \) and \( c_j = 0 \) for \( j \neq 1 \). We notice that an arbitrary circulant matrix \( C \) can be written as a polynomial in \( J \)

\[ C = \sum_{j=0}^{N-1} c_j J^j. \]

As \( J^N = I \), the eigenvalues of \( J \) are the \( N^{th} \) roots of unity that are given by \( \omega^k = e^{2i\pi kN/N} \). We also observe that \( E_k \) defined in (2.19) verifies \( JE_k = \omega^k E_k \) which means that \( E_k \) is an eigenvector of \( J \) associated to the eigenvalue \( \omega^k \).

This implies that

\[ CE_k = \sum_{j=0}^{N-1} c_j J^j E_k = \sum_{j=0}^{N-1} c_j e^{2i\pi jk/N} E_k, \]

and so the eigenvalues of \( C \) associated to the eigenvectors \( E_k \) are those defined in (2.24).

Proposition 4. Any circulant matrix \( C \) can be written in the form \( C = P\Lambda P^* \) where \( P \) is the matrix of the discrete Fourier transform and \( \Lambda \) is the diagonal matrix of the eigenvalues of \( C \). In particular all circulant matrices have the same eigenvectors (which are the columns of \( P \)), and any matrix of the form \( P\Lambda P^* \) is circulant.
Corollary 2. We have the following properties:

- The product of two circulant matrix is a circulant matrix.
- A circulant matrix whose eigenvalues are all non vanishing is invertible and its inverse is circulant.

Proof. The key point is that all circulant matrices can be diagonalized in the same basis of eigenvectors. If $C_1$ and $C_2$ are two circulant matrices, we have $C_1 = PA_1 P^*$ and $C_2 = PA_2 P^*$ so $C_1 C_2 = PA_1 A_2 P^*$.

If all eigenvalues $\lambda_i$ of $C = PA P^*$ are non vanishing, $A^{-1}$ is well defined and its diagonal coefficients are $1/\lambda_i$, $0 \leq i \leq N - 1$ and

$$PAP^* P^{-1} P^* = I.$$ 

So the inverse of $C$ is the circulant matrix $PA^{-1} P^*.$

Now applying the discrete Fourier transform to our generic scheme (2.22)

$$P^* A U^{n+1} = P^* B U^n \iff P^* A P^* U^{n+1} = P^* B P P^* U^n$$

which is equivalent to

$$\Lambda_A \hat{U}^{n+1} = \Lambda_B \hat{U}^n,$$

where $\Lambda_A$ and $\Lambda_B$ are the diagonal matrices containing the eigenvalues of the circulant matrices $A$ and $B$ which are given explicitly from the matrix coefficients. This implies that

$$\|\hat{U}^{n+1}\|^2 \leq \|\Lambda_A^{-1} \Lambda_B \hat{U}^n\|^2 \leq \max_i \left( \frac{|\lambda_{B,i}|}{|\lambda_{A,i}|} \right)^2 \|\hat{U}^n\|^2,$$

which follows because $\|\hat{U}\| = \|U\|$ for any vector $U$ that the scheme is $L^2$ stable if $\max_i \frac{|\lambda_{B,i}|}{|\lambda_{A,i}|} \leq 1.$

$L^2$ stability of the implicit upwind scheme Let us now apply this technique to prove the stability of the implicit upwind scheme (2.9):

Proposition 5. The scheme (2.9) is stable for the $L^2$ norm for all strictly positive values of $\Delta x$ and $\Delta t$.

Proof. Let us denote by $\alpha = a \frac{\Delta t}{\Delta x}$. We notice that matrix $B$ is circulant with $c_0 = 1 + \alpha$, $c_{N-1} = -\alpha$, the other $c_i$ being 0.

The eigenvalues of $B$ thus are $\lambda_k = c_0 + c_{N-1} e^{-2i k \pi / N}$. Which implies that

$$\text{Re}(\lambda_k) = 1 + \alpha (1 - \cos \frac{2k\pi}{N}) \geq 1,$$

as $\alpha \geq 0$. It follows that all eigenvalues of $B$ have a modulus larger or equal to 1, which implies that $B$ is invertible. Moreover the eigenvalues of its inverse all have modulus less or equal to 1, which implies the $L^2$ stability of the scheme.

Proposition 6. The centred scheme second order in space and first order in time:

$$u_{j}^{n+1} = u_{j}^{n} - a \frac{\Delta t}{2 \Delta x} (u_{j+1}^{n} - u_{j-1}^{n}).$$

is unstable in $L^2$. 

15
Proof. Let us denote by \( \alpha = a \frac{\Delta t}{\Delta x} \). The first order in time centred scheme becomes in matrix form
\[ U^{n+1} = AU^n \]
where \( A \) is the circulant matrix with three non vanishing diagonals corresponding to \( c_0 = 1, c_1 = -c_{N-1} = \frac{\alpha}{2} e^{\frac{2\pi i k}{N}} - e^{-\frac{2\pi i k}{N}} = 1 + i\alpha \sin \frac{2\pi k}{N} \). Hence its eigenvalues are
\[ \lambda_k = 1 - \alpha^2 (e^{\frac{2\pi i k}{N}} - e^{-\frac{2\pi i k}{N}}) = 1 + i\alpha \sin \frac{2\pi k}{N} \]
so that \( |\lambda_k| > 1 \) for all \( k \) such that \( \sin \frac{2\pi k}{N} \neq 0 \). Hence the scheme is unstable.

Corollary 3. If the exact solution is of class \( C^3 \), the schemes (2.7) and (2.9) converge.

Proof. This follows immediately from the Lax theorem by applying propositions 1 and 2.

2.1.7 High-order time schemes

We consider here an approach based on the method of lines, where we first perform a semi-discretization in space and then look for an appropriate time discretization.

When working with linear homogeneous equations with no source term, the simplest way to derive high order time schemes is to use a Taylor expansion in time and plug in the expression of the successive time derivatives obtained from the differential system resulting from the semi-discretization in space. Consider for example that after semi-discretization in space using Finite Differences (or any other space discretization method) we obtain the differential systems
\[
\frac{dU}{dt} = LU,
\]
with \( U = \begin{pmatrix} u_0(t) \\ \vdots \\ u_{N-1}(t) \end{pmatrix} \)
and \( L \) the appropriate matrix coming from the semi-discretization in space. Then a Taylor expansion in time up to order \( p \) yields
\[
U(t_{n+1}) = U(t_n) + \Delta t \frac{dU}{dt}(t_n) + \cdots + \frac{\Delta t^p}{p!} \frac{d^p U}{dt^p}(t_n) + O(\Delta t^{p+1}).
\]
Now if \( L \) does not depend on time and \( U \) verifies \( \frac{dU}{dt} = LU \), we get that
\[
\frac{d^p U}{dt^p} = L^p U, \text{ for any integer } p.
\]
Hence, denoting \( U^n \) an approximation of \( U(t_n) \), we get a time scheme of order \( p \) using the formula
\[
U^{n+1} = U^n + \Delta t L U^n + \cdots + \frac{\Delta t^p}{p!} L^p U^n = (I + \Delta t L + \cdots + \frac{\Delta t^p}{p!} L^p)U^n. \tag{2.25}
\]
For \( p = 1 \) this boils down to the standard explicit Euler scheme.

Writing \( U^n \) the solution in vector form at time \( t_n \), we define the propagation matrix
\[
A = I + \Delta t L + \cdots + \frac{\Delta t^p}{p!} L^p
\]
such that
\[
U^{n+1} = AU^n.
\]

The propagation matrix defines the fully discrete explicit scheme, which advances the solution from time \( t_n \) to time \( t_{n+1} \). Assuming the space discretization to be consistent, as the temporal scheme has been derived using a Taylor expansion it is also consistent. So the scheme will be convergent if it is stable.

Definition 5. The numerical scheme defined by the propagation matrix \( A \) is stable if there exists \( \tau > 0 \) such that \( U^n \) is bounded for all \( n \in \mathbb{N} \) and \( \Delta t < \tau \).
The criterion we shall use to determine the stability of a scheme is given by the following property:

**Proposition 7.** The numerical scheme defined by the propagation matrix $A$ is stable if there exists $\tau > 0$ such that for all $\Delta t < \tau$ all eigenvalues of $A$ are of modulus less or equal to 1.

**Stability of Taylor schemes.** For a Taylor scheme of order $p$ applied to $\frac{dU}{dt} = LU$, we have $A = I + \Delta t L + \cdots + \frac{\Delta t^p}{p!} L^p$. Then denoting by $\lambda$ an eigenvalue of $L$, the corresponding eigenvalue of $A$ is $\mu = 1 + \lambda \Delta t + \cdots + \lambda^p \frac{\Delta t^p}{p!}$.

**Definition 6.** The stability region of an explicit scheme defined by the propagation matrix $A = I + \Delta t L + \cdots + \frac{\Delta t^p}{p!} L^p$ is the region of the complex plane

$$\{ z \in \mathbb{C} | |\mu(z)| = |1 + z + \cdots + \frac{\Delta z^p}{p!}| \leq 1 \}$$

This means that the time scheme used for the semi-discrete model $\frac{dU}{dt} = LU$ is stable provided all the eigenvalues $\lambda$ of $L$ are such that $\lambda \Delta t$ is in the stability region.

**Examples.**

1. The Upwind scheme reads $\frac{du_i(t)}{dt} = -a \frac{u_i(t) - u_{i-1}(t)}{\Delta x}$ for each grid point $x_i$ and corresponds to the circulant matrix $L$ with $c_0 = -\frac{a}{\Delta x} = -c_{N-1}$. So its eigenvalues verify

$$\lambda_k \Delta t = -\frac{a \Delta t}{\Delta x} \left(1 - e^{-2\pi i k/N}\right).$$

Obviously, for any integer value of $k$, $\lambda_k \Delta t$ is on a circle in the complex plane of radius $\alpha = \frac{a \Delta t}{\Delta x}$ centered at $(-\frac{a \Delta t}{\Delta x}, 0)$. The stability region of the explicit Euler method is the circle of radius 1 centred at $(-1, 0)$, so that in this case we see again that the scheme is stable provided $\frac{a \Delta t}{\Delta x} \leq 1$.

The stability region for the explicit Euler scheme and the location of the eigenvalues of the upwind schemes for different values of $\alpha = \frac{a \Delta t}{\Delta x}$ are shown in Fig. 2.2.

![Figure 2.2: Stability region for the explicit Euler scheme and the location of the eigenvalues of the upwind schemes for $\alpha = \frac{a \Delta t}{\Delta x}$](image)

For the higher order schemes, the radius corresponding to the maximal stability can be found by computing the second real root (in addition to 0) $r_2$ of the equation $|\mu(\lambda \Delta t)| = 1$, see Fig. 2.3 for a graphical representation of the stability regions. Indeed, the stability regions are symmetric with respect to the real axis and the eigenvalue corresponding to the maximum modulus is on the real axis. For the second order Taylor scheme

$$\mu(z) = 1 + z + z^2/2.$$
The Stability limit is reached on the real axis for \( x \in \mathbb{R} \) such that \( |\mu(x)| = 1 \). The first case \( \mu(x) = 1 \) corresponds to \( x(1 + x/2) = 0 \), which has the roots \( x = 0 \) and \( x = -2 \). For \( \mu(x) = -1 \), we get \( 2 + x + x^2/2 = (x + 2)^2 = 0 \). There is only one real root \( x = -2 \). So in this case the stability interval on the real axis is \([-2, 0]\). The real parts of the eigenvalues of the upwind scheme verify \( \text{Re}(\lambda_k \Delta t) = -\alpha(1 - \cos(2\pi k/\Delta x)) \), whose minimum value is \(-2\alpha\). So, all eigenvalues are in the stability region if \( 0 \leq \alpha \leq 1 \).

For the order 3 scheme the stability region corresponds to \( |\mu(\lambda \Delta t)| \leq 1 \) for 
\[
\mu(z) = 1 + z + z^2/2 + z^3/6.
\]

As previously the stability limit is reached on the real axis for \( |\mu(x)| = 1 \). For \( \mu(x) = 1 \) we find \( x(1 + x/2 + x^2/6) = 0 \) whose only real root is \( x = 0 \). For \( \mu(x) = -1 \) we get \( 2 + x + x^2/2 + x^3/6 = 0 \) which has one real root \( r_2 \approx -2.5127 \). The same reasoning as before shows that the scheme is stable if \( 0 \leq \alpha \leq r_2/2 \approx 1.25637 \).

In the same way, we find that the upwind scheme of order 5 in time is stable if \( \alpha \leq r_2/2 \approx 1.6085 \) and for the upwind scheme of order 6 in time if \( \alpha \leq r_2/2 \approx 1.7767 \).

The value of \( \alpha \) corresponds to the diameter of the largest circle of eigenvalues that is still completely enclosed in the stability region. We notice that the maximal stable time step is larger for the higher order time schemes. These results are displayed in Fig. 2.3.

![Figure 2.3: Stability regions for Taylor time schemes of order 2, 3 and 4 (from left to right). Location of eigenvalues (circles) of upwind scheme \((N = 16)\) for different values of \( \alpha = \frac{a \Delta t}{\Delta x} \).](image)

2. The centered scheme: 
\[
\frac{du_i(t)}{dt} = -a \frac{u_{i+1}(t) - u_{i-1}(t)}{2\Delta x}
\]

Corresponds to the circulant matrix \( \mathcal{L} \) with 
\[
c_1 = -\frac{a}{2\Delta x} = -c_{N-1}.
\]

The corresponding eigenvalues are such that 
\[
\lambda_k \Delta t = -\frac{a \Delta t}{2\Delta x} \left( e^{\frac{2\pi i k}{N}} - e^{-\frac{2\pi i k}{N}} \right) = \frac{ia \Delta t}{\Delta x} \sin \frac{2\pi k}{N}.
\]

Hence the eigenvalues are all purely imaginary and the modulus of the largest one is \( \frac{a \Delta t}{\Delta x} \). Hence a fully discrete scheme will be stable if the chosen time scheme has a stability region including a non empty interval on the imaginary axis. The stability regions of the first and second order schemes are strictly on the negative part of the complex plane with only the origin \( z = 0 \) being in the stability region on the imaginary axis. See Fig. (2.4). So the corresponding schemes are unstable for any \( \Delta t > 0 \).
The third order Taylor scheme is stable for values $iy$ on the imaginary axis if $|\mu(iy)|^2 \leq 1$. We have

$$|\mu(iy)|^2 - 1 = \left| 1 + iy - \frac{y^2}{2} - \frac{i y^3}{6} \right|^2 - 1 = \frac{1}{12} y^4 \left( \frac{y^2}{3} - 1 \right).$$

Hence the scheme is stable for the centered discretization in space if $\alpha = \frac{a \Delta t}{\Delta x} \leq \sqrt{3}$.

A similar computation shows that the fourth order Taylor scheme is stable for the centered discretization in space if $\alpha = \frac{a \Delta t}{\Delta x} \leq 2\sqrt{2}$. These results are displayed in Fig. 2.5.

Let us conclude with the stability regions of the order 5 and 6 Taylor schemes in time. Both time schemes are stable for the upwind schemes with maximum values of $\alpha = \frac{a \Delta t}{\Delta x}$ of approximately 1.61 and 1.78 respectively.

As for the stability region on the imaginary axis needed for the centered scheme, the $5^{th}$ order scheme is peculiar as it indeed has a part of the stability region on the imaginary axis, but
these intervals are not around 0. So that the order 5 scheme is unstable for the centered scheme for all \( \Delta t > 0 \). The order 6 Taylor scheme is also unstable for the centered scheme for all \( \Delta t > 0 \), but its stability region does not contain any part of the imaginary axis.

These results are represented in Figures (2.6) and (2.7). The latter corresponds to a zoom around the imaginary axis of the former one.

**Remark 2.** The instability of the 5\textsuperscript{th} order Taylor scheme in time associated to the centered scheme in space is peculiar as the zooms around the imaginary axis of Figure 2.7 tell us. Even though there is a part of the imaginary axis in the stability zone, there is also a part in the neighborhood of 0 which is not. Therefore the small eigenvalues of \( A \) will lead to instability on longer time scales unlike the usual case, where the largest eigenvalues are unstable. This is problematic, as unlike usual Courant condition instability problems which reveal themselves very fast, this leads to a small growth in time.

![Figure 2.6: Stability regions of Taylor schemes of order 5 and 6 and eigenvalues of centered scheme and upwind scheme for \( N = 16 \).](image)

**2.1.8 The Cauchy-Kowalevski procedure**

The Cauchy-Kowalevski procedure consists in writing a Taylor expansion in time to update the scheme from time \( t_n \) to \( t_{n+1} \) and then to use the equation to replace the time derivatives in the Taylor expansion by spatial derivatives. This is particularly simple for the linear advection, but can be used in principle (with more complicated computations) for any hyperbolic equation or system.

Let us now apply this procedure for our 1D linear advection: (we denote as usual by \( u_i^n = u(t_n, x_i) \))

\[
\begin{align*}
    u_i^{n+1} &= u_i^n + \Delta t \frac{\partial u}{\partial t}(t_n, x_i) + \frac{\Delta t^2}{2} \frac{\partial^2 u}{\partial t^2}(t_n, x_i) + O(\Delta t^3). \\
\end{align*}
\]  

(2.28)

Then using the equation \( \frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0 \) we get

\[
\begin{align*}
    \frac{\partial u}{\partial t}(t_n, x_i) &= -a \frac{\partial u}{\partial x}(t_n, x_i), \\
    \frac{\partial^2 u}{\partial t^2}(t_n, x_i) &= \frac{\partial}{\partial t}(-a \frac{\partial u}{\partial x}) = -a \frac{\partial}{\partial x} \frac{\partial u}{\partial t} = a^2 \frac{\partial^2 u}{\partial x^2}(t_n, x_i),
\end{align*}
\]
Figure 2.7: Stability regions of Taylor schemes of order 5 and 6 and eigenvalues of centered scheme and upwind scheme for \( N = 16 \). Zoom around the imaginary axis.

and now we use centered schemes for the first and second order spatial derivatives of \( u \):

\[
\frac{\partial u}{\partial x}(t_n, x_i) = \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} + O(\Delta x^2), \quad \frac{\partial^2 u}{\partial x^2}(t_n, x_i) = \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2} + O(\Delta x^2).
\]

Plugging this into the Taylor expansion (2.28), we get:

\[
u_{i+1}^{n+1} = u_i^n - \frac{a\Delta t}{2\Delta x}(u_{i+1}^n - u_{i-1}^n) + \frac{a^2\Delta t^2}{2\Delta x^2}(u_{i+1}^n - 2u_i^n + u_{i-1}^n) + \Delta tO(\Delta t^2 + \Delta x^2),
\]

which yields the numerical scheme

\[
u_{i+1}^{n+1} = u_i^n - \frac{a\Delta t}{2\Delta x}(u_{i+1}^n - u_{i-1}^n) + \frac{a^2\Delta t^2}{2\Delta x^2}(u_{i+1}^n - 2u_i^n + u_{i-1}^n)
\]

This scheme, which is second order in space and time, is known as the Lax-Wendroff scheme.

Let us perform the von Neumann stability analysis. We see that the transfer matrix is tridiagonal circulant with 
\( c_0 = 1 - \frac{a^2\Delta t^2}{\Delta x^2} \), \( c_1 = -\frac{a\Delta t}{2\Delta x} + \frac{a^2\Delta t^2}{2\Delta x^2} \), and \( c_{N-1} = \frac{a\Delta t}{2\Delta x} + \frac{a^2\Delta t^2}{2\Delta x^2} \). So its eigenvalues are given by

\[
\lambda_k = 1 - \frac{a^2\Delta t^2}{\Delta x^2} + \left( \frac{a^2\Delta t^2}{2\Delta x^2} - \frac{a\Delta t}{2\Delta x} \right) e^{\frac{2\pi ik}{N}} + \left( \frac{a\Delta t}{2\Delta x} + \frac{a^2\Delta t^2}{2\Delta x^2} \right) e^{-\frac{2\pi ik}{N}}
\]

\[
= 1 - \frac{a^2\Delta t^2}{\Delta x^2} + \frac{a^2\Delta t^2}{\Delta x^2} \cos \frac{2\pi k}{N} - i \frac{a\Delta t}{\Delta x} \sin \frac{2\pi k}{N}.
\]

We recognize that \( \lambda_k \) is on an ellipse oriented along the real and imaginary axes and centred at the point \((1 - \frac{a^2\Delta t^2}{\Delta x^2}, 0)\). Its diagonal on the real axis is \( 2\frac{a^2\Delta t^2}{\Delta x^2} \) and on the imaginary axis it is \( \frac{2a\Delta t}{\Delta x} \).

Note that the real parts of \( \lambda_k \) go from \( 1 - 2\frac{a^2\Delta t^2}{\Delta x^2} \) to \( 1 \). So a necessary condition for stability being that the modulus of the real part is smaller than \( 1 \), we find that for this to be realized we need \( \frac{a^2\Delta t^2}{\Delta x^2} \leq 1 \) or equivalently for \( a \geq 0 \) that \( \frac{a\Delta t}{\Delta x} \leq 1 \).
Conversely assuming that $0 \leq \frac{a \Delta t}{\Delta x} \leq 1$, we find
\[ |\lambda_k| \leq 1 - \frac{a^2 \Delta t^2}{\Delta x^2} + \left| \frac{a^2 \Delta t^2}{\Delta x^2} \cos \frac{2\pi j k}{N} - i \frac{a \Delta t}{\Delta x} \sin \frac{2\pi j k}{N} \right| \leq 1 - \frac{a^2 \Delta t^2}{\Delta x^2} + \frac{a \Delta t}{\Delta x} \leq 1. \]

Finally the Lax-Wendroff scheme is stable under the same CFL condition as the upwind scheme $\frac{a \Delta t}{\Delta x} \leq 1$.

2.1.9 Modified equation analysis

Another useful way to understand the properties of a numerical scheme in addition to its convergence and stability properties as we have seen them up to now is to find a modified equation that the solution of the numerical scheme satisfies exactly. This is widely used for the discretization of Hamiltonian systems for which some classes of numerical schemes, called symplectic, can be shown to be the exact solution of another hamiltonian system with a modified hamiltonian expressed as an infinite series, as discussed in [5]. This method is also widely used to get a better grasp at the properties of numerical schemes for hyperbolic partial differential equations, which we are interested in this lecture. We will use consider the method only to get a qualitative insight on the behavior of the solution of the numerical scheme. It can be made rigorous in different ways depending on the precise feature that one is interested in. See [4] for a discussion about the scope of the method.

Starting from an approximation of a function $u$ by its values $u^n_j$ at grid points $x_j$ and at discrete times $t_n$. We define a function $\tilde{u}$ such that
\[ \tilde{u}(t_n, x_j) = u^n_j \text{ for } j = 0, \ldots, N - 1, \quad n = 0, \ldots, M. \] (2.30)

Modified equation for the explicit upwind scheme

Let us consider the explicit Euler upwind scheme
\[ u^n_{j+1} = u^n_j - \frac{a \Delta t}{\Delta x} (u^n_j - u^n_{j-1}), \] (2.31)

and consider $\tilde{u}$ verifying (2.30). Taylor expansions of $\tilde{u}$ around $(t_n, x_j)$ yield
\[ \tilde{u}(t_{n+1}, x_j) = \tilde{u}(t_n, x_j) + \Delta t \frac{\partial \tilde{u}}{\partial t}(t_n, x_j) + \frac{\Delta t^2}{2} \frac{\partial^2 \tilde{u}}{\partial t^2}(t_n, x_j) + \sum_{p \geq 2}^\infty \frac{\Delta t^p}{p!} \frac{\partial^p \tilde{u}}{\partial t^p}(t_n, x_j), \] (2.32)
\[ \tilde{u}(t_n, x_{j\pm1}) = \tilde{u}(t_n, x_j) \pm \Delta x \frac{\partial \tilde{u}}{\partial x}(t_n, x_j) + \frac{\Delta x^2}{2} \frac{\partial^2 \tilde{u}}{\partial x^2}(t_n, x_j) + \sum_{p \geq 2} (\pm 1)^p \frac{\Delta x^p}{p!} \frac{\partial^p \tilde{u}}{\partial x^p}(t_n, x_j). \] (2.33)

The Taylor series might not converge, but we will only need a few terms. Using (2.31) divided by $\Delta t$ we get
\[ \frac{\partial \tilde{u}}{\partial t} + a \frac{\partial \tilde{u}}{\partial x} + \frac{\Delta t}{2} \frac{\partial^2 \tilde{u}}{\partial t^2} - a \frac{\Delta x}{2} \frac{\partial^2 \tilde{u}}{\partial x^2} = O(\Delta x^2). \] (2.34)
As $\Delta t$ and $\Delta x$ are related by the CFL number, we assume that they are of the same order. Moreover, we assume that $\tilde{u}$ verifies this modified equation for all $t$ and $x$. In order to remove the higher order time derivatives, we take the time derivative of (2.34) $\frac{\partial^2 \tilde{u}}{\partial t^2}(t, x)$
\[ \frac{\partial^2 \tilde{u}}{\partial t^2} + a \frac{\partial^2 \tilde{u}}{\partial x \partial t} + \frac{\Delta t}{2} \frac{\partial^3 \tilde{u}}{\partial t^3} - a \frac{\Delta x}{2} \frac{\partial^3 \tilde{u}}{\partial x^2 \partial t} = O(\Delta x^2) \] (2.35)

and in the same way we take the $x$ derivative of (2.34)
\[ \frac{\partial^2 \tilde{u}}{\partial t \partial x} + a \frac{\partial^2 \tilde{u}}{\partial x^2} + \frac{\Delta t}{2} \frac{\partial^3 \tilde{u}}{\partial t^2 \partial x} - a \frac{\Delta x}{2} \frac{\partial^3 \tilde{u}}{\partial x^3} = O(\Delta x^2). \] (2.36)
Plugging the expression of \( \frac{\partial^2 \tilde{u}}{\partial x^2} \) from (2.36) into (2.35), we find

\[
\frac{\partial^2 \tilde{u}}{\partial t^2} - a^2 \frac{\partial^2 \tilde{u}}{\partial x^2} = O(\Delta x).
\]

(2.37)

We can now plug this equation (2.37) into (2.34) to get the final modified equation

\[
\frac{\partial \tilde{u}}{\partial t} + a \frac{\partial \tilde{u}}{\partial x} - \left(1 - \frac{a\Delta t}{\Delta x}\right) \frac{a\Delta x}{2} \frac{\partial^2 \tilde{u}}{\partial x^2} = O(\Delta x^2).
\]

(2.38)

We can observe that the dominating correction term corresponds to a diffusion term with coefficient

\[
\left(1 - \frac{a\Delta t}{\Delta x}\right) \frac{a\Delta x}{2}.
\]

For the solution to be bounded, we need that \( \frac{a\Delta t}{\Delta x} \leq 1 \), which is again our CFL condition. There is no diffusion when the CFL number is exactly equal to one and the diffusion coefficient becomes larger when the CFL number gets closer to 0. Moreover the diffusion coefficient is proportional to \( \Delta x \) so that it gets smaller when the grid is refined as expected.

\section*{Modified equation for the Lax-Wendroff scheme}

The Lax-Wendroff scheme (2.29) reads

\[
u_j^{n+1} = u_j^n - \frac{a\Delta t}{2\Delta x} (u_{j+1}^{n} - u_{j-1}^{n}) + \frac{a^2 \Delta t^2}{2\Delta x^2} (u_{j+1}^{n} - 2u_{j}^{n} + u_{j-1}^{n})
\]

(2.39)

Using the Taylor expansions (2.32) and (2.33), we get

\[
u_j^{n+1} - u_j^n = \frac{\partial \tilde{u}}{\partial t} + \frac{\Delta t}{2} \frac{\partial^2 \tilde{u}}{\partial t^2} + \frac{\Delta t^2}{6} \frac{\partial^3 \tilde{u}}{\partial t^3} + O(\Delta t^3),
\]

(2.40)

\[
u_{j+1}^{n} - u_j^n = \frac{\partial \tilde{u}}{\partial x} + \frac{\Delta x}{6} \frac{\partial^3 \tilde{u}}{\partial x^3} + O(\Delta x^3),
\]

(2.41)

\[
u_{j+1} - 2u_{j}^{n} + u_{j-1}^{n} = \frac{\partial^2 \tilde{u}}{\partial x^2} + O(\Delta x^2).
\]

(2.42)

Plugging these into the Lax-Wendroff scheme (2.39) divided by \( \Delta t \) we get

\[
\frac{\partial \tilde{u}}{\partial t} + \frac{\Delta t}{2} \frac{\partial^2 \tilde{u}}{\partial t^2} + \frac{\Delta t^2}{6} \frac{\partial^3 \tilde{u}}{\partial t^3} + a \frac{\partial \tilde{u}}{\partial x} + \frac{a\Delta x}{6} \frac{\partial^3 \tilde{u}}{\partial x^3} - \frac{a^2 \Delta t^2}{2} \frac{\partial^2 \tilde{u}}{\partial x^2} = O(\Delta x^3)
\]

(2.43)

where we still assume that \( \Delta t \) and \( \Delta x \) are related by a constant factor so that \( O(\Delta x) = O(\Delta t) \). As for the upwind scheme, we now want to express the higher order time derivatives with respect to space derivatives. Taking the time derivative of (2.43), multiplying by \( \Delta t \) and putting the order 3 terms on the right-hand-side we get

\[
\Delta t \frac{\partial^2 \tilde{u}}{\partial t^2} + \frac{\Delta t^2}{2} \frac{\partial^3 \tilde{u}}{\partial t^3} + a\Delta t \frac{\partial^2 \tilde{u}}{\partial x^2 t} - \frac{a^2 \Delta t^2}{2} \frac{\partial^3 \tilde{u}}{\partial x^2 \partial t} = O(\Delta x^3).
\]

(2.44)

Taking another time derivative and multiplying again by \( \Delta t \)

\[
\Delta t^2 \frac{\partial^3 \tilde{u}}{\partial t^3} + a\Delta t^2 \frac{\partial^3 \tilde{u}}{\partial x \partial t^2} = O(\Delta x^3).
\]

(2.45)

Now subtracting 1/12 of (2.45) from 1/2 of (2.44) gives

\[
\frac{\Delta t}{2} \frac{\partial^2 \tilde{u}}{\partial t^2} + \frac{\Delta t^2}{6} \frac{\partial^3 \tilde{u}}{\partial t^3} + a\Delta t \frac{\partial^2 \tilde{u}}{\partial x t} - \frac{a^2 \Delta t^2}{4} \frac{\partial^3 \tilde{u}}{\partial x^2 t} - \frac{a\Delta t^2}{12} \frac{\partial^3 \tilde{u}}{\partial x \partial t} = O(\Delta x^3).
\]

(2.46)
We now need to express the mixed derivatives above with respect to spatial derivatives. Taking
the derivative in \(x\) of (2.43) and multiplying by \(a \Delta t\) yields
\[
a \Delta t \frac{\partial^2 \tilde{u}}{\partial x \partial t} + \frac{a \Delta t^2}{2} \frac{\partial^3 \tilde{u}}{\partial x \partial t^2} + a^2 \Delta t \frac{\partial^2 \tilde{u}}{\partial x^2} - \frac{a^3 \Delta t^2}{2} \frac{\partial^3 \tilde{u}}{\partial x^3} = O(\Delta x^3),
\] (2.47)
taxing the \(x\) derivative of (2.47) and multiplying by \(a \Delta t\) yields
\[
a^2 \Delta t^2 \frac{\partial^3 \tilde{u}}{\partial x^2 \partial t} + a^3 \Delta t^2 \frac{\partial^3 \tilde{u}}{\partial x^3} = O(\Delta x^3),
\] (2.48)
Finally, taking the \(t\) derivative of (2.47) and multiplying by \(\Delta t\) yields
\[
a \Delta t^2 \frac{\partial^3 \tilde{u}}{\partial x \partial t^2} + a^2 \Delta t^2 \frac{\partial^3 \tilde{u}}{\partial x^2 \partial t} = O(\Delta x^3),
\] (2.49)
which implies, using (2.48) that
\[
a \Delta t^2 \frac{\partial^3 \tilde{u}}{\partial x \partial t^2} - a^3 \Delta t^2 \frac{\partial^3 \tilde{u}}{\partial x^3} = O(\Delta x^3).
\] (2.50)
We now plug this expression into (2.47) to get
\[
a \Delta t \frac{\partial^2 \tilde{u}}{\partial x \partial t} + a^2 \Delta t \frac{\partial^2 \tilde{u}}{\partial x^2} = O(\Delta x^3).
\] (2.51)
With this, we can express all the mixed derivatives with respect to spatial derivatives, which we
plug into (2.46)
\[
\frac{\Delta t}{2} \frac{\partial^2 \tilde{u}}{\partial t^2} + \frac{\Delta t^2}{6} \frac{\partial^3 \tilde{u}}{\partial t^3} - \frac{a^2 \Delta t^2}{2} \frac{\partial^2 \tilde{u}}{\partial x^2} + \frac{a^3 \Delta t^2}{4} \frac{\partial^3 \tilde{u}}{\partial x^3} - \frac{a^3 \Delta t^2}{12} \frac{\partial^3 \tilde{u}}{\partial x^3} = O(\Delta x^3).
\] (2.52)
We can now plug this expression of the higher order time derivatives into (2.43) to get the following
modified equation
\[
\frac{\partial \tilde{u}}{\partial t} + a \frac{\partial \tilde{u}}{\partial x} + \frac{a \Delta x^2}{6} \left(1 - \frac{a^2 \Delta t^2}{\Delta x^2}\right) \frac{\partial^3 \tilde{u}}{\partial x^3} = O(\Delta x^3)
\] (2.53)
We see here that the dominating error term in the modified equation is in \(\Delta x^2\) as expected for a
second order scheme and it is in factor of a third order derivative, such that the error is dominated
by dispersion rather than dissipation, which we have for second order space derivatives as in the
modified equation of the explicit upwind scheme (2.31).

**Fourier analysis of linear evolution equations**

In order to better understand the qualitative behavior of evolution partial differential equations with
different order spatial derivatives, let us now compute the exact solutions on a periodic domain as
a Fourier series.

1. **Pure advection equation**
\[
\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0.
\] (2.54)
An analytical solution for an initial condition \(u_0\) reads
\[
u(t, x) = u_0(x - at).
\] (2.55)
2. Addition of a second order partial derivative in space. This leads to the advection diffusion equation which reads in 1D:
\[
\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} - \nu \frac{\partial^2 u}{\partial x^2} = 0.
\] (2.56)

Given an initial condition \( u_0 \) and a periodic domain of length \( L \) the solution can be sought as a Fourier series:
\[
u(t, x) = \sum_{k \in \mathbb{Z}} u_k(t) e^{\frac{2i\pi k}{L} x} \] (2.57)

with
\[
u_k = \frac{1}{L} \int_0^L u(t, x) e^{-\frac{2i\pi k}{L} x} dx \] (2.58)

Then applying the this relation defining the Fourier coefficients on (2.56) we get
\[
\frac{d \nu_k}{dt} + \left[ a \frac{2i\pi k}{L} - \nu \left( \frac{2\pi k}{L} \right)^2 \right] \nu_k = 0, \] (2.59)

whose solution is
\[
u_k(t) = \nu_k(0) e^{-\frac{2i\pi k}{L} at} e^{-\left( \frac{2\pi k}{L} \right)^2 \nu t}. \] (2.60)

Plugging the expression of the Fourier coefficients back into the Fourier series (2.57), the solution of (2.59) can be expressed as
\[
u(t, x) = \sum_{k \in \mathbb{Z}} \nu_k(0) e^{\frac{2i\pi k}{L} (x-at)} e^{-\left( \frac{2\pi k}{L} \right)^2 \nu t}. \] (2.61)

We observe that without diffusion, \( \nu = 0 \), we get \( \nu(t, x) = \nu(0, x-at) = \nu_0(x-at) \) as expected, and with diffusion the Fourier coefficients are damped by the factor
\[
\left( \frac{2\pi k}{L} \right)^2 \nu t,
\]

which becomes larger when \( |k| \) is larger. The highest modes corresponding to the shortest wavelengths are damped faster, which leads to a smoothing of the solution.

3. Addition of a third order partial derivative in space. This leads to a linear Korteweg–de Vries (KdV) type equation which reads in 1D:
\[
\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} + \nu \frac{\partial^3 u}{\partial x^3} = 0.
\] (2.62)

Note that the classical non linear KdV equation reads in 1D
\[
\frac{\partial u}{\partial t} + 6u \frac{\partial u}{\partial x} + \frac{\partial^3 u}{\partial x^3} = 0.
\] (2.63)

In the same way as for the advection-diffusion equation, the equation on the Fourier coefficients reads for (2.62)
\[
\frac{d \nu_k}{dt} + \left[ a \frac{2i\pi k}{L} - i\nu \left( \frac{2\pi k}{L} \right)^3 \right] \nu_k = 0, \] (2.64)

whose solution is
\[
u_k(t) = \nu_k(0) e^{-\frac{2i\pi k}{L} at} e^{i\left( \frac{2\pi k}{L} \right)^3 \nu t}. \] (2.65)
Plugging the expression of the Fourier coefficients back into the Fourier series (2.57), the solution of (2.59) can be expressed as

\[ u(t, x) = \sum_{k \in \mathbb{Z}} u_k(0) e^{\frac{2\pi ik}{L} \left[ x - \left( a - \left( \frac{2\pi k}{L} \right)^2 \nu \right) t \right]} \]  

(2.66)

Here again, we observe that for \( \nu = 0 \), we get as expected that \( u(t, x) = u_0(t - ax) \), but now for \( \nu \neq 0 \) there is no diffusion, the amplitude of each Fourier mode is conserved, but the phase velocity (the velocity at which the wave is propagating), which is \( a \) for pure advection is now different for each Fourier mode, namely

\[ a - \left( \frac{2\pi k}{L} \right)^2 \nu. \]

This phenomenon is called dispersion. So the wave is propagating slower for positive \( \nu \), the more that \( |k| \) is larger.

**Remark 3.** The computations performed above can be extended to partial derivatives of any order, and we can easily check that for any \( p \in \mathbb{N} \). The equation

\[ \frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} - \nu \frac{\partial^{2p+1} u}{\partial x^{2p+1}} = 0 \]

is dispersive and the damping coefficient of the mode number \( k \) is

\[ \left( \frac{2\pi k}{L} \right)^{2p} \nu, \]

which implies that higher mode numbers are damped even faster for larger \( p \). The equation

\[ \frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} + \nu \frac{\partial^{2p+1} u}{\partial x^{2p+1}} = 0 \]

is dispersive and the new phase velocity for the mode number \( k \) is

\[ a - \left( \frac{2\pi k}{L} \right)^{2p} \nu. \]

**Dominating error for odd and even order numerical schemes**

We noticed in the modified equation that the main error term for the 1st order explicit upwind scheme was a second order partial derivative in \( x \) which has diffusive effects. This is similar for all odd order schemes with the corresponding even partial derivatives appearing as the first correction in the modified equation.

On the other hand, for even order schemes, like the Lax-Wendroff scheme the dominating term in the modified equation corresponds to an odd order partial derivative which induces a dispersive behavior of the errors.

Finally unstable schemes can be stabilized by adding a viscosity coefficient with a small partial derivative of even order (in factor of \( \Delta x \) such that it vanishes when the grid is refined) in the spirit of the additional terms in the modified equation. In particular the Lax-Wendroff scheme can be viewed as stabilized centered scheme.
2.2 The Finite Volume method

2.2.1 The first order Finite Volume schemes

Let us introduce the Finite Volume method on the generic scalar conservation law of the form

\[
\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0. \tag{2.67}
\]

In the case of our linear advection equation, we have \( f(u) = au \).

In the Finite Volume method, the computational domain is divided into cells (intervals in 1D) and the unknown quantity that is numerically computed is the cell average of \( u \) on each cell. Recall that for Finite Differences the unknowns were the point values of \( u \) at the grid points. We need to number the cells. In 1D a convenient way to do it in order to avoid confusion with the grid points, is to assign half integers. Let us denote by

\[
u_{i+\frac{1}{2}}(t) = \frac{1}{x_{i+1} - x_i} \int_{x_i}^{x_{i+1}} u(t, x) \, dx.
\]

The Finite Volume numerical scheme is then obtained by integrating the original equation on each cell of the domain. As for the time scheme there are at least two classical ways, the first is to also integrate in time between \( t_n \) and \( t_{n+1} \) and then use a quadrature formula to compute the integral, for example, the left hand rectangle rule yields a first order explicit formula. The second is to use the method of lines and separate space discretization from time discretization. Then standard ODE discretization schemes can be used. This is what we shall do mostly in this lecture.

So integrating (2.67) on the cell \([x_i, x_{i+1}]\) and dividing by \( \Delta x_{i+\frac{1}{2}} = x_{i+1} - x_i \) yields

\[
\frac{du_{i+\frac{1}{2}}(t)}{dt} + \frac{1}{\Delta x_{i+\frac{1}{2}}} (f(u(t, x_{i+1})) - f(u(t, x_i))) = 0.
\]

Here we see that a second ingredient is needed in order to define the algorithm. We only know the cell averages of \( u \), how do we define the value at \( u \) at the cell interfaces. The simplest scheme, which is first order accurate in space consists in assuming that \( u \) is constant on each cell and thus equal to its cell average. But it is not defined at the cell interface. In order to complete the Finite Volume scheme we need to define a so called numerical flux at each cell interface denoted by \( g_i \) that needs to be consistent with \( f(x) \), i.e. \( g_i = f(u(x_i)) + O(\Delta x^p) \) for some positive \( p \). A numerical flux of order 2 is the centred flux \( g_i = \frac{1}{2}(f(u_{i-\frac{1}{2}}) + f(u_{i+\frac{1}{2}})) \). This yields the following scheme for a uniform \( \Delta x \):

\[
\frac{du_{i+\frac{1}{2}}(t)}{dt} + \frac{(f(u_{i+\frac{1}{2}}) - f(u_{i-\frac{1}{2}}))}{2\Delta x} = 0.
\]

Coupling it with an explicit Euler scheme in time this becomes, and applying it to the linear advection (\( f(u) = au \)) we get

\[
u_{i+\frac{1}{2}}^{n+1} = \frac{\Delta t}{2\Delta x} (u_{i+\frac{1}{2}}^n - u_{i-\frac{1}{2}}^n).
\]

We recognize here the centred Finite Difference scheme shifted to the cell centres. Remember that this scheme is unstable, so that it cannot be used in practice. In order to get stable scheme, we need to introduce the notion of upwinding like for Finite Differences. This can be done very easily in the definition of the numerical flux by simply choosing the value of \( u \) in the upwind cell only to define the numerical flux. We have \( \frac{\partial f(u)}{\partial x} = f'(u) \frac{\partial u}{\partial x} \). This means that locally at each cell interface the direction of the transport is defined by the sign of \( f'(u) \) (in the case of the linear advection...
Let $W$.

Then we take for $\tilde{x}$.

Remark 4. Using the midpoint rule we also recognize here the first order in time and space upwind scheme shifted to the cell centres.

Again, combining the Finite Volume scheme with an upwind flux and an explicit Euler time discretization yields for the linear advection with $a > 0$

$$u^{n+1}_{i+\frac{1}{2}} = u^n_{i+\frac{1}{2}} - \frac{a\Delta t}{\Delta x} (u_{i+\frac{1}{2}} - u_{i-\frac{1}{2}}). \tag{2.69}$$

We also recognize here the first order in time and space upwind scheme shifted to the cell centres.

**Remark 4. Using the midpoint rule**

$$u_{i+\frac{1}{2}} = \frac{1}{\Delta x} \int_{x_i}^{x_{i+1}} u(x) \, dx = u(x_{i+\frac{1}{2}}) + O(\Delta x^2).$$

The we can reinterpret the Finite Volume as a Finite Difference scheme at the cell centres, which explains that we get the same formulas. However this is not true for higher orders, for which Finite Volume and Finite Difference schemes are genuinely different.

### 2.2.2 Higher order schemes

In order to get high order Finite Volume schemes, the idea is to reconstruct polynomials of some given degree from the cell averages that are obtained with the Finite Volume procedure. The main idea for doing this is to construct an interpolation polynomial for the primitive of the polynomial we are looking for.

At time step $t_n$ we know $u^n_{j+\frac{1}{2}}$ known average value of $u^n$ on cell $[x_j, x_{j+1}]$ of length $\Delta x_{j+\frac{1}{2}} = x_{j+1} - x_j$. We want to construct a polynomial $p_m(x)$ of degree $m$ such that

$$\frac{1}{\Delta x_{j+\frac{1}{2}}} \int_{x_j}^{x_{j+1}} p_m(x) \, dx = u^n_{j+\frac{1}{2}}.$$

To this aim we look for $\tilde{p}_m(x)$ such that $\frac{d}{dx} \tilde{p}_m(x) = p_m(x)$. Then

$$\Delta x_{j+\frac{1}{2}} u^n_{j+\frac{1}{2}} = \int_{x_j}^{x_{j+1}} p_m(x) \, dx = \tilde{p}_m(x_{j+1}) - \tilde{p}_m(x_j).$$

Let $W(x) = \int_{x_0}^{x} \tilde{u}^n(x) \, dx$ a primitive of the piecewise constant function $\tilde{u}^n$ with value $u^n_{j+\frac{1}{2}}$ on $[x_j, x_{j+1}]$. Then $W(x_{j+1}) = \sum_{k=1}^{j} h_{k+\frac{1}{2}} u^n_{k+\frac{1}{2}}$ and

$$W(x_{j+1}) - W(x_j) = \Delta x_{j+\frac{1}{2}} u^n_{j+\frac{1}{2}} = \tilde{p}_m(x_{j+1}) - \tilde{p}_m(x_j).$$

Then we take for $\tilde{p}_m$ an interpolating polynomial at points $x_j$ of $W$ so that

$$\frac{1}{\Delta x_{j+\frac{1}{2}}} \int_{x_j}^{x_{j+1}} p_m(x) \, dx = \frac{1}{\Delta x_{j+\frac{1}{2}}} (\tilde{p}_m(x_{j+1}) - \tilde{p}_m(x_j)) = \frac{1}{\Delta x_{j+\frac{1}{2}}} (W(x_{j+1}) - W(x_j)) = u^n_{j+\frac{1}{2}}.$$
There are many ways to choose an interpolating polynomial, one could use spline interpolation or Hermite interpolation, but the simplest and most used choice is to use a Lagrange interpolation polynomial. This being said, a Lagrange interpolating polynomial of degree \( k \) is defined with \( k + 1 \) interpolation points. So we need to use as many values in neighboring cells as needed.

In order to reconstruct a polynomial of a given degree in a given cell there are many possible stencils, i.e. ensembles of cells, that can be used. For the reconstruction of a polynomial of degree \( k \) exactly \( k \) average values corresponding to \( k \) neighboring cells are needed. The only constraint is that the value on the cell where the polynomial being reconstructed is used. High-order methods are prone to oscillations especially around discontinuities. So one good idea is to use the stencil which minimizes the oscillations. This can be easily done by choosing automatically the stencil based on the Newton divided differences which can be used to construct the interpolating polynomial. This method is called ENO (Essentially Non Oscillatory).

The high order scheme is obtained by using a high order approximation of the numerical flux obtained by evaluating the reconstructed polynomial \( p_{m,i+\frac{1}{2}} \) at the cell interface. The reconstruction is done cell by cell. This means in particular, when computing the numerical flux \( g_i \) that \( p_{m,i+\frac{1}{2}}(x_i) \neq p_{m,i-\frac{1}{2}}(x_i) \), where \( p_{m,i+\frac{1}{2}} \) is the reconstructed polynomial on the cell \([x_i, x_{i+1}]\). The difference is of the order of the scheme. The centered flux \( g_i \) is then defined as the average of \( p_{m,i+\frac{1}{2}}(x_i) \) and \( p_{m,i-\frac{1}{2}}(x_i) \) and the upwind flux as the value coming from the upwind cell.

**Newton’s divided differences interpolation formula** Consider a function \( f \) for which we want to construct an interpolation polynomial of degree \( k \) such at the points \((x_i, f(x_i))\), \( 0 \leq i \leq k \). Newton’s divided differences are defined recursively as follows:

\[
\begin{align*}
  f[x_0] &= f(x_0), f[x_0, x_1] = \frac{f[x_1] - f[x_0]}{x_1 - x_0}, \ldots, \\
  f[x_0, \ldots, x_k] &= \frac{f[x_1, \ldots, x_k] - f[x_0, \ldots, x_{k-1}]}{x_k - x_0} 
\end{align*}
\] (2.70)

The Lagrange interpolation polynomial of \( f \) at the points \( x_0, \ldots, x_k \) can be defined using the divided differences:

\[
p_k(x) = f[x_0] + (x - x_0)f[x_0, x_1] + \cdots + (x - x_0)\cdots(x - x_k)f[x_0, \ldots, x_k]
\]

In the ENO method the stencil for the reconstruction polynomial \( \tilde{p}_m \) is then constructed using the following algorithm:

- start with constant value \( W(x_i) = 0 \), \( W[x_i, x_{i+1}] = u_{i+\frac{1}{2}}^{n} \)
- compute \( W[x_{i-1}, x_i, x_{i+1}] \) and \( W[x_i, x_{i+1}, x_{i+2}] \)
  - Extend stencil with \( x_{i-1} \) if \( |W[x_{i-1}, x_i]| < |W[x_i, x_{i+1}]| \)
  - Extend stencil with \( x_{i+1} \) else
- proceed in same way with next divided difference

See also [7] for a detailed description of the ENO method.

The ENO method can be still improved by taking all possible stencils but putting a weight on each of the polynomials obtained. This is called the WENO method (Weighted Essentially Non Oscillatory). A good review of this technique is given in [8].
2.3 The Finite Element method

The Finite Element method is mathematically more involved than the Finite Difference and Finite Volume methods. The discretization idea is based on the Galerkin principle of approximating the function space in which the solution is defined by a finite dimensional subspace. It uses also a weak (or variational form of the equation). In order to obtain the weak form, the idea is to multiply by a smooth test function and integrate over the whole domain, with a possible integration by parts to eliminate the highest derivatives.

Let us describe it on the advection-diffusion problem (assuming periodic boundary conditions on the domain $[0, L]$):

$$
\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} - \nu \frac{\partial^2 u}{\partial x^2} = 0.
$$

Multiplying by a test function $v$ which does not depend on $t$ and integrating, with an integration by parts in the last integral and periodic boundary conditions, yields

$$
\frac{d}{dt} \int_0^L uv \, dx + a \int_0^L \frac{\partial u}{\partial x} v \, dx + \nu \int_0^L \frac{\partial u}{\partial x} \frac{\partial v}{\partial x} \, dx = 0.
$$

The natural space in which this formulation is defined is

$$
H^1([0, L]) = \{ u \in L^2([0, L]) \mid \frac{\partial u}{\partial x} \in L^2([0, L]) \}.
$$

The variational problem thus reads

Find $u \in H^1$ such that

$$
\frac{d}{dt} \int_0^L uv \, dx + a \int_0^L \frac{\partial u}{\partial x} v \, dx + \nu \int_0^L \frac{\partial u}{\partial x} \frac{\partial v}{\partial x} \, dx = 0 \quad \forall v \in H^1([0, L]).
$$

Now in order to define a Finite Element approximation, we need to construct a sequence of subspaces $V_h$ of $H^1([0, L])$ which is dense in $H^1([0, L])$ (in order to get convergence). One subspace is of course enough to compute a numerical approximation. There are many ways to construct such subspaces.

The classical Lagrange Finite Element method consists in building a mesh of the computational domain and to assume that the approximating function is polynomial say of degree $k$ in each cell.

For the piecewise polynomial function space to be a included in $H^1([0, L])$ the only additional requirement is that the functions are continuous at the cell interfaces. So the subspace $V_h$ on the mesh $x_0 < x_1 < \cdots < x_{N-1}$ is defined by

$$
V_h = \{ v_h \in C^0([a, b]) \mid v_h|_{[x_i, x_{i+1}]} \in \mathbb{P}_k([x_i, x_{i+1}]) \}.
$$

In order to express a function $v_h \in V_h$ we need a basis of $V_h$. This can be constructed easily combining bases of $\mathbb{P}_k([x_i, x_{i+1}])$. In order to impose the continuity requirement at the cell interface, the simplest is to use a Lagrange basis $\mathbb{P}_k([x_i, x_{i+1}])$ with interpolation points on the edge of the intervals. Given $k + 1$ interpolation points $x_i = y_0 < y_1 < \cdots < y_k = x_{i+1}$ the Lagrange basis functions of degree $k$ denoted by $\ell_{k,i}$, $0 \leq i \leq k$, are the unique polynomials of degree $k$ verifying $\ell_{k,i}(y_j) = \delta_{i,j}$. Because of this property, any polynomial $p(x) \in \mathbb{P}_k([x_i, x_{i+1}])$ can be expressed as $p(x) = \sum_{j=0}^k p(y_j) \ell_{k,j}(x)$ and conversely any polynomial $p(x) \in \mathbb{P}_k([x_i, x_{i+1}])$ is uniquely determined by its values at the interpolation points $y_j$, $0 \leq j \leq k$. Hence in order to ensure the continuity of the piecewise polynomial at the cell interface $x_i$ it is enough that the values of the polynomials on both sides of $x_i$ have the same value at $x_i$. This constraint removes one degree of freedom in each cell, so that the total dimension of $V_h$ is $Nk$ and the functions of $V_h$ are uniquely defined in each cell by their value at the interpolation points in all the cells. The basis functions denoted of $V_h$ denoted by $(\varphi_i)_{0 \leq j \leq Nk}$ are such that their restriction on each cell is a Lagrange basis function.
Note that for \( k = 1 \), corresponding to \( P_1 \) finite elements, the degrees of freedom are just the values at the grid points. For higher order finite elements internal degrees of freedom are needed. For stability and convenience issues these are most commonly taken to be the Gauss-Lobatto points on each cell. The basis functions for \( P_3 \) elements on a mesh of two cells is represented in Figure 2.8 and the positions of the Gauss-Lobatto points and their weights are given in Table 2.1 for \( 1 \leq k \leq 4 \).

![Figure 2.8: Graphical representation of the \( P_3 \) basis functions on 2 elements.](image)

<table>
<thead>
<tr>
<th>( k )</th>
<th>( x_{GL,0} )</th>
<th>( x_{GL,1} )</th>
<th>( x_{GL,2} )</th>
<th>( x_{GL,3} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>-1</td>
<td>-\sqrt{\frac{1}{5}}</td>
<td>\sqrt{\frac{1}{5}}</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>-1</td>
<td>-\sqrt{\frac{3}{7}}</td>
<td>0</td>
<td>\sqrt{\frac{3}{7}}</td>
</tr>
</tbody>
</table>

Table 2.1: Gauss-Lobatto points in interval \([-1, 1]\)

In order to obtain a discrete problem that can be solved on a computer, the Galerkin procedure consist in replacing \( H^1 \) by \( V_h \) in the variational formulation. The discrete variational problem thus reads: Find \( u_h \in V_h \) such that

\[
\frac{d}{dt} \int_0^L u_h v_h \, dx + \int_0^L \frac{\partial u_h}{\partial x} v_h \, dx + \nu \int_0^L \frac{\partial u_h}{\partial x} \frac{\partial v_h}{\partial x} \, dx = 0 \quad \forall v_h \in V_h.
\]

Now expressing \( u_h \) (and \( v_h \)) in the basis of \( V_h \) as \( u_h(t, x) = \sum_{j=1}^{N_k-1} u_j(t) \varphi_j(x), v_h(x) = \sum_{j=1}^{N_k-1} v_j(t) \varphi_j(x) \) and plugging these expressions into the variational formulation, denoting by \( U = (u_0, u_1, \cdots, u_{N_k-1})^\top \) and similarly for \( V \) yields: Find \( U \in \mathbb{R}^{N_k} \) such that

\[
\frac{d}{dt} \sum_{i,j} u_{ij}v_i \int_0^L \varphi_i(x) \varphi_j(x) \, dx + a \sum_{i,j} u_{ij}v_i \int_0^L \frac{\partial \varphi_j(x)}{\partial x} \varphi_i(x) \, dx
\]

\[
\quad + \nu \sum_{i,j} u_{ij}v_i \int_0^L \frac{\partial \varphi_i(x)}{\partial x} \frac{\partial \varphi_j(x)}{\partial x} \, dx = 0 \quad \forall V \in \mathbb{R}^{N_k},
\]
which can be expressed in matrix form
\[ V^T(M \frac{dU}{dt} + DU + AU) = 0 \quad \forall V \in \mathbb{R}^{Nk}, \]
which is equivalent to
\[ M \frac{dU}{dt} + DU + AU = 0 \]
where the square \( Nk \times Nk \) matrices are defined by
\[ M = \left( \int_0^L \varphi_j(x) \varphi_i(x) \, dx \right)_{i,j}, \quad D = \left( \int_0^L \frac{\partial \varphi_j(x)}{\partial x} \varphi_i(x) \, dx \right)_{i,j}, \]
\[ A = \left( \int_0^L \frac{\partial \varphi_i(x)}{\partial x} \frac{\partial \varphi_j(x)}{\partial x} \, dx \right)_{i,j}. \] (2.71)

Note that these matrices can be computed exactly as they involve integration of polynomials on each cell. Moreover because the Gauss-Lobatto quadrature rule is exact for polynomials of degree up to \( 2k - 1 \), both \( A \) and \( D \) can be computed exactly with the Gauss-Lobatto quadrature rule. Moreover, approximating the mass matrix \( M \) with the Gauss-Lobatto rule introduces an error which does not decrease the order of accuracy of the scheme [2] and has the big advantage of yielding a diagonal matrix. This is what is mostly done in practice.

**Matrix Assembly.** Usually for Finite Elements the matrices \( M, D \) and \( A \) are computed from the corresponding elementary matrices which are obtained by change of variables onto the reference element \([-1,1]\) for each cell. So
\[ \int_0^L \varphi_i(x) \varphi_j(x) \, dx = \sum_{\nu=0}^{N-1} \int_{x_{\nu}}^{x_{\nu+1}} \varphi_i(x) \varphi_j(x) \, dx, \]
and performing the change of variable \( x = \frac{x_{\nu+1}-x_{\nu}}{2} \hat{x} + \frac{x_{\nu+1}+x_{\nu}}{2} \), we get
\[ \int_{x_{\nu}}^{x_{\nu+1}} \varphi_i(x) \varphi_j(x) \, dx = \frac{x_{\nu+1} - x_{\nu}}{2} \int_{-1}^{1} \hat{\ell}_\alpha(\hat{x}) \hat{\ell}_\beta(\hat{x}) \, d\hat{x}, \]
where \( \hat{\ell}_\alpha(\hat{x}) = \varphi_i(\frac{2x_{\nu+1}-x_{\nu}}{2} \hat{x} + \frac{x_{\nu+1}+x_{\nu}}{2}) \). The local indices \( \alpha \) on the reference element go from 0 to \( k \) and the global numbers of the basis functions not vanishing on element \( \nu \) are \( j = k\nu + \alpha \). The \( \hat{\ell}_\alpha \) are the Lagrange polynomials at the Gauss-Lobatto points in the interval \([-1,1]\).

The mass matrix in \( V_h \) can be approximated with no loss of order of the finite element approximation using the Gauss-Lobatto quadrature rule. Then because the products \( \hat{\ell}_\alpha(\hat{x}) \hat{\ell}_\beta(\hat{x}) \) vanish for \( \alpha \neq \beta \) when \( \hat{x} \) is a Gauss-Lobatto point by definition of the \( \hat{\ell}_\alpha \) which are the Lagrange basis functions at the Gauss-Lobatto points, the elementary matrix \( M \) is diagonal and we have
\[ \int_{-1}^{1} \hat{\ell}_\alpha(\hat{x})^2 \, d\hat{x} \approx \sum_{\beta=0}^{k} w_{\beta}^{GL} \varphi_\alpha(\hat{x}_\beta)^2 = w_{\alpha}^{GL} \]
using the quadrature rule, where \( w_{\alpha}^{GL} \) is the Gauss-Lobatto weight at Gauss-Lobatto point \( (\hat{x}_\alpha) \in [-1,1] \). So that finally \( M = \text{diag}(w_0^{GL}, \ldots, w_k^{GL}) \) is the matrix with \( k+1 \) lines and columns with the Gauss-Lobatto weights on the diagonal.

Let us now compute the elements of \( D \). As previously we go back to the interval \([-1,1]\) with the change of variables \( x = \frac{x_{\nu+1}-x_{\nu}}{2} \hat{x} + \frac{x_{\nu+1}+x_{\nu}}{2} \) and we define \( \hat{\ell}_\alpha(\hat{x}) = \varphi_i(\frac{2x_{\nu+1}-x_{\nu}}{2} \hat{x} + \frac{x_{\nu+1}+x_{\nu}}{2}) \).
Note that a global basis function $\varphi_i$ associated to a grid point has a support which overlaps two cells and is associated to two local basis functions. Thus one needs to be careful to add the two contributions as needed in the final matrix.

We get $\hat{\ell}_\alpha(x) = \frac{x+1-x}{2} \varphi_i'\left(\frac{x+1-x}{2}(x+1) + x\right)$. It follows that

$$
\int_{x_{\nu+1}}^{x_{\nu+1}} \varphi_j'(x)\varphi_i(x) \, dx = \int_{-1}^{1} \frac{2}{x_{\nu+1} - x_{\nu}} \hat{\ell}_\beta(x)\hat{\ell}_\alpha(x) \frac{x_{\nu+1} - x_{\nu}}{2} \, dx = \int_{-1}^{1} \hat{\ell}_\beta(x)\hat{\ell}_\alpha(x) \, dx.
$$

The polynomial $\hat{\ell}_\alpha(x)$ is of degree $k$ so that $\hat{\ell}_\beta(x)$ is of degree $k-1$ so that the Gauss-Lobatto quadrature rule with $k+1$ points is exact for the product which is of order $2k-1$. Using this rule

$$
\int_{-1}^{1} \hat{\ell}_\beta(x)\hat{\ell}_\alpha(x) \, dx = \sum_{m=0}^{k} w_m^{GL} \hat{\ell}_\beta(x_m)\hat{\ell}_\alpha(x_m) = w_m^{GL} \hat{\ell}_\beta(x_m).
$$

As before, because $\hat{\ell}_\alpha$ are the Lagrange polynomials at the Gauss-Lobatto points, only the value at $x_\alpha$ in the sum is one and the others are 0. On the other hand evaluating the derivatives of the Lagrange polynomial at the Gauss-Lobatto points can be done using the formula

$$
\hat{\ell}_\alpha'(x_\beta) = \frac{p_\beta/p_\alpha}{x_\beta - x_\alpha} \text{ for } \beta \neq \alpha \text{ and } \hat{\ell}_\alpha'(x_\alpha) = -\sum_{\beta \neq \alpha} \hat{\ell}_\beta(x_\alpha),
$$

where $p_\alpha = \prod_{\beta \neq \alpha} (x_\alpha - x_\beta)$. This formula is obtained straightforwardly by taking the derivative of the explicit formula for the Lagrange polynomial

$$
\hat{\ell}_\alpha(x) = \prod_{\beta \neq \alpha} \frac{(x - x_\beta)}{(x_\alpha - x_\beta)}
$$

and using this expression at the Gauss-Lobatto point $x_\beta \neq x_\alpha$. We refer to [1] for a detailed description.

We can now conclude with the computation of the stiffness matrix $A$. Having already computed the expression of the change of variable of the derivatives we can quickly go to the result. We have in each element

$$
\int_{x_{\nu+1}}^{x_{\nu+1}} \varphi_j'(x)\varphi_i(x) \, dx = \int_{-1}^{1} \left(\frac{2}{x_{\nu+1} - x_{\nu}}\right)^2 \hat{\ell}_\beta(x)\hat{\ell}_\alpha(x) \frac{x_{\nu+1} - x_{\nu}}{2} \, dx = \frac{2}{x_{\nu+1} - x_{\nu}} \int_{-1}^{1} \hat{\ell}_\beta(x)\hat{\ell}_\alpha(x) \, dx = \frac{2}{x_{\nu+1} - x_{\nu}} \sum_{m=0}^{k} w_m^{GL} \hat{\ell}_\beta(x_m)\hat{\ell}_\alpha(x_m).
$$

As the polynomial being integrated is of degree $2(k-1) = 2k-2$ the Gauss-Lobatto quadrature is exact. Here no simplifications occurs in the sum, which has to be computed. Still the expressions of the derivatives at the Gauss-Lobatto points computed above can then be plugged in.

**Time advance and stability.** At the end we get as for the finite difference and finite volume methods a system of differential equations that can be solved with any ODE solver. Let us make a few remarks concerning the stability of the scheme (once discretized in time). As we saw previously this depends on whether the eigenvalues of the matrix $L$ is included in the stability zone of the ODE solver. Here $L = -M^{-1}(D+A)$. Note that the matrices $M$ and $A$ are obviously symmetric and thus have only real eigenvalues. On the other hand, for periodic boundary conditions and integration by
parts, yields that \( \int \frac{\partial \varphi_j(x)}{\partial x} \varphi_i(x) \, dx = - \int \varphi_j(x) \frac{\partial \varphi_i(x)}{\partial x} \, dx \). Hence \( D \) is skew symmetric and has only imaginary eigenvalues. Remember that the stability zones of our explicit ODE solvers lie mostly on the left-hand side of the imaginary axis in the complex plane, and that only the third and fourth order schemes have a stability zone on the imaginary axis. Hence in the pure advection case \( \nu = 0 \) \( \mathcal{L} \) has purely imaginary eigenvalues and the order one and two time schemes are always unstable. In order to stabilize the method a procedure that is often used with a finite element discretization of a pure advection problem is to add a diffusive term that goes to 0 with the cell size, i.e. take \( \nu = \alpha \Delta x \), in this case a small negative real part is added to the eigenvalues which are thus pushed into the left half of the complex plane and the stability zone is enhanced.

Stabilizing a Finite Element scheme is quite a bit more involved than straightforward upwinding as is done in Finite Volume schemes, even though good solutions exist, in particular by using a Petrov-Galerkin method, which in contrast to the classical Galerkin method uses two different spaces for the trial and test functions.

### 2.4 The Discontinuous Galerkin (DG) method

The DG method represents the unknowns like the Finite Element method by piecewise polynomial functions, but unlike Finite Element methods the polynomials are discontinuous at the cell interfaces and a numerical flux is defined at the cell interface in the same way as for Finite Volume methods.

So on each cell the discrete unknown \( u_h \) is represented as a linear combination of well chosen basis functions of the space of polynomials of degree \( k \) \( \mathbb{P}_k \). The dimension of this space is \( k + 1 \).

As no continuity is enforced at the element interface, there is no constraint on the basis functions and generally two kinds of basis functions are used: either the Legendre polynomials which form an orthogonal basis, we then speak of modal DG or one can use a Lagrange basis defined on a set of interpolation points within the cell we then speak of nodal DG. The interpolation points are generally chosen to be either the Gauss points or the Gauss-Lobatto points which are both very convenient as they allow to express the integrals appearing in the formulation exactly (for Gauss) or very accurately without loss of order in the DG method (for Gauss-Lobatto) using the associated quadrature rule.

For the derivation of a DG scheme, the equation is multiplied by a polynomial test function on each cell and integration by parts is used so that a boundary term appears which will allow the coupling between two neighboring cells. Let us apply it here to the conservation law

\[
\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0.
\]

We then get

\[
\frac{d}{dt} \int_{x_\nu}^{x_{\nu+1}} u v \, dx + \int_{x_\nu}^{x_{\nu+1}} \frac{\partial f(u)}{\partial x} v \, dx
= \frac{d}{dt} \int_{x_\nu}^{x_{\nu+1}} u v \, dx - \int_{x_\nu}^{x_{\nu+1}} f(u) \frac{\partial v}{\partial x} \, dx + (f(u(x_{\nu+1}))v(x_{\nu+1}) - f(u(x_\nu))v(x_\nu)) = 0. \tag{2.72}
\]

The DG method has then two key ingredients.

1. Choose on each cell a finite dimensional representation, usually by a polynomial as said previously.

2. Define a unique numerical flux denoted by \( g_\nu = g(u_L(x_\nu), u_R(x_\nu)) \) at the cell interface which is constructed from the two values coming from the cells sharing the interface on the left and on the right. Indeed the approximation of \( u \) being discontinuous at the cell interface, the values \( u(x_\nu) \) and \( f(u(x_\nu)) \) are not defined in a natural way and ingredient of the scheme is to approximate the flux \( f(u(x_\nu)) \) by the numerical flux \( g_\nu \)
At the interface between two cells \( x_\nu \), the Discontinuous Galerkin approximation provides two values of \( u(x_\nu) \), \( u_L \) coming from the approximation of \( u \) on the left of \( x_\nu \) and \( u_R \) corresponding to the value at \( x_\nu \) from the right-hand cell. The numerical flux at each cell interface \( g_\nu \) needs to be consistent with \( f(x_\nu) \), i.e. \( g_\nu = f(u(x_\nu)) + O(\Delta x^p) \) for some positive integer \( p \). A numerical flux of order 2 is the centred flux \( g_\nu = \frac{1}{2} (f(u_L) + f(u_R)) \). The centred flux amounts to projecting the discontinuous approximation to a continuous Finite Element basis and with yield a skew symmetric derivative matrix. Thus this scheme is unstable for explicit time discretizations of order 1 and 2. In order to get stable scheme in this case, we need to introduce the notion of unwinding like for Finite Differences or Finite Volumes. This can be done very easily in the definition of the numerical flux by simply choosing the value of \( u \) in the upwind cell only to define the numerical flux. We have \( \frac{df(u)}{dx} = f'(u) \frac{d u}{d x} \). This means that locally at each cell interface the direction of the transport is defined by the sign of \( f'(u) \) (in the case of the linear advection \( f'(u) = a \) and the upwind direction is determined by the sign of \( a \)). So the upwind numerical flux is defined by

\[
g_\nu = g(u_L(x_\nu), u_R(x_\nu)) = \begin{cases} 
    f(u_L) & \text{if } f'(\frac{u_L+u_R}{2}) \geq 0, \\
    f(u_R) & \text{if } f'(\frac{u_L+u_R}{2}) < 0.
\end{cases}
\]

Figure 2.9: Representation of the \( P_3 \) basis functions on reference element \([-1, 1]\) for Gauss points (left) and Gauss-Lobatto points (right).

Choosing as local representation for \( u \) and the test function \( v \) the Lagrange polynomials at the Gauss-Lobatto points simplifies the computation of the fluxes, as in this case only the Lagrange polynomial associated to the edge node does not vanish at the edge. This situation is different when using Legendre polynomials or Lagrange polynomials at only interior nodes (like the Gauss points). Note however that Legendre polynomials have the advantage of having exactly a diagonal mass matrix. This is obtained also with Lagrange polynomials at the Gauss-Lobatto points but in this case at the price of a small quadrature error.

As opposite to the Finite Element method, only local matrices on each element, in practice only the elementary matrices on the \([-1, 1]\) reference interval need to be assembled. The elementary mass matrix \( \hat{M} \) on cell on the reference interval has the components

\[
\hat{M}_{\alpha, \beta} = \int_{-1}^{1} \hat{\ell}_\alpha(\hat{x}) \hat{\ell}_\beta(\hat{x}) \, d\hat{x}, \quad 0 \leq \alpha, \beta \leq k.
\]

When the basis functions are the Legendre polynomials which form an orthonormal basis.

The mass matrix in \( V_h \) can be approximated with no loss of order of the finite element approximation using the Gauss-Lobatto quadrature rule. Then because the products \( \hat{\ell}_\alpha(\hat{x}) \hat{\ell}_\beta(\hat{x}) \) vanish for \( \alpha \neq \beta \) at the Gauss-Lobatto points by definition of the \( \hat{\ell}_\alpha \) which are the Lagrange basis functions.
at these points, the elementary matrix $M$ is diagonal and we have

$$
\int_{-1}^{1} \ell_\alpha(\hat{x})^2 \, d\hat{x} \approx \sum_{\beta=0}^{k} w^{GL}_\beta \varphi_\beta(\hat{x}_\beta)^2 = \nu^{GL}_\alpha
$$

using the quadrature rule, where $w^{GL}_\beta$ is the Gauss-Lobatto weight at Gauss-Lobatto point $(\hat{x}_\alpha) \in [-1, 1]$. So that finally $\hat{M} = \text{diag}(w^{GL}_0, \ldots, w^{GL}_k)$ is the matrix with $k + 1$ lines and columns with the Gauss-Lobatto weights on the diagonal. From this matrix, the local mass matrix $M_{\nu+\frac{1}{2}}$ on cell $[x_\nu, x_{\nu+1}]$ can be expressed as

$$
M_{\nu+\frac{1}{2}} = \frac{x_{\nu+1} - x_\nu}{2} \hat{M}.
$$

Let us denote by $K_{\nu+\frac{1}{2}}$ the local matrix containing the derivative of $v$. In order to get an expression for the components of $K_{\nu+\frac{1}{2}}$, let us introduce the local basis functions and compute using again the affine change of variable to the reference interval $[-1, 1]:$

$$
\int_{x_\nu}^{x_{\nu+1}} f(u) \frac{\partial \varphi_i}{\partial x} \, dx = \int_{-1}^{1} f(u(x)) \frac{2}{x_{\nu+1} - x_\nu} \frac{x_{\nu+1} - x_\nu}{2} \, dx = \int_{-1}^{1} f(u(x)) \ell_\alpha(\hat{x}) \, d\hat{x}.
$$

Then using the Gauss-Lobatto quadrature rule this becomes

$$
\int_{-1}^{1} f(u(\hat{x})) \ell_\alpha(\hat{x}) \, d\hat{x} \approx \sum_{\beta=0}^{k} w^{GL}_\beta f(u(\hat{x}_\beta)) \ell_\alpha(\hat{x}_\beta) = \sum_{\beta=0}^{k} w^{GL}_\beta f(u_\beta) \ell_\alpha(\hat{x}_\beta),
$$

where $u_\beta = u(\hat{x}_\beta)$ is the $\beta$th component of $u$ on the Lagrange basis. Denoting $U_{\nu+\frac{1}{2}} = (u_0, u_1, \ldots, u_k)$ on the cell $[x_\nu, x_{\nu+1}]$, thus defining the component of matrix $K_{\nu+\frac{1}{2}}$ at line $\alpha$ and column $\beta$ as being $w^{GL}_\beta \ell_\alpha(\hat{x}_\beta)$, we get that

$$
\int_{-1}^{1} f(u(\hat{x})) \ell_\alpha(\hat{x}) \, d\hat{x} \approx \sum_{\beta=0}^{k} w^{GL}_\beta f(u(\hat{x}_\beta)) \ell_\alpha(\hat{x}_\beta) = (K_{\nu+\frac{1}{2}} f(U_{\nu+\frac{1}{2}}))_\alpha,
$$

where $f(U_{\nu+\frac{1}{2}}) = (f(u_0), f(u_1), \ldots, f(u_k))$.

**Remark 5.** The Gauss-Lobatto quadrature is exact for $f(u) = au$ for some given constant $a$, we notice that the matrix $K_{\nu+\frac{1}{2}}$ is exactly the matrix associated to the components

$$
\int_{x_\nu}^{x_{\nu+1}} \varphi_j(x) \frac{\partial \varphi_i}{\partial x} \, dx = \int_{-1}^{1} \varphi_\beta(\hat{x}) \ell_\alpha(\hat{x}) \, d\hat{x} = w^{GL}_\beta \ell_\alpha(\hat{x}_\beta).
$$

We also notice that this matrix does not depend on the specific interval and is equal to the matrix on the reference element $\hat{K} = K_{\nu+\frac{1}{2}}$ for all $\nu$.

Now plugging all this into the formula (2.72) we get on each cell

$$
V^T_{\nu+\frac{1}{2}} M_{\nu+\frac{1}{2}} \frac{dU_{\nu+\frac{1}{2}}}{dt} = V^T_{\nu+\frac{1}{2}} \hat{K} f(U_{\nu+\frac{1}{2}}) - (g_{\nu+1} v_k - g_\nu v_0).
$$

Then introducing the vector $G_{\nu+\frac{1}{2}} \in \mathbb{R}^k$ whose only non zero components are the first which is $g_\nu$ and the last which is $g_{\nu+1}$, we get the following system of ODE

$$
\frac{x_{\nu+1} - x_\nu}{2} \hat{M} \frac{dU_{\nu+\frac{1}{2}}}{dt} = \hat{K} f(U_{\nu+\frac{1}{2}}) + G_{\nu+\frac{1}{2}}.
$$
The numerical flux $g_{i} \nu$ depends on values of $u$ coming from the neighboring cell, this is where the coupling between the cells takes place. The matrix $\hat{M}$ being diagonal there is no linear system to solve. Simple examples of fluxes in the linear case $f(u) = au$ are the same as for the finite volume method with the centred or upwind fluxes, the two values being used here are the values of $u$ on the interface coming from the two cells sharing the interface, this will be the local value of $u_k$ from the left cell and the local value of $u_0$ from the right cell.
Chapter 3

Linear hyperbolic systems

Let us first consider linear systems of conservation laws in 1D. This can be written in the general form

$$\frac{\partial U}{\partial t} + A \frac{\partial U}{\partial x} = 0,$$

where $U(t, x)$ is a vector in $\mathbb{R}^n$ and $A$ a given matrix with constant coefficients. Our aim is to derive a numerical scheme for this system. We will focus in the following on the Discontinuous Galerkin method, which includes the first order Finite Volume method when polynomials of degree 0, i.e. constants are taken on each cell. For higher order finite volumes, high order polynomial approximations of the unknowns can be reconstructed locally by proceeding component by component like in the scalar case.

### 3.1 The Riemann problem

The main numerical issue when constructing a Finite Volume or Discontinuous Galerkin scheme is to find a good numerical flux that is consistent (i.e. converges towards the exact flux when the cell size goes to 0) and stable. As we saw previously in the linear scalar case enhanced stability is given by upwinding. We now need to generalize the notion of upwinding to the case of systems.

The construction of a numerical flux is a local procedure at the interface between two cells, where a different value is given on the left hand side and on the right hand side from the polynomial approximation (or reconstruction for Finite Volumes). In order to get information from the equation itself the idea is to solve it locally in the neighborhood of the interface using an initial condition which is a step function. The corresponding initial value problem is called the Riemann problem

$$\frac{\partial U}{\partial t} + A \frac{\partial U}{\partial x} = 0,$$

$$U(0, x) = \begin{cases} U_L & \text{if } x < 0, \\ U_R & \text{if } x \geq 0, \end{cases}$$

where $U_L$ and $U_R$ are two given constant vectors in $\mathbb{R}^n$.

The system being hyperbolic implies that $A$ has real eigenvalues and can be diagonalized. Hence $A = \Lambda P^{-1}$, where $\Lambda$ is the diagonal matrix containing the eigenvalues. Then introducing the so-called characteristic variables $V = P^{-1}U$, and multiplying the system by $P^{-1}$ on the left we get

$$P^{-1} \frac{\partial U}{\partial t} + P^{-1} A P P^{-1} \frac{\partial U}{\partial x} = \frac{\partial V}{\partial t} + \Lambda \frac{\partial V}{\partial x} = 0.$$

So in the variable $V$ the system is diagonal and reduces to the set of linear advection equations

$$\frac{\partial v_i}{\partial t} + \lambda_i \frac{\partial v_i}{\partial x} = 0, \quad 1 \leq i \leq n$$
where the \( v_i \) are the components of \( V \) and the \( \lambda_i \) the eigenvalues of \( A \). The exact solution of these

\[
v_i(t, x) = \begin{cases} v_{i,L} & \text{if } x < \lambda_i t, \\ v_{i,R} & \text{if } x \geq \lambda_i t. \end{cases}
\]

In practice we want to use the Riemann problem to determine the value of \( V \) (and \( U \)) at the cell interface, corresponding to \( x = 0 \), the discontinuity point at any strictly positive time. And we deduce from the previous solution that

\[
v_i(t, 0) = \begin{cases} v_{i,L} & \text{if } 0 < \lambda_i, \\ v_{i,R} & \text{if } 0 \geq \lambda_i. \end{cases}
\]

In order to get a vector expression, we introduce the diagonal matrices \( \Lambda^+ \) where the negative eigenvalues are replaced by 0 and \( \Lambda^- \) where the positive eigenvalues are replaced by 0. Obviously

\[
\Lambda = \Lambda^+ + \Lambda^-.
\]

Then for \( t > 0 \) we have

\[
\Lambda V(t, 0) = \Lambda^+ V(t, 0) + \Lambda^- V(t, 0) = \Lambda^+ V_L + \Lambda^- V_R,
\]

as for all positive eigenvalues the corresponding component of \( V(t, 0) \) is \( v_{i,L} \) and for all negative eigenvalues the corresponding component of \( V(t, 0) \) is \( v_{i,R} \). Note that as \( V(t, 0) \) is multiplied by \( \Lambda \) the components of \( V(t, 0) \) corresponding to 0 eigenvalues do not need to be considered as they are multiplied by 0 anyway. So the side where the strict inequality is used for the initial condition of the Riemann problem plays no role.

Denoting by \( A^+ = P \Lambda^+ P^{-1} \) and \( A^- = P \Lambda^- P^{-1} \) the flux \( AU(t, 0) \) associated to the solution of the Riemann problem at the cell interface can also be expressed conveniently directly in terms of \( U \)

\[
AU(t, 0) = P \Lambda^+ V(t, 0) + P \Lambda^- V(t, 0) = P \Lambda^+ V_L + P \Lambda^- V_R = A^+ U_L + A^- U_R.
\]

This expression \( AU(t, 0) = A^+ U_L + A^- U_R \) can be used to define the numerical flux at the cell interface, using the value \( U_L \) coming from the left-hand side of the interface and \( U_R \) coming from the right-hand side of the interface. For actual computations, the matrices \( A^+ \) and \( A^- \) need to be computed explicitly from the eigenvalues and eigenvectors of the matrix \( A \). Notice that in the case of a scalar equation the matrix \( A \) is reduced to the scalar \( a \) which is then obviously the only eigenvalue of the \( 1 \times 1 \) matrix and if \( a > 0 \) we have \( A^+ = a \) and \( A^- = 0 \), so that the numerical flux becomes \( au(t, 0) = au_L \) and the same way if \( a < 0 \) \( au(t, 0) = au_R \), so that the numerical flux obtained from the solution of the Riemann problem reduces to the upwind flux.

**Example.** We consider the 1D Maxwell equations which can be written in dimensionless units:

\[
\begin{align*}
\frac{\partial E}{\partial t} + \frac{\partial B}{\partial x} &= 0, \\
\frac{\partial B}{\partial t} + \frac{\partial E}{\partial x} &= 0.
\end{align*}
\]

This can be written in the form of a linear system

\[
\frac{\partial U}{\partial t} + A \frac{\partial U}{\partial x} = 0, \quad \text{with } U = \begin{pmatrix} E \\ B \end{pmatrix}, \quad A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.
\]
The eigenvalues of $A$ are the solutions of $\det(A - \lambda I) = 0$, i.e. $\lambda^2 = 1$. So the eigenvalues are $\lambda_1 = -1$ and $\lambda_2 = 1$. They are real and distinct so that the system is strictly hyperbolic. Let $V_i$ be a normalized eigenvector associated to the eigenvalue $\lambda_i$, $i = 1, 2$. We have $AV_1 = -V_1$ so that $V_1 = \frac{1}{\sqrt{2}}(1, -1)^\top$ and $AV_2 = V_2$ so that $V_2 = \frac{1}{\sqrt{2}}(1, 1)^\top$. We define $P$ the matrix whose columns are $V_1$ and $V_2$. $P$ is obviously orthonormal, so that its inverse is its transpose. Then we have $PA = \Lambda P$.

So that we can define:

$$A_+ = PA_+P^\top = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix},$$

$$A_- = PA_-P^\top = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix}$$

Hence, the upwind flux is given by

$$AU(t, 0) = A_+U_L + A_-U_R = \frac{1}{2} \begin{pmatrix} U_{L,1} + U_{L,2} + (-U_{R,1} + U_{R,2}) \\ U_{L,1} + U_{L,2} + (U_{R,1} - U_{R,2}) \end{pmatrix}.$$

**Remark 6.** As for the scalar problem, the numerical flux can be taken as a linear combination of the centred flux and the upwind flux (solution the Riemann problem):

$$G_j = \frac{1}{2}A(U_L + U_R) + (1 - \mu)(A_+U_L + A_-U_R), \quad 0 \leq \mu \leq 1$$

corresponding to partial upwinding. The pure upwind flux is obtained for $\mu = 0$ and the centered flux for $\mu = 1$.

### 3.2 The discontinuous Galerkin method for linear systems

#### 3.2.1 General formulation

The discontinuous Galerkin method can be generalized to a system in a straightforward manner. Each component of the approximate solution vector that we shall denote by $U_h$ is defined locally in each cell as a polynomial of degree $k$, i.e. an element of $\mathbb{P}_k[x_\nu, x_{\nu+1}]$. Denoting by $\varphi_0, \ldots, \varphi_k$ a basis of $\mathbb{P}_k[x_\nu, x_{\nu+1}]$, the restriction of $U_h$ to the cell $[x_\nu, x_{\nu+1}]$, that we shall denote by $U_{h,k}^\nu$ can be expressed as

$$U_{h,k}^\nu(t, x) = \sum_{j=0}^{k} \varphi_j(x)U_j^\nu(t), \quad \text{with} \quad U_j^\nu(t) = \begin{pmatrix} u_{j,1}^\nu(t) \\ \vdots \\ u_{j,n}^\nu(t) \end{pmatrix}.$$ 

The unknown function $U_h(t, x)$ is completely determined by $U_j^\nu(t)$. In order to determine those we proceed like in the scalar case and plug this expression in the equation, multiply by each of the basis functions $\varphi_i$ and integrate the derivative term by parts to let the flux through the interface appear:

$$\sum_{j=0}^{k} \left( \frac{dU_j^\nu(t)}{dt} \int_{x_\nu}^{x_{\nu+1}} \varphi_i(x)\varphi_j(x) \, dx - \int_{x_\nu}^{x_{\nu+1}} AU_j^\nu(t)\varphi_i'(x)\varphi_j(x) \, dx \right) + G_{\nu+1}\varphi_i(x_{\nu+1}) - G_{\nu}\varphi_i(x_\nu) = 0.$$
The numerical flux $G_\nu$ is a consistent approximation of the real flux $A(U_{\nu-1}(t, x_\nu) + U_\nu(t, x_\nu))$ at the cell interface $x_\nu$ which is identical for both cells sharing $x_\nu$.

**Choice of the numerical flux**: As in the scalar case, there are many possible choices of the numerical flux.

1. Centered flux: Use an average from the fluxes coming from both cells

$$G_\nu = \frac{1}{2} A (U_{\nu-1}(t, x_\nu) + U_\nu(t, x_\nu)).$$

2. Solution of the Riemann problem corresponding to $U_L = U_{\nu-1}(t, x_\nu)$ and $U_R = U_\nu(t, x_\nu)$. This is the generalization of the upwind flux to systems. Better stability properties but more diffusive.

3. Linear combination of the above to combine good stability properties of the latter and lower diffusion properties of former.

4. Generalized Lax-Friedrichs flux (also called Rusanov flux).

$$G_\nu = \frac{1}{2} [A(U_{\nu-1}(t, x_\nu) + U_\nu(t, x_\nu)) - \alpha_\nu (U_\nu(t, x_\nu) - U_{\nu-1}(t, x_\nu))],$$

with $\alpha_\nu = \max|\lambda_k|$, where the $\lambda_k$ are the eigenvalues of $A$. This solver has good stability properties without needing to solve the Riemann problem. Especially interesting in the non linear case when the Riemann problem is hard to solve.

### 3.2.2 Derivation of DG scheme for Maxwell’s equations

We consider the 1D Maxwell equations:

$$\begin{align*}
\frac{\partial E_j}{\partial t} + \frac{\partial B_j}{\partial x} &= 0, \\
\frac{\partial B_j}{\partial t} + \frac{\partial E_j}{\partial x} &= 0.
\end{align*}$$

(3.1)

(3.2)

Rather than using abstract formalism, we treat separately each component. Both are expressed on the same basis in each cell, e.g. the Lagrange-Gauss-Lobatto basis

$$E^\nu_j(t, x) = \sum_{i=1}^{k} e_j^\nu(t) \varphi_i^\nu(x), \quad B^\nu_j(t, x) = \sum_{i=1}^{k} b_j^\nu(t) \varphi_i^\nu(x)$$

Plug the expression of $E_j^\nu$ and $B_j^\nu$ into (3.1), multiply by the basis function $\varphi_i^\nu$, $0 \leq i \leq k$, and integrate by parts the derivative term

$$\sum_{j=1}^{k} \frac{d e_j^\nu}{dt} \int_{x_j}^{x_{j+1}} \varphi_j^\nu(x) \frac{\partial}{\partial x} \varphi_i^\nu(x) \, dx - \sum_{j=1}^{k} b_j^\nu(t) \int_{x_j}^{x_{j+1}} \varphi_j^\nu(x) \frac{\partial}{\partial x} \varphi_i^\nu(x) \, dx - \frac{d b_j^\nu}{dt} (t) \varphi_i^\nu(x_{j+1}) - \frac{d b_j^\nu}{dt} (t) \varphi_i^\nu(x_j) = 0.$$  

(3.3)

Note that the expression of the boundary is like that because we are using the Lagrange-Gauss-Lobatto basis functions for which only $\varphi_0^\nu(x_j) = 1$ does not vanish at $x_j$ and only $\varphi_k^\nu(x_{j+1}) = 1$ does not vanish at $x_{j+1}$. This term will be replaced in the second step of the DG scheme by the numerical flux coupling the current cell with the neighboring cells.

We proceed in the same way for (3.2)

$$\sum_{j=1}^{k} \frac{d b_j^\nu}{dt} \int_{x_j}^{x_{j+1}} \varphi_j^\nu(x) \varphi_i^\nu(x) \varphi_j^\nu(x) \, dx - \sum_{j=1}^{k} e_j^\nu(t) \int_{x_j}^{x_{j+1}} \varphi_j^\nu(x) \varphi_i^\nu(x) \varphi_j^\nu(x) \, dx + \frac{d e_j^\nu}{dt} (t) \varphi_i^\nu(x_{j+1}) - \frac{d e_j^\nu}{dt} (t) \varphi_i^\nu(x_j) = 0.$$  

(3.4)
The cell mass and derivative matrices are computed on the reference cell $[-1, 1]$ by the change of variable

$$x = \frac{x_{\nu+1} - x_{\nu}}{2} \hat{x} + \frac{x_{\nu+1} + x_{\nu}}{2}.$$  

Then, denoting by $J_\nu = \frac{x_{\nu+1} - x_{\nu}}{2}$, the basis functions $\varphi^i_\nu$ on the cell $[x_{\nu}, x_{\nu+1}]$ are defined with respect to the Lagrange basis functions $\ell_i$ at the Gauss-Lobatto points in the reference cell by

$$\varphi^i_\nu(x) = \ell_i(\hat{x})$$

and consequently $(\varphi^i_\nu)'(x) = \ell_i'(\hat{x}) / J_\nu$.

Using this change of variables we find

$$\int_{x_{\nu}}^{x_{\nu+1}} \varphi_j(x) \varphi_i(x) \, dx = J_\nu \int_{-1}^{1} \ell_j(\hat{x}) \ell_i(\hat{x}) \, d\hat{x},$$

and so, denoting by $e^\nu = (e^\nu_0, \ldots, e^\nu_k)^\top$, $b^\nu = (b^\nu_0, \ldots, b^\nu_k)^\top$, and by

$$M = \left( \int_{-1}^{1} \ell_i'(\hat{x}) \ell_j(\hat{x}) \, d\hat{x} \right)_{0 \leq i, j \leq k},$$

$$D = \left( \int_{-1}^{1} \ell_j'(\hat{x}) \ell_i(\hat{x}) \, d\hat{x} \right)_{0 \leq i, j \leq k},$$

the mass and derivative matrices on the reference element. Expressions (3.3) and (3.4) can be rewritten in matrix form

$$J_\nu M \frac{de^\nu}{dt} - D b^\nu + G^\nu_{1+1} - G^\nu_1 = 0 \quad (3.5)$$

$$J_\nu M \frac{db^\nu}{dt} - D e^\nu + G^\nu_{2+1} - G^\nu_2 = 0 \quad (3.6)$$

where $G^\nu_1$ and $G^\nu_2$ are the two components of the numerical flux defined at the cell interface $x_{\nu}$.

Different options for the computations of this numerical flux, upwind, centered, partially upwind and Rusanov have been discussed. The expression of the upwind flux obtained by solving the Riemann problem has been given at the end of section (3.1).

### 3.3 The Discontinuous Galerkin method for linear systems in 2D

The DG method can be extended in a natural way to multiple dimensions. We shall describe it for a system of 2D generalized Maxwell equations, which includes as well gradient, curl and divergence operators. This system can be used when the sources $\rho$ and $J$ do not verify exactly the continuity equations $\frac{\partial \rho}{\partial t} + \nabla \cdot J = 0$, which can occur in a numerical scheme when Maxwell’s equations are coupled with the Vlasov equation. The unknown $p$, which can be proved to be 0 at the continuous level if the continuity equation $\frac{\partial \rho}{\partial t} + \text{div} J = 0$ is satisfied, has been added to absorb the numerical errors occurring from the continuity equation only being satisfied approximately at the discrete level. The $\gamma$ factor in front of $c^2 \nabla p$ enables to fix the velocity of the wave transporting $p$ out of the computational domain. In general it is taken slightly larger than 1, so that this non physical wave propagates a little bit faster than the electromagnetic waves.

The generalized Maxwell equations in 3D write:

$$\frac{\partial \mathbf{E}}{\partial t} - c^2 \nabla \times \mathbf{B} + \gamma c^2 \nabla p = -\frac{1}{\varepsilon_0} \mathbf{J}, \quad (3.7)$$

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \times \mathbf{E} = 0, \quad (3.8)$$

$$\frac{\partial p}{\partial t} + \nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0}. \quad (3.9)$$
In 2D when all the unknowns depend only on two variables \((x, y)\) and not on \(z\) the Maxwell equations decouple into two subsystems known as the transverse electric (TE) mode and the transverse magnetic (TM) mode.

The TE mode reads

\[
\begin{align*}
\frac{\partial \mathbf{E}}{\partial t} - c^2 \text{curl} \mathbf{B} + \gamma c^2 \nabla p &= -\frac{1}{\varepsilon_0} \mathbf{J}, \\
\frac{\partial \mathbf{B}}{\partial t} + \text{curl} \mathbf{E} &= 0, \\
\frac{\partial p}{\partial t} + \text{div} \mathbf{E} &= \frac{\rho}{\varepsilon_0}.
\end{align*}
\]

(3.10) (3.11) (3.12)

and for the TM mode we have

\[
\begin{align*}
\frac{\partial \mathbf{E}}{\partial t} - c^2 \text{curl} \mathbf{B} &= -\frac{1}{\varepsilon_0} \mathbf{J}, \\
\frac{\partial \mathbf{B}}{\partial t} + \text{curl} \mathbf{E} &= 0.
\end{align*}
\]

(3.13) (3.14)

with \(\mathbf{E} = (E_x, E_y)^\top\), \(\mathbf{B} = (B_x, B_y)^\top\), \(\mathbf{B}_z = (\partial_y B_x - \partial_x B_y)^\top\), \(\text{curl} \mathbf{E} = \partial_x E_y - \partial_y E_x\), and \(\text{div} \mathbf{E} = \partial_x E_x + \partial_y E_y\).

Let us now describe in details the DG method for the TE mode of the 2D Maxwell equations. The DG method is most naturally expressed using hyperbolic conservation laws.

### 3.3.1 Derivation of the discrete equations

In order to define a Discontinuous Galerkin method, one needs a mesh of the computational domain consisting of disjoint cells. Then the unknown is approximated on each cell in a finite dimensional linear space (generally polynomial) which could be different in each cell. We shall consider only approximations based on \((P_k)^p\) for an unknown with values in \(\mathbb{R}^p\). In our case, for the two-dimensional Maxwell equations we consider polynomials of two variables with \(p = 3\), the unknown being \(\mathbf{u} = (E_x, E_y, B_z)^\top\). We denote by \(P_k\) the space of polynomials of degree less or equal to \(k\) in 2D here: \(P_k = \text{span}\{x^n y^m, \ n + m \leq k\}\). In two space dimensions where we work the dimension of \(P_k\) is \(\frac{(k+1)(k+2)}{2}\). On a given cell \(K\) we look for an approximation \(\mathbf{u}_h = (E_{h,x}, E_{h,y}, B_{h,z})^\top \in (P_k(K))^3\).

Using the Galerkin principle we shall derive an equation on \(\mathbf{u}_h\) by multiplying the equations by a test function belonging to the same space. However in the Discontinuous Galerkin case, the integration shall take place on one unique cell instead of the whole computational domain for the standard Galerkin method. We shall detail the derivation for the TE mode. It can be performed in a similar way for the TM mode.

The polynomial space \(P_k\) is the most natural when we consider a mesh of triangles. In a mesh of quads, in particular for a cartesian grid, we would rather consider the space of tensor product of 1D polynomials generally denoted by \(Q_k = \text{span}\{x^n y^m, \ n, m \leq k\}\). The derivation of the DG method is identical in the two cases. An example of unstructured mesh and a schematic representation of the \(P_3\) Finite Element on the reference cell are show in Figure 3.1.

**Remark 1.** The specificity of the Discontinuous Galerkin method with respect to the standard Galerkin method is that the weak form is defined on each cell with no continuity constraint between two neighboring cells. A definition of a unique numerical flux at the cell interface is needed to transfer the information from one cell to the other in a consistent manner. This numerical flux and the cell approximation are the building blocks of a DG method.

Let us consider a test function \(\mathbf{v}_h = (F_{h,x}, F_{h,y}, C_{h,z}, q_h)^\top \in (P_k(K))^4\), and take the dot product of the Ampere equation (3.10) with \(\mathbf{F}_h = (F_{h,x}, F_{h,y})^\top\), multiply the Faraday equation (3.11) by...
We also end up with a total of 6 basis functions, with each basis having a value 1 at its assigned node and a value of zero at all the other mesh nodes in the element.

Again this has the form 

so we can easily determine the coefficients of all 6 basis functions in the quadratic case by inverting the matrix $M$. Each column of the inverted matrix is a set of coefficients for a basis function.

**P3 Cubic Elements**

In the case of cubic finite elements, the basis functions are cubic polynomials of degree 3 with the designation $P_3$

We now have 10 unknowns so we need a total of 10 node points on the triangle element to determine the coefficients of the basis functions in the same manner as the $P_2$ case.

**Experiment 1**

In this experiment we will solve Poisson’s equation over a square region using Finite Elements with linear, quadratic and cubic elements and compare the results. We will solve

$$
\frac{d}{dt} \int_K \mathbf{E}_h \cdot \mathbf{F}_h - c^2 \int_K \text{curl} \mathbf{B}_{h,z} \cdot \mathbf{F}_h + \gamma c^2 \int_K \nabla p_h \cdot \mathbf{F}_h = -\frac{1}{\varepsilon_0} \int_K \mathbf{J}_h \cdot \mathbf{F}_h,
$$

$$
\frac{d}{dt} \int_K \mathbf{B}_{h,z} \cdot \mathbf{B}_{h,z} + \int_K \text{curl} \mathbf{E}_h \mathbf{C}_{h,z} = 0,
$$

$$
\frac{d}{dt} \int_K p_h q_h + \int_K \text{div} \mathbf{E}_h q_h = \frac{1}{\varepsilon_0} \int_K \rho q_h.
$$

In order to introduce the numerical flux which will link local solutions on neighboring cells, it is necessary to integrate all those equations by parts in order to let a boundary term appear. To this aim we shall need the following Green formulas:

$$
\int_K \mathbf{F} \cdot \text{curl} \mathbf{C} - \int_K \text{curl} \mathbf{F} \mathbf{C} = \int_{\partial K} (\mathbf{F} \cdot \mathbf{\tau}) \mathbf{C} \quad \forall \mathbf{F} \in H(\text{curl}, K), \mathbf{C} \in H^1(K), \quad (3.15)
$$

$$
\int_K \mathbf{F} \cdot \nabla q + \int_K \text{div} \mathbf{F} q = \int_{\partial K} (\mathbf{F} \cdot \mathbf{n}) q \quad \forall \mathbf{F} \in H(\text{div}, K), q \in H^1(K), \quad (3.16)
$$

where $\mathbf{\tau} = (n_y, -n_x)^\top$ is the tangent vector on the cell boundary $\partial K$, $\mathbf{n} = (n_x, n_y)^\top$ defining the outbound normal vector. We thus obtain

$$
\int_K \text{curl} \mathbf{B}_{h,z} \cdot \mathbf{F}_h = \int_K \mathbf{B}_{h,z} \text{curl} \mathbf{F}_h + \int_{\partial K} (\mathbf{F}_h \cdot \mathbf{\tau}) \mathbf{B}_{h,z},
$$

$$
\int_K \text{curl} \mathbf{E}_h \mathbf{C}_{h,z} = \int_K \mathbf{E}_h \cdot \text{curl} \mathbf{C}_{h,z} - \int_{\partial K} (\mathbf{E}_h \cdot \mathbf{\tau}) \mathbf{C}_{h,z},
$$

$$
\int_K \nabla p_h \cdot \mathbf{F}_h = -\int_K p_h \text{div} \mathbf{E}_h + \int_{\partial K} p_h (\mathbf{F}_h \cdot \mathbf{n}),
$$

$$
\int_K \text{div} \mathbf{E}_h q_h = -\int_K \mathbf{E}_h \cdot \nabla q_h + \int_{\partial K} (\mathbf{E}_h \cdot \mathbf{n}) q_h.
$$

The unknowns being inherently discontinuous at the cell interfaces in the DG method, the boundary terms are not well defined, at least not consistently on the left hand side and the right hand side of
the interface. In order to specify the DG method a numerical approximation of these fluxes which is identical on both sides of the interface needs to be defined. These are the numerical fluxes. A simple an natural definition of the numerical flux consists in taking the average value of the left hand side and the right hand side values. This yields a centered flux which has some advantages for Maxwell’s equations. This flux will lead to an exact energy conservation at the discrete level as is the case for standard elements. In this case as we will see, the matrix form of the DG discretization is identical to what we obtained for Finite Elements. However due to the discontinuities introduced in the DG formulation, it can be good idea to add a dissipation mechanism by considering upwind fluxes. In order to keep the freedom of specifying the numerical flux later, let us simply denote by $B_N, E_N$ and $p_N$ the values of the approximate electromagnetic flux and of $p$ defined by the specific numerical flux. We then obtain the following discrete formulations on each element $K$:

$$\frac{d}{dt} \int_K \mathbf{E}_h \cdot \mathbf{F}_h = -c^2 \int_K B_{h,z} \text{curl} \mathbf{F}_h - \gamma c^2 \int_K p_h \text{div} \mathbf{F}_h$$

$$- c^2 \int_{\partial K} (\mathbf{F}_h \cdot \mathbf{\tau}) B_N + \gamma c^2 \int_{\partial K} p_N (\mathbf{F}_h \cdot \mathbf{n}) = -\frac{1}{\varepsilon_0} \int_K \mathbf{J}_h \cdot \mathbf{F}_h, \quad \forall \mathbf{F}_h \in \mathbb{P}_k^2,$$  \tag{3.17}

$$\frac{d}{dt} \int_K B_{h,z} C_{h,z} + \int_K \mathbf{E}_h \cdot \text{curl} C_{h,z} - \int_{\partial K} (\mathbf{E}_N \cdot \mathbf{\tau}) C_{h,z} = 0 \quad \forall C_{h,z} \in \mathbb{P}_k,$$ \tag{3.18}

$$\frac{d}{dt} \int_K p_h \mathbf{q}_h - \int_K \mathbf{E}_h \cdot \nabla \mathbf{q}_h + \int_{\partial K} (\mathbf{E}_N \cdot \mathbf{n}) \mathbf{q}_h = \frac{1}{\varepsilon_0} \int_K \rho \mathbf{q}_h \quad \forall \mathbf{q}_h \in \mathbb{P}_k.$$ \tag{3.19}

### 3.3.2 Matrix formulation of the discrete problem

#### Integrals over cells

A classical Finite Element is defined on a cell by a triple $(K, P, \Sigma)$, where $K$ is the cell, $P$ a finite dimensional function space of dimension $N_k$ on $K$ (in our case $\mathbb{P}_k(K)$) and $\Sigma = \{\sigma_1, \ldots, \sigma_{N_k}\}$ is a set of $N_k$ linear forms enabling to identify uniquely an element of $P$. Thus to the linear forms of $\Sigma$ is associated a unique basis $(\varphi_j)_{1 \leq j \leq N_k}$ of $P$ such that $\sigma_i(\varphi_j) = \delta_{i,j}$ where $\delta_{i,j}$ is the Kronecker symbol. We consider here only Lagrange Finite Elements where the linear forms $(\sigma_i)_{1 \leq i \leq N_{k}}$ are associated to interpolation points $(x_i)_{1 \leq i \leq N_k}$ in $K$. In this case we simply have for a function $G_h \in P$, $\sigma_i(G) = G(x_i)$. Other classical degrees of freedom are edge, face or volume integrals or polynomial moments. Any element $G_h$ of our finite dimensional function space $P$ is thus defined uniquely by $(\sigma_i(G_h))_{1 \leq i \leq N_k}$ and we have

$$G_h(x) = \sum_{j=1}^{N_k} \sigma_j(G_h) \varphi_j(x) = \sum_{j=1}^{N_k} G_h(x_j) \varphi_j(x),$$  \tag{3.20}

the last equality holding for Lagrange Finite Elements. Let us denote by $\mathbf{g} = (\sigma_1(G_h), \ldots, \sigma_{N_k}(G_h))^\top$ the vector whose components are the degrees of freedom and the same for $\mathbf{e}, \mathbf{f}, \mathbf{b}, \mathbf{c}$. In the case of Lagrange Elements we have $\mathbf{g} = (G_h(x_1), \ldots, G_h(x_{N_k}))^\top$, where $x_1, \ldots, x_{N_k}$ are the interpolation points characterizing the Lagrange basis. As we are here in 2D, $x_i = (x_{i1}, y_{i})$ has two components. We then obtain a matrix expression of the relations (3.17)-(3.19) by plugging in the expression of the unknowns and test functions for each component separately in the basis, as in (3.20). We first have

$$\int_K \mathbf{E}_h \cdot \mathbf{F}_h \, dx = \int_K (E_{h,x} F_{h,x} + E_{h,y} F_{h,y}) \, dx$$

$$= \sum_{1 \leq i, j \leq N_k} \sigma_j(E_{h,x}) \sigma_i(F_{h,x}) \int_K \varphi_i \varphi_j \, dx + \sum_{1 \leq i, j \leq N_k} \sigma_j(E_{h,y}) \sigma_i(F_{h,y}) \int_K \varphi_i \varphi_j \, dx$$

$$= \mathbf{f}_x^\top M_K \mathbf{e}_x + \mathbf{f}_y^\top M_K \mathbf{e}_y,$$
where the mass matrix on element $K$ is defined by

$$M_K = \left( \int_K \varphi_i \varphi_j \, dx \right)_{1 \leq i, j \leq N_k}.$$  

In the same way

$$\int_K B_{h,z} C_{h,z} \, dx = \sum_{1 \leq i, j \leq N_k} \sigma_j(B_{h,z})\sigma_i(C_{h,z}) \int_K \varphi_i \varphi_j \, dx = c_z^\top M_K b_z.$$  

All the interior terms involving derivatives, can be expressed using the two derivative matrices in the $x$ and $y$ directions

$$D^x_K = \left( \int_K \varphi_j \partial_x \varphi_i \, dx \right)_{1 \leq i, j \leq N_k}, \quad D^y_K = \left( \int_K \varphi_j \partial_y \varphi_i \, dx \right)_{1 \leq i, j \leq N_k}.$$  

First the term involving the scalar curl can be written

$$\int_K B_{h,z} \text{curl} \mathbf{F}_h = \int_K B_{h,z} \left( \partial_x F_{h,y} - \partial_y F_{h,x} \right) \, dx$$

$$= \sum_{1 \leq i, j \leq N_k} \sigma_j(B_{h,z})\sigma_i(F_{h,y}) \int_K \varphi_j \partial_x \varphi_i \, dx - \sum_{1 \leq i, j \leq N_k} \sigma_j(B_{h,z})\sigma_i(F_{h,x}) \int_K \varphi_j \partial_y \varphi_i \, dx$$

$$= f_x^y D^y_K b_z - f_x^x D^y_K b_z.$$  

The term involving the vector curl on element $K$ is

$$\int_K \mathbf{E}_h \cdot \text{curl} C_{h,z} = \int_K \left( E_{h,z} \partial_y C_{h,z} - E_{h,y} \partial_z C_{h,z} \right) \, dx$$

$$= \sum_{1 \leq i, j \leq N_k} \sigma_i(C_{h,z})\sigma_j(E_{h,x}) \int_K \varphi_j \partial_y \varphi_i \, dx - \sum_{1 \leq i, j \leq N_k} \sigma_i(C_{h,z})\sigma_j(E_{h,y}) \int_K \varphi_j \partial_y \varphi_i \, dx$$

$$= c_z^\top D^y_K e_x - c_z^\top D^x_K e_y.$$  

The term involving the divergence becomes

$$\int_K p_h \text{div} \mathbf{F}_h = \int_K p_h \left( \partial_x F_{h,x} + \partial_y F_{h,y} \right) \, dx$$

$$= \sum_{1 \leq i, j \leq N_k} \sigma_j(p_h)\sigma_i(F_{h,x}) \int_K \varphi_j \partial_x \varphi_i \, dx + \sum_{1 \leq i, j \leq N_k} \sigma_j(p_h)\sigma_i(F_{h,y}) \int_K \varphi_j \partial_y \varphi_i \, dx$$

$$= f_x^y D^x_K p + f_x^y D^y_K p.$$  

and finally the term involving the gradient can be written as

$$\int_K \mathbf{E}_h \cdot \nabla q_h = \int_K \left( E_{h,x} \partial_x q_h + E_{h,y} \partial_y q_h \right) \, dx$$

$$= \sum_{1 \leq i, j \leq N_k} \sigma_i(q_h)\sigma_j(E_{h,x}) \int_K \varphi_j \partial_x \varphi_i \, dx + \sum_{1 \leq i, j \leq N_k} \sigma_i(q_h)\sigma_j(E_{h,y}) \int_K \varphi_j \partial_y \varphi_i \, dx$$

$$= q^\top D^x_K e_x + q^\top D^y_K e_y.$$  

46
Integrals over cell boundaries

Let us now handle the flux terms. Let us denote by \( f \) an internal or external face of the mesh for which the tangent vector \( \tau \) is constant and \( \tilde{N}_k \), the number of degrees of freedom on the face. Then

\[
\int_f (F_h \cdot \tau) B_N = \int_f (F_{h,x}\tau_x + F_{h,y}\tau_y) B_N \, d\sigma
\]

\[
= \sum_{1 \leq i,j \leq \tilde{N}_k} \sigma_j(B_N)\sigma_i(F_{h,x}) \int_f \varphi_i \varphi_j \, d\sigma + \sum_{1 \leq i,j \leq \tilde{N}_k} \sigma_j(B_N)\sigma_i(F_{h,y}) \int_f \varphi_i \varphi_j \, d\sigma
\]

\[
= \tau_{f,x} f_x^\top M_f b_N + \tau_{f,y} f_y^\top M_f b_N,
\]

where \( M_f = \int_f \varphi_i \varphi_j \, d\sigma \) is the mass matrix on face \( f \). In the same way

\[
\int_f (E_N \cdot \tau) C_{h,z} = \int_f (E_{N,x}\tau_x + E_{N,y}\tau_y) C_{h,z} \, d\sigma
\]

\[
= \tau_{f,x} e^\top M_f e_{N_x} + \tau_{f,y} e^\top M_f e_{N_y}.
\]

**Remark 2.** Let us notice that the matrix relation representing the terms interior to an element only involves the degrees of freedom of the considered element. The coupling between different elements appears only in the flux terms. The notations \( B_N \) and \( e_N \) hide a coupling between the values of the two elements (in the case of a conforming mesh) sharing the same face. For example in the case of a centered flux, on the face \( f \), \( b_N = \frac{1}{2}(b_K + b_L) \) and \( e_N = \frac{1}{2}(e_K + e_L) \), where \( K \) and \( L \) denote the two elements sharing face \( f \).

We can thus write a matrix relation enabling to compute the degrees of freedom freedom by element, knowing that there exists a coupling through the terms \( B_N \) and \( e_N \). For conciseness we only write the terms in \( E \) and \( B \) coming from the standard Maxwell equations, setting \( p = 0 \). Plugging the above formulations in (3.17)-(3.19) we get

\[
\frac{d}{dt} \left( f_x^\top M_K e_x + f_y^\top M_K e_y \right) - c^2 \left( f_y^\top D_y^f - f_x^\top D_x^f \right) b_z - c^2 \sum_{f \in \partial K} \left( \tau_{f,x} f_x^\top + \tau_{f,y} f_y^\top \right) M_f b_N
\]

\[
= - \frac{1}{\varepsilon_0} \left( f_x^\top M_K j_x + f_y^\top M_K j_y \right) \quad \forall (f_x, f_y) \in \mathbb{R}^{2N_K}, \quad (3.21)
\]

\[
\frac{d}{dt} \left( c_z^\top M_K b_z \right) + c_z^\top D_y^f e_x - c_z^\top D_x^f e_y - \sum_{f \in \partial K} c_z^\top M_f (e_{N,x} \tau_{f,x} + e_{N,y} \tau_{f,y}) = 0 \quad \forall c_z \in \mathbb{R}^{N_k}, \quad (3.22)
\]

which is equivalent to

\[
\begin{pmatrix}
M_K & 0 \\
0 & M_K
\end{pmatrix}
\frac{d}{dt}
\begin{pmatrix}
e_x \\
e_y
\end{pmatrix}
- c^2
\begin{pmatrix}
-D_y^f \\
D_x^f
\end{pmatrix}
b_z - c^2 \sum_{f \in \partial K} \left( \tau_{f,x} M_f \right) b_N
= - \frac{1}{\varepsilon_0}
\begin{pmatrix}
M_K & 0 \\
0 & M_K
\end{pmatrix}
\begin{pmatrix}
\dot{j}_x \\
\dot{j}_y
\end{pmatrix}
\]

\[
M_K \frac{d}{dt} b_z + \left( D_y^f - D_x^f \right)
\begin{pmatrix}
e_x \\
e_y
\end{pmatrix}
- \sum_{f \in \partial K} \left( \tau_{f,x} M_f \tau_{f,y} M_f \right)
\begin{pmatrix}
e_{N,x} \\
e_{N,y}
\end{pmatrix}
= 0. \quad (3.23)
\]

In view of implementation, it is more convenient to rewrite this expression of the matrices on \( K \) using only integrals computed on a reference element \( \tilde{K} \) and a change of variables. We consider
here a mesh of triangles and a reference element \( \hat{K} \) with vertices \((0,0), (1,0), (0,1)\), and an arbitrary element \( K \) with vertices \((x_1, y_1), (x_2, y_2), (x_3, y_3)\), the affine transformation of the plane

\[
\begin{pmatrix}
\hat{x} \\
\hat{y}
\end{pmatrix} \mapsto A \begin{pmatrix}
\hat{x} \\
\hat{y}
\end{pmatrix} + b,
\]

where

\[
A = \begin{pmatrix}
x_2 - x_1 & x_3 - x_1 \\
y_2 - y_1 & y_3 - y_1
\end{pmatrix}
\quad \text{and} \quad
b = \begin{pmatrix}
x_1 \\
y_1
\end{pmatrix},
\]

transforms the reference triangle \( \hat{K} \) onto the triangle \( K \). Using the corresponding change of variables for computing the integrals on \( K \) used in the definition of the matrices, we get only integrals on \( \hat{K} \). Thus

\[
\int_K \phi_i \phi_j \, dx \, dy = \det(A) \int_{\hat{K}} \hat{\phi}_i \hat{\phi}_j \, d\hat{x} \, d\hat{y},
\]

and so \( M_K = \det(A) M_{\hat{K}} \). For the integrals involving a derivative, we use the chain rule which yields \( \nabla \hat{\phi}_j = A^T \nabla \phi_j \) and so \( \nabla \phi_j = A^{-T} \nabla \hat{\phi}_j \) where

\[
A^{-T} = (A^{-1})^T = \frac{1}{\det A} \begin{pmatrix}
y_3 - y_1 & y_1 - y_2 \\
x_1 - x_3 & x_2 - x_1
\end{pmatrix}.
\]

Thus as \( \det A \) of the jacobian cancels with the one from \( A^{-T} \), we have

\[
\int_K \phi_j \partial_x \phi_i \, dx = (y_3 - y_1) \int_{\hat{K}} \hat{\phi}_j \partial_{\hat{x}} \hat{\phi}_i \, d\hat{x} \, d\hat{y} + (y_1 - y_2) \int_{\hat{K}} \hat{\phi}_j \partial_{\hat{y}} \hat{\phi}_i \, d\hat{x} \, d\hat{y},
\]

and

\[
\int_K \phi_j \partial_y \phi_i \, dx = (x_1 - x_3) \int_{\hat{K}} \hat{\phi}_j \partial_{\hat{x}} \hat{\phi}_i \, d\hat{x} \, d\hat{y} + (x_2 - x_1) \int_{\hat{K}} \hat{\phi}_j \partial_{\hat{y}} \hat{\phi}_i \, d\hat{x} \, d\hat{y}.
\]

It follows that

\[
D_{\hat{K}}^x = (y_3 - y_1)D_{\hat{K}}^{\hat{x}} + (y_1 - y_2)D_{\hat{K}}^{\hat{y}} \quad \text{and} \quad
D_{\hat{K}}^y = (x_1 - x_3)D_{\hat{K}}^{\hat{x}} + (x_2 - x_1)D_{\hat{K}}^{\hat{y}}.
\]

Finally the integrals on the faces are computed using the same principle by a change of variables for the mass matrix on a face:

\[
\int_f \phi_i \phi_j \, d\sigma = |f| \int_{\hat{f}} \hat{\phi}_i \hat{\phi}_j \, d\hat{\sigma}
\]

where \(|f|\) is the length of face \( f \) and \( M_f \) the mass matrix on the reference element of faces.

Plugging these expressions in the matrix formulation (3.23)-(3.24), we obtain a matrix formulation uniquely based on the elementary matrices on the reference element

\[
\det(A) \begin{pmatrix}
M_K & 0 \\
0 & M_{\hat{K}}
\end{pmatrix} \frac{d}{dt} \begin{pmatrix}
ex \\
ey
\end{pmatrix} - c^2 \begin{pmatrix}
-(x_1 - x_3)D_{\hat{K}}^{\hat{x}} + (x_2 - x_1)D_{\hat{K}}^{\hat{y}} \\
y_3 - y_1)D_{\hat{K}}^{\hat{x}} + (y_1 - y_2)D_{\hat{K}}^{\hat{y}}
\end{pmatrix} b_z
\]

\[
- c^2 \sum_{f \in \partial K} |f| \begin{pmatrix}
\tau_f e \cdot M_f \\
\tau_f \hat{b} \cdot M_f
\end{pmatrix} b_N = - \frac{1}{\varepsilon_0} \det(A) \begin{pmatrix}
M_K & 0 \\
0 & M_{\hat{K}}
\end{pmatrix} \begin{pmatrix}
1_x \\
1_y
\end{pmatrix},
\]

(3.25)
\[ \det(A)M_{\hat{K}} \frac{d}{dt} b_z + \left( (x_1 - x_3) D_{\hat{K}}^x + (x_2 - x_1) D_{\hat{K}}^y - (y_3 - y_1) D_{\hat{K}}^x + (y_1 - y_2) D_{\hat{K}}^y \right) \begin{pmatrix} e_x \\ e_y \end{pmatrix} \]

\[ - \sum_{f \in \partial K} |f| \left( \tau_{f,x} M_{\hat{f}} \tau_{f,y} M_{\hat{f}} \right) \begin{pmatrix} e_N_x \\ e_N_y \end{pmatrix} = 0. \quad (3.26) \]

To conclude we need to give an expression of the 2D Lagrange basis functions that can be used to compute the elementary matrices. For simplices, which are triangles in 2D and tetrahedra in 3D, it is convenient to use the barycentric coordinates associated to the vertices of the simplex. We shall denote them by \( x_0, \ldots, x_n \in \mathbb{R}^n \). The barycentric coordinate, classically denoted by \( \lambda_i \), \( 0 \leq i \leq n \), associated to the vertex \( x_i \) is the affine function with value 1 at \( x_i \) and 0 at the \( n \) other vertices.

In 2D on our reference element where \( x_0 = (0,0) \), \( x_1 = (1,0) \) and \( x_2 = (0,1) \), the barycentric coordinates are

\[ \lambda_0(x) = 1 - x - y, \quad \lambda_1(x) = x, \quad \lambda_2(y). \quad (3.27) \]

The coefficients \( (\alpha_{i,l})_{0 \leq l \leq n} \) being determined by the relations \( \lambda_i(a_j) = \delta_{ij} \).

The dimension of \( P_k \) is \( \frac{(k+1)(k+2)}{2} \). Putting the Lagrange interpolation points on a lattice with \( k + 1 \) points on the edges and then one point less at each level starting from one edge, we get \( (k+1) + k + (k-1) + \cdots + 1 = (k+1)(k+2)/2 \) interpolation points and so also degrees of freedom, which is the same as the dimension of the space.

1. Lagrange \( P_1 \): The dimension is 3 and the degrees of freedom are the vertices of the triangles. The Lagrange basis functions are then the barycentric coordinate.

2. Lagrange \( P_2 \): The dimension is 6 and the degrees of freedom are the vertices of the triangles and the midpoints of each edge. The Lagrange basis functions are obtained by factoring the lines where the basis functions should be 0 and then normalizing. The degrees of freedom are numbered starting from the bottom and left to right. We then obtain

\[ \hat{\varphi}_1 = 2\lambda_0(\lambda_0 - \frac{1}{2}) \]
\[ \hat{\varphi}_2 = 4\lambda_1\lambda_0 \]
\[ \hat{\varphi}_3 = 2\lambda_1(\lambda_1 - \frac{1}{2}) \]
\[ \hat{\varphi}_4 = 4\lambda_2\lambda_0 \]
\[ \hat{\varphi}_5 = 4\lambda_1\lambda_2 \]
\[ \hat{\varphi}_6 = 2\lambda_2(\lambda_2 - \frac{1}{2}) \]

3. Lagrange \( P_3 \): The dimension is 10. We can use as degrees of freedom the vertices, two equispaced points on each edge and the barycenter of the triangle. Then we get

49
Figure 3.2: Graphical representation of the 2D $P_2$ (left) and $P_3$ (right) degrees of freedom.

4.

\[ \hat{\phi}_1 = \frac{9}{2}\lambda_0(\lambda_0 - \frac{1}{3})(\lambda_0 - \frac{2}{3}) \] (3.34)

\[ \hat{\phi}_2 = \frac{27}{2}\lambda_0\lambda_1(\lambda_0 - \frac{1}{3}) \] (3.35)

\[ \hat{\phi}_3 = \frac{27}{2}\lambda_1\lambda_0(\lambda_1 - \frac{1}{3}) \] (3.36)

\[ \hat{\phi}_4 = \frac{9}{2}\lambda_1(\lambda_1 - \frac{1}{3})(\lambda_1 - \frac{2}{3}) \] (3.37)

\[ \hat{\phi}_5 = \frac{27}{2}\lambda_0\lambda_2(\lambda_0 - \frac{1}{3}) \] (3.38)

\[ \hat{\phi}_6 = 27\lambda_0\lambda_1\lambda_2 \] (3.39)

\[ \hat{\phi}_7 = \frac{27}{2}\lambda_1\lambda_2(\lambda_1 - \frac{1}{3}) \] (3.40)

\[ \hat{\phi}_8 = \frac{27}{2}\lambda_2\lambda_0(\lambda_2 - \frac{1}{3}) \] (3.41)

\[ \hat{\phi}_9 = \frac{27}{2}\lambda_2\lambda_1(\lambda_2 - \frac{1}{3}) \] (3.42)

\[ \hat{\phi}_{10} = \frac{9}{2}\lambda_2(\lambda_2 - \frac{1}{3})(\lambda_2 - \frac{2}{3}) \] (3.43)

### 3.3.3 Computation of the fluxes

In a Discontinuous Galerkin method, the inter-element coupling occurs through the numerical flux which corresponds to the boundary terms in equations (3.17)-(3.19). The discrete fields in the Discontinuous Galerkin method being discontinuous at the inter-element interfaces, these boundary terms could be expressed either using the values on one side of the boundary or on the other. In the case of a continuous solution, these values will obviously differ only by an error corresponding to the order of the method. A natural way to define these fluxes is to use the half sum of the terms on both sides. This corresponds to the centered flux. On the other hand in order to stabilize the method for long time computations it is often useful to use an upwind flux, which will introduce a dissipation mechanism. This upwinding can be partial using a parameter $\alpha \in [\frac{1}{2}, 1]$ which enables
to build a variable amount of upwinding into the scheme. This flux will then be the centered for \( \alpha = \frac{1}{2} \) and the fully upwind flux for \( \alpha = 1 \).

In order to determine the upwind direction, it is necessary to find the direction of propagation of the flow in the vicinity of the interface. To this aim we need to separate the different waves propagating in the direction of the normal vector \( \mathbf{n} \). To this aim we write the Maxwell equations (3.10)-(3.11) in the form of a hyperbolic system

\[
\frac{\partial \mathbf{u}}{\partial t} + A_x \frac{\partial \mathbf{u}}{\partial x} + A_y \frac{\partial \mathbf{u}}{\partial y} = 0.
\]

with

\[
\mathbf{u} = \begin{pmatrix} E_x \\ E_y \\ B_z \end{pmatrix}, \quad A_x = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & c^2 \\ 0 & 1 & 0 \end{pmatrix}, \quad A_y = \begin{pmatrix} 0 & 0 & -c^2 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix}.
\]

We decompose the flux in the direction \( \mathbf{n} = (n_x, n_y) \top \) in order to get the upwind flux using the signs of the eigenvalues of the matrix \( A_n = n_x A_x + n_y A_y \). This brings us back to the 1D Riemann problem. The eigenvalues of \( A_n \) are here \( 0, c \) and \( -c \). Denote by

\[
P = \begin{pmatrix} \frac{n_x}{n_y} & -n_y c & n y c \\ 1 & n_x c & -n_x c \\ 0 & 1 & 1 \end{pmatrix}
\]

the matrix whose columns are the eigenvectors. This will be the transformation matrix into the diagonalization basis. Its inverse is

\[
P^{-1} = \begin{pmatrix} n_y n_x & n_y^2 & 0 \\ -\frac{n_y}{2c} & \frac{n_x}{2c} & \frac{1}{2} \\ \frac{n_x}{2c} & -\frac{n_y}{2c} & \frac{1}{2} \end{pmatrix}.
\]

Denote by

\[
\Lambda = \begin{pmatrix} 0 & 0 & 0 \\ 0 & c & 0 \\ 0 & 0 & -c \end{pmatrix}
\]

the associated diagonal eigenvalue matrix. We then write \( \Lambda = \Lambda^+ + \Lambda^- \) the first matrix containing the positive eigenvalues and the second one the negative. Denoting by \( \mathbf{u}_L \) and \( \mathbf{u}_R \) the values of the unknowns on the left hand side and the right hand side of the interface, the upwind flux is defined by

\[
A_n \mathbf{u}^* = A_n^+ \mathbf{u}_L + A_n^- \mathbf{u}_R,
\]

with \( A_n^+ = P \Lambda^+ P^{-1} \) and \( A_n^- = P \Lambda^- P^{-1} \).

The partially upwind flux is then obtained thanks to the formula

\[
A_n \mathbf{u}^*_\alpha = A_n^+ (\alpha \mathbf{u}_L + (1 - \alpha) \mathbf{u}_R) + A_n^- (\alpha \mathbf{u}_R + (1 - \alpha) \mathbf{u}_L).
\]
Thus denoting for a quantity \( v \), its average at the interface by \( \{ v \} = \frac{1}{2}(v_R + v_L) \) and its jump at the boundary by \( [v] = v_R - v_L \), the partially upwind flux for the Maxwell equations writes

\[
A_n u_\alpha^* = \begin{pmatrix} -cn_y(c\{B_z\}) + (\alpha - \frac{1}{2})(n_y[Ex] - n_x[Ey]) \\ cn_x(c\{B_z\}) + (\alpha - \frac{1}{2})(n_y[Ex] - n_x[Ey]) \\ n_x[E_y] - n_y[E_x] + (\alpha - \frac{1}{2})c[B_z] \end{pmatrix}.
\]

And as we already mentioned we recover the centered flux for \( \alpha = \frac{1}{2} \) and the standard upwind flux for \( \alpha = 1 \).

### 3.3.4 Semi-discrete energy conservation

Let us consider the homogeneous equations (3.17)-(3.18) with \( \gamma = 0 \) and indexing the different elements and the discrete fields attached to the elements by the same index

\[
\frac{d}{dt} \int_{K_i} E_{h,i} \cdot F_h - c^2 \int_{K_i} B_{h,z,i} \text{curl} F_h - c^2 \int_{\partial K_i} (F_h \cdot \tau) B_N = 0, \quad \forall F_h \in P^2(K_i), \quad (3.44)
\]

\[
\frac{d}{dt} \int_{K_i} B_{h,z,i} C_{h,z} + \int_{K_i} E_{h,i} \cdot \text{curl} C_{h,z} - \int_{\partial K_i} (E_N \cdot \tau) C_{h,z} = 0 \quad \forall C_{h,z} \in P_k(K_i). \quad (3.45)
\]

**Proposition 8.** Assume periodic boundary conditions or \( E \cdot \tau = 0 \) on the boundary then the semi-discrete energy

\[
\int_\Omega (||E_h||^2 + c^2 B_{h,z}^2) \, dx
\]

is conserved in time.

**Proof.** Take \( F_h = E_{h,i} \) and \( C_{h,z} = B_{h,z,i} \) and sum the contributions over all the cells considering a centered flux, so that if \( K_i \) and \( K_j \) share an edge we have \( B_N = \frac{1}{2}(B_{h,z,i} + B_{h,z,j}) \) and \( E_N = \frac{1}{2}(E_{h,i} + E_{h,j}) \) Note also that, as the tangent vector \( \tau \) is linked to the outbound normal we have \( \tau_i = -\tau_j \) on the face shared by \( K_i \) and \( K_j \). We then get

\[
\sum_i \frac{d}{dt} \left( \int_{K_i} ||E_{h,i}||^2 + c^2 \int_{K_i} B_{h,z,i}^2 \right) - c^2 \int_{K_i} B_{h,z,i} \text{curl} E_{h,i} + c^2 \int_{K_i} E_{h,i} \cdot \text{curl} B_{h,z,i} - c^2 \int_{\partial K_i} (E_{h,i} \cdot \tau_i) B_N - c^2 \int_{\partial K_i} (E_N \cdot \tau_i) B_{h,z,i} = 0.
\]

Then using the Green formula (3.15)

\[
\frac{1}{2} \int_{K_i} E_{h,i} \cdot \text{curl} B_{h,z,i} = \int_{K_i} B_{h,z,i} \text{curl} E_{h,i} + \int_{\partial K_i} (E_{h,i} \cdot \tau_i) B_{h,z,i},
\]

so that, putting in the centered fluxes, with index \( j \) denoting an arbitrary element sharing an interface with \( K_i \) the boundary terms become

\[
- \frac{1}{2} \int_{\partial K_i} (E_{h,i} \cdot \tau_i) (B_{h,z,i} + B_{h,z,j}) - \frac{1}{2} \int_{\partial K_i} (E_{h,i} + E_{h,j}) \cdot \tau_i B_{h,z,i} + \int_{\partial K_i} (E_{h,i} \cdot \tau_i) B_{h,z,i} = - \frac{1}{2} \int_{\partial K_i} (E_{h,i} \cdot \tau_i) B_{h,z,j} - \frac{1}{2} \int_{\partial K_i} E_{h,j} \cdot \tau_i B_{h,z,i}
\]
so that
\[
\sum_i \frac{1}{2} \frac{d}{dt} \left( \int_{K_i} \| \mathbf{E}_{h,i} \|^2 + c^2 \int_{K_i} B_{h,z,i}^2 \right) - \frac{c^2}{2} \int_{\partial K_i} (\mathbf{E}_{h,i} \cdot \mathbf{\tau}_i) B_{k,z,j} - \frac{c^2}{2} \int_{\partial K_i} (\mathbf{E}_{h,j} \cdot \mathbf{\tau}_i) B_{h,z,i} = 0.
\]

Finally the remaining boundary terms vanish as for a shared boundary between \( K_i \) and \( K_j \) they appear on both sides with an opposite sign as \( \mathbf{\tau}_i = -\mathbf{\tau}_j \) and on the domain boundary they vanish because of the boundary conditions (periodic or \( \mathbf{E} \cdot \mathbf{\tau} = 0 \)). We thus get the desired result as the global square of the \( L^2 \) norm on \( \Omega \) is the sum of the square of the \( L^2 \) norms on the elements. \( \square \)

### 3.3.5 A leap-frog time scheme

For the time discretization, we can use any scheme for an ODE. Let us consider here the classical leap-frog scheme which computes the electrical field at integer time steps \( t_n = n \Delta t \) and the magnetic field at half integer time steps \( t_{n+\frac{1}{2}} = (n + \frac{1}{2}) \Delta t \). We notice that on each face \( \tau_{f,x} = n_{f,y} \) and \( \tau_{f,y} = -n_{f,x} \). We then get from equations (3.23)-(3.24), the following time scheme:

\[
M_K \mathbf{e}_{x}^{n+1} = M_K \mathbf{e}_{x}^{n} - \Delta t \left( c^2 D_{K}^{y} \mathbf{b}_{z}^{n+\frac{1}{2}} - \sum_{f \in \partial K} c n_{f,y} M_f \left( c \{ B_{z}^{n+\frac{1}{2}} \} + (\alpha - \frac{1}{2})(n_{f,y} \{ E_{x}^{n} \} - n_{f,x} \{ E_{y}^{n} \}) \right) + \frac{1}{\epsilon_0} M_K \mathbf{j}_x \right)
\]

\( (3.46) \)

\[
M_K \mathbf{e}_{y}^{n+1} = M_K \mathbf{e}_{y}^{n} - \Delta t \left( -c^2 D_{K}^{x} \mathbf{b}_{z}^{n+\frac{1}{2}} + \sum_{f \in \partial K} c n_{f,x} M_f \left( c \{ B_{z}^{n+\frac{1}{2}} \} + (\alpha - \frac{1}{2})(n_{f,y} \{ E_{x}^{n} \} - n_{f,x} \{ E_{y}^{n} \}) \right) + \frac{1}{\epsilon_0} M_K \mathbf{j}_y \right)
\]

\( (3.47) \)

\[
M_K \mathbf{b}_{z}^{n+\frac{1}{2}} = M_K \mathbf{b}_{z}^{n+\frac{1}{2}} - \Delta t \left( D_{K}^{y} \mathbf{e}_{x}^{n+1} - D_{K}^{x} \mathbf{e}_{y}^{n+1} + \sum_{f \in \partial K} M_f \left( n_{f,x} \{ E_{y}^{n+1} \} - n_{f,y} \{ E_{x}^{n+1} \} + c(\alpha - \frac{1}{2}) \{ B_{z}^{n+\frac{1}{2}} \} \right) \right)
\]

\( (3.48) \)

In order to keep a completely explicit time scheme, we have expressed the part of the flux involving the electric field in Ampère’s law at time \( t_n \) and the part of the flux involving the magnetic field in Faraday’s law at time \( t_{n+\frac{1}{2}} \). This scheme is thus only first order when upwinding is performed. These parts cancel for the centered flux in which case the scheme is of order 2 in time.
Chapter 4

Non linear conservation laws

4.1 Characteristics

We consider a generic scalar conservation law of the form

\[
\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0,
\]

with the initial condition \( u(0, x) = u_0(x) \).

Assuming \( f \in C^1(\mathbb{R}) \) and denoting by \( a(u) = f'(u) \), \( u \) also verifies

\[
\frac{\partial u}{\partial t} + a(u) \frac{\partial u}{\partial x} = 0.
\]

We now define the characteristic curves (or characteristics) associated to the conservation law as the solution of the differential equation

\[
\frac{dX}{dt} = a(u(t, X(t))).
\]

Then using the chain rule we have

\[
\frac{d}{dt}(u(t, X(t))) = \frac{\partial u}{\partial t}(t, X(t)) + \frac{dX}{dt} \frac{\partial u}{\partial x}(t, X(t)) = \frac{\partial u}{\partial t}(t, X(t)) + a(u(t, X(t)) \frac{\partial u}{\partial x}(t, X(t)) = 0,
\]

if \( u \) is a solution of the equation. From this we deduce that \( u(t, X(t)) = u(0, X(0)) = u_0(X(0)) \) which is independent of \( t \). It follows that the characteristic curves are the straight lines in the \((t, x)\) plane of equation:

\[
X(t) = X(0) + a(u_0(X(0)))t.
\]

And it follows that the solutions of the conservation law satisfy \( u(t, X(t)) = u_0(X(0)) \). This allows us to get the solution at a given point \( x \) and time \( t \) if the characteristic curve can be traced back in the \((t, x)\) plane to the line \( t = 0 \). This is not always the case when \( f \) is non linear.

Remark 7. In the linear case we have \( f(u) = au \), then the characteristics are the parallel lines of slope \( a \). They obey the equation \( X(t) = X(0) + at \). So they never cross and so taking \( X(t) = x \), we have \( X(0) = x - at \), and we recover the classical solution of the linear advection equation

\[
u(t, x) = u_0(x - at).
\]

A representation of the characteristics in the \((x, t)\) plane and of the propagation of a sine wave is represented in figure 4.1.
Figure 4.1: Representation of characteristics in the \((x,t)\) plane (left) and of the propagation of a sine wave (right).

Figure 4.2: Representation of characteristics in the \((x,t)\) plane for \(u_0(x) = \sin(2\pi x)\), (left) and of the propagation of the solution (right).

**Example.** The Burgers equation reads

\[
\frac{\partial u}{\partial t} + \frac{1}{2} \frac{\partial u^2}{\partial x} = 0.
\]

It corresponds to \(f(u) = \frac{1}{2} u^2\), so that \(a(u) = f'(u) = u\). Then the characteristic \(X(t; x)\) such that \(X(0) = x\) satisfies \(X(t; x) = x + tu_0(x)\). Hence if we consider \(x_2 > x_1\), we have

\[
X(t; x_2) - X(t; x_1) = x_2 - x_1 + t(u_0(x_2) - u_0(x_1)).
\]

Then if \(u_0\) is non decreasing we have \(X(t; x_2) > X(t; x_1)\) for all positive times, but if \(u_0\) is strictly decreasing then \(X(t; x_2) = X(t; x_1)\) for \(t_* = \frac{x_2 - x_1}{u_0(x_2) - u_0(x_1)}\). So the characteristics can cross and in this case the method of characteristics cannot be used to compute the solution which is then no more \(C^1\). A representation of the characteristics in the \((x,t)\) plane and of the propagation of a sine wave is represented in figure 4.2.

### 4.2 Weak solutions

#### 4.2.1 Definition

As we have seen in the case of Burgers equation, the characteristics associated to a non linear conservation law can cross, in which case the method of characteristics can non longer be used to find a solution. Actually when this happens the solution is no longer of class \(C^1\) and can even become discontinuous. We thus need another form of the equation \(\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0\) associated to the initial condition \(u(0, x) = u_0(x)\) to make it well defined for such a solution.

Let us first recall the definition of a classical solution:

**Definition 7.** Let \(u_0 \in C^1(\mathbb{R})\), then \(u\) is called a classical solution if \(u \in C^1(\mathbb{R}^+ \times \mathbb{R})\) and \(u\) satisfies

\[
\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \quad \text{and} \quad u(0, x) = u_0(x)
\]
in the classical sense.

Now we can define the notion of weak solution:

**Definition 8.** Let \( u_0 \in L^\infty(\mathbb{R}) \), then \( u \in L^\infty(\mathbb{R}^+ \times \mathbb{R}) \) is called a weak solution of our scalar conservation law if \( u \) satisfies

\[
\int_0^T \int_{-\infty}^{+\infty} u \frac{\partial \varphi}{\partial t} + f(u) \frac{\partial \varphi}{\partial x} \, dt \, dx + \int_{-\infty}^{+\infty} u_0(x) \varphi(0, x) \, dx = 0 \quad \forall \varphi \in C^1_c([0, T] \times \mathbb{R}).
\]

Let \( u \in C^1 \) be a classical solution of \( \frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0 \). Multiplying this equation by \( \varphi \in C^1_c([0, T] \times \mathbb{R}) \) and integrating by parts, we get

\[
0 = \int_0^T \int_{-\infty}^{+\infty} (\partial u/\partial t + \partial f(u)/\partial x) \, \varphi \, dt \, dx = -\int_0^T \int_{-\infty}^{+\infty} \left( u \frac{\partial \varphi}{\partial t} + f(u) \frac{\partial \varphi}{\partial x} \right) \, dx \, dt + \int_{-\infty}^{+\infty} u_0(x) \varphi(0, x) \, dx
\]

an thus \( u \) is also a weak solution. So the notion of weak solutions generalizes the notion of classical solutions.

### 4.2.2 The Rankine-Hugoniot condition

In most physical cases the solution is actually piecewise \( C^1 \) and there is only one (or a few) lines of discontinuities in the space-time plane. The Rankine-Hugoniot condition gives a constraint on the discontinuity along a line for the piecewise smooth solution to be a weak solution of the equation.

**Theorem 2.** Assume the half space \( \mathbb{R}^+ \times \mathbb{R} \) is split into two parts \( M_1 \) and \( M_2 \) by a smooth curve \( S \) parametrized by \( (t, \sigma(t)) \) with \( \sigma \in C^1(\mathbb{R}^+) \). We also assume that \( u \in L^\infty(\mathbb{R}^+ \times \mathbb{R}) \) and that \( u_1 = u|_{M_1} \in C^1(\bar{M}_1) \) and \( u_2 = u|_{M_2} \in C^1(\bar{M}_2) \) with \( u_1 \) and \( u_2 \) two classical solutions of our equation in respectively \( M_1 \) and \( M_2 \).

Then \( u \) is a weak solution if and only if

\[
[u_1(t, \sigma(t)) - u_2(t, \sigma(t))] \sigma'(t) = f(u_1(t, \sigma(t))) - f(u_2(t, \sigma(t))) \quad \forall t \in \mathbb{R}^+.
\]

Relation (4.2) is called the Rankine-Hugoniot condition. It is often written in the simplified manner

\[
(u_1 - u_2)s = f(u_1) - f(u_2),
\]

where \( s = \sigma'(t) \) is the propagation speed of the discontinuity.

**Proof.** Assume \( u \) is a weak solution of our equation. Then by definition

\[
\int_0^T \int_{-\infty}^{+\infty} u \frac{\partial \varphi}{\partial t} + f(u) \frac{\partial \varphi}{\partial x} \, dt \, dx + \int_{-\infty}^{+\infty} u_0(x) \varphi(0, x) \, dx = 0 \quad \forall \varphi \in C^1_c([0, T] \times \mathbb{R}).
\]

Then we can split the first double integral into an integral on \( M_1 \) and an integral on \( M_2 \) and integrate these integrals by parts as \( u \) is \( C^1 \) on each of these domains. We then get using Green’s divergence formula, with \( \nu \) denoting the outward unit normal vector,

\[
\int_{\Omega} \psi \nabla \cdot \mathbf{v} \, d\mathbf{x} = -\int_{\Omega} \mathbf{v} \cdot \nabla \psi \, d\mathbf{x} \int_{\partial \Omega} \mathbf{v} \cdot \nu \psi \, ds
\]

that

\[
\int_{M_1} u \frac{\partial \varphi}{\partial t} + f(u) \frac{\partial \varphi}{\partial x} \, dt \, dx = -\int_{M_1} \left( \frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} \right) \varphi \, dt \, dx + \int_{\partial M_1} \left( \frac{u}{f(u)} \right) \cdot \nu_1 \varphi \, ds
\]

56
Because \( u = u_1 \) is a classical solution in \( M_1 \) the first term on the right hand side vanishes. The boundary term is composed of a part on the \( t = 0 \) line which cancels with the part of the integral on the initial condition which is in \( M_1 \) and a part that can be parametrized using \( \sigma \). A tangent vector to the parametrized curve \( S \) is given by \((1, \sigma'(t))\). Hence the outward unit normal is given by

\[
\nu_1 = \frac{1}{\sqrt{1 + \sigma'(t)^2}} \begin{pmatrix} \sigma'(t) \\ -1 \end{pmatrix}.
\]

Recall that the integral over a curve parametrized by \( \gamma: [a, b] \to \mathbb{R}^n \) is defined by

\[
\int_S F \, ds = \int_a^b F(\gamma(t)) \| \dot{\gamma}(t) \|_2 \, dt.
\]

Thus the part of the boundary integral corresponding to the integral on \( S \), which is parametrized by \( \gamma(t) = (t, \sigma(t)) \) becomes

\[
\int_S \left( \begin{array}{c} u_1 \\ f(u_1) \end{array} \right) \cdot \nu_1 \varphi \, ds = \int_0^T \frac{1}{\sqrt{1 + \sigma'(t)^2}} \left[ \sigma'(t)u_1(t, \sigma(t)) - f(u_1(t, \sigma(t))) \right] \varphi(t, \sigma(t)) \sqrt{1 + \sigma'(t)^2} \, dt,
\]

\[
= \int_0^T \left[ \sigma'(t)u_1(t, \sigma(t)) - f(u_1(t, \sigma(t))) \right] \varphi(t, \sigma(t)) \, dt.
\]

In the same way for the part on \( M_2 \) for which the outward normal is \( \nu_2 = -\nu_1 \) we get

\[
\int_S \left( \begin{array}{c} u_1 \\ f(u_1) \end{array} \right) \cdot \nu_1 \varphi \, ds = \int_0^T \left[ -\sigma'(t)u_2(t, \sigma(t)) + f(u_2(t, \sigma(t))) \right] \varphi(t, \sigma(t)) \, dt.
\]

Adding the two pieces we find

\[
\int_0^T \left[ \sigma'(t)(u_1(t, \sigma(t)) - u_2(t, \sigma(t))) + f(u_2(t, \sigma(t))) - f(u_1(t, \sigma(t))) \right] \varphi(t, \sigma(t)) \, dt = 0.
\]

This is true for all \( C^1 \) functions \( \varphi \), hence its factor in the integral vanishes. This corresponds to the Rankine-Hugoniot relation.

Conversely, the same calculation proves that if the Rankine-Hugoniot relation is satisfied then \( u \) is a weak solution.

**Example:** Consider the Burgers equation which corresponds to \( f(u) = \frac{u^2}{2} \):

\[
\frac{\partial u}{\partial t} + \frac{1}{2} \frac{\partial u^2}{\partial x} = 0,
\]

with the initial condition

\[
u_0(x) = \begin{cases} 
0 & \text{if } x < 0, \\
1 & \text{if } x \geq 0. 
\end{cases}
\]

1. Using the Rankine-Hugoniot relation let us see under what condition a piecewise constant solution can be a weak solution of this problem. Because the characteristics of the Burgers equation are straight lines, it is natural to look for piecewise smooth solutions which are
separated by a straight line in the $t - x$ plane. A straight line can be parametrized by $(t, \alpha t)$.

Then the candidate piecewise solution writes

$$u(t, x) = \begin{cases} 0 & \text{if } x < \alpha t, \\ 1 & \text{if } x \geq \alpha t. \end{cases}$$

As $\sigma(t) = \alpha t$, we have $s = \sigma'(t) = \alpha$ and so the Rankine-Hugoniot condition $(u_1 - u_2)s = f(u_1) - f(u_2)$ becomes

$$(0 - 1)\alpha = \frac{1}{2}(0 - 1)$$

and thus the only weak solution consisting of two constant states is $u$ corresponding to $\alpha = \frac{1}{2}$. Such a discontinuous solution is called a shock.

2. Let us now define

$$u(t, x) = \begin{cases} 0 & \text{if } x < 0, \\ x/t & \text{if } 0 \leq x < t, \\ 1 & \text{if } x \geq t. \end{cases}$$

The two constant values are obviously classical solutions in their part of the domain. On the other hand $u(t, x) = x/t$ verifies $\frac{\partial u}{\partial t}(t, x) = -\frac{x}{t^2}$ and $\frac{\partial^2 u}{\partial x^2}(t, x) = \frac{2}{t^2}$ so that

$$\frac{\partial u}{\partial t}(t, x) + \frac{1}{2} \frac{\partial^2 u}{\partial x^2}(t, x) = -\frac{x}{t^2} + \frac{1}{2} \frac{2x}{t^2} = 0$$

and $u(t, x) = x/t$ is also a classical solution on its domain of definition. It is straightforward to verify that this solution is continuous on the lines $x = 0$ and $x = t$, so that the Rankine-Hugoniot conditions are automatically verified (no jump) but not $C^1$. This solution which goes continuously from one constant state to the other, but is not $C^1$ on two lines is called a rarefaction wave.

So we proved that this is also a weak solution of the Burgers equation, which means that the equation we are considering in this example has at least two weak solutions. In practice it is possible to construct other weak solutions with several constant states separated by a line of discontinuity (see [3] for examples).

### 4.2.3 Entropy solution

The last example shows us that the uniqueness of a weak solution is not guaranteed. However the physical solution is unique. This means that there is a piece missing in our theory that will enable us to characterize the physically correct weak solution. The idea that will lead us there is that an exact conservation law is unlikely to be physical. There is always a small amount of dissipation. Mathematically this can be modelled by adding a small diffusion term to our conservation law: for a small positive $\epsilon$ we consider

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} - \epsilon \frac{\partial^2 u}{\partial x^2} = 0. \tag{4.3}$$

Associated to a smooth initial condition this equation has a unique smooth solution. And the physically correct solution of our conservation law, will be the unique limit of this solution (which depends on $\epsilon$) when $\epsilon \to 0$. This unique solution can be characterized using the notion of entropy.

**Definition 9.** Let $U, F \in C^2(\mathbb{R})$ such that

(i) $U$ is strictly convex,
(ii) \( F' = U' f' \)

Then \( U \) is called an entropy and \( F \) an entropy flux associated to the conservation law

\[
\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0.
\]

**Theorem 3.** Assume the conservation law \( \frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0 \) can be associated to an entropy \( U \) with entropy flux \( F \). Then if \( (u_\epsilon)_\epsilon \) is a sequence of smooth solutions of (4.3) such that \( \|u_\epsilon\|_{L^\infty} \) is bounded uniformly in \( \epsilon \) and \( u_\epsilon \to u \) almost everywhere then \( u \) is a weak solution of the conservation law and verifies the entropy condition

\[
\frac{\partial U(u)}{\partial t} + \frac{\partial F(u)}{\partial x} \leq 0
\]

in the weak sense, i.e.

\[
\int_0^T \int_{-\infty}^{+\infty} (U(u) \frac{\partial \varphi}{\partial t} + F(u) \frac{\partial \varphi}{\partial x}) \, dt \, dx \geq 0 \quad \forall \varphi \in C^1_c([-\infty, T]\times\mathbb{R}), \varphi \geq 0.
\]

**Definition 10.** A weak solution that verifies the entropy condition (4.4) is called an entropy solution.

It has been proven by Kruzhkov (see [3] and references therein) that for a bounded initial condition \( u_0 \) and a smooth flux \( f(u) \), there exists a unique entropy solution of the scalar conservation law.

In the case of a strictly convex flux \( (f''(u) > 0) \) which is verified by the Burgers equation, there is a simple characterization of the entropy condition for a shock, which is called the Lax entropy condition:

\[
f'(u_L) > s > f'(u_R), \quad \text{with } s = \frac{f(u_R) - f(u_L)}{u_R - u_L}.
\]

This is the Rankine-Hugoniot setting and \( s \) is the shock speed. This tells us that the shock is the entropy solution provided the characteristics lines cross at the shock line \( x = st \). Else the entropy solution is a rarefaction wave.

**Remark 8.** Because \( f \) is strictly convex \( f' \) is increasing and the entropy condition can be satisfied only if \( u_L > u_R \) and in this case because of the strict convexity, \( s \) which is the slope of the line joining \((u_L, f(u_L))\) and \((u_R, f(u_R))\) on the graph of \( f \) lies necessarily between \( f'(u_R) \) and \( f'(u_L) \).

**Example:** Consider again the Burgers equation \( \frac{\partial u}{\partial t} + \frac{1}{2} \frac{\partial u^2}{\partial x} = 0 \) with piecewise constant initial conditions \( u_0(x) = u_L \) for \( x < 0 \) and \( u_0(x) = u_R \) for \( x \geq 0 \). We computed previously the characteristics for Burgers equation which are given by \( X(t; x) = x + tu_0(x) \). So in our case we have two families of characteristics, those issued from the negative values of \( x \) at \( t = 0 \) which become \( X(t; x) = x + tu_L \) and those issued from the positive values of \( x \) at \( t = 0 \) which become \( X(t; x) = x + tu_R \). The flux of the Burgers equation \( f(u) = \frac{1}{2}u^2 \) is strictly convex, so the Lax criterion applies and the Lax entropy condition can be satisfied only if \( u_L > u_R \) which is the case when the characteristics cross.

**Remark 9.** In the case of a linear flux \( f(u) = au \) for some constant \( a \). If \( U \) is an entropy, the associated entropy flux is \( F = au \) and we have, as \( u \) satisfies \( \frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0 \),

\[
\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} = \frac{\partial U}{\partial t} + a \frac{\partial U}{\partial x} = U'(t)(\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x}) = 0.
\]

So the entropy condition is always verified in this case.

This is a simple case, which also can appear for some of the waves in non linear systems. In this case two different values are just propagated side by side without interfering. This case is called a contact discontinuity.
We will restrict in this lecture to the two simplest cases of scalar conservation laws, namely the cases of a strictly convex flux and of a linear flux. These often occur in applications, but not always. A more complex study is needed in other cases. We refer to the book of Leveque [7] for details.

4.3 The Riemann problem

Let us here compute the exact entropy solution of the Riemann problem for a strictly convex flux i.e. \( f''(u) > 0 \).

The Riemann problem for the 1D scalar non linear conservation laws reads

\[
\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0,
\]

\[
u_0(x) = \begin{cases} 
  u_L & \text{if } x < 0, \\
  u_R & \text{if } x \geq 0.
\end{cases}
\]

**Case 1: \( u_L < u_R \).** We saw previously that in this case the shock does not satisfy the Lax entropy condition and is not entropic. We thus look for a rarefaction wave which is continuous across each of the characteristics issued from the left and right of 0. Between these two characteristics, we look for an explicit solution of the form \( u(t, x) = v(x/t) \). Plugging this function into our equation, we get

\[
-\frac{x}{t^2} v'(\frac{x}{t}) + f'(v(\frac{x}{t})) \frac{1}{t} v'(\frac{x}{t}) = 0.
\]

Setting \( \xi = x/t \) this becomes

\[
(f'(v(\xi)) - \xi)v'(\xi) = 0.
\]

So we get two kinds of solution, either \( v(\xi) = C \) (\( C \) constant), or \( v \) such that \( f'(v(\xi)) - \xi = 0 \). The constant solution is not possible as it would yield non entropic shocks. As \( f'' > 0 \), \( f' \) is strictly increasing and thus invertible. Hence the solution can be expressed as \( v(\xi) = (f')^{-1}(\xi) \).

Finally the entropy solution of the Riemann problem in this case is the rarefaction wave defined by

\[
u(t, x) = \begin{cases} 
  u_L & \text{if } x < f'(u_L)t, \\
  (f')^{-1}(x/t) & \text{if } f'(u_L)t \leq x < f'(u_R)t, \\
  u_R & \text{if } x \geq f'(u_R)t.
\end{cases}
\]

**Case 2: \( u_L > u_R \).** In this case we saw that the shock separating the two constant states \( u_L \) and \( u_R \) and propagating at the speed defined by the Rankine-Hugoniot condition

\[
s = \frac{f(u_L) - f(u_R)}{u_L - u_R}
\]

is entropic and so is the unique physical solution. This is defined by

\[
u(t, x) = \begin{cases} 
  u_L & \text{if } x < st, \\
  u_R & \text{if } x \geq st.
\end{cases}
\]
4.4 Numerical methods

4.4.1 The Godunov method

The idea of the Godunov method is that if one is looking for piecewise constant solutions then as the solution is propagating at a finite speed which is known, the solution can be computed exactly by solving a Riemann problem for each cell interface. If \( s = \max |f'(u)| \) denotes the fastest possible wave then the neighboring Riemann problems do not interact provided the time step \( \Delta t \) verifies \( s \Delta t \leq \frac{1}{2} \Delta x \). In principle the solution at time \( t_{n+1} \) could be computed by integrating over the solutions of all the Riemann problems at this time to find the new constant value on each cell by averaging. However, this could be quite complicated. The Godunov method can be expressed in a much simpler way by integrating the starting equation in space and time over one cell and one time step:

\[
\int_{t_n}^{t_{n+1}} \int_{x_i}^{x_{i+1}} \left( \frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} \right) \, dx \, dt = 0.
\]

Denoting by \( u_{i+\frac{1}{2}}^{n+1} = \frac{1}{\Delta x} \int_{x_i}^{x_{i+1}} u(t_n, x) \, dx \) the average of \( u \) over a cell at time \( t_n = n \Delta t \), this yields

\[
\Delta x (u_{i+\frac{1}{2}}^{n+1} - u_{i+\frac{1}{2}}^{n}) + \int_{t_n}^{t_{n+1}} (f(u(t, x_{i+1})) - f(u(t, x_i))) \, dt = 0.
\]

Now as \( u \) is assumed to be constant on each cell and equal to its average, the value of the flux \( f(u(t, x)) \) can be exactly computed by solving the Riemann problem between the left value \( u_{i-\frac{1}{2}}^{n} \) and the right value \( u_{i+\frac{1}{2}}^{n} \) at the interface \( x_i \). Note that we only need the solution of the Riemann problem exactly on the interface and this solution does not depend on time as long as there is no interaction with the neighboring problem on the interface, which cannot occur if \( s \Delta t \leq \Delta x \). Here because we are only interested in the solution directly at the interface we gain a factor \( \frac{1}{2} \) as we do not care about other interactions.

As we saw previously, assuming that the flux \( f(u) \) is strictly convex, the entropy solution of the Riemann problem can be either a shock or a rarefaction wave, but \( u(t, x_i) \) will take a value different from \( u_{i-\frac{1}{2}}^{n} \) or \( u_{i+\frac{1}{2}}^{n} \) only if the rarefaction wave has an interaction with \( x = x_i \) which is the case only if \( f'(u_L) < 0 \) and \( f'(u_R) > 0 \). Let us clearly distinguish the cases and express the solution of the local Riemann problem at the interface \( x = x_i \). Recall that we consider the local Riemann problem with the initial condition (at time \( t_n \))

\[
\begin{aligned}
  u(t_n, x) &= \begin{cases} 
    u_{i-\frac{1}{2}}^{n} & \text{if } x < x_i, \\
    u_{i+\frac{1}{2}}^{n} & \text{if } x \geq x_i.
  \end{cases}
\end{aligned}
\]

If \( u_{i-\frac{1}{2}}^{n} < u_{i+\frac{1}{2}}^{n} \) the entropy solution is a rarefaction wave so that

\[
\begin{aligned}
  u(t, x_i) &= \begin{cases} 
    u_{i-\frac{1}{2}}^{n} & \text{if } f'(u_{i-\frac{1}{2}}^{n}) > 0, \\
    (f')^{-1}(0) & \text{if } f'(u_{i-\frac{1}{2}}^{n}) \leq 0 \leq f'(u_{i+\frac{1}{2}}^{n}), \\
    u_{i+\frac{1}{2}}^{n} & \text{if } f'(u_{i+\frac{1}{2}}^{n}) < 0.
  \end{cases}
\end{aligned}
\]

If \( u_{i-\frac{1}{2}}^{n} > u_{i+\frac{1}{2}}^{n} \) the entropy solution is a shock wave so that

\[
\begin{aligned}
  u(t, x_i) &= \begin{cases} 
    u_{i-\frac{1}{2}}^{n} & \text{if } s > 0, \\
    u_{i+\frac{1}{2}}^{n} & \text{if } s < 0,
  \end{cases}
\end{aligned}
\]
where $s$ is the shock speed given by the Rankine-Hugoniot condition: 
\[ s = \frac{f(u_L) - f(u_R)}{u_L - u_R}. \]

Using the exact solution of the Riemann problem, the numerical flux defined by $g^n_i = g(u^n_{i-\frac{1}{2}}, u^n_{i+\frac{1}{2}})$ is taken to be the flux associated to the exact solution of the Riemann problem, i.e. $f(u(t, x_i))$ for $x = x_i$ and $t > t_n$. Noticing that for a strictly convex flux, the minimum of $f$ is reached at the points $(f')^{-1}(0)$ and going through the different cases we find that

\[ g^n_i = \begin{cases} 
\min_{u \in [u^n_{i-\frac{1}{2}}, u^n_{i+\frac{1}{2}}]} f(u) & \text{if } u_{i-\frac{1}{2}} < u_{i+\frac{1}{2}}, \\
\max_{u \in [u^n_{i-\frac{1}{2}}, u^n_{i+\frac{1}{2}}]} f(u) & \text{if } u_{i-\frac{1}{2}} > u_{i+\frac{1}{2}}.
\end{cases} \]

With this, the flux at the interface can be computed and the Godunov scheme is well defined (here in the case of a strictly convex flux). In the case of a linear flux, the Godunov scheme amounts to the first order upwind scheme. For other smooth fluxes, the Riemann problem can generally also be solved even though it is more complicated.

### 4.4.2 Approximate Riemann solvers

Computing the exact solution of the local Riemann problems is in principle the best method. However it can be complicated or numerically expensive. Moreover, as there are other sources of numerical errors anyway, it might be numerically as good or almost as good to replace the solution of the exact Riemann problem by some approximation.

1. The Rusanov flux (also called Local Lax-Friedrichs flux). The idea behind this flux, instead of approximating the exact Riemann solver, is to recall that the entropy solution is the limit of viscous solutions and to take a centred flux to which some viscosity (with the right sign) is added. This flux is given by

\[ g^n_i = g(u^n_{i-\frac{1}{2}}, u^n_{i+\frac{1}{2}}) = \frac{1}{2} \left[ f(u_{i-\frac{1}{2}}) + f(u_{i+\frac{1}{2}}) - \max_{u \in [u_{i-\frac{1}{2}}, u_{i+\frac{1}{2}}]} |f'(u)| (u_{i+\frac{1}{2}} - u_{i-\frac{1}{2}}) \right]. \]

Taking the viscosity parameter as the largest local wave speed guarantees the stability of the scheme. On the other hand taking the largest local wave speed (instead of globally) adds less viscosity in regions where the solution is smooth and $u_{i-\frac{1}{2}}$ is close to $u_{i+\frac{1}{2}}$.

2. The Roe flux (also called Murman or Murman-Roe in the scalar case): the idea here is to linearize the flux $f(u)$ around the cell interface and then use the upwind flux. Assuming two constant values $u_L$ and $u_R$ on each side, this amounts to replacing $f(u)$ by $a(u_L, u_R)u$ with a well chosen velocity $a(u_L, u_R)$ that will enable us to get a flux which is close to the flux given by the solution of the exact Riemann problem. Looking at the solution of the exact Riemann problem an approximation that looks interesting for both rarefaction wave and shocks is to take the Rankine-Hugoniot velocity

\[ a(u_L, u_R) = \frac{f(u_L) - f(u_R)}{u_L - u_R}, \]

if there is a discontinuity i.e. $u_L \neq u_R$, and simply $a(u_L, u_R) = f'(u_L) = f'(u_R)$ if $u_L = u_R$. Indeed if $u_L$ and $u_R$ are close, which will be the case in smooth regions $a(u_L, u_R)$ defines a good approximation of both $f'(u_L)$ and $f'(u_R)$. Moreover if both $f'(u_L)$ and $f'(u_R)$ have the same sign $a(u_L, u_R)$ will also have this sign, so that the same value of the numerical flux will be obtained as in the exact Riemann problem was solved. The only case where this does not
yield the same solution as the exact Riemann solver is the case of a rarefaction wave with $f'(u_L)$ and $f'(u_R)$ of different signs. In this case, it is possible that the solution obtained by the scheme is not the correct entropy solution, the non entropic shock being approximated instead of the rarefaction wave. So a fix, known as entropy fix, is needed. For this let us express the Murman-Roe flux as a centred flux with some added viscosity.

The Murman-Roe flux is an upwind flux for the linearized problem we just introduced. The case $u_L = u_R$ being trivial, we consider only $u_L \neq u_R$. Then

$$g(u_L, u_R) = \begin{cases} f(u_L) & \text{if } a(u_L, u_R) > 0, \\ f(u_R) & \text{if } a(u_L, u_R) \leq 0. \end{cases}$$

If $a(u_L, u_R) = \frac{f(u_L) - f(u_R)}{u_L - u_R} > 0$ and $u_L \neq u_R$ we have

$$f(u_L) = \frac{1}{2} \left( f(u_L) + f(u_R) - \frac{f(u_R) - f(u_L)}{u_R - u_L} (u_R - u_L) \right),$$

and if $a(u_L, u_R) = \frac{f(u_L) - f(u_R)}{u_L - u_R} < 0$ and $u_L \neq u_R$ we have

$$f(u_R) = \frac{1}{2} \left( f(u_L) + f(u_R) + \frac{f(u_R) - f(u_L)}{u_R - u_L} (u_R - u_L) \right),$$

so that we can define the numerical flux in all cases by

$$g(u_L, u_R) = \frac{1}{2} \left( f(u_L) + f(u_R) - |a(u_L, u_R)|(u_R - u_L) \right).$$

Here we also see that the numerical viscosity vanishes when $a(u_L, u_R) \approx 0$ which can happen close to the minimum of the convex function $f$. Then a non entropic shock might be selected by the scheme. A simple fix, introduced by Harten, consists in smoothing the graph of the absolute value close to 0 (see [7] for details). This consists in replacing the absolute value in the formula defining the flux by

$$\phi(\lambda) = \begin{cases} |\lambda| & |\lambda| \geq \epsilon, \\ (\lambda^2 + \epsilon^2)/(2\epsilon) & |\lambda| < \epsilon. \end{cases}$$

This ensures that $\phi(\lambda) \geq \epsilon$ and that there is always some dissipation. This works and yields the correct entropy solution provided $\epsilon$ is well tuned to the problem at hand.

### 4.4.3 Higher order methods

Higher order methods can be designed using the Finite Volume or the Discontinuous Galerkin methodology. In both cases what needs to be defined is the numerical flux constructed from the two values $u_L$ and $u_R$ on each side of the interface. The same numerical fluxes as previously can be used in this case. However for the scheme to be stable close to discontinuities a limiting procedure needs to be introduced so that the scheme falls back to first order in the cells where a discontinuity is detected. There is a vast literature on limiters that we shall not detail here.

### 4.4.4 Strong stability preserving (SSP) Runge-Kutta methods.

Fluxes are generally constructed so that the associated scheme has some stability properties when used with a first order explicit Euler time solver. When higher order methods in $x$ are used, it makes sense to use also higher order methods in time. A specific class of Runge-Kutta methods has been developed to keep the strong stability properties of the explicit Euler solver at each stage (See [6]).
4.5 Nonlinear systems of conservation laws

Going from the scalar case to systems in the non-linear case, is similar to what is done in the linear case. The hyperbolicity of the system is essential so that the system can be locally diagonalized and the eigenvalues explicitly used in the definition of the flux.

The derivation of a Finite Volume or Discontinuous Galerkin scheme can be done component by component and so reduces to the scalar case except for the definition of the numerical flux which in general mixes the different components and needs to be specific to the system at hand. We shall restrict in this lecture to the introduction of two of the most used numerical fluxes, namely the Rusanov (or local Lax-Friedrichs) flux and the Roe flux.

4.5.1 The Rusanov flux

As in the scalar case, the main idea here is to use a centred flux to which just enough dissipation is added to ensure stability in all cases. In the scalar case the needed viscosity was given by the largest local wave speed. A system of \(n\) components corresponds to the superposition of \(n\) waves the local speed of each being given by the corresponding eigenvalue. So taking the viscosity coefficient in the flux as the maximum over all eigenvalues should do the job. This yields the Rusanov flux for systems, which is the simplest stable flux. It is defined for a linear system of the form

\[
\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x} = 0,
\]

as

\[
\mathbf{G}(\mathbf{U}_L, \mathbf{U}_R) = \frac{1}{2} \left( \mathbf{F}(\mathbf{U}_L) + \mathbf{F}(\mathbf{U}_R) - \max_{\lambda \in [\mathbf{U}_L, \mathbf{U}_R]} |\lambda(\mathbf{F}'(\mathbf{U}))| (\mathbf{U}_R - \mathbf{U}_L) \right),
\]

where \(\max_{\mathbf{U} \in [\mathbf{U}_L, \mathbf{U}_R]} |\lambda(\mathbf{F}'(\mathbf{U}))|\) denotes the maximum modulus of the eigenvalues of the Jacobian matrix \(\mathbf{F}'(\mathbf{U})\).

4.5.2 The Roe flux

Roe’s method consists in locally linearizing the non-linear flux with a well chosen procedure. The linearized matrix between two constant states \(\mathbf{U}_L\) and \(\mathbf{U}_R\) is denoted by \(A(\mathbf{U}_L, \mathbf{U}_R)\) and constructed such that the following properties are verified:

- \(\mathbf{F}(\mathbf{U}_R) - \mathbf{F}(\mathbf{U}_L) = A(\mathbf{U}_L, \mathbf{U}_R)(\mathbf{U}_R - \mathbf{U}_L)\).
- \(A(\mathbf{U}, \mathbf{U}) = \mathbf{F}'(\mathbf{U})\).
- \(A(\mathbf{U}_L, \mathbf{U}_R)\) is diagonalizable, has real eigenvalues and a complete system of eigenvectors.

Such a matrix is not always easy to find, but there are procedures, described in [7] for example, to construct them. Moreover classical Roe matrices are known for the most usual systems [7].

Once the Roe matrix is defined, the flux can be computed by solving the corresponding linear Riemann problem that we treated in Chapter 3. Let us rewrite the formula, so that we can also include the entropy fix, which as in the scalar case is needed for non-linear systems to make sure that the scheme always converges to the correct entropy solution.

In the case of linear systems, the flux was defined as \(A\mathbf{U}(t, 0) = A_+\mathbf{U}_L + A_-\mathbf{U}_R\). Defining the absolute value of a matrix as \(|A| = A_+ - A_-\), the flux can also be expressed as

\[
A\mathbf{U}(t, 0) = \frac{1}{2} (A\mathbf{U}_L + A\mathbf{U}_R - |A|(\mathbf{U}_R - \mathbf{U}_L)).
\]
Using the properties of the Roe matrix in the non linear case, the same expression will be used to define the Roe flux:

$$G(U_L, U_R) = \frac{1}{2} (A(U_L, U_R)U_L + A(U_L, U_R)U_R - |A(U_L, U_R)|(U_R - U_L)).$$

The same entropy fix consisting in replacing the absolute value by the function $\phi$ which is bounded away from 0 can be applied in this formula.
Bibliography


