NON OSCILLATORY INTERPOLATION METHODS APPLIED TO KINETIC EQUATIONS FOR PLASMAS

Trabajo de investigación

dirigido por

Prof. José Antonio Carrillo de la Plata

presentado el 26 de septiembre de 2005 por

Francesco Vecil.

Composición del tribunal.

Presidente	Prof. Lluís Alsedà Soler	Universitat Autònoma de Barcelona
Secretario	Prof. Francesc Xavier Mora Giné	Universitat Autònoma de Barcelona
Vocal	Prof. Josep Maria Mondelo González	Universitat Autònoma de Barcelona

Contents

Introduction

The goal of this work is to devise new numerical schemes to simulate two kinds of objects:

- An *electronic device* is a physical solid state device. It has a fixed electronic lattice, where impurities are injected in order to modify its electric properties. A standard semiconductor is made of Silicon, which is a tetravalent atom and might be doped by injecting Phosphoros (P) or Arsenic (As) to obtain a negative doping (there is an excess of free electrons), or by injecting Boron (B), which is electron-deficient (it possesses a vacant p-orbital), to obtain a positive doping: a sort of excess of positive charges is produced, which is in fact an excess of electron holes. To give an idea of the dimensions of the doping phenomenon, in 1 cm^3 of Silicon there are order 10^{24} atoms; a low doping means injecting 10^{13} atoms per cm^3, while a high doping 10^{20} atoms per cm^3.
  Commercial electronic devices such as MOSFETs, MESFETs, pn-junctions and diodes are based on several semiconductors: Silicon (Si), Galium Arsenide (GaAs), Silicon Carbide (SiC),...; with different dopings. The doping of semiconductors is essential in order to create a potential barrier high enough to induce an electron current.

- A *plasma* is a ionized gas, where the electrons of the most external orbits are separated from the atom. Plasma is the fourth state of matter after gas, and is obtained by warming it $10^6 - 10^8$ Celsius degrees: a dissociation between positive, negative and neutral charges happens. 99% of the matter is made of plasmas. They are commonly used in fluorescent lamps and neon signs, and are a central point of research in fusion energy.

The numerical schemes and simulations have four main aspects to model and combine:

- *Vlasov*'s operator, for the kinetic description of the motion of the carriers under the effects of a positional force field F and the free motion. As a huge number of particles is involved, a dynamical description is unrealizable and must be replaced by a probabilistic description.

- *Poisson*'s equation, for the computation of both the auto-consistent and the external electric potential.

- *Boltzmann*'s operator, for the description of collisional models: we suppose the particles to have interactions, called collisions, with the other particles or with the fixed lattice.

- *Maxwell*'s equations, for the electro-magnetic coupling and the computation of magnetic fields.

The main properties of these equations are reviewed in Chapter 1.

A central instrument for both solving the advection part (i.e. performing a time step in Vlasov's equation) and for the coupling of Vlasov's and Boltzmann's equations is *Strang's splitting* scheme, which was originally introduced in 1976 by Cheng and G. Knorr [7] based on a Strang's article appeared in 1968: the performance of a time step in a 2-dimensional system is reduced to the solution in either direction (at intermediate steps) and to a suitable recombination of the results. For instance, the advection step is obtained by performing a $\frac{\Delta t}{2}$-time step in x-direction, then a Δt-time step in v-direction, and finally a $\frac{\Delta t}{2}$-time step in x-direction. These issues will be dealt within Chapters 3 and 4 in detail.

Several problems might appear during the numerical computation. One of them is called *filamentation* of the phase space: this word means the formation of strong gradients and oscillations (see Figure where an example is shown) in the (x, v)-space. This is a physical phenomenon, essential in the evolution of Vlasov-based models, and must be treated using numerical methods which take it into account in the "cleanest" way.

The numerical scheme must not add *spurious oscillations*, numerical, non-physical phenomena. Lagrange interpolation polynomials do not behave in a proper way: even if they provide a very accurate reconstruction for smooth functions, when high gradients appear (which is the case of almost all the simulations of Vlasov-Maxwell models) they produce noise in a non-acceptable quantity. We propose to replace Lagrange interpolation by WENO (Weighted Essentially Non Oscillatory) interpolation methods, developed to control the total variation of the interpolation. This interpolation procedure is introduced, developed and adapted to our purposes in Chapter 2.

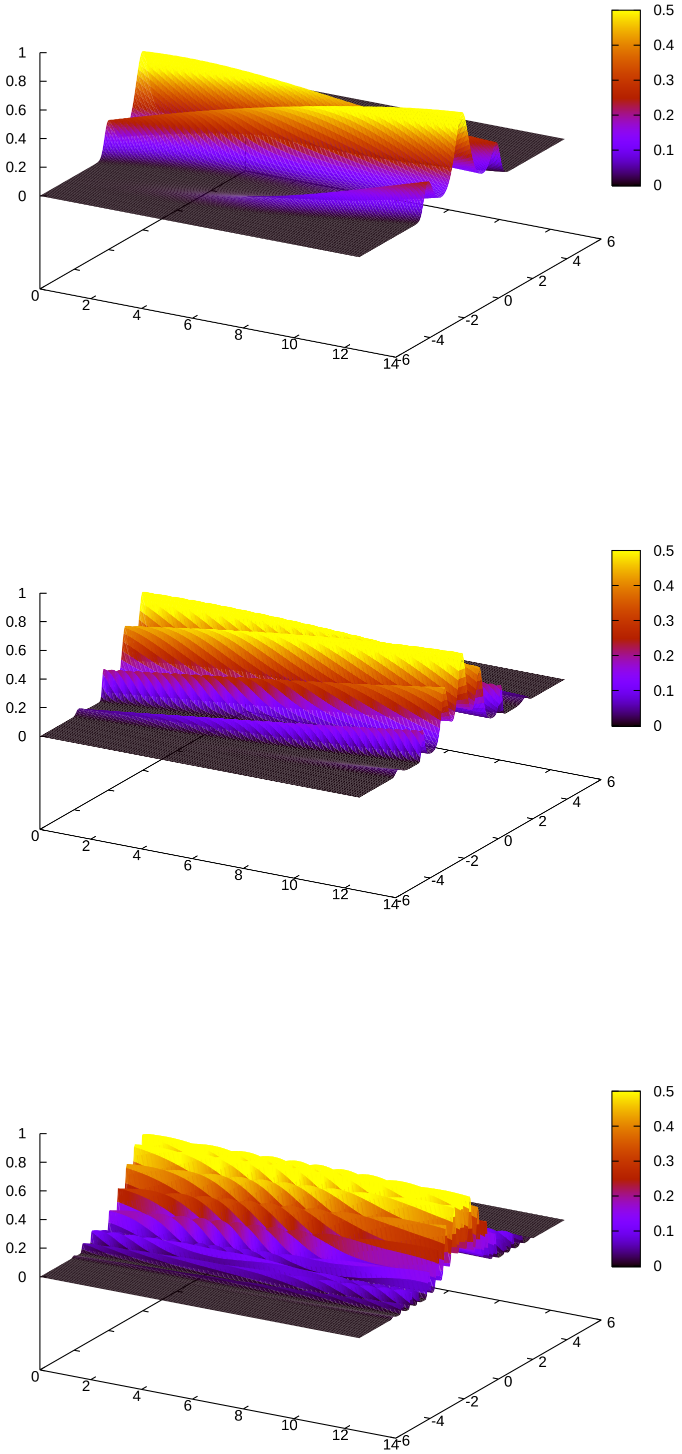
We should also be careful about the *conservation or monotone evolution* of magnitudes like the L^p norms, the total energy and the entropy. In this sense, we shall expose advantages and disadvantages of Semi-Lagrangian methods with respect to Flux Balance methods. Both methods are proposed for advection equations and thus, by splitting, for Vlasov's like models in Chapter 3.

The main tests we consider are the relaxation to equilibrium in a linear Vlasov-Boltzmann equation, the $1D$-Landau damping and a laser-plasma interaction. This is the objective of the final Chapter 5.

The first test is used to check that the numerical scheme give the expected known results about the speed of convergence towards equilibrium in simple linear collisional models in charge particle transport subjected to external forces.

The second one is a Vlasov-Poisson system where wild gradients occur due to the filamentation: we shall check whether the method takes it into account and is able to properly treat it without creating numerical oscillations. Moreover, we expect from the theory that the electric energy has an initial exponential decay followed by a stationary regime, and that the total energy is conserved, as well as the L^1 and L^2 norms.

The third test is a simplified quasi-relativistic Vlasov-Maxwell system, where violent oscillations appear: the method shall not add spurious oscillations and conserve the total energy (given by kinetic and potential part of both the electrostatic and the magnetic fields).



fi

Figure 1: The filamentation of the phase space in Landau damping.

Chapter 1

Kinetic equations in charged particles transport

The fundamental instrument used to describe the motion of charged particles in an electronic device or a plasma is Vlasov's equation. In this chapter, we shall introduce this kinetic equation and the other operators and equations we need to compute electric potentials (Poisson's equation), to describe collisional models (Boltzmann's operator) and to couple and compute electric and magnetic phenomena (Maxwell's equations). Thereafter, we shall review the state of the art regarding the existence and uniqueness of solutions for Vlasov's like systems and their main qualitative properties. To conclude, we shall introduce some models we are going to focus on in next chapters.

1.1 Vlasov's and Poisson's equations

1.1.1 Vlasov's equation

The *Vlasov* equation

$$\begin{cases} \dfrac{\partial f}{\partial t} + v \cdot \nabla_x f + F \cdot \nabla_v f = 0 \\ f(0,x,v) = f_0(x,v) \end{cases} \tag{1.1} \qquad \boxed{\texttt{liouv}}$$

is a probabilistic description of the the flow of charged particles which, subjected to a force field per unit mass $F(t,x)$ and given the initial distribution $f_0(x,v)$, satisfy the fundamental laws of classical Newton mechanics

$$\begin{cases} \dfrac{dx}{dt} = v \\ \dfrac{d^2x}{dt^2} = F(t,x). \end{cases} \tag{1.2} \qquad \boxed{\texttt{classmech}}$$

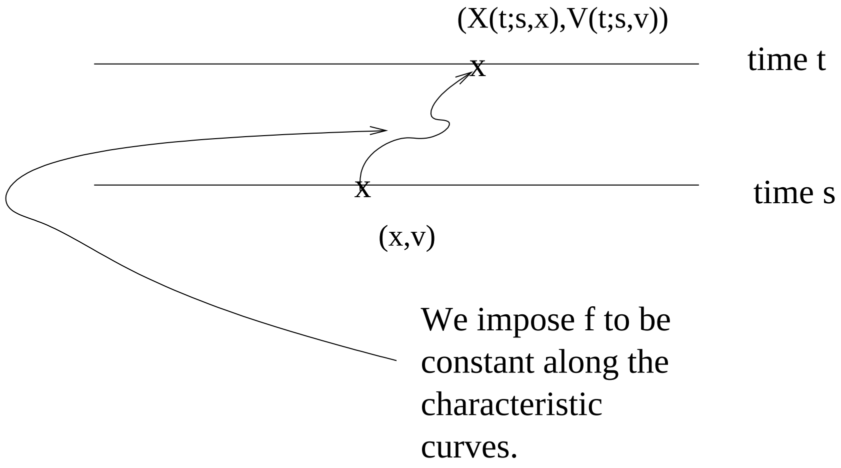


Figure 1.1: Vlasov's equation comes from imposing this condition of f. fconstchar

The function $f(t,x,v)$ describes the number density of particles at position $x \in \Omega \subset \mathbb{R}^N$ with velocity $v \in \mathbb{R}^N$ at time $t \geq 0$. In some cases, the velocity variable v will be replaced by another parameter identifying the state of the particles, like the momentum p or the wave vector k. For instance, wave vector variables are more adapted when talking about interactions that are modelled through quantum mechanics, they are related by De Broglie's identity $p = mv = hk$ in simplified models of the energy-band for semiconductors.

Derivation of Vlasov's equation

Vlasov's equation (liouv 1.1), also called Liouville's equation, comes from imposing that the probability function $f(t,x,v)$ is constant along the characteristic curves of the system

$$\begin{cases} \dfrac{d}{dt}\mathcal{X}(t;s,x) = v \\ \dfrac{d}{dt}\mathcal{V}(t;s,v) = F \\ \mathcal{X}(s;s,x) = x \\ \mathcal{V}(s;s,v) = v, \end{cases}$$

i.e. by imposing for any arbitrary time s, position x and velocity v, that

$$\frac{d}{dt} f\left[t, \mathcal{X}(t;s,x), \mathcal{V}(t;s,v)\right] = 0,$$

as we see in Figure fconstchar 1.1.

1.1.2 Poisson's equation

In order to compute both the self-consistent and the external potential, *Poisson*'s equation is used: given the density

$$\rho(x) = \int_{\mathbb{R}^N} f(t,x,v)\,dv$$

and, if the case, an external density $n^{ext}(x)$ which might be a stabilizing term (background ion density in a plasma) or the doping profile (impurities added to a semiconductor material), Poisson's equation returns the electric potential inside our object given by

$$\epsilon_0 \Delta_x \Phi = e\left[n^{ext} - \rho\right], \qquad x \in \Omega$$

where ϵ_0 is the electric permittivity of the material and e the elementary charge. The electric field E is obtained from the potential, $E = -\nabla_x \Phi$.

Poisson's equation has to be equipped with boundary conditions. Periodic boundary conditions for the potential and the electric field on a N-dimensional cube, $\Omega = [0,L]^N$, are standard in plasmas since they imply neutrality of the charged particles ensemble, i.e.,

$$\int_\Omega n^{ext}(x)\,dx = \int_\Omega \rho(t,x)\,dx,$$

for all times. In semiconductors, the electron current is produced due to a potential drop applied to the device, and thus we should solve Poisson's equation with fixed potential values at the contacts and homogeneous Neumann conditions on the insulating boundaries.

In case we do not consider any other force acting on particles, the force field on Vlasov's equation will be obtained for the self-consistent electric field E from Poisson's equation and an eventual external potential $\Phi^{ext}(x)$ acting on the system, and thus $F(t,x) = \frac{e}{m}\left[E(t,x) - \nabla_x \Phi^{ext}\right]$.

1.1.3 Existence of solutions for Vlasov-Poisson system

Vlasov-Poisson systems are used to describe the flow of particles under the effect of a force field $F(t,x)$ (attractive or repulsive depending on the charge of the electric carriers). The kinetic aspect is given by Vlasov's equation, while the force field is computed by Poisson's equation.

Classical solutions

We shall call *classical* a solution for which the derivatives hold in the classical sense. The first result of existence of solutions for Vlasov-Poisson system was given in 1961 by S. V. Iordanskii iordanskii [22], where he proved local existence of classical solutions, but they were not proved to be global in time nor were they shown to be unique. S. Ukai and T. Okabe ukaiokabe [38] in 1978 proved the existence and uniqueness of global in time classical solutions in $\mathbb{R}^2$, and locally in time in $\mathbb{R}^3$. In 1981 E. Horst horst [19] gave necessary conditions for global existence in $\mathbb{R}^3$ and examples of global non-existence in dimension $N \geq 4$. In 1985 P. Degond and C. Bardos degondbardos [8] proved the existence of global in time solutions for small initial data. Finally the problem in $\mathbb{R}^3$ was solved by Pfaffelmoser 41 [31], who obtained the long-sought result about the existence of global solutions for arbitrary initial data. J. Schaeffer schaeffer [34] gave few years later a clear short proof of existence and uniqueness of global classical solutions in $\mathbb{R}^3$. Here, we report Schaeffer's main result:

Theorem 1.1.1. *If $f_0 : \mathbb{R}^3 \times \mathbb{R}^3 \longrightarrow [0, \infty[$ is a smooth nonnegative function of compact support, then the problem possesses a global solution which is $\mathcal{C}^\infty$. Moreover, for $p > \frac{33}{17}$, there exists a constant C_p such that*

$$1 + \sup \{|v| : \exists (x, s) \in \Omega \times [0, t] \ s.t. \ f(s, x, v) \neq 0\} = Q(t) \leq C_p(1 + t)^p,$$

for all times $t \geq 0$.

Weak solutions

We shall call *weak* a solution which holds in the distributional sense, because f_0 is not regular enough to provide the application of the classical characteristics methods. We refer for a detailed definition of weak solutions to bouchut,mjcaceres [3, 4]. The first result about the existence of weak solutions was obtained by A. A. Arsen'ev in 1973 arsenev [1]. In 1984, E. Hörst and R. Hunze 34 [21] extended the set of initial functions and proved the existence of global weak solutions in $\mathbb{R}^3$. In 1988 R. J. DiPerna and P. L. Lions diperna [10] gave results of global existence of weak solutions in arbitrary dimension, and in 1991 F. Bouchut bouchut1991 [2] proved the global existence of weak solutions for a system with two kinds of charged particles.

1.1.4 Qualitative properties of the Vlasov-Poisson system

Maximum principle

The maximum principle asserts that whenever

$$0 \leq f_0(x, v) \leq M \Longrightarrow 0 \leq f(t, x, v) \leq M.$$

In the particular case in which f is normalized as a probability density, the maximum principle is interpreted as Pauli's principle which asserts that two electrons cannot have the same state (x, v) at the same time t. We refer to markowich [29] for the proof in the case of classical solutions. For weak solutions it is a simple consequence of lower weak-* semicontinuity of the L^∞-norm for bounded functions.

Mass conservation

Both classical and weak solutions of the Vlasov-Poisson system have the property of being mass conservative whenever posed in the whole space or in a N-dimensional cube with periodic boundary conditions:

$$\frac{d}{dt}\int_\Omega \int_{\mathbb{R}^N} f(t,x,v)\,dvdx = 0.$$

Total energy conservation

Classical solutions of the Vlasov-Poisson system posed in the whole space or with periodic boundary conditions have the property of conserving the total energy, which is given by the sum of the kinetic energy

$$KE(t) = \int_\Omega \int_{\mathbb{R}^N} \frac{|v|^2}{2} f(t,x,v)\,dvdx$$

and the potential energy

$$PE(t) = \frac{1}{2}\int_\Omega \rho\Phi^{self}\,dx + \int_\Omega \rho\Phi^{ext}\,dx, \tag{1.3}$$

totalenergyvlasovpoisson

where Φ^{self} is computed from Poisson's equation. Here, we are assuming that $n^{ext} = 0$ and we take into account external effects through Φ^{ext}. The total conservation of energy

$$\frac{d}{dt}\left[KE(t) + PE(t)\right] = 0.$$

reflects the Hamiltonian character of this equation. Weak solutions only achieve an energy inequality, for $0 \le s \le t$,

$$KE(t) + PE(t) \le KE(s) + PE(s).$$

Entropy conservation

Classical solutions of the Vlasov-Poisson system preserves the physical entropy of the system, that is,

$$\frac{d}{dt}\int_\Omega \int_{\mathbb{R}^N} f(t,x,v)\log[f(t,x,v)]\,dvdx = 0$$

while weak solutions only verify an inequality, for $0 \le s \le t$,

$$\int_\Omega \int_{\mathbb{R}^N} f(t,x,v)\log[f(t,x,v)]\,dvdx \le \int_\Omega \int_{\mathbb{R}^N} f(s,x,v)\log[f(s,x,v)]\,dvdx.$$

Conservation of L^p norms

Classical solutions of Vlasov-Poisson systems have the property of conserving the L^p norms:

$$\frac{d}{dt}\|f(t,x,v)\|_{L^p(\Omega\times\mathbb{R}^N)} = 0$$

for $1 \leq p \leq \infty$, while weak solutions have non-increasing L^p norms.

General conservation

If $\beta : [0,\infty) \to \mathbb{R}$ is convex, and $\beta(f_0) \in L^1(\Omega\times\mathbb{R}^N)$, then classical solutions of the Vlasov-Poisson system conserves

$$\int_\Omega \int_{\mathbb{R}^N} \beta(f(t,x,v))\, dvdx.$$

For weak solutions the inequality

$$\int_\Omega \int_{\mathbb{R}^N} \beta(f(t,x,v))\, dvdx \leq \int_\Omega \int_{\mathbb{R}^N} \beta(f(s,x,v))\, dvdx$$

holds for $0 \leq s \leq t$.

1.2 Collisions

We may assume that particles undergo “collisions”, between themselves or with the fixed atomic lattice. This word does not mean that physical shocks happen; it means that the carriers, passing close to another carrier, are deviated due to short-range interaction forces, in contrast with long-range interaction forces as the Coulomb interaction, and this phenomenon seems like if a ball hit another ball and both were pushed to other directions following laws of conservation of energies and momenta. The description of this kind of interactions is modelled through *Boltzmann*’s operators $\mathcal{Q}[f]$ as the right hand side of Vlasov’s equation:

$$\frac{\partial f}{\partial t} + v\cdot\nabla_x f + F\cdot\nabla_v f = \mathcal{Q}[f].$$

The collision rate will be influenced by the lattice temperature θ_0 (the higher it is the more probable the collisions are), by the particle density ρ (where there are more particles there will be more collisions), and by a typical relaxation time τ, which depends on the material and even on the state of the particle.

1.2.1 Introduction to the collision operator

Boltzmann's collision operator describes how carrier particles are deviated by the interaction with other particles or with the environment. Therefore, we consider in this probabilistic description how many particles pass from one state (x,k) to another state (x,k'), where x is the position and k is the wave vector; that basically means that at a certain time a particle at position x changes its direction of motion: wave vector is oriented as the velocity. Boltzmann's operator has the following form:

$$\mathcal{Q}[f](x,k,t) = \int_{\mathbb{R}^N} \left[s(x,k',k)f(k')(1-f(k)) - s(x,k,k')f(k)(1-f(k')) \right] dk'$$

where $s(x,k,k')$ is called the *collision rate* and represents the probability of passing from state (x,k) to state (x,k'). Its integral

$$\lambda(x,k) = \int_{\mathbb{R}^N} s(x,k,k')dk'$$

is called *collision frequency*, because it represents the collision mean frequency for a particle having state (x,k). Its reciprocal

$$\tau(x,k) = \frac{1}{\lambda(x,k)}$$

is called *relaxation time*: it represents the mean time passing from one collision and the successive one for a particle with state (x,k). The collision operator should typically model the process of relaxation towards thermodynamical equilibrium in the absence of external forces and under homogeneous densities, being the thermodynamic equilibria given by the kernel of the Boltzmann operator $\mathcal{Q}[f]$.

1.2.2 Linear Boltzmann's operator

Boltzmann's operator can describe different kinds of interactions by choosing in a convenient way the collision rate. We have neglected collisions between the particles, just taking into account collisions with the fixed lattice. Let us simplify its form by assuming a low-density regime (markowich[29, page 33]). Since f is very small in many real devices, we are going to ignore the quadratic terms in operator $\mathcal{Q}[f]$ to obtain

$$\mathcal{Q}[f](x,k,t) = \int_{\mathbb{R}^N} \left[s(x,k',k)f(k') - s(x,k,k')f(k) \right] dk'.$$

Assuming the simple relation $mv = hk$, writing the collision operator in terms of the velocity variable v and choosing the following collision rate:

$$s = \frac{1}{\tau} M_{\theta_0}(v),$$

where M_{θ_0} is the normalized Maxwellian distribution

$$M_{\theta_0}(v) = \left(\frac{1}{2\pi\theta_0}\right)^{\frac{N}{2}} e^{-\frac{|v|^2}{2\theta_0}},$$

and τ is an averaged relaxation time, we obtain the so-called relaxation time operator:

$$\mathcal{Q}[f](t,x,v) = \frac{1}{\tau}\left[\rho(t,x)M_{\theta_0}(v) - f(t,x,v)\right].$$

The thermodynamic equilibrium is obviously given by the normalized Maxwellian distribution $M_{\theta_0}(v)$.

1.3 Vlasov-Poisson-Boltzmann systems

Vlasov-Poisson-Boltzmann systems

$$\begin{cases} \dfrac{\partial f}{\partial t} + v \cdot \nabla_x f + F \cdot \nabla_v f = \mathcal{Q}[f] \\ \epsilon_0 \Delta\Phi = e(n^{ext} - \rho) \\ E = -\nabla_x \Phi \end{cases}$$

describe interactions and transport of charged particles in a semiconductor device or plasma where the force field is given by the self-consistent potential, computed by Poisson's equation and an external one equivalent to the present of the external background density $n^{ext}(x)$.

1.3.1 Properties of Vlasov-Poisson-Boltzmann systems

The collision operator verifies

$$\int_{\mathbb{R}^N} \mathcal{Q}[f]\, dv = 0$$

and thus, these systems are mass-conservative. On the other hand, they are dissipative, i.e., multiplying the operator by the derivative of any convex function $\beta(f)$, we get

$$\int_{\mathbb{R}^N} \beta'(f)\mathcal{Q}[f]\, dv \leq 0,$$

in particular, for the relevant case of the physical entropy $\beta(f)) = f \log f$. This property means that these quantities are dissipated for homogeneous initial data.

1.4 Vlasov-Maxwell system

When describing plasmas, particles have typically to be considered relativistic, and thus it is easier to consider f as a function of position x and of momentum p. In Vlasov equation the force field will be given by both electric and magnetic fields:

$$F = e(E + v \wedge B).$$

Electric and magnetic fields are related by a set of equations, called Maxwell's equations. They are, in its differential form:

1. Ampere's law. It describes the line integral of the magnetic field:

$$\frac{\partial E}{\partial t} = c^2 \nabla \wedge B. \tag{1.4}$$

$\boxed{\texttt{ampere}}$

2. Faraday's law of induction. It describes the line integral of the electric field being proportional to the rate of change of the magnetic flux through the surface enclosed by the line:

$$\frac{\partial B}{\partial t} = -\nabla \wedge E. \tag{1.5}$$

$\boxed{\texttt{faraday}}$

3. Gauss's law for electricity. It describes the area integral of the electric flux through a closed surface:

$$\nabla \cdot E = \frac{e}{\epsilon_0}\left[n^{ext} - n\right]. \tag{1.6}$$

$\boxed{\texttt{gausselec}}$

4. Gauss's law for magnetism. It describes the area integral of the magnetic flux through a closed surface:

$$\nabla \cdot B = 0. \tag{1.7}$$

$\boxed{\texttt{gaussmagn}}$

Relativistic particles means that we distinguish between their velocity and momentum:

$$v = \frac{p}{m\sqrt{1 + \frac{|p|^2}{m^2c^2}}}.$$

As the magnetic field admits a vector potential A because of ($\overset{\texttt{gaussmagn}}{1.7}$), if we call Φ the electrostatic potential, then

1. $$B = \nabla \wedge A. \tag{1.8}$$

$\boxed{\texttt{magnetic}}$

2. $$E = -\frac{\partial A}{\partial t} - \nabla\Phi. \tag{1.9}$$

$\boxed{\texttt{electric}}$

We shall use these equations to simulate the action of a laser wave (called pump wave) penetrating into a plasma. In this case, the electromagnetic fields to consider are the self-consistent electric field due to the electrons (given by Poisson's equation), the electromagnetic field due to the laser wave and the electrostatic field due to the ions in the plasma (given by Poisson's equation).

1.4.1 Existence of solutions of Vlasov-Maxwell systems

Studies about the existence of solutions for Vlasov-Maxwell equations are more recent than for Vlasov-Poisson. In 1986 R. Glassey and W. Strauss glassey1 [16] gave results about the existence and uniqueness of global classical solutions for the relativistic case, for the system with a mixture of several types of particles and local existence for the non-relativistic case with smooth compactly supported initial data. In 1989 glassey2 [15] they extended their previous proof for initial distributions with certain decay conditions. In 1996 Glassey glassey3 [14] extended the study to the research of weak solutions. In 1997 Glassey and his former student Schaeffer published glassey4 [17] results about the existence of classical solutions for the relativistic case. Here we report the main result about the existence of classical solutions exposed in glassey1 [16]:

Theorem 1.4.1. *Let $f_0 \in \mathcal{C}^1(\mathbb{R}^3 \times \mathbb{R}^3)$ a compactly supported positive function, let the electromagnetic fields $E_0(x) \in \mathcal{C}^2(\mathbb{R}^3)$ and $B_0(x) \in \mathcal{C}^2(\mathbb{R}^3)$. If there exists a continuous function $\alpha(t)$ such that $f(t,x,v) = 0$ for $|v| > \alpha(t)$, then the Vlasov-Maxwell system admits a unique solution in the space $\mathcal{C}^1([0,\infty) \times \mathbb{R}^3 \times \mathbb{R}^3)$, and both $E(t)$ and $B(t)$ belong to $\mathcal{C}^1([0,\infty) \times \mathbb{R}^3)$.*

Here we report the result achieved in 1989 by R. DiPerna and P. L. Lions dipernamaxwell [9] about the existence of weak solutions:

Theorem 1.4.2. *Let $f_0 \in L^1 \cap L^\infty(\mathbb{R}^3 \times \mathbb{R}^3)$ a positive function such that*

$$\int_{\mathbb{R}^3\times\mathbb{R}^3} f_0|v|^2 dxdv + \frac{1}{2}\int_{\mathbb{R}} |E_0|^2 + |B_0|^2 dx \leq \infty.$$

Then there exists a weak solution of Vlasov-Maxwell system in the space $\mathcal{C}([0,\infty), L^\infty(\mathbb{R}^3 \times \mathbb{R}^3) - w^*)$, and both E and B belong to $\mathcal{C}([0,\infty), L^2(\mathbb{R}^3) - w)$.

1.4.2 Simplifying assumptions

We shall make some simplifying assumptions in order to get an easier equation and to preserve the characteristics of these phenomena. In order to simplify the problem, we shall assume without loss of generality that the pump wave is linearly polarized in y-direction, that is,

$$A_x = A_z = 0.$$

We shall also neglect the effects of the laser wave in y and z directions in the plasma, because the heating effect is much slower than in the direction of propagation (x). Equations (1.8) and (1.9) become

$$\begin{cases} B_x = 0 \\ B_y = \dfrac{\partial A_y}{\partial x} \\ B_z = 0 \\ E_x = -\dfrac{\partial \Phi}{\partial x} \\ E_y = -\dfrac{\partial A_y}{\partial t} \\ E_z = 0 \end{cases}$$

Putting these assumptions into Vlasov's equation, in (1.4)-(1.7) and in Poisson's equation, we get:

$$\begin{cases} \dfrac{\partial f}{\partial t} + \dfrac{p_x}{m\gamma}\dfrac{\partial f}{\partial x} - e\left(E_x + \dfrac{eA_y}{m\gamma}\dfrac{\partial A_y}{\partial x}\right)\dfrac{\partial f}{\partial p_x} = 0 \\ \dfrac{\partial^2 A_y}{\partial t^2} - c^2\dfrac{\partial^2 A_y}{\partial x^2} = -\dfrac{e}{m\epsilon_0}\displaystyle\int_{\mathbb{R}} \dfrac{1}{\gamma} f dp_x \\ \dfrac{\partial E_x}{\partial t} = \dfrac{e}{\epsilon_0}\displaystyle\int_{\mathbb{R}} \dfrac{p_x}{m\gamma} f dp_x \\ \dfrac{\partial E_x}{\partial x} = \dfrac{e}{\epsilon_0}\left(n_{ext} - \displaystyle\int_{\mathbb{R}} f dp_x\right) \end{cases}$$

where the factor

$$\gamma = \sqrt{1 + \frac{p_x^2}{m^2c^2} + \frac{e^2A_y^2}{m^2c^2}}$$

and

$$\{n, n_\gamma, j_x\} = \int_{\mathbb{R}} \left\{1, \frac{1}{\gamma}, \frac{p_x}{m\gamma}\right\} f dp_x.$$

We shall consider this system posed either in the whole space $x \in \mathbb{R}$ or in $x \in [0, L]$ with periodic boundary conditions.

1.4.3 Quasi-relativistic model

Quasi-relativistic model consists in approximating γ by $\sqrt{1 + \frac{p^2}{m^2c^2}}$ in the terms $\frac{p_x}{m\gamma}\frac{\partial f}{\partial x}$ and in j_x and approximating γ by 1 in the other cases. Scaling the set of equations by

$$\begin{cases} x = L\tilde{x} \\ t = \frac{L}{c}\tilde{t} \\ p = (mc)\tilde{p} \\ E = \sqrt{\frac{mc^2\bar{n}}{\epsilon_0}}\tilde{E} \\ A = \left(\frac{mc}{e}\right)\tilde{A} \\ f = \left(\frac{\bar{n}}{mc}\right)\tilde{f} \\ \Phi = \left(\frac{c^2 m}{e}\right)\tilde{\Phi} \\ \eta = \frac{\bar{x}}{L} = \frac{1}{L}\frac{c}{e}\sqrt{\frac{m\epsilon_0}{\bar{n}}} \end{cases}$$

where L and $\bar{n}$ are the typical length and density respectively, and then dropping the tildes to simply the notation, we obtain the dimensionless quasi-relativistic Vlasov-Maxwell model:

$$\frac{\partial f}{\partial t} + \frac{p}{\sqrt{1+p^2}}\frac{\partial f}{\partial x} - \left(\eta^{-1}E + A\frac{\partial A}{\partial x}\right)\frac{\partial f}{\partial p} = 0. \tag{1.10}$$ QRVM

The electrostatic potential Φ is the solution of Poisson's equation

$$\Phi_{xx} = \eta^{-2}\left[n^{ext} - \rho\right],$$

where n^{ext} is an external density used to get a neutral electric field. The vector magnetic potential A satisfies the following wave-like equation

$$\frac{\partial^2 A}{\partial t^2} - \frac{\partial^2 A}{\partial x^2} = -\eta^{-2}\rho A.$$

To simplify the notation we will denote by E the x-component of the electrostatic field given by Poisson's equation, $E = -\Phi_x$, $\mathcal{E}$ the y-componenent of the electric field and B the y-componenent of the magnetic field that due to Maxwell's equation verify

$$\begin{cases} \mathcal{E} = -\dfrac{\partial A}{\partial t} \\ \dfrac{\partial \mathcal{E}}{\partial x} = -\dfrac{\partial B}{\partial t} \\ -\dfrac{\partial B}{\partial x} = -\eta^{-2}A\displaystyle\int_{\mathbb{R}} f dp + \dfrac{\partial \mathcal{E}}{\partial t}. \end{cases}$$

Global existence and uniqueness of classical solutions for this system has recently been obtained in laserplasma [6].

1.4.4 Total energy conservation

The total energy present in this model naturally split into two parts: the transversal energy WT and the longitudinal energy WL, both composed of a kinetic and a potential part.

$$\begin{cases} WT(t) = \dfrac{1}{2}\displaystyle\int_0^1 \rho A^2\,dx + \dfrac{1}{2}\eta^2 \int_0^1 \left[\mathcal{E}^2 + B^2\right]\,dx \\ WL(t) = \displaystyle\int_0^1 \int_{\mathbb{R}} \sqrt{1+p^2} f\,dpdx + \dfrac{1}{2}\eta^2 \int_0^1 E^2\,dx. \end{cases}$$

In laserplasma [6] it is proven that the total energy $WT(t) + WL(t)$ is constant

$$\frac{d}{dt}\left[WT(t) + WL(t)\right] = 0$$

for classical solutions.

Chapter 2

WENO interpolations

In this chapter, we are going to describe the PWENO (Pointwise Weighted Essentially Non Oscillatory) and FDWENO (Finite Differences Weighted Essentially Non Oscillatory) interpolation methods: they are developed to avoid Lagrange interpolation oscillations where high derivatives appear, and they are based on making a suitable average of Lagrange interpolations. The difference between these two methods is that while the first one respects the values of the function f_i on the grid points x_i (the purpose is to compute a direct interpolation), the second one respects the averages of the function $\bar{f}_i$ in the intervals $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ (the purpose is not to make a direct interpolation but to use the reconstructed values to give high order accurate approximations of derivatives).

2.1 Introduction

We have an interval $[x_L, x_R]$, a division into a grid of N points

$$x_L = x_0 < x_1 < ... < x_{N-1} = x_R$$

and the values of a function f in these points:

$$\{f_i = f(x_i)\}_{i=0,...,N-1}.$$

Our goal is to reconstruct the function in the whole interval in a "non-oscillatory" way near the points where the function has high gradients or discontinuities.

In the numerical solution of conservation laws, hyperbolic and transport equations, the sharp shape of the solutions (shocks in conservation laws) and the total variation of the function have to be controlled.

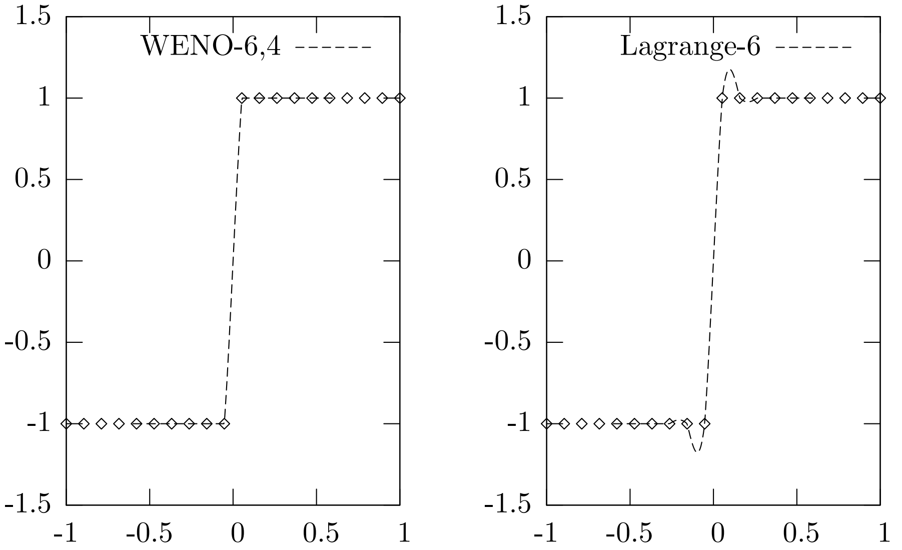


Figure 2.1: Comparison between an oscillating and a non-oscillating interpolation method

comparison

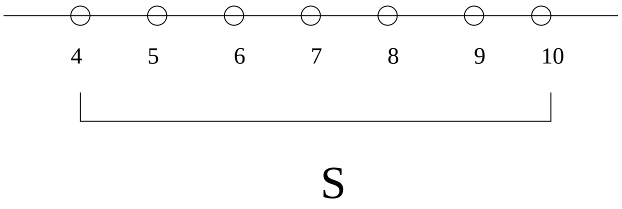


Figure 2.2: The main stencil $\mathcal{S}$

Essegrande

2.2 Description of the method

We have an interval $[x_L, x_R]$ divided into a grid of N points

$$x_L = x_0 < x_1 < x_2 < ... < x_{N-1} = x_R$$

$$\{f_i = f(x_i)\}_{i=0,...,N-1}.$$

We choose a stencil of *ntot* points (see Figure Essegrande 2.2)

$$\mathcal{S} = \{x_{first}, ..., x_{last}\} = \{x_{first}, ..., x_{first+ntot-1}\}$$

and *nlp* substencils (see Figure Essegrander 2.3)

$$\mathcal{S}_r = \{x_{last-lpo+1-r}, ..., x_{last-r}\} = \{x_{i-r}, ..., x_{last-r}\}$$

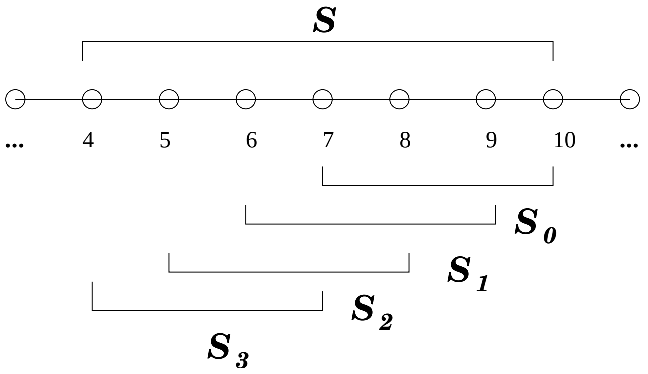


Figure 2.3: The substencils $\mathcal{S}_r$

Essegrander

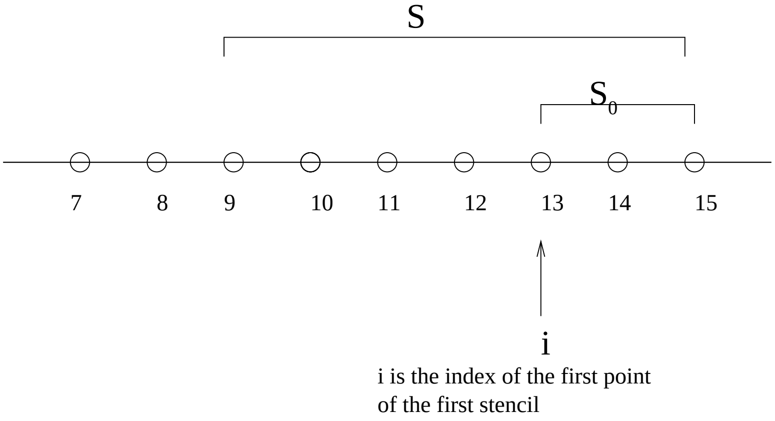


Figure 2.4: The parameter i

I

for $r = 0, ..., nlp - 1$, where $lpo := ntot - nlp + 1$ (see Figure lpo 2.8) and $i := last - lpo + 1$ (see Figure I 2.4).

The reconstructed value at points x will be a convex combination of the values given by the Lagrange polynomials in the stencils $\mathcal{S}_r$:

$$\sum_{r=0}^{nlp-1} \omega_r(x) p_r(x)$$

with p_r the Lagrange polynomials of degree $lpo - 1$ interpolating the stencil $\mathcal{S}_r$

$$p_r(x) = p_r^{lpo}(x) = \sum_{j=0}^{lpo-1} f_{i-r+j} c_{r,j}(x) \qquad (2.1)$$

lp

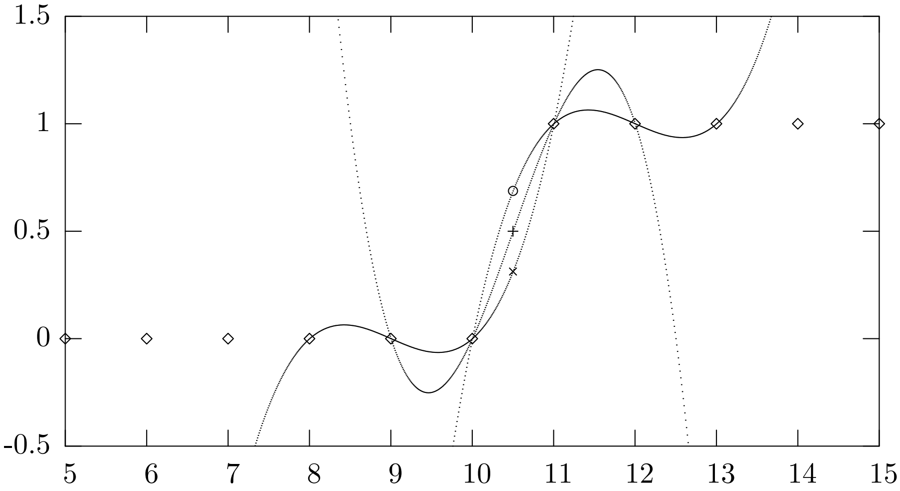


Figure 2.5: This figure shows how PWENO works with $ntot = 6$ and $lpo = 4$: it takes the (three, in this case) reconstructions given by the (three) Lagrange polynomials and then it will make an average between them

Boh

where

$$c_{r,j}(x) = \prod_{l=0, l\neq j}^{lpo-1} \frac{x - x_{i-r+l}}{x_{i-r+j} - x_{i-r+l}}.$$

$\omega_r(x)$ are weights which give more or less relevance to the stencils where the $p_r(x)$ are more or less "regular". Both the words "non-oscillatory" and "regular" will be rigorously defined in the next section.

The meaning of the parameters

1. cardinality of the main stencil $\mathcal{S}$ (**ntot**).
2. number of Lagrange polynomials (**nlp**).
3. Lagrange polynomial order (**lpo**), i.e. the number of points each polynomial interpolates.

2.3 The weights $\omega_r(x)$

WENO interpolation is given by

$$\sum_{r=0}^{nlp-1} \omega_r(x) p_r(x)$$

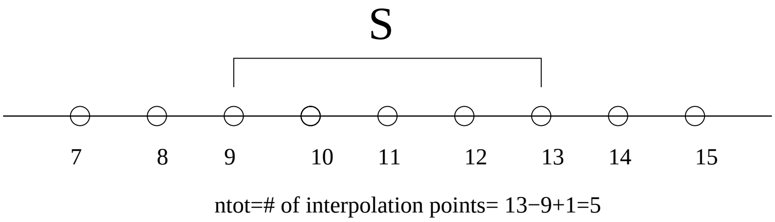


Figure 2.6: The parameter *ntot*

`ntot`

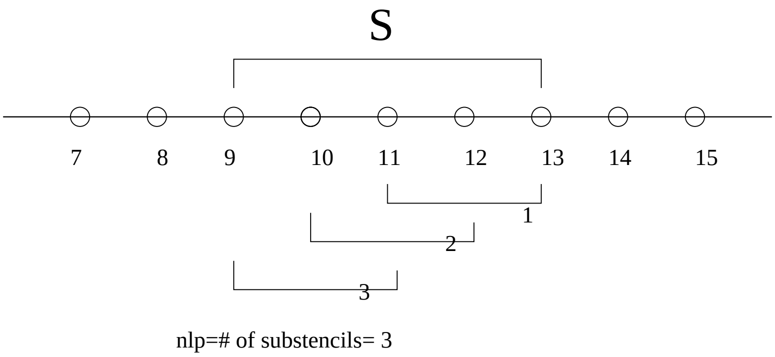


Figure 2.7: The parameter *nlp*

`nlp`

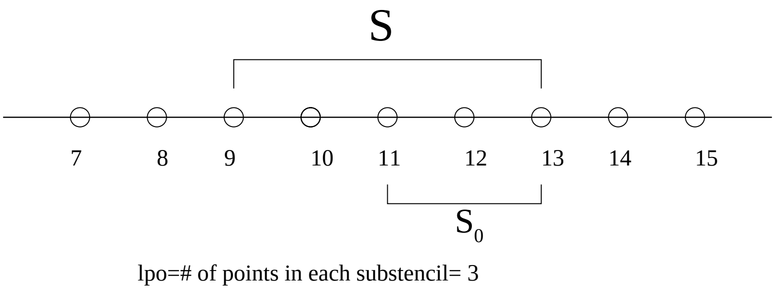


Figure 2.8: The parameter *lpo*

`lpo`

where $p_r(x)$ is defined in (2.1).

We need now to define the coefficients ω_r. First of all we need a measurement of the regularity of the Lagrange polynomials near x.

2.3.1 The smoothness indicators β_r

We shall call smoothness of the polynomial $p_r(x)$ a measure of its derivatives in the interval:

$$\mathcal{E} = [\mathcal{E}_L, \mathcal{E}_R] = \begin{cases} \left[x_{first+\left[\frac{ntot}{2}\right]-\frac{1}{2}}, x_{first+\left[\frac{ntot}{2}\right]+\frac{1}{2}}\right] & \text{if } ntot \text{ is odd} \\ \left[x_{first+\left[\frac{ntot}{2}\right]-1}, x_{first+\left[\frac{ntot}{2}\right]}\right] & \text{if } ntot \text{ is even} \end{cases} \tag{2.2}$$

centerint

where the dependencies of $\mathcal{E}_L$ and $\mathcal{E}_R$ will be omitted and $[x]$ means the integer part of x. Using the notation

$$i^* = first + \left[\frac{ntot}{2}\right]$$

we can also write

$$[\mathcal{E}_L, \mathcal{E}_R] = \begin{cases} \left[x_{i^*-\frac{1}{2}}, x_{i^*+\frac{1}{2}}\right] & \text{if } ntot \text{ is odd} \\ [x_{i^*-1}, x_{i^*}] & \text{if } ntot \text{ is even} \end{cases}$$

If the derivatives are large, the smoothness indicator is wanted to be large, and viceversa. The following measurement is proposed by Jiang and Shu in [23, page 207]:

$$\beta_r = \sum_{l=1}^{lpo-1} \int_{\mathcal{E}_L}^{\mathcal{E}_R} \Delta x^{2l-1} \Big(D^l p_r\Big)^2 dx$$

This is a weighted sum of L^2-norms of the derivatives, which we can see also as a weighted Sobolev norm of Dp_r in the interval $[\mathcal{E}_L, \mathcal{E}_R]$

$$\beta_r = \sum_{l=1}^{lpo-1} \Delta x^{2l-1} \Big\| D^l p_r \Big\|^2_{L^2(\mathcal{E}_L, \mathcal{E}_R)}$$

The weights Δx^{2l-1} are needed to make the terms of the sum independent of Δx, i.e., to make them all be of the same order. This will be clarified below. Other measurements would be possible, but we shall omit discussing this point.

2.3.2 Protoweights $\tilde{\omega}_r(x)$

Once we have computed the smoothness indicators, we define the $\tilde{\omega}_r(x)$ as

$$\tilde{\omega}_r(x) = \frac{d_r(x)}{(\epsilon + \beta_r)^p} \tag{2.3}$$

omgtld

where $d_r(x)$ are some weights we need to optimize the order of the method (we shall discuss it later), and ϵ is a constant to avoid the denominator to be zero (in the code $\epsilon = 10^{-6}$ is used). The choice of $1 \leq p < \infty$ has no influence on the order of the method. In all the tests we have set $p = 2$. If we chose a greater p we would decide to give less weight to the stencils where the reconstruction is more irregular, and viceversa.

2.3.3 Weights $\omega_r(x)$

To get the weights $\omega_r(x)$ we just have to normalize the $\tilde{\omega}_r(x)$ given by ($\overset{\texttt{omgtld}}{2.3}$).

$$\omega_r(x) = \frac{\tilde{\omega}_r(x)}{\sum_{j=0}^{nlp-1} \tilde{\omega}_j(x)}. \qquad (2.4) \quad \boxed{\texttt{omg}}$$

Still we have to find weights $d_r(x)$ to get the highest order method.

2.3.4 The weights $d_r(x)$

In order to get a high order method (if the function is smooth enough), we need coefficients $d_r(x)$ such that:

$$p(x) = \sum_{r=0}^{nlp-1} d_r(x) p_r(x) \qquad (2.5) \quad \boxed{\texttt{dr}}$$

where $p(x)$ is the $ntot - 1$-degree Lagrange polynomial interpolating the whole stencil $\mathcal{S}$

$$p(x) = \sum_{j=0}^{ntot-1} f_{first+j} \prod_{l=0,j\neq j}^{ntot-1} \frac{x - x_{first+j}}{x_{first+j} - x_{first+l}}. \qquad (2.6) \quad \boxed{\texttt{p}}$$

Lagrange interpolation gives a *ntot*-order method; by mean of these coefficients we want PWENO-*ntot*,*lpo* interpolation to approach a *ntot*-order method non oscillatory for homogeneously regular functions, i.e., whenever all the weights β_r have the same order of magnitude.

Proposition 2.3.1 (Existence and uniqueness of the weights $d_r(x)$)**.** *Let $\mathcal{I} = [x_L, x_R] \subset \mathbb{R}$ be an interval, and let*

$$x_L = x_0 < x_1 < ... < x_{N-1} = x_R$$

be the grid. If x is not a point of the grid, then the weights $d_r(x)$ defined by ($\overset{\texttt{dr}}{2.5}$) *are unique.*

Proof. The *ntot*-order Lagrange interpolation is exact on $\mathbb{P}^{ntot-1}$, i.e.,

$$f \in \mathbb{P}^{ntot-1} \rightarrow p[f](x) = f(x)$$

where $p[f]$ is the Lagrange polynomial which interpolates f in $ntot$ points. If $\mathcal{B} = \{b_i\}_{i=0}^{ntot-1}$ is a basis of $\mathbb{P}^{ntot-1}$, $f(x) = \sum_{i=0}^{ntot-1} f_i b_i(x)$ and $c_{r,j}(x)$ are given by ($\overset{\texttt{cr}}{2.2}$),

$$\begin{aligned} p\left[f\right](x) &= p\left[\sum f_i b_i\right](x) = \sum_j \sum_i f_i b_i(x_j) c_{r,j}(x) && (2.7)\\ &= \sum_i f_i \sum_j b_i(x_j) c_{r,j}(x) = \sum_i f_i p\left[b_i\right](x), && (2.8) \end{aligned}$$

so we only need to impose condition ($\overset{\texttt{dr}}{2.5}$) on the elements of a basis. Take as a basis of $\mathbb{P}^{ntot-1}$

$$\mathcal{B} = \{b_0, b_1, ..., b_{ntot-1}\} = \left\{1, \prod_{l=0}^{m}(x - x_{first+l}), m \in \{0, ..., ntot-2\}\right\}.$$

For $deg \leq lpo - 1$, condition ($\overset{\texttt{dr}}{2.5}$) gives $b_{deg}(x) = \sum_{r=0}^{nlp-1} d_r(x) b_{deg}(x)$ which means

$$\sum_{r=0}^{nlp-1} d_r(x) = 1.$$

For $lpo \leq deg \leq ntot - 1$, condition ($\overset{\texttt{dr}}{2.5}$) gives

$$b_{deg}(x) = \sum_{r=0}^{nlp-1} d_r(x) p_{r,deg}(x)$$

where $p_{r,deg}(x)$ is the Lagrange polynomial interpolating polynomial $b_{deg}(x)$ at points $\{x_{i-r}, ..., x_{i-r+lpo-1}\}$, i.e.,

$$p_{r,deg}(x) = \sum_{j=0}^{lpo-1} b_{deg}(x_{i-r+j}) c_{r,j}(x).$$

We get the following linear system

$$L = \left(\begin{array}{llll|l} 1 & 1 & ... & 1 & 1\\ p_{0,lpo}(x) & p_{1,lpo}(x) & ... & p_{nlp-1,lpo}(x) & b_{lpo}(x)\\ p_{0,lpo+1}(x) & p_{1,lpo+1}(x) & ... & p_{nlp-1,lpo+1}(x) & b_{lpo+1}(x)\\ \vdots & \vdots & \vdots & \vdots & \vdots\\ p_{0,ntot-1}(x) & p_{1,ntot-1}(x) & ... & p_{nlp-1,ntot-1}(x) & b_{ntot-1} \end{array}\right).$$

Note that $b_{deg}(x_{i-r+j}) = 0$ for $i - r + j \leq deg - 1$, which implies that the matrix has the following appearance

$$L = \left(\begin{array}{lllll|l} 1 & 1 & 1 & ... & 1 & 1\\ p_{0,lpo}(x) & p_{1,lpo}(x) & ... & p_{nlp-2,lpo}(x) & 0 & b_{lpo}(x)\\ p_{0,lpo+1}(x) & p_{1,lpo+1}(x) & ... & 0 & 0 & b_{lpo+1}(x)\\ \vdots & \vdots & \vdots & \vdots & \vdots & \\ p_{0,ntot-1}(x) & 0 & 0 & ... & 0 & b_{ntot-1}(x) \end{array}\right).$$

Call $L = (L_1|L_2)$. The computation of the determinant of L_1 is straightforward

$$\det(L_1) = \prod_{s=0}^{nlp-2} p_{s,(ntot-1)-s}(x) \neq 0$$

which is non-zero because none of the terms can be zero. Since

$$p_{s,(ntot-1)-s}(x) = \sum_{j=0}^{lpo-1} b_{(ntot-1)-s}(x_{i-s+j})c_{s,j}(x)$$

and $b_{(ntot-1)-s}(x_{i-s+j}) = 0$, for $j = 0, ..., lpo-2$, then

$$p_{s,(ntot-1)-s}(x) = b_{(ntot-1)-s}(x_{i-s+lpo-1})c_{s,lpo-1}(x) \neq 0$$

because $b_{(ntot-1)-s}(x_{i-s+lpo-1}) \neq 0$; if not, we should get a contradiction: a non-zero polynomial of degree $(ntot-1)-s$ would have $ntot-1$ zeros. $c_{s,lpo-1}(x) \neq 0$ because x does not belong to the points of the grid. In this way, the existence and uniqueness of the $d_r(x)$ has been proven. □

Remark. If x is a grid point, the uniqueness would not be needed, because in fact any linear combination of the Lagrange polynomials interpolating that point would be suitable.

2.3.5 Code implementation of the weights $d_r(x)$

Imposing the definition of Lagrange polynomials (lp 2.1) in (dr 2.5), we get

$$\sum_{j=0}^{ntot-1} f_{i-r^*+j}c^{ntot}_{r^*,j}(x) = \sum_{r=0}^{nlp-1} d_r(x) \sum_{j=0}^{lpo-1} f_{i-r+j}c^{nlp}_{r,j}(x)$$

where r^* is defined in Figure rstar 2.9. As the $d_r(x)$ do not depend on the values of f, we have to impose for every $0 \leq s \leq ntot-1$ the coefficients of f_{i-r^*+s} to be equal:

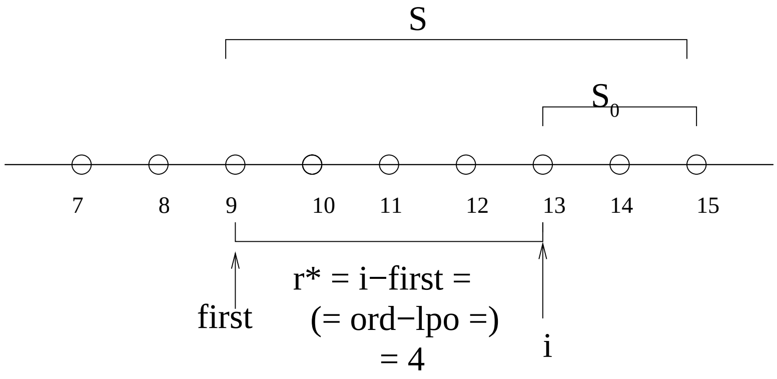
$$c^{ntot}_{r^*,s}(x) = \sum_{r,j \ \ s.t. \ \ -r+j=s} d_r(x)c^k_{r,j}(x).$$

So, the linear system to be solved is represented by the following $ntot \times (nlp+1)ntot$ matrix:

$$(ls)_{i,j} = \begin{cases} c^{lpo}_{j,i+j-(nlp-1)}(x) & \text{if } nlp-1 \leq i+j \leq lpo-1+nlp-1 \\ 0 & \text{else} \end{cases},$$

with the known terms

$$(ls)_{i,nlp} = c^{ntot}_{r^*,i}.$$



r* is the difference between the first point of the substencil S0 and the first point of the main stencil S

Figure 2.9: The parameter r^* rstar

Take now its submatrix $LS \in \mathbb{M}_{nlp \times nlp+1}$:

$$(LS)_{i,j} = \begin{cases} c^{lpo}_{j,i+j-(nlp-1)}(x) & if\ nlp-1 \leq i+j \leq lpo-1+nlp-1 \\ 0 & else \end{cases} .$$

Here we have nlp conditions for nlp unknowns, and the linear system is represented by an upper triangular matrix, which can be solved directly by a recursive procedure starting from the first line ($i = 0$).

2.3.6 A way for calculating explicitly the weights $d_r(x)$ as polynomials

We are able to get an explicit formula for the polynomials $d_r(x)$ by an iterative method. Assume the interpolation points are $\{x_0, x_1, ..., x_n\}$. Suppose we have two polynomials: $p(x)$ interpolates a function f at the points $x_0, x_1, ..., x_{n-1}$, and $q(x)$ interpolates at the points $x_1, x_2, ..., x_n$. A simple exactness argument (Aitken-Neville method, see calculonumerico1 [30, page 56]) allows to check that the polynomial interpolating f at the points $x_0, x_1, ..., x_n$ is given by

$$r(x) = \frac{p(x)(x - x_n) - q(x)(x - x_0)}{x_0 - x_n}.$$

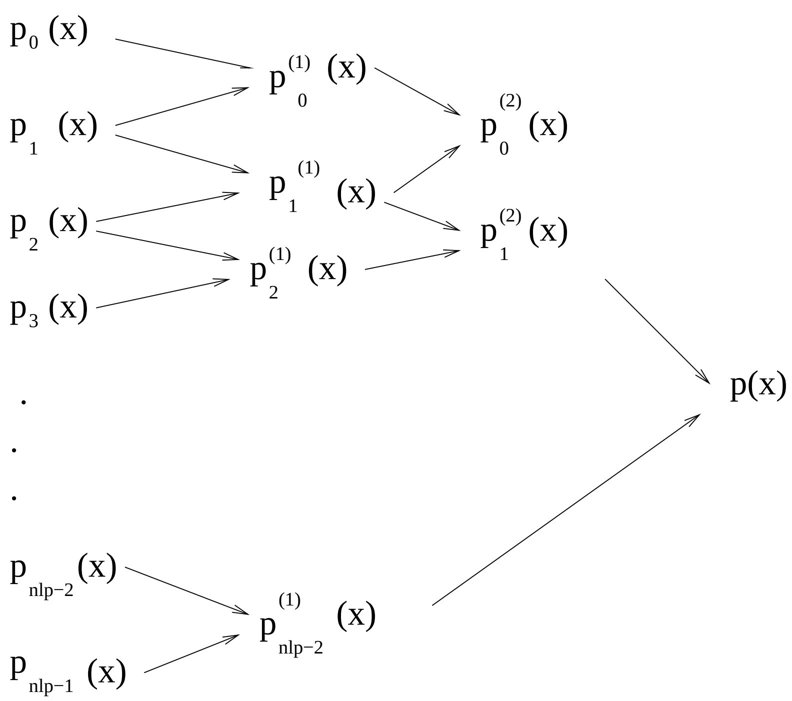


Figure 2.10: Explicit construction of the polynomials $\tilde{d}_r(x)$ by recursion dierre

An explicit recursion procedure based on the previous formula gives us the explicit value of $\tilde{d}_r(x)$. In the case of three points ($n = 2$) we get:

$$\begin{cases} \tilde{d}_0(x) = \frac{(x-x_{i-1})(x-x_{i-2})}{(x_{i-1}-x_{i+2})(x_{i-2}-x_{i+2})} \\ \tilde{d}_1(x) = -\left[\frac{(x-x_{i+2})(x-x_{i-2})}{(x_{i-1}-x_{i+2})(x_{i-2}-x_{i+2})} + \frac{(x-x_{i-2})(x-x_{i+2})}{(x_{i-2}-x_{i+1})(x_{i-2}-x_{i+2})}\right] \\ \tilde{d}_2(x) = \frac{(x-x_{i+1})(x-x_{i+2})}{(x_{i-2}-x_{i+1})(x_{i-2}-x_{i+2})} \end{cases}$$

Proposition 2.3.2 (Uniqueness of the weights $d_r(x)$). *Let $\mathcal{I} = [x_L, x_R] \subset \mathbb{R}$ be an interval, and let*

$$x_L = x_0 < x_1 < ... < x_{N-1} = x_R$$

be the grid. The weights $d_r(x)$, recursively constructed as polynomials like in Figure dierre 2.10, are unique in $\mathbb{P}^{ntot-lpo}$.

Proof. Polynomials $\tilde{d}_r(x)$ are explicitly constructed by the recursive method shown in Figure dierre 2.10, so they exist (no denominator can be zero because $i \neq j \Rightarrow x_i \neq x_j$), are unique and their degree is $(ntot - 1) - (lpo - 1) = ntot - lpo$. Moreover, $\forall x \in \mathbb{R} \setminus \{x_i\}_{i=0,...,N-1}$,

$$\tilde{d}_r(x) = d_r(x)$$

for construction (they must verify the same property), then the weights $d_r(x)$, constructed as polynomials in $\mathbb{P}^{ntot-lpo}$, are unique $\forall x \in \mathbb{R}$. □

2.3.7 The choice of $d_r(x)$

mainprop **Proposition 2.3.3.** *If the weights $d_r(x)$ satisfty*

$$\sum_{r=0}^{nlp-1} d_r(x) = 1 \tag{2.9}$$

qwert

and

$$\omega_r(x) - d_r(x) = O(\Delta x^n),$$

then PWENO-ntot,lpo gives a $(lpo + n)$-order reconstruction.

The proof is developed in the following section.

The choice we have made for the weights $d_r(x)$ satisfy property 2.9, but it was also meant to approach the best order and the best accuracy. On one hand, if f is a homogeneous regular function (which means that all the β_r are of the same order), by this choice the $\omega_r(x)$ approach the $d_r(x)$, i.e. $p^{PWENO}(x)$ approaches $p(x)$ (defined in (2.6)). Even if PWENO is not *ntot*-order, we force it to behave like Lagrange, which is of highest order, in case of regular functions.

On the other hand, if the function is not regular, then the weights $\omega_r(x)$ are very different from the weights $d_r(x)$, like it had to be, because we want PWENO to behave differently from Lagrange near high-gradients.

2.4 The order of the method

First of all we recall a standard result about the error committed by Lagrange interpolation (see [24, page 291]):

standardlagrange **Proposition 2.4.1.** *Let $\mathcal{I} = [x_L, x_R] \subset \mathbb{R}$ be an interval. If $f \in \mathcal{C}^{n+1}[x_L, x_R]$ and $p \in \mathbb{P}^n$ is the polynomial interpolating f in $n+1$ different points $\{x_0, x_1, ..., x_n\}$ in $[x_L, x_R]$. Then, $\forall x \in [x_L, x_R]$ there is a $\xi_x \in]x_L, x_R[$ such that*

$$f(x) - p(x) = \frac{1}{(n+1)!} f^{(n+1)}(\xi_x) \prod_{i=0}^{n} (x - x_i).$$

In the case of a regular grid this means that

$$p(x) = f(x) + O(\Delta x^{n+1}).$$

We shall now introduce a simple lemma.

omegamenodr **Lemma 2.4.2.** *If*

$$\beta_r = A(1 + O(\Delta x^n)) \tag{2.10}$$

order

where A is a non-zero quantity independent of r, then

$$\omega_r(x) - d_r(x) = O(\Delta x^n).$$

Proof of lemma $\overset{\texttt{omegamenodr}}{\textbf{2.4.2}}$. By straighforward calculations,

$$\frac{1}{{\beta_r}^p} = \frac{1}{D^p} + O(\Delta x^n)$$

and

$$\tilde{\omega_r} = \frac{d_r}{{\beta_r}^p} = d_r\left[\frac{1}{D^p} + O(\Delta x^n)\right] = \frac{1}{D^p}d_r + O(\Delta x^n)$$

with

$$\sum_j \tilde{\omega_j} = \frac{1}{D^p}d_r + O(\Delta x^n).$$

Finally,

$$\omega_r = \frac{\tilde{\omega_r}}{\sum_j \tilde{\omega_j}} = \frac{\frac{1}{D^p}d_r + O(\Delta x^n)}{\frac{1}{D^p} + O(\Delta x^n)} = d_r + O(\Delta x^n).$$

□

2.4.1 Proof of proposition $\overset{\texttt{mainprop}}{\textbf{2.3.3}}$

Since

$$\begin{aligned} p^W(x) - f(x) &= p^W(x) - p^L(x) + p^L(x) - f(x) = \\ &= \sum \omega_r(x)p_r(x) - \sum d_r(x)p_r(x) + \sum d_r(x)p_r(x) - f(x) \\ &= \sum \omega_r(x)p_r(x) - \sum d_r(x)p_r(x) + O(\Delta x^{ntot}), \end{aligned}$$

we already know that the method cannot be more than *ntot*-order, and we need to calculate

$$\mathcal{I} = \sum_{r=0}^{nlp-1} \omega_r(x)p_r(x) - \sum_{r=0}^{nlp-1} d_r(x)p_r(x)$$

to check which is the order. By simple manipulations

$$\begin{aligned} \mathcal{I} &= \sum_{r=0}^{nlp-1} \omega_r(x)p_r(x) - \sum_{r=0}^{nlp-1} d_r(x)p_r(x) \\ &= \sum_{r=0}^{nlp-1} \left[\omega_r(x) - d_r(x)\right] p_r(x) \\ &= \sum_{r=0}^{nlp-1} \left[\omega_r(x) - d_r(x)\right] \left[p_r(x) - f(x)\right]. \end{aligned}$$

Due to (qwert 2.9), we get

$$\begin{aligned}
\mathcal{I} &= \sum_{r=0}^{nlp-1} \left[\omega_r(x) - d_r(x)\right]\left[p_r(x) - f(x)\right] \\
&= \sum_{r=0}^{nlp-1} \left[\omega_r(x) - d_r(x)\right] O(\Delta x^{lpo})
\end{aligned}$$

and because of proposition standardlagrange 2.4.1 and lemma order 2.10, we finally deduce

$$\begin{aligned}
\mathcal{I} &= \sum_{r=0}^{nlp-1} \left[\omega_r(x) - d_r(x)\right] O(\Delta x^{lpo}) \\
&= \sum_{r=0}^{nlp-1} O(\Delta x^{n}) O(\Delta x^{lpo}).
\end{aligned}$$

□

2.4.2 Manipulation on the β_r

Our problem is now translated to the Taylor expansion of the smoothness indicators β_r. Once we have been able to compute n in (order 2.10), then the method will be order $lpo + n$. The weights can be expressed as

$$\begin{aligned}
\beta_r &= \sum_{l=1}^{lpo-1} \int_{\mathcal{E}} \Delta x^{2l-1} \left[D^l p_r(x)\right]^2 dx \\
&= \sum_{l} \Delta x^{2l-1} \int_{\mathcal{E}} \left[D^l \sum_{j=0}^{nlp-1} c_{r,j}^{lpo}(x) f_{i-r+j}\right]^2 dx \\
&= \sum_{l} \Delta x^{2l-1} \int_{\mathcal{E}} \left[\sum_{j} f_{i-r+j} D^l c_{r,j}(x)\right]^2 dx \\
&= \sum_{l} \Delta x^{2l-1} \int_{\mathcal{E}} \sum_{j,k=0}^{lpo-1} f_{i-r+j} f_{i-r+k} D^l c_{r,j}(x) D^l c_{r,k}(x) dx \\
&= \sum_{l} \Delta x^{2l-1} \sum_{j,k} f_{i-r+j} f_{i-r+k} \int_{\mathcal{E}} D^l c_{r,j}(x) D^l c_{r,k}(x) dx \\
&= \sum_{j,k} f_{i-r+j} f_{i-r+k} \sum_{l=1}^{lpo-1} \int_{\mathcal{E}} \Delta x^{2l-1} D^l c_{r,j}(x) D^l c_{r,k}(x) dx \\
&= \sum_{j,k} f_{i-r+j} f_{i-r+k} K_{j,k}^{r}
\end{aligned}$$

where K^r is a symmetric matrix defined as

$$K^r_{j,k} = \sum_{l=1}^{lpo-1} \int_{\mathcal{E}} \Delta x^{2l-1} D^l c_{r,j} D^l c_{r,k} dx.$$

We can now expand $f(x)$ around the point x_i:

$$f(x) = \sum_n \frac{1}{n!} f^{(n)}(x_i)(x - x_i)^n$$

so that

$$\begin{aligned} f_{i-r+j} &= \sum_n \frac{1}{n!} f^{(n)}(x_i)(\Delta x(i - r + j - i))^n && (2.11) \\ &= \sum_n \frac{1}{n!} f^{(n)}(x_i)(j - r)^n \Delta x^n. && (2.12) \end{aligned}$$

Therefore,

$$\begin{aligned} \beta_r &= \sum_{j,k} K^r_{j,k} \sum_{n,m} \frac{1}{n!m!} f^{(n)}(x_i) f^{(m)}(x_i)(j - r)^n (k - r)^m \Delta x^{n+m} K \\ &= \sum_{n,m} \Delta x^{n+m} f^{(n)}(x_i) f^{(m)}(x_i) \frac{1}{n!m!} \sum_{j,k} K^r_{j,k}(j - r)^n (k - r)^m \\ &= \sum_{n,m} \Delta x^{n+m} f^{(n)} f^{(m)} \mathcal{D}^r_{n,m} \end{aligned}$$

where $\mathcal{D}^r$ is a symmetric matrix defined as

$$\mathcal{D}^r_{n,m} = \frac{1}{n!m!} \sum_{j,k=0}^{lpo-1} K^r_{j,k}(j - r)^n (k - r)^m.$$

Numerical computation of $\mathcal{D}^r$ for method WENO-6,4 are given in (d064 2.13)-(d264 2.15) in the appendix of this chapter. These computations show that

$$\forall 0 \leq r \leq 2, \qquad \mathcal{D}^r_{1,1} = 1, \mathcal{D}^r_{2,2} = \frac{4}{3}$$

and therefore, if $f' \neq 0$ then

$$\beta_r = (f'(x_i)\Delta x)^2 (1 + O(\Delta x))$$

(where $A = (f'(x_i)\Delta x)^2$, $n = 1$ in (order 2.10)), and if $f'(x_i) = 0$ then

$$\beta_r = \frac{4}{3}(f''(x_i)\Delta x^2)^2 (1 + O(\Delta x))$$

($A = \frac{4}{3}(f''(x_i)\Delta x^2)^2$, $n = 1$ in (order (2.10)), i.e. that the method is $lpo + 1 = 4 + 1 = 5$ order.
As for WENO-5,3, we see from (d053 2.16)-(d253 2.18) that if $f'(x_i) \neq 0$ then

$$\beta_r = (f'(x_i)\Delta x)^2(1 + O(\Delta x^2))$$

(where $A = (f'(x_i)\Delta x)^2$, $n = 2$ in (order (2.10)). This implies that WENO-5,3 is $lpo + 2 = 3 + 2 = 5$ order. □

Numerical example

Take $u_0(x) = \exp(x)$ in $[-1; 1]$, reconstruct the value at $x = 0$ (if the number of points is even it will never belong to the grid) by PWENO-6,4 method and compute the difference $|\exp(0) - num.val.|$:

points	$L^\infty - error$	$L^\infty - order$
20	6.88×10^{-9}	
40	8.95×10^{-11}	6.263432
80	1.28×10^{-12}	6.119669
160	1.93×10^{-14}	6.059640
320	2.22×10^{-16}	6.442943

While at least 5 was expected, we see that we get 6, due to the homogeneous regularity of the function.

2.5 WENO for Finite Differences

The main difference between PWENO and FDWENO is that while the first one is a direct interpolation method (i.e. it reconstructs the values of the function), the second one is used to compute derivatives: what it reconstructs are not the values of the function f but the flux $\hat{f}$. The goal is to therefore compute (with an appropriate upwinding of the interpolations that we do not develop here)

$$\frac{\partial f}{\partial x}(x_i) \simeq \frac{\hat{f}_{i+\frac{1}{2}} - \hat{f}_{i-\frac{1}{2}}}{\Delta x}.$$

2.5.1 Introduction

FDWENO is a generalization of the method of Shu chiwangshu [35], with the only difference of allowing a free choice of the stencil $\mathcal{S}$ (see Figure Essegrande 2.2) and the substencils $\mathcal{S}_r$ (see Figure Essegrander 2.3) instead of only admitting k substencils of k points each.

As for the construction, while PWENO respects the values $f(x_i)$, FD-WENO respects the averages of f on the intervals $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$, i.e. while in the first case we impose

$$p^{PWENO}(x_i) = f(x_i),$$

in the second case we want

$$\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} p^{FDWENO}(x)dx = \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} f(x)dx.$$

2.5.2 Procedure

Once we have chosen a stencil $\mathcal{S}$ (see Figure 2.2) and into how many substencils to divide it (i.e. we have chosen nlp), we can proceed.

1. We construct a primitive of the function f,

$$F(x) = \int_{x_L}^{x} f(\xi)d\xi$$

2. We interpolate F by Lagrange polynomials

$$P_r(x) = \sum_{j=0}^{lpo} F_{i-\frac{1}{2}-r+j} C_{r,j}(x)$$

   for $r = 0, ..., nlp-1$, where

$$C_{r,j}(x) = \prod_{l=0, l\neq j}^{lpo} \frac{x - x_{i-\frac{1}{2}-r+j}}{x_{i-\frac{1}{2}-r+j} - x_{i-\frac{1}{2}-r+l}}$$

3. We differentiate polynomials $P_r(x)$

$$p_r(x) = \frac{d}{dx} P_r(x)$$

   Polynomials $p_r(x)$ satisfy

$$\int_{x_{i-\frac{1}{2}-r+j}}^{x_{i+\frac{1}{2}-r+j}} p_r(x) = F_{i+\frac{1}{2}-r+j} - F_{i-\frac{1}{2}-r+j}$$

4. We compute smoothness indicators

$$\beta_r = \sum_{l=0}^{lpo-1} \int_{\mathcal{E}} \Delta x \left(D^l p_r(x) \right)^2 dx$$

   where $\mathcal{E}$ is defined in (2.2).

5. We compute coefficients $d_r(x)$ defined in the same way as ($\overset{\texttt{dr}}{2.5}$).

6. We compute weights $\omega_r(x)$ like in ($\overset{\texttt{omgtld}}{2.3}$) and ($\overset{\texttt{omg}}{2.4}$).

7. We reconstruct

$$\hat{f}(x) = \sum_{r=0}^{nlp-1} \omega_r(x) p_r(x)$$

2.6 Appendix

2.6.1 Results for matrix $\mathcal{D}$

For WENO-6,4 (the values are given in absolute values):

$$\mathcal{D}^0 = \begin{bmatrix} 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 1.000 & 0.500 & 0.166 & 0.041 & 0.008 & 0.001 & 0.000 \\ 0.000 & 0.500 & 1.333 & 0.625 & 0.159 & 0.402 & 0.312 & 0.167 \\ 0.000 & 0.166 & 0.625 & 1.383 & 1.543 & 1.154 & 0.657 & 0.305 \\ 0.000 & 0.041 & 0.159 & 1.543 & 2.472 & 2.101 & 1.272 & 0.610 \\ 0.000 & 0.008 & 0.402 & 1.154 & 2.101 & 1.847 & 1.134 & 0.548 \\ 0.000 & 0.001 & 0.312 & 0.657 & 1.272 & 1.134 & 0.700 & 0.339 \\ 0.000 & 0.000 & 0.166 & 0.305 & 0.610 & 0.548 & 0.339 & 0.165 \end{bmatrix}, \quad (2.13) \quad \boxed{\texttt{d064}}$$

$$\mathcal{D}^1 = \begin{bmatrix} 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 1.000 & 0.500 & 0.166 & 0.041 & 0.008 & 0.001 & 0.000 \\ 0.000 & 0.500 & 1.333 & 0.625 & 0.381 & 0.139 & 0.048 & 0.013 \\ 0.000 & 0.166 & 0.625 & 1.383 & 0.729 & 0.340 & 0.114 & 0.033 \\ 0.000 & 0.041 & 0.381 & 0.729 & 0.393 & 0.181 & 0.061 & 0.018 \\ 0.000 & 0.008 & 0.139 & 0.340 & 0.181 & 0.084 & 0.028 & 0.008 \\ 0.000 & 0.001 & 0.048 & 0.114 & 0.061 & 0.028 & 0.009 & 0.002 \\ 0.000 & 0.000 & 0.013 & 0.033 & 0.018 & 0.008 & 0.002 & 0.000 \end{bmatrix}, \quad (2.14) \quad \boxed{\texttt{d164}}$$

$$\mathcal{D}^2 = \begin{bmatrix} 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 1.000 & 0.500 & 0.166 & 0.041 & 0.008 & 0.001 & 0.000 \\ 0.000 & 0.500 & 1.333 & 0.625 & 0.159 & 0.139 & 0.041 & 0.013 \\ 0.000 & 0.166 & 0.625 & 1.383 & 0.625 & 0.340 & 0.111 & 0.033 \\ 0.000 & 0.041 & 0.159 & 0.625 & 0.300 & 0.157 & 0.052 & 0.015 \\ 0.000 & 0.008 & 0.139 & 0.340 & 0.157 & 0.084 & 0.027 & 0.008 \\ 0.000 & 0.001 & 0.041 & 0.111 & 0.052 & 0.027 & 0.009 & 0.002 \\ 0.000 & 0.000 & 0.013 & 0.033 & 0.015 & 0.008 & 0.002 & 0.000 \end{bmatrix}. \quad (2.15) \quad \boxed{\texttt{d264}}$$

For WENO-5,3:

$$\mathcal{D}^0 = \begin{bmatrix} 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 1.000 & 0.000 & 0.333 & 0.250 & 0.116 & 0.041 & 0.012 \\ 0.000 & 0.000 & 1.083 & 1.083 & 0.631 & 0.270 & 0.093 & 0.027 \\ 0.000 & 0.333 & 1.083 & 1.194 & 0.715 & 0.309 & 0.107 & 0.031 \\ 0.000 & 0.250 & 0.631 & 0.715 & 0.431 & 0.187 & 0.064 & 0.018 \\ 0.000 & 0.116 & 0.270 & 0.309 & 0.187 & 0.081 & 0.028 & 0.008 \\ 0.000 & 0.041 & 0.093 & 0.107 & 0.064 & 0.028 & 0.009 & 0.002 \\ 0.000 & 0.012 & 0.027 & 0.031 & 0.018 & 0.008 & 0.002 & 0.000 \end{bmatrix}, \quad (2.16) \quad \boxed{\texttt{d053}}$$

$$\mathcal{D}^1 = \begin{bmatrix} 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 1.000 & 0.000 & 0.166 & 0.000 & 0.008 & 0.000 & 0.000 \\ 0.000 & 0.000 & 1.083 & 0.000 & 0.090 & 0.000 & 0.003 & 0.000 \\ 0.000 & 0.166 & 0.000 & 0.027 & 0.000 & 0.001 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.090 & 0.000 & 0.007 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.008 & 0.000 & 0.001 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.003 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \end{bmatrix}, \qquad (2.17) \quad \boxed{\texttt{d153}}$$

$$\mathcal{D}^2 = \begin{bmatrix} 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\ 0.000 & 1.000 & 0.000 & 0.333 & 0.250 & 0.116 & 0.041 & 0.012 \\ 0.000 & 0.000 & 1.083 & 1.083 & 0.631 & 0.270 & 0.093 & 0.027 \\ 0.000 & 0.333 & 1.083 & 1.194 & 0.715 & 0.309 & 0.107 & 0.031 \\ 0.000 & 0.250 & 0.631 & 0.715 & 0.431 & 0.187 & 0.064 & 0.018 \\ 0.000 & 0.116 & 0.270 & 0.309 & 0.187 & 0.081 & 0.028 & 0.008 \\ 0.000 & 0.041 & 0.093 & 0.107 & 0.064 & 0.028 & 0.009 & 0.002 \\ 0.000 & 0.012 & 0.027 & 0.031 & 0.018 & 0.008 & 0.002 & 0.000 \end{bmatrix}. \qquad (2.18) \quad \boxed{\texttt{d253}}$$

Chapter 3

Linear advection and tests

In this chapter, the resolution of the fundamental instrument, the step in linear advection, is explained and comparisons between several methods are performed, in order to remark advantages and disadvantages of each one.

3.1 Time step in linear advection

In this section, we are presenting three algorithms for performing a time step in linear advection:

$$\begin{cases} \dfrac{\partial f}{\partial t} + a\dfrac{\partial f}{\partial x} = 0 \\ f(t_0, x) = f_0(x) \end{cases}$$

Given $f(t^n, x)$ we want to compute $f(t^{n+1}, x)$.

The first and the second methods are based in different methods following the characteristics while the third one is based on the method of lines by direct approximation of the spatial derivatives and Runge-Kutta methods for solving in time.

In order to present the first two methods, a mathematical introduction is needed.

3.1.1 Transport equation

The transport (or advection) equation is:

$$\begin{cases} \dfrac{\partial f}{\partial t} + a(t,x) \cdot \nabla_x f = 0, \quad (t,x) \in [0,T] \times \mathbb{R}^N \\ f(0,x) = f_0(x) \end{cases}$$

where $a : [0,T] \times \mathbb{R}^N \longrightarrow \mathbb{R}^N$.

We want to give results about existence and uniqueness of the solutions for such an equation. In order to do this, first of all we need to introduce the definition of characteristics.

uniqchar **Proposition 3.1.1** (Uniqueness of characteristic $\mathcal{X}$)**.** *If*

$$a \in \mathcal{C}^1([0,T] \times \mathbb{R}^N), \tag{3.1}$$ aisdiff

for all $T > 0$*, and there exists* $k > 0$ *such that*

$$|a(t,x)| \leq k(1+|x|), \ \forall (t,x) \in \mathbb{R}_{\geq 0} \times \mathbb{R}^N, \tag{3.2}$$ agrowslin

then there exists a unique solution,

$$\mathcal{X}(s;t,x) \in \mathcal{C}^1([0,T] \times [0,T] \times \mathbb{R}^N),$$

for all $T > 0$*, of the Cauchy problem*

$$\begin{cases} \dfrac{d\mathcal{X}}{dt} = a(t, \mathcal{X}(t;s,x)) \\ \mathcal{X}(s;s,x) = x \end{cases}$$

The proof can be found in any standard analysis book for ODE's systems and in this particular case in [3].

Going back to problem (3.1.1) the following theorem can be stated:

Theorem 3.1.2 (Existence and uniqueness of strong solutions)**.** *Given the advection field* $a(t,x)$ *satisfying* (3.1) *and* (3.2)*, and the initial data* $f_0 \in \mathcal{C}^1(\mathbb{R}^N)$*, then there exists a unique solution of the Cauchy problem* (3.1.1)*, given by*

$$f(t,x) = f(s, \mathcal{X}(s;t,x)).$$

so, in particular,

$$f(t,x) = f_0(\mathcal{X}(0;t,x)).$$

Proof. By hypotheses (3.1) and (3.2) Proposition 3.1.1 applies, so characteristics $\mathcal{X}(s;t,x)$ exist globally and are unique. We define

$$f(t,x) = f_0(\mathcal{X}(0;t,x)).$$

This $f(t,x)$ is a solution because of the semigroup property of $\mathcal{X}(s)$, and uniqueness comes from proving that if $f(t,x)$ is a solution, then $f(t,x)$ is constant over characteristics, that is,

$$\frac{d}{ds} f(s, \mathcal{X}(s;t,x)) = 0$$

and thus, it must be of the form we have written (see [3] for details).

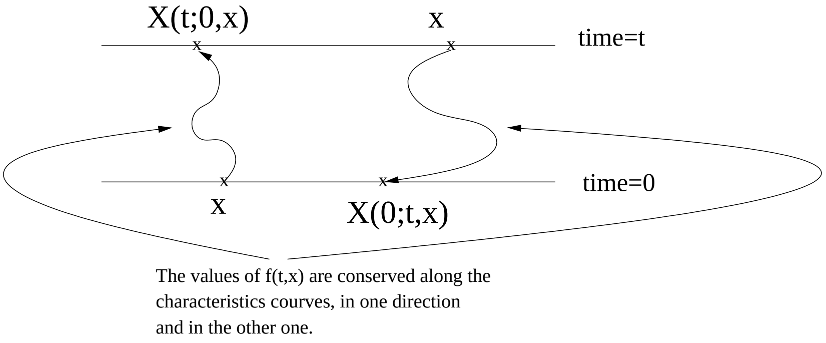

X(t;0,x)
x
time=t
time=0
x
X(0;t,x)
The values of f(t,x) are conserved along the
characteristics courves, in one direction
and in the other one.

Figure 3.1: This is why this equation is called transport equation: the values of $f(t,x)$ are transported along the characteristics.

sindes

3.1.2 Linear advection

The computation of characteristics in case $a(t,x)$ is a real constant is straighforward:

$$\begin{cases} \dfrac{d\mathcal{X}}{ds} = a \\ \mathcal{X}(t) = x \end{cases}$$

gives

$$\mathcal{X}(s;t,x) = x + a(s-t)$$

so that the solution of the initial value problem

$$\begin{cases} \dfrac{\partial f}{\partial t} + a\dfrac{\partial f}{\partial x} = 0 \\ f(t_0,x) = f_0(x) \end{cases}$$

is

$$f(t,x) = f_0(x - a(t-t_0))$$

This is the only result that we need to implement in all the routines concerning advection: the different reconstructions of f_0 will give different properties, like mass conservation or total variation control.

Mass conservation

It is trivial to remark that linear advection is mass-conservative:

$$\int_{\mathbb{R}} f(t,x)dx = \int_{\mathbb{R}} f_0(x-at)dx = \int_{\mathbb{R}} f_0(x)dx = M$$

That is why we would like numerical methods to preserve this property.

3.1.3 Linear advection via Semi Lagrangian Method

Knowing $f(t^n, x_i)$, we want to compute $f(t^{n+1}, x_i)$. Following the characteristics, we know that

$$f(t^{n+1}, x_i) = f(t^n, x_i - a\Delta t)$$

which means that we have to reconstruct the values of $f(t^n, \cdot)$ (of which we do not dispose). In our case, this will be made by

- Lagrange interpolation
- PWENO interpolation

This method has an advantage, which is its easiness of implementation, but has also an important disadvantage: it is not conservative.

The α parameter

The parameter

$$\alpha = a\frac{\Delta t}{\Delta x}$$

describes how close to the grid points we are interpolating: from $f(t^{n+1}, x_i) = f(t^n, x_i - a\Delta t)$,

$$x_i - a\Delta t = x_i - \alpha\Delta x$$

which means that the nearer is α to an integer number, the better is hoped the interpolation to be.

3.1.4 Linear advection via Flux Balance Method

FBM (Flux Balance Method) is used in filbet [13] to construct a conservative method. We already know that

$$f(t + \Delta t, x) = f(t, x - a\Delta t).$$

Now, let us integrate over an interval $[b_1, b_2]$, to get

$$\begin{aligned}
\int_{b_1}^{b_2} f(t + \Delta t, \xi)d\xi &= \int_{b_1}^{b_2} f(t, \xi - v\Delta t)d\xi \\
&= \int_{b_1 - a\Delta t}^{b_2 - a\Delta t} f(t, \xi)d\xi \\
&= \int_{b_1 - a\Delta t}^{b_1} f(t, \xi)d\xi + \int_{b_1}^{b_2} f(t, \xi)d\xi - \int_{b_2 - a\Delta t}^{b_2} f(t, \xi)d\xi.
\end{aligned}$$

If we use as notation

$$\Phi(t, x) = \int_{x - a\Delta t}^{x} f(t, \xi)d\xi,$$

we get

$$\int_{b_1}^{b_2} f(t+\Delta t,\xi)d\xi = \int_{b_1}^{b_2} f(t,\xi)d\xi + \Phi(t,b_1) - \Phi(t,b_2),$$

and dividing by $\Delta = b_2 - b_1$

$$\frac{\int_{b_1}^{b_2} f(t+\Delta t,\xi)d\xi}{\Delta} = \frac{\int_{b_1}^{b_2} f(t,\xi)d\xi}{\Delta} + \frac{\Phi(t,b_1) - \Phi(t,b_2)}{\Delta},$$

which means

$$\bar{f}_{(b_1,b_2)}(t+\Delta t) = \bar{f}_{(b_1,b_2)}(t) + \frac{\Phi(t,b_1) - \Phi(t,b_2)}{\Delta}.$$

This is the local description of mass conservation.
Call $F(t,\cdot)$ the primitive of $f(t,\cdot)$, the numerical method we get is the following:

$$f_i^{n+1} = f_i^n + \frac{\Phi^n(x_{i-\frac{1}{2}}) - \Phi^n(x_{i+\frac{1}{2}})}{\Delta x},$$

where

$$\Phi^n(x_{i-\frac{1}{2}}) = \int_{x_{i-\frac{1}{2}}-a\Delta t}^{x_{i-\frac{1}{2}}} f(t^n,\xi)d\xi = F(x_{i-\frac{1}{2}}) - F(x_{i-\frac{1}{2}} - a\Delta t).$$

Now we need some method to reconstruct what we do not have: either directly $\Phi^n(x_{i-\frac{1}{2}})$ or $F(_{i-\frac{1}{2}}-a\Delta t)$ (if we can compute $F(x_{i-\frac{1}{2}})$).

Reconstruction of $F(t,\cdot)$

We can compute $F(t,x_{i+\frac{1}{2}})$ by putting $F(t,x_{i+\frac{1}{2}}) = \sum_{j=0}^{i} f(t,x_i)\Delta x$ and reconstruct the values $F(t,x_{i-\frac{1}{2}} - a\Delta t)$ and $F(t,x_{i+\frac{1}{2}} - a\Delta t)$ by using

- Lagrange interpolation
- PWENO interpolation

The problems we could find by using these methods is that the positivity is not guaranteed and the oscillations could be uncontrolled, especially by using Lagrange method.

PFC-3 method

In order to assure the positivity and the control of the oscillations in the reconstruction, this method has been introduced in filbet [13, page 70-72]. It is a third order method, and the flux $\Phi_{i+\frac{1}{2}}$ is directly computed: if the wind propagation velocity is positive, let j be the index of the cell which contains

$x_{i+\frac{1}{2}} - a\Delta t$, let $\alpha_i = x_{j+\frac{1}{2}} - (x_{i+\frac{1}{2}} - a\Delta t)$, compute slope correctors ϵ_i^+ and ϵ_i^-, then

$$\begin{aligned}\Phi_{i+\frac{1}{2}} &= \Delta x \sum_{k=j+1}^{i} f_k + \alpha_i \left[f_j + \frac{\epsilon_i^+}{6}\left(1 - \frac{\alpha_i}{\Delta x}\right)\left(2 - \frac{\alpha_i}{\Delta x}\right)(f_{j+1} - f_j) \right. \\ &+ \left. \frac{\epsilon_i^-}{6}\left(1 - \frac{\alpha_i}{\Delta x}\left(1 + \frac{\alpha_i}{\Delta x}\right)(f_j - f_{j-1})\right)\right].\end{aligned}$$

Otherwise, if the wind propagation velocity is negative, let $\alpha_i = x_{j-\frac{1}{2}} - (x_{i+\frac{1}{2}} - a\Delta t)$, then

$$\begin{aligned}\Phi_{i+\frac{1}{2}} &= \Delta x \sum_{k=i+1}^{j-1} f_k + \alpha_i \left[f_j - \frac{\epsilon_i^+}{6}\left(1 - \frac{\alpha_i}{\Delta x}\right)\left(1 + \frac{\alpha_i}{\Delta x}\right)(f_{j+1} - f_j) \right. \\ &- \left. \frac{\epsilon_i^-}{6}(2 + \frac{\alpha_i}{\Delta x}(1 + \frac{\alpha_i}{\Delta x})(f_j - f_{j-1})\right].\end{aligned}$$

Correctors ϵ_i^+ and ϵ_i^- are defined

$$\epsilon_i^+ = \begin{cases} \min\left(1; 2\frac{f_i}{f_{i+1} - f_i}\right) & f_{i+1} > f_i \\ \min\left(1; -2\frac{f_\infty - f_i}{f_{i+1} - f_i}\right) & f_{i+1} < f_i \end{cases}$$

$$\epsilon_i^- = \begin{cases} \min\left(1; 2\frac{f_\infty - f_i}{f_i - f_{i-1}}\right) & f_i > f_{i-1} \\ \min\left(1; -2\frac{f_i}{f_i - f_{i-1}}\right) & f_i < f_{i-1} \end{cases}$$

where $f_\infty = \sup_{i=0}^{N-1} f_i$.

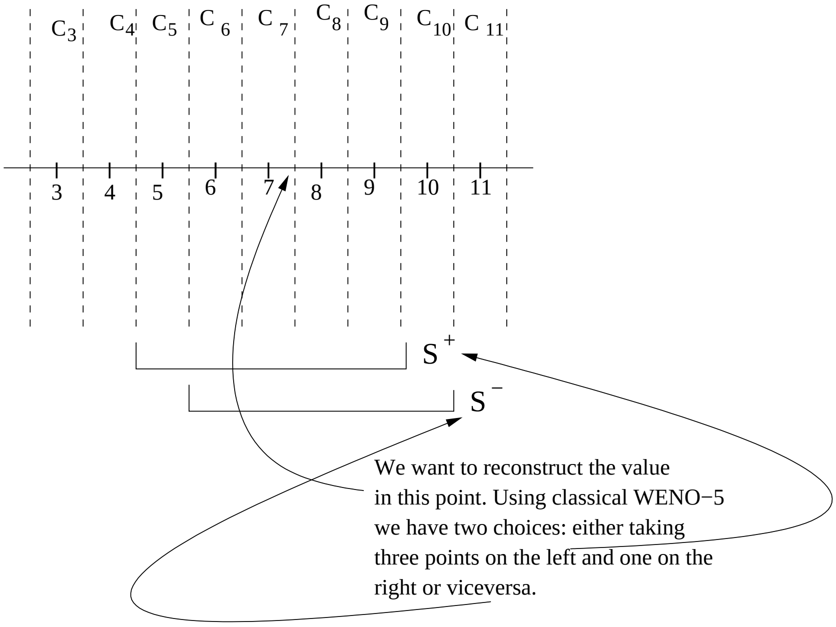
3.1.5 Linear Advection via Flux Reconstruction

Finally, we want to apply the procedure in chiwangshu [35, page 357] for the equation

$$\begin{cases} \dfrac{\partial f}{\partial t} + \dfrac{\partial}{\partial x}[af] = 0 \\ f(0, x) = f_0(x) \end{cases}$$

where a is the wind propagation velocity.
Our strategy in this method will be to approximate $\frac{\partial f_i}{\partial t}$ and then perform a Runge-Kutta step in time. This will be the procedure we follow:

1. Upwinding. The approximation of derivative $\frac{\partial f}{\partial x}(x_i)$ must take into account the direction of propagation. We have two choices, like in Figure wind 3.2. The direction of propagation is given by the sign of a. When the wind a is positive, the information run from the left to the right, and viceversa, so that if $a > 0$ we shall choose $\mathcal{S}^+$ and if $a < 0$ we shall choose $\mathcal{S}^-$ (like in Figure wind 3.2).



We want to reconstruct the value in this point. Using classical WENO−5 we have two choices: either taking three points on the left and one on the right or viceversa.

Figure 3.2: The procedure of choosing the correct stencil $\mathcal{S}^+$ or $\mathcal{S}^-$ is called upwinding. This must be done for the sake od stability of the method. wind

2. Flux reconstruction. Denote by g the flux function, i.e.,

$$g_i = af_i$$

and then compute

$$\hat{f}_{i+\frac{1}{2}} = \begin{cases} g^+_{i+\frac{1}{2}} & a < 0 \\ g^-_{i+\frac{1}{2}} & a \geq 0 \end{cases}$$

where $g^+_{i+\frac{1}{2}}$ and $g^-_{i+\frac{1}{2}}$ are FDWENO reconstructions of g.

3. Runge-Kutta step. Form scheme

$$\frac{df_i}{dt} = -\frac{1}{\Delta x}\left[\hat{f}_{i+\frac{1}{2}} - \hat{f}_{i-\frac{1}{2}}\right]$$

and perform a Runge-Kutta step in time to get $f(t+\Delta t, x_i)$.

Runge-Kutta methods

Runge-Kutta methods are well-known multi-stage algorithms used to solve initial value problems for ordinary differential equations.

$$\begin{cases} \dfrac{du}{dt} = f(t,u) \\ u(t_0) = u_0 \end{cases}$$

The classical fourth order Runge-Kutta requires four evaluations of f:

$$k_1 = \Delta t f(t_0, u_0), \qquad k_2 = \Delta t f(t_0 + \frac{1}{2}\Delta t, u_0 + \frac{1}{2}k_1),$$

$$k_3 = \Delta t f(t_0 + \frac{1}{2}\Delta t, u_0 + \frac{1}{2}k_2), \qquad k_4 = \Delta t f(t_0 + \Delta t, u_0 + k_3),$$

with $u(t_0 + \Delta t) = u_0 + \frac{1}{6}\left[k_1 + 2k_2 + 2k_3 + k_4\right]$ (see lambert [26] for details).

It is also useful to have implemented Runge-Kutta-3 algorithm: it has the advantage of being of decreasing total variation [1], say it is TVD, i.e.

$$TV[f(t)] \leq TV[f_0]$$

which means that it does not create oscillations. Refer to shu-osher [32] for details. The classical Runge-Kutta-4 does not give certainty of that.

Runge-Kutta-3

1. First stage: compute
$$k_1 = \Delta t f(t_0, u)$$
2. Second stage: compute
$$k_2 = \Delta t f(t_0 + \frac{1}{2}\Delta t, u_0 + \frac{1}{2}k_1)$$
3. Third stage: compute
$$k_3 = \Delta t f(t_0 + \Delta t, u_0 - k_1 + 2k_2)$$
4. compute
$$u(t_0 + \Delta t) = u(t_0) + \frac{1}{6}\left[k_1 + 4k_2 + k_3\right]$$

3.2 Tests

In this chapter, the attention will be focused on the difference between the results given by different interpolation methods. The Cauchy problem

$$\begin{cases} \dfrac{\partial f}{\partial t} + \dfrac{\partial f}{\partial x} = 0 \\ f(0, x) = f_0(x) \end{cases}$$

has as exact solution

$$f(t, x) = f_0(x - t).$$

[1] Discrete total variation is defined

$$TV[f] = \sum_{i=0}^{N-2} |f_{i+1} - f_i|$$

.

3.2.1 Accuracy

In the following test we have computed the L^{∞} error after time $t_{max} = 10$ ($\Delta t = 0.05$) of the numerical solution with respect to the exact solution. Initial function is set to $f_0(x) = \sin^2(x)$. SL-WENO-6,4 works as a 6th-order method, and its accuracy grows as the number of points grows, analogously for the FBM-WENO-6,4 method. FBM-PFC-3 is very low order and has an irregular behaviour; anyway its main property is the control of oscillations, and not the accuracy.

pts	$SL-Lagr{-}6$		$SL-WENO{-}6{,}4$	
40	$4.543e-05$	$---$	$1.594e-04$	$---$
80	$6.660e-07$	6.091	$2.330e-06$	6.095
160	$1.005e-08$	6.050	$3.336e-08$	6.126
320	$1.542e-10$	6.026	$3.248e-10$	6.682
640	$2.391e-12$	6.011	$2.674e-12$	6.924

pts	$FBM-Lagr{-}6$		$FBM-WENO{-}6{,}4$		$FBM-PFC{-}3$	
40	$4.543e-05$	$---$	$1.179e-04$	$---$	$7.724e-01$	$---$
80	$6.660e-07$	6.091	$1.281e-06$	6.523	$7.494e-03$	6.687
160	$1.005e-08$	6.050	$1.124e-08$	6.832	$1.866e-03$	2.005
320	$1.543e-10$	6.025	$1.543e-10$	6.187	$4.650e-04$	2.005
640	$3.687e-12$	5.386	$3.006e-12$	5.682	$3.247e-04$	0.518

3.2.2 Total variation control

The Discrete Total Variation is defined as

$$\sum_{i=0}^{N-2} \|f_i^n - f_{i+1}^n\|$$

We have compared the results given by several methods for two initial functions: a step and a peak,

$$f^{step}(x) = \begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases}$$

$$f^{peak}(x) = \begin{cases} 1 + \alpha x & -\frac{1}{\alpha} \leq x \leq 0 \\ 1 - \alpha x & 0 \leq x \leq \frac{1}{\alpha} \end{cases}$$

As the solution of linear advection at time t is just a translation of initial solution, theoretically the total variation is constant.

In the evolution of the step, Lagrange interpolation never conserves or controls the total variation, and strong oscillations are produced, like in Figure totv 3.3. SL-WENO-6,4 conserves the total variation, while FBM-WENO-6,4 does not.

In the evolution of the peak, we see in Figure ahiahiahi 3.4 that PWENO, in both semi-lagrangian and FB methods, conserves the total variation better than PFC-3 method, for which it decreases strongly. From this point of view, FBM-PWENO-6,4 is a bit better than SL-WENO-6,4.

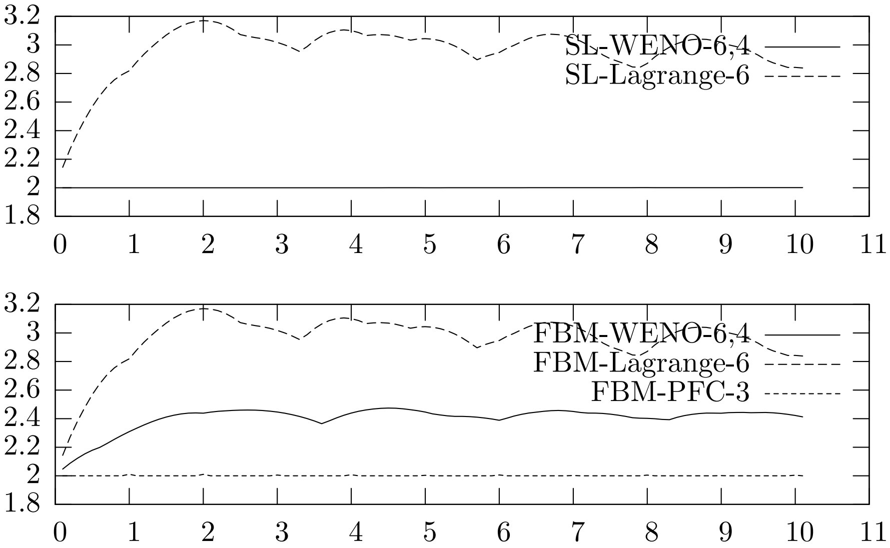


Figure 3.3: The evolution of Discrete Total Variation against time. In this test, $N = 100$, $x \in [-\pi, \pi]$, $\Delta t = 0.1$, $t_{max} = 10$, $f_0(x) = f^{step}(x)$. totv

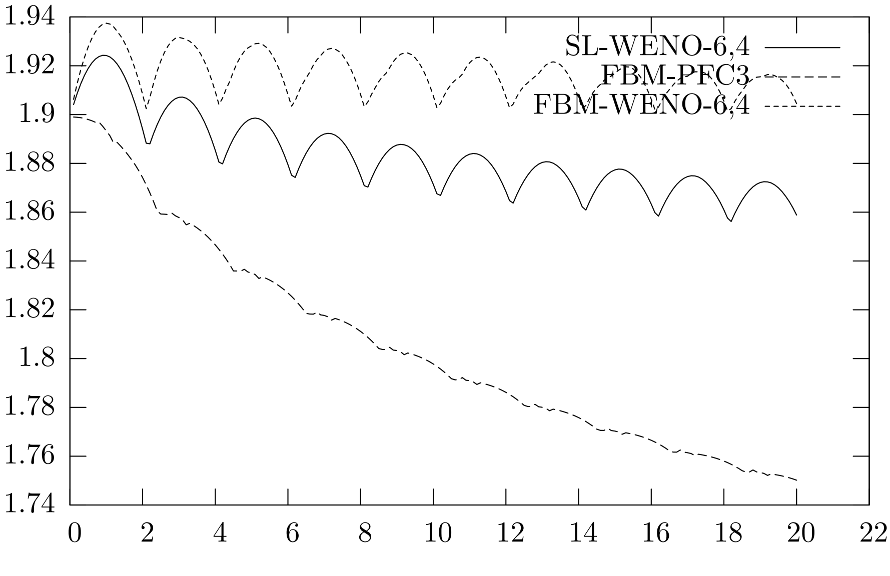


Figure 3.4: The evolution of Discrete Total Variation against time. In this test, $N = 100$, $x \in [-\pi, \pi]$, $\Delta t = 0.1$, $t_{max} = 20$, $f_0(x) = f^{peak}(x)$. ahiahiahi

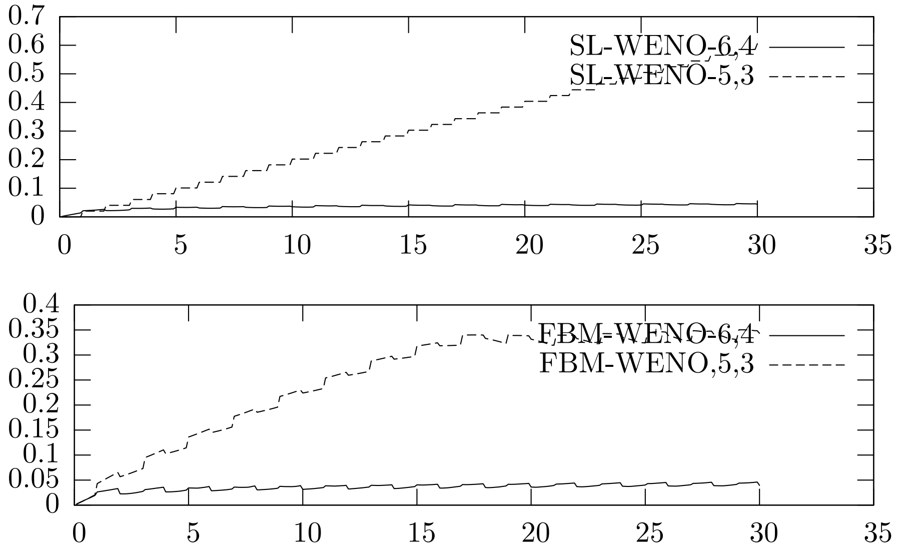


Figure 3.5: The evolution of L^1-norm error against the time for $N = 100$, $x \in [-\pi, \pi]$, $\Delta t = 0.1$, $t_{max} = 30$, $f_0(x) = f^{shock}(x)$. dsfgf

3.2.3 Avoiding disphasement errors

We want to focus now the attention on how important is the choice of the substencils. In Figure dsfgf 3.5, we see that WENO-6,4 produces less error than WENO-5,3 both in semi-lagrangian method and in flux balance methods. This difference is due to the fact that in WENO-5,3 not all the substencils "feel" that there is a jump point, like in Figure jump53 3.6. Suppose we are interpolating close to x_i, and suppose the jump point is situated between x_i and x_{i+1}. If there is a substencil which does not contain the irregularity, WENO method will give weights 0 to all of them but this one. This means that the jump has parcourred the distance Δx in the time Δt (see Figure dtjump53 3.7), i.e., it runs with wrong speed $\frac{\Delta x}{\Delta t}$ with an error with respect to the real speed of

$$\begin{aligned} \Delta &= a - \frac{\Delta x}{\Delta t} = \alpha \frac{\Delta x}{\Delta t} - \frac{\Delta x}{\Delta t} \\ &= \frac{\Delta x}{\Delta t} [\alpha - 1] \end{aligned}$$

which, after time T gives a phase error of

$$\Delta \Phi = \frac{\Delta x}{\Delta t} [\alpha - 1] \times T.$$

This induces to think that it is essential to use a WENO method such that every substencil contains the discontinuity, like WENO-6,4. In general, it

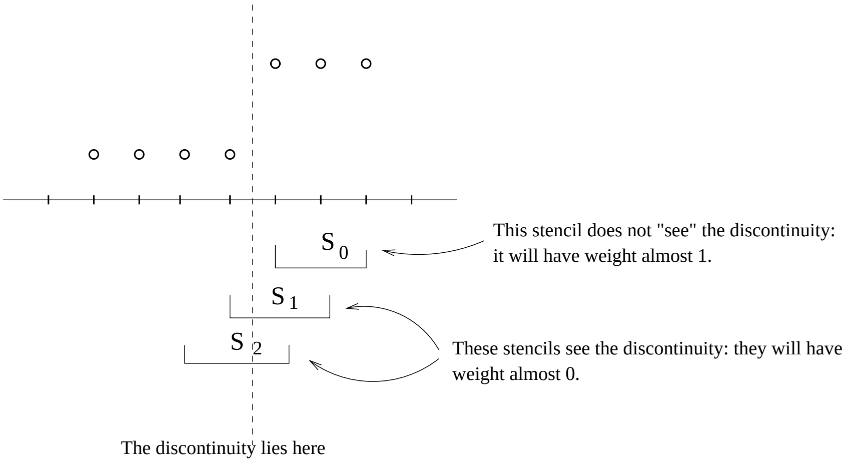

S 0
This stencil does not "see" the discontinuity: it will have weight almost 1.
S 1
S 2
These stencils see the discontinuity: they will have weight almost 0.
The discontinuity lies here

Figure 3.6: In WENO-5,3 problems are caused by not all the substencils feeling that high derivatives occur: this makes the weights neglect the substencils where the discontinuity is produced. jump53

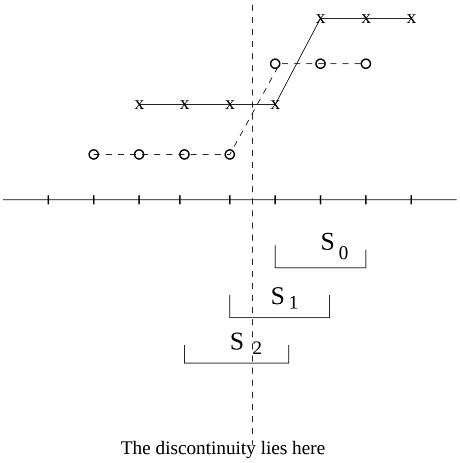


Figure 3.7: At next time step this is the distance parcoured by the jump: in time Δt the displacement in Δx. dtjump53

must be

$$lpo \geq \left[\frac{ntot+1}{2}\right]+1.$$

3.2.4 Amplification factor and disphasement error

We want to perform a test on the harmonics of the solution, using a Discrete Fourier Transform (DFT). The expansion is defined this way

$$f_j^n = \sum_{k=0}^{N-1} \hat{f}_k^n \exp(ikx_j),$$

where

$$\hat{f}_k^n = \frac{1}{N}\sum_{j=0}^{N-1} f_j^n \exp(-ikx_j)$$

where the values f_j^n represents the numerical solution of the problem at time t_n and spatial grid point x_j.

The exact solution for a single mode K in advection equation can be easily computed. Imposing that

$$f(t,x) = \sum_{k=0}^{N-1} \hat{f}_k(t)\exp(ikx)$$

must satisfy the advection equation $\frac{\partial f}{\partial t} + a\frac{\partial f}{\partial x} = 0$, we get

$$\sum_{k=0}^{N-1} \frac{d}{dt}\hat{f}_k(t)\exp(ikx) + a\sum_{k=0}^{N-1} \hat{f}_k(t) ik \exp(ikx) = 0.$$

Multiplying by $\exp(iKx)$ and then integrating over $[-\pi,\pi]$ in dx, we deduce

$$\frac{d}{dt}\hat{f}_K(t)\exp(iKx) + iKa\hat{f}_K(t)\exp(iKx) = 0.$$

Simplifying,

$$\frac{d}{dt}\hat{f}_K(t) + iKa\hat{f}_K(t) = 0,$$

and thus,

$$\frac{d}{dt}\hat{f}_K(t) = -iKa\hat{f}_K(t),$$

so

$$\hat{f}_K(t) = \hat{f}_K(0)\exp(-iKx).$$

The test will be the following one:

1. Choose
$$(\hat{f}_0^0, \hat{f}_1^0, \hat{f}_2^0, ..., \hat{f}_{N-1}^0) \in \mathbb{C}^N.$$

2. Construct as initial function

$$f_0(x) = \sum_{k=0}^{N-1} \hat{f}_k^0 \exp(ikx).$$

3. Consider separately the evolution via characteristics method of $\Re(f_0)$ and $\Im(f_0)$. Adding them, we get $f(t^n, x_j)$.

4. Fix one mode K to study.

5. Consider the exact solution for that mode

$$\hat{f}_K^n = \hat{f}_K^0 \exp(-iKat^n).$$

6. Consider the numerical solution for that mode

$$\tilde{f}_K^n = \frac{1}{N} \sum_{j=0}^{N-1} f_j^n \exp(-iKx_j).$$

7. Compute the amplification error

$$a.e. = \frac{\|\hat{f}_K^n\|_{\mathbb{C}}}{\|\tilde{f}_K^n\|_{\mathbb{C}}}.$$

8. Compute the disphasement error

$$ph.e. = \arg(\hat{f}_K^n) - \arg(\tilde{f}_K^n).$$

3.2.5 Dispersion and diffusion

Both the amplitude and the disphasement error are computed as a function of the parameter α which measures how good the interpolation is going to be for the linear advection equation. We shall show both errors for the case of $f_0(x) = e^{iKx}$ as initial data.

The amplitude error measures the numerical diffusion error, that is, how much the harmonics decrease in the evolution of the numerical solution. As wave number k gets smaller, the error increases, like we can see in Figures ampli ampli_8 3.8-3.11.

The phase error measures the dispersion error, i.e. how much the harmonics run with a wrong velocity. As wave number k gets larger, the error increases, like we can see in Figures phase phase_8 3.12-3.15.

For this number of points, Lagrange reconstructions are better, both in semi-lagrangian and in FB methods. PWENO behaves better in semi-lagrangian method than in FBM, where it is more dispersive.

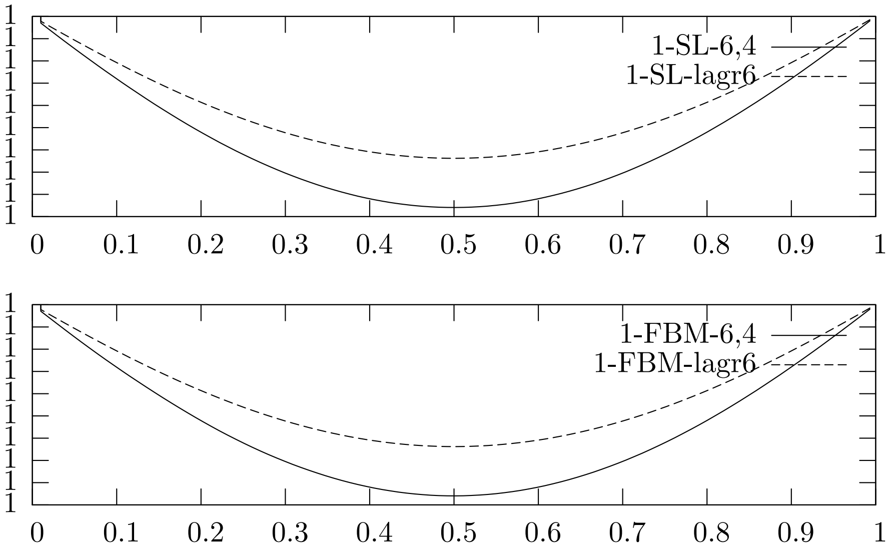


Figure 3.8: Amplitude error for mode $K = 1$, $N = 100$, $\Delta t = 0.05$, $t_{max} = 1$ `ampli_1`

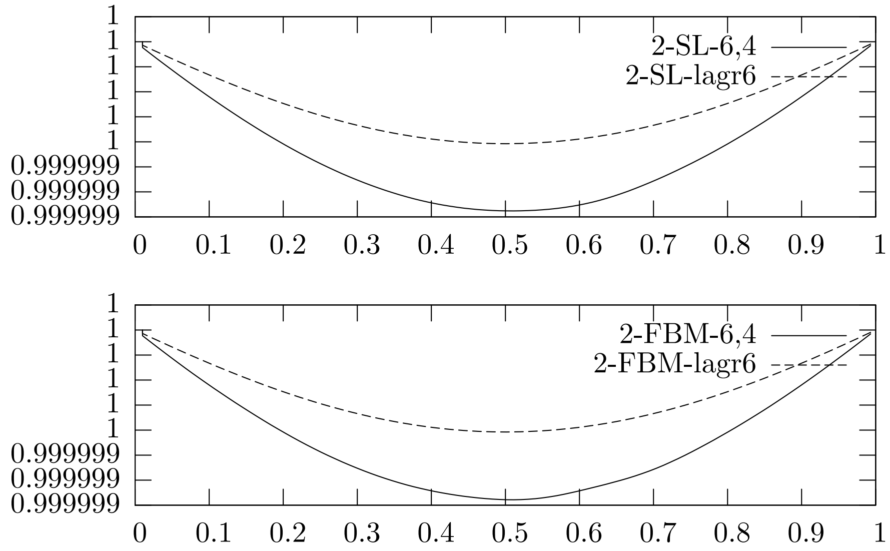


Figure 3.9: Amplitude error for mode $K = 2$, $N = 100$, $\Delta t = 0.05$, $t_{max} = 1$ `ampli_2`

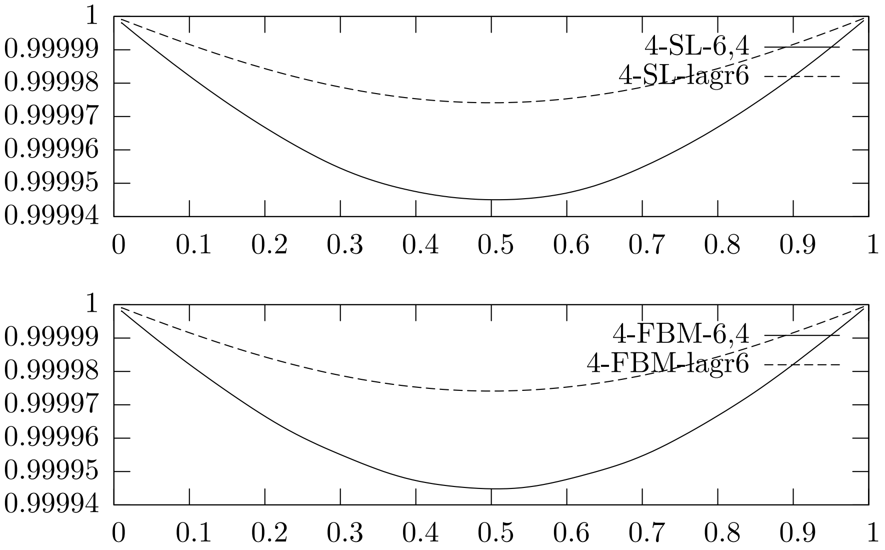


Figure 3.10: Amplitude error for mode $K = 4$, $N = 100$, $\Delta t = 0.05$, $t_{max} = 1$ `ampli_4`

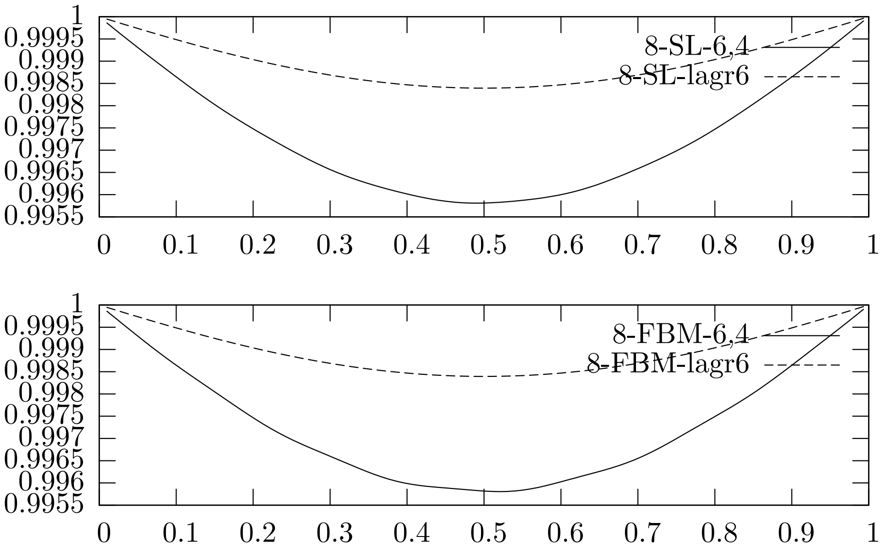


Figure 3.11: Amplitude error for mode $K = 8$, $N = 100$, $\Delta t = 0.05$, $t_{max} = 1$ `ampli_8`

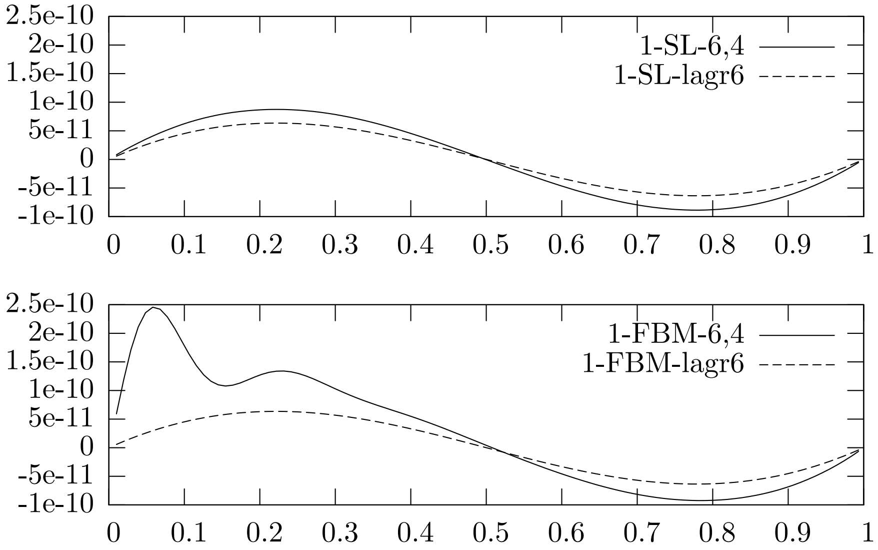


Figure 3.12: Phase error for mode $K = 1$, $N = 100$, $\Delta t = 0.05$, $t_{max} = 1$ `phase_1`

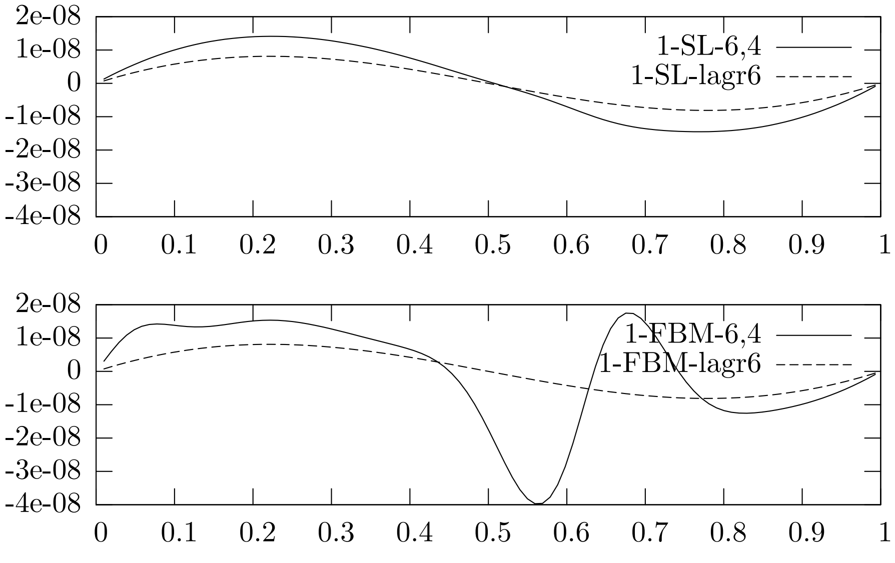


Figure 3.13: Phase error for mode $K = 2$, $N = 100$, $\Delta t = 0.05$, $t_{max} = 1$ `phase_2`

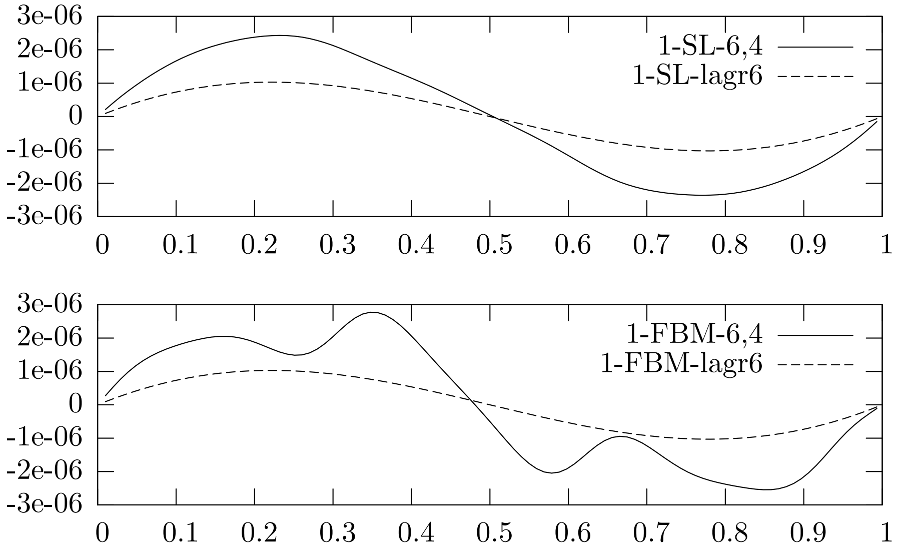


Figure 3.14: Phase error for mode $K = 4$, $N = 100$, $\Delta t = 0.05$, $t_{max} = 1$ phase_4

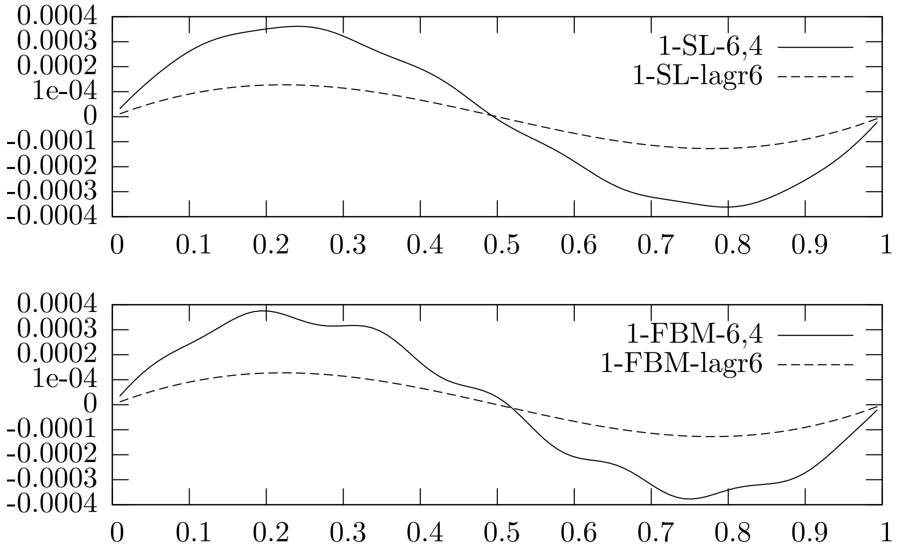


Figure 3.15: Phase error for mode $K = 8$, $N = 100$, $\Delta t = 0.05$, $t_{max} = 1$ phase_8

Chapter 4

Solving Vlasov's systems

In this section, the techniques for solving several kinds of Vlasov-Poisson-Boltzmann equations are explained. The fundamental instruments are the time-splitting procedures.

4.1 Strang's splittings

Take the equation

$$\frac{\partial \Psi}{\partial t} = \mathcal{L}\Psi$$

where $\mathcal{L}$ is a linear operator generator of a $\mathcal{C}_0$-semigroup. Its formal solution is

$$\Psi(t) = \exp(t\mathcal{L})\Psi(0).$$

We split the operator $\mathcal{L}$ into two parts:

$$\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2$$

and we take them as the infinitesimal generators of the $\mathcal{C}_0$-semigroups

$$\mathcal{F}_i(t) = \exp\left(t\mathcal{L}_i\right).$$

Note that $\mathcal{L}_1 + \mathcal{L}_2$ is also the infinitesimal generator of a $\mathcal{C}_0$-semigroup because of Trotter's product theorem. *Strang's splitting* consists in taking

$$\mathcal{F}(\Delta t) = \mathcal{F}_1\left(\frac{\Delta t}{2}\right)\mathcal{F}_2\left(\Delta t\right)\mathcal{F}_1\left(\frac{\Delta t}{2}\right).$$

This can be proven to be a second order scheme in the sense that

$$\mathcal{F}\left(\Delta t\right) = \exp\left(\Delta t\mathcal{L}\right) + O\left(\Delta t^3\right).$$

Consider now two cases of Strang's splittings, the two we are going to use.

4.1.1 Strang's time splitting between Vlasov and Boltzmann

Given equation

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f + F \cdot \nabla_v f = \mathcal{Q}[f]$$

where the force field $F(t,x)$ and the Boltzmann operator $\mathcal{Q}[f]$ are known, we advance a step in time by advancing separately in Vlasov and in Boltzmann parts, i.e., given

$$f(t^n, x_i, v_j)$$

we proceed in this way:

1. Perform a $\frac{\Delta t}{2}$ time step in Vlasov part, i.e., solving

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f + F \cdot \nabla_v f = 0$$

to get

$$f^*(t^n, x_i, v_j).$$

2. Perform a Δt time step in Boltzmann part, i.e., solving

$$\frac{\partial f}{\partial t} = \mathcal{Q}[f]$$

to get

$$f^{**}(t^n, x_i, v_j).$$

3. Perform a $\frac{\Delta t}{2}$ time step in Vlasov part, i.e., solving

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f + F \cdot \nabla_v f = 0$$

to get

$$f(t^{n+1}, x_i, v_j).$$

like in Figure firsttimesplitting 4.1.

Remark. In our previous notation, we split the operator

$$\mathcal{L} = -v \cdot \nabla_x - F \cdot \nabla_v + \mathcal{Q}$$

into

$$\begin{cases} \mathcal{L}_1 = -v \cdot \nabla_x - F \cdot \nabla_v \\ \mathcal{L}_2 = \mathcal{Q} \end{cases}.$$

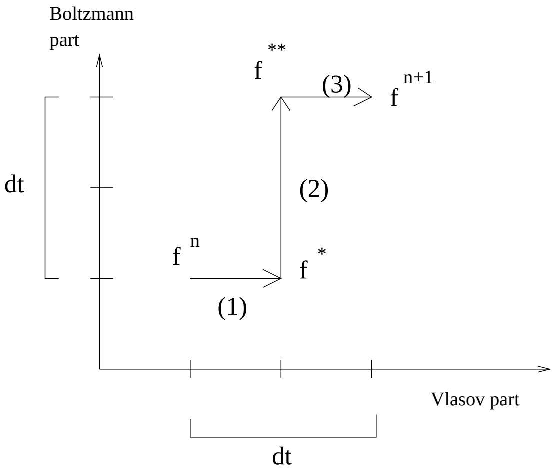


Figure 4.1: Time splitting between Vlasov and Boltzmann parts. firsttimesplitting

4.1.2 Strang's splitting between x and v

Our purpose is now to solve

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f + F \cdot \nabla_v f = 0$$

given the force field $F(t, x)$. The procedure (which was originally introduced by Cheng and Knorr timesplitting [7]) is to split the Vlasov equation into either phases x and v, in this way: given $f\left(t^n, x, v\right)$,

1. Consider v fixed, and take the free transport equation

   $$\frac{\partial f}{\partial t} + v \frac{\partial f}{\partial x} = 0$$

   and perform a $\frac{\Delta t}{2}$ time step in the x-direction to get

   $$f^*\left(t^n, x, v\right) = f\left(t^n, x - v\frac{\Delta t}{2}, v\right)$$

   by one of the advection algorithms previously described.

2. Compute the force field $F\left(f^*\left(t^n, x, v\right)\right)$.

3. Consider x fixed, and take equation

   $$\frac{\partial f^*}{\partial t} + F \frac{\partial f^*}{\partial v} = 0$$

   and perform a Δt time step in the v-direction to obtain

   $$f^{**}\left(t^n, x, v\right) = f^*\left(t^n, x, v - F\Delta t\right).$$

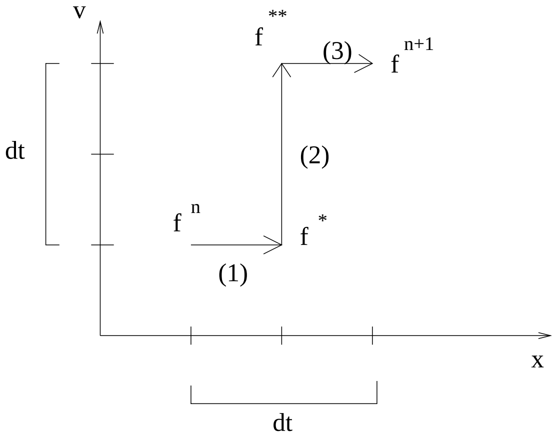


Figure 4.2: Time splitting between phases x and v in $\mathbb{R}^N$. 2ts

4. Consider v fixed, and take the free transport equation

$$\frac{\partial f^{**}}{\partial t} + v\frac{\partial f^{**}}{\partial x} = 0$$

and perform a $\frac{\Delta t}{2}$ time step in the x-direction to obtain

$$f\left(t^{n+1}, x, v\right) = f^{**}\left(t^n, x - v\frac{\Delta t}{2}, v\right).$$

This scheme is second order in time.

Remark. In our former terms, we split the operator

$$\mathcal{L} = -v \cdot \nabla_x - F \cdot \nabla_v$$

into

$$\left\{\begin{array}{l} \mathcal{L}_1 = -v \cdot \nabla_x \\ \mathcal{L}_2 = -F \cdot \nabla_v \end{array}\right. .$$

A similar scheme is used to solve the relativistic Vlasov equation (see chapter 1):

$$\frac{\partial f}{\partial t} + \frac{p}{(1+p^2)^{\frac{1}{2}}}\frac{\partial f}{\partial x} - \left(\eta^{-1}E + A\frac{\partial A}{\partial x}\right)\frac{\partial f}{\partial p} = 0$$

assuming both E and A are given functions just by substituting v by the relativistic velocity

$$\frac{p}{(1+p^2)^{\frac{1}{2}}}$$

and the force field by $F(t,x) = \eta^{-1}E + A\frac{\partial A}{\partial x}$.

4.2 Computing the force field

The force field can be of different kinds; in our cases it will be the solution of Poisson's equation (electrostatic field) or a combination of Poisson's and Maxwell's equations (coupling electrostatic and magnetic fields).

4.2.1 Resolution of $1D$ Poisson equation

$1D$-Poisson equation is straightforward computed, for instance by numerical integration. Given

$$\frac{\partial^2 \Phi}{\partial x^2} = \Psi,$$

we solve it by

$$\Phi(x) = \int_{\eta=0}^{\eta=x} \int_{\xi=0}^{\xi=\eta} \Psi(\xi) d\xi d\eta + Ax + B$$

where A and B are constants we shall define in order to satisfy some properties, like periodicity, boundary conditions or the global neutrality of electric field.

4.2.2 1D Maxwell's equations

In this section, we shall explain the instruments we are going to use to compute the force field for the quasi-relativistic Vlasov-Maxwell model (QRVM 1.10).

Let us summarize here the magnitudes we need to evolve the vector potential A for Maxwell's equation:

1. The y-component of the electric field $\mathcal{E}$ (see chapter 1):

$$\mathcal{E} = -\frac{\partial A}{\partial t} \tag{4.1}$$

ele

2. The y-component of the magnetic field B (see chapter 1):

$$\frac{\partial \mathcal{E}}{\partial x} = -\frac{\partial B}{\partial t} \tag{4.2}$$

Bi

3. The final equation for the vector potential A substituting the values of the $\mathcal{E}$ and B:

$$-\frac{\partial B}{\partial x} = -\eta^{-2} A \int f dp + \frac{\partial \mathcal{E}}{\partial t}. \tag{4.3}$$

last

There is no problem in the computation of the x-component of the electrostatic field E given by 1D Poisson's equation since the advection step in p-direction does not modify the density (see Figure nomodifydensity 4.3) and therefore, the force field is constant in time $E[f^*]$ during the advection step in p-direction.

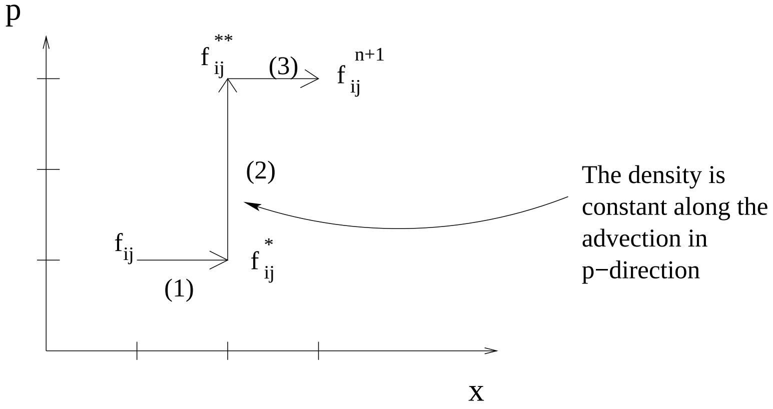


Figure 4.3: The density $\int f dp$ does not change along the advection steps in p-direction

nomodifydensity

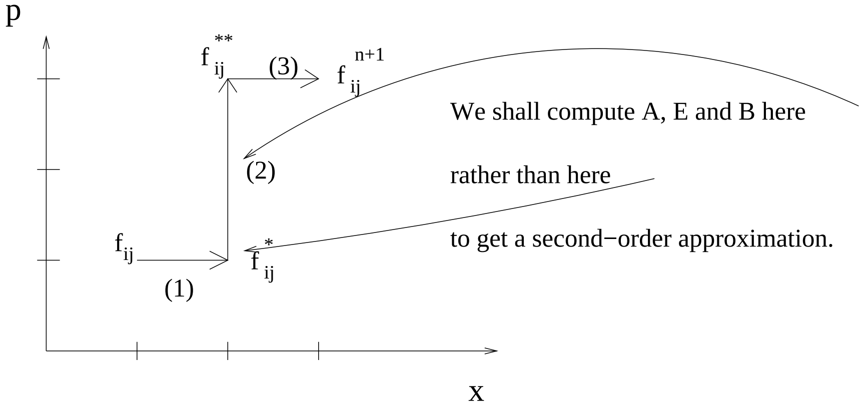

p
f ij **
(3)
f ij n+1
We shall compute A, E and B here
(2)
rather than here
f ij
f ij *
to get a second−order approximation.
(1)
x

Figure 4.4: We have to take care where we compute the magnitudes, because we can gain one order in time.

here

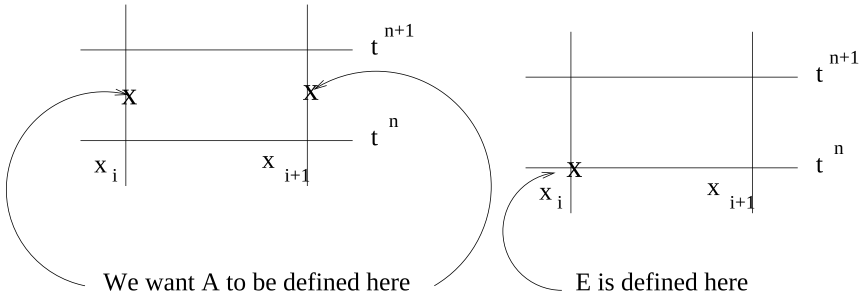


Figure 4.5: A and $\mathcal{E}$ are defined in the marked positions. $\boxed{\texttt{AmE}}$

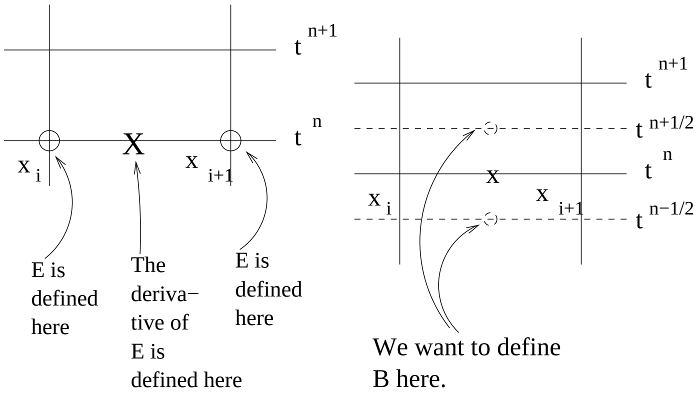


Figure 4.6: $\frac{\partial\mathcal{E}}{\partial x}$ is approximated in the marked points to define B at $x_{i+\frac{1}{2}}$ and $t^{n+\frac{1}{2}}$. $\boxed{\texttt{B}}$

In order to get a second-order in time approximation in p-direction advection, we shall compute A and $\frac{\partial A}{\partial t}$ at time marked in Figure $\overset{\texttt{here}}{\text{4.4}}$. We want to compute $A(t^{n+\frac{1}{2}}, x_i)$, like in figure $\overset{\texttt{AmE}}{\text{4.5}}$. All the other magnitudes shall be centered in order to achieve this goal. Therefore, we discretize ($\overset{\texttt{ele}}{\text{4.1}}$) this way

$$\frac{A_i^{n+\frac{1}{2}} - A_i^{n-\frac{1}{2}}}{\Delta t} = -\mathcal{E}_i^n.$$

$\mathcal{E}_i^n$ is defined like in figure $\overset{\texttt{AmE}}{\text{4.5}}$. Then, we discretize ($\overset{\texttt{Bi}}{\text{4.2}}$) as

$$-\frac{\mathcal{E}_{i+1}^n - \mathcal{E}_i^n}{\Delta x} = \frac{B_{i+\frac{1}{2}}^{n+\frac{1}{2}} - B_{i+\frac{1}{2}}^{n-\frac{1}{2}}}{\Delta t}.$$

We want now to define B like in Figure $\overset{\texttt{B}}{\text{4.6}}$. Discretization of last condition ($\overset{\texttt{last}}{\text{4.3}}$) gives

$$\frac{\mathcal{E}_i^{n+1} - \mathcal{E}_i^n}{\Delta t} = \eta^{-2} A_i^{n+\frac{1}{2}} \rho_i^* - \frac{B_{i+\frac{1}{2}}^{n+\frac{1}{2}} - B_{i-\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta x}.$$

Resumé

The three conditions ($\overset{\texttt{ele}}{4.1}$)-($\overset{\texttt{last}}{4.3}$) give

1.
$$A_i^{n+\frac{1}{2}} = A_i^{n-\frac{1}{2}} - \mathcal{E}_i^n \Delta t \tag{4.4}$$ $\boxed{\texttt{eqA}}$

2.
$$B_{i+\frac{1}{2}}^{n+\frac{1}{2}} = B_{i+\frac{1}{2}}^{n-\frac{1}{2}} - \frac{\Delta t}{\Delta x}\left(\mathcal{E}_{i+1}^n - \mathcal{E}_i^n\right) \tag{4.5}$$ $\boxed{\texttt{eqB}}$

3.
$$\mathcal{E}_i^{n+1} = \mathcal{E}_i^n + \eta^{-2} A_i^{n+\frac{1}{2}} \rho_i^* \Delta t - \frac{\Delta t}{\Delta x}\left(B_{i+\frac{1}{2}}^{n+\frac{1}{2}} - B_{i-\frac{1}{2}}^{n+\frac{1}{2}}\right). \tag{4.6}$$ $\boxed{\texttt{eqC}}$

Procedure scheme

Knowing $f_{i,j}^n$, $\mathcal{E}_i^n$, $A_i^{n-\frac{1}{2}}$ and $B_{i+\frac{1}{2}}^{n-\frac{1}{2}}$ we shall perform a time-step:

1. From $\mathcal{E}_i^n$ and $A_i^{n-\frac{1}{2}}$ in ($\overset{\texttt{eqA}}{4.4}$) we get $A_i^{n+\frac{1}{2}}$.
2. From $\mathcal{E}_i^n$ and $B_{i+\frac{1}{2}}^{n-\frac{1}{2}}$ in ($\overset{\texttt{eqB}}{4.5}$) we get $B_{i+\frac{1}{2}}^{n+\frac{1}{2}}$.
3. From $\mathcal{E}_i^n$, previous steps and ($\overset{\texttt{eqC}}{4.6}$) we get $\mathcal{E}_i^{n+1}$.
4. Compute
$$B_i^{n+\frac{1}{2}} = \frac{\left(B_{i-\frac{1}{2}}^{n+\frac{1}{2}} + B_{i+\frac{1}{2}}^{n+\frac{1}{2}}\right)}{2}.$$
5. Compute
$$F_i^* = \eta^{-1} E_i^* + A_i^{n+\frac{1}{2}} B_i^{n+\frac{1}{2}}.$$

4.3 Collision

The linear Boltzmann homogeneous equation

$$\frac{\partial f}{\partial t} = \frac{1}{\tau}\left[\rho(t,x) M_{\theta_0}(v) - f(t,x,v)\right]$$

is an ODE for f(t) in which (x, v) are parameters. This ODE is of the form $x' = A - Bx$, whose solution is $x(t) = e^{-Bt} x_0 + \frac{A}{B}\left(1 - e^{-Bt}\right)$, therefore

$$f(t^n + \Delta t, x, v) = e^{-\frac{1}{\tau}\Delta t} f(t^n, x, v) + \rho(t^n, x) M_{\theta_0}(v)\left(1 - e^{-\frac{1}{\tau}\Delta t}\right).$$

Chapter 5

Performing PWENO method

In this chapter, we apply our numerical method to solve Vlasov-Poisson-Boltzmann equations in several problems of interest in plasma physics and charge particle transport in semiconductors.

5.1 $1D$ Vlasov-Boltzmann equation with a confining potential

We solve the $1D$ Vlasov-Boltzmann equation with a confining potential $\Phi_0(x)$ and the simplest linear collision operator:

$$\begin{cases} \dfrac{\partial f}{\partial t} + v\dfrac{\partial f}{\partial x} - \dfrac{\partial \Phi_0}{\partial x}\dfrac{\partial f}{\partial v} = \dfrac{1}{\tau}\left[\rho M_1 - f\right] \\ f(0,x) = f_0(x) \end{cases}$$

From carrillocaceresgoudon [5] we know that the solution tends to a global equilibrium given by

$$f_s = M\left(\int_{\mathbb{R}} \exp\left(-\Phi_0(x)\right)dx\right)^{-1} \exp\left(-\Phi_0(x)\right) M_1(v)$$

in L^1 norm, whenever the external potential verifies the confinement conditions:

$$\begin{cases} \bullet\, \Phi_0 \geq 0, \Phi_0 \in C^\infty(\mathbb{R}), \\ \bullet\, \exp(-\Phi_0(x)) \in L^1(\mathbb{R}), \\ \bullet\, \Phi_0 \text{ is a bounded perturbation of a uniformly convex potential on } \mathbb{R}\text{:} \\ \Phi_0 = \Phi_0^{uc} + \Phi_0^{bp} \text{ such that} \\ \qquad\qquad \text{there exists } \lambda_1 > 0 \text{ such that} \frac{\partial^2}{\partial^2 x}\Phi_0^{uc}(x) \geq \lambda_1,\ \forall x \in \mathbb{R}, \\ \text{and} \\ \qquad\qquad \text{there exists } a \text{ and } b \text{ such that } 0 < a \leq \Phi_0^{bp}(x) \leq b < \infty,\ \ \forall x \in \mathbb{R}. \end{cases}$$

The decay rate was proved to be "almost exponential" (see carrillocaceresgoudon [5]), i.e.,

$$\|f - f_s\|_{L^1}^2 \leq H[f; f_s] \leq C(\epsilon, f_0)t^{-\frac{1}{\epsilon}}, \tag{5.1} \quad \boxed{\texttt{bound}}$$

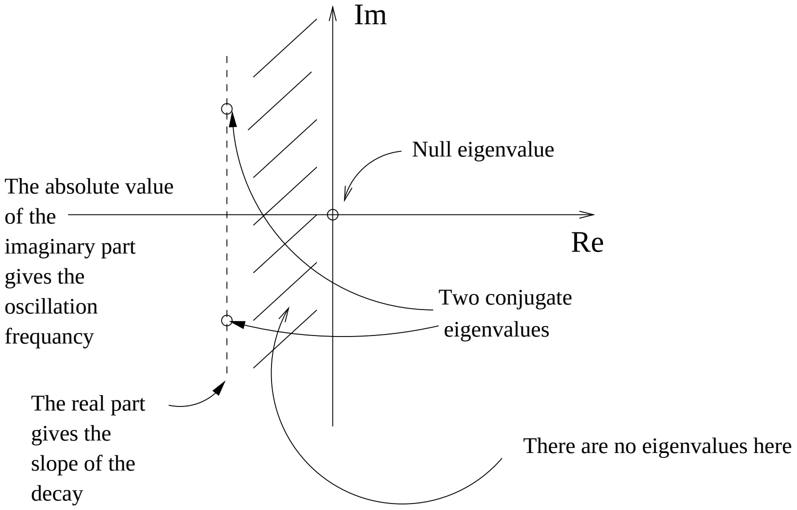

Im
Re
Null eigenvalue
The absolute value of the imaginary part gives the oscillation frequancy
Two conjugate eigenvalues
The real part gives the slope of the decay
There are no eigenvalues here

Figure 5.1: The spectrum of operator (5.1). oscilla

for all $\epsilon > 0$. Global and local relative entropies are measures of how far is f from the global equilibrium f_s and the local equilibrium $\rho(t,x)M_1$.

$$\begin{cases} H[f;f_s] = \displaystyle\int_{\mathbb{R}}\int_{\mathbb{R}} \frac{|f-f_s|^2}{f_s}\,dvdx \\ \tilde{H}[f;\rho M_1] = \displaystyle\int_{\mathbb{R}}\int_{\mathbb{R}} \frac{|f-\rho M_1|^2}{f_s}\,dvdx \end{cases}.$$

Global and local relative entropies satisfy the ODE inequalities system (see [5]):

$$\begin{cases} -\dfrac{d}{dt}H[f;f_s] \geq K\tilde{H}[f;\rho M_{\theta_0}] \\ \dfrac{d^2}{dx^2}\tilde{H}[f;\rho M_{\theta_0}] \geq K' H[f;f_s] - C(f,\epsilon)\tilde{H}[f;\rho M_{\theta_0}] \end{cases}. \qquad (5.2)$$ h-htilda

We will show numerical results in the particular case of $\Phi_0(x) = \frac{x^2}{2}$. In this case, the Vlasov part gives a rotation of the initial function $f_0(x,v)$ while the collision part thermalizes the velocity distribution towards $M_1(v)$. Moreover, an analysis of the spectrum of operator

$$f \longmapsto -v\frac{\partial f}{\partial x} + x\frac{\partial f}{\partial v} + \frac{1}{\tau}\left[\rho M_1 - f\right]$$

has recently been done in [18] showing the existence of an spectral gap in a suitable L^2-weighted space and thus, of the exponential convergence in that space and as a consequence in L^1 towards f_s.

Both the previous ODEs inequalities (5.2) and the spectral analysis suggest the appearance of oscillations in the trend of solutions towards global

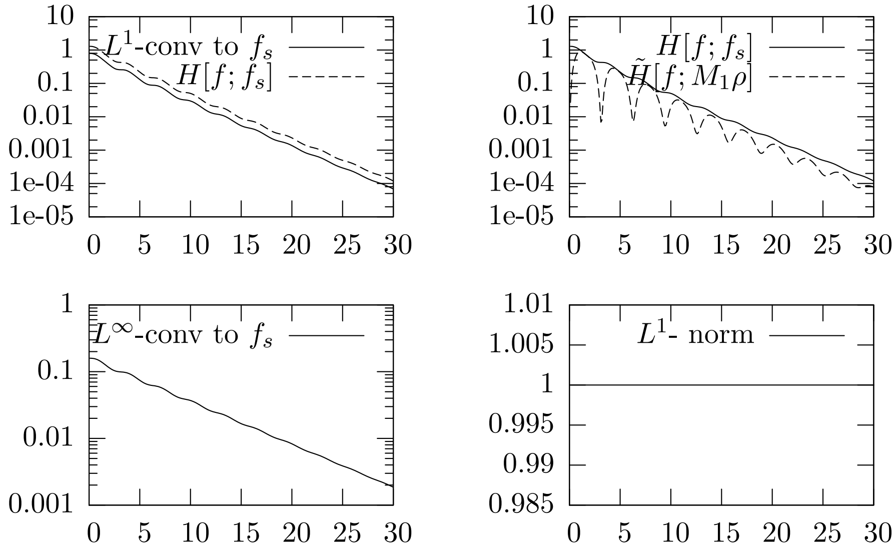


Figure 5.2: Test performed with 256×256 points, $\Delta t = 0.1$, SL method with PWENO-6,4, $f_0^{(1)}(x) = Z_1 \sin^2\left(\frac{x^2}{2}\right) e^{-\frac{x^2+v^2}{2}}$. twopeaks

equilibrium. The ODEs inequalities (5.2) shows that the trend of convergence towards local equilibria is compensated by the transport term that should push the solution out of the local equilibria manifold whenever the solution approaches a local equilibria which is not the global equilibrium. This fact suggests an oscillation both in the local and global relative entropy. On the other hand, assuming the first non-zero eigenvalues in the spectrum are given by a pair of conjugate eigenvalues λ_1 and λ_2 then, we expect oscillations of the L^2-weighted norm with a slope decay given by $\Re(\lambda_i) < 0$ and oscillation frequency given by the absolute value of $\Im(\lambda_i)$. We refer to [18] for details. The spectrum looks like in Figure 5.1.

In the upper left graphs of Figures 5.2 and 5.3 we see, for two different initial functions normalized to have unit mass, that the L^1-convergence of f towards f_s is led by the decay of $H[f; f_s]$, like in (5.1). The convergence is clearly exponential as the results of [18] prove for the L^1-norm.

In the lower left graphs we see that L^∞-convergence is also expected to be exponential, although this result has not been proven yet. In the upper right graphs we see that the oscillations of global and relative entropies correspond, due to the couplings (5.2). In the lower right graphs we observe that the mass seems pretty well conserved.

We shall now perform tests with different domain lengths and different initial functions: the oscillation rate and the decay slope should not change, because we are not using periodic conditions. In fact, we are solving the

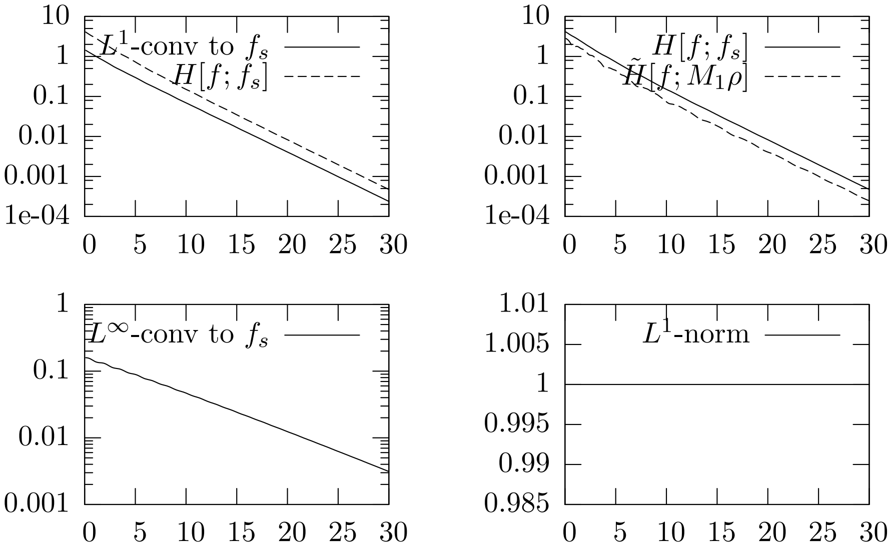


Figure 5.3: Test performed with 256 × 256 points, $\Delta t = 0.1$, SL method with PWENO-6,4, $f_0^{(2)}(x) = Z_2 \sin^2\left(\frac{x^2}{2}\right) sin^2\left(\frac{v^2}{2}\right) e^{-\frac{x^2+v^2}{2}}$. fourpeaks

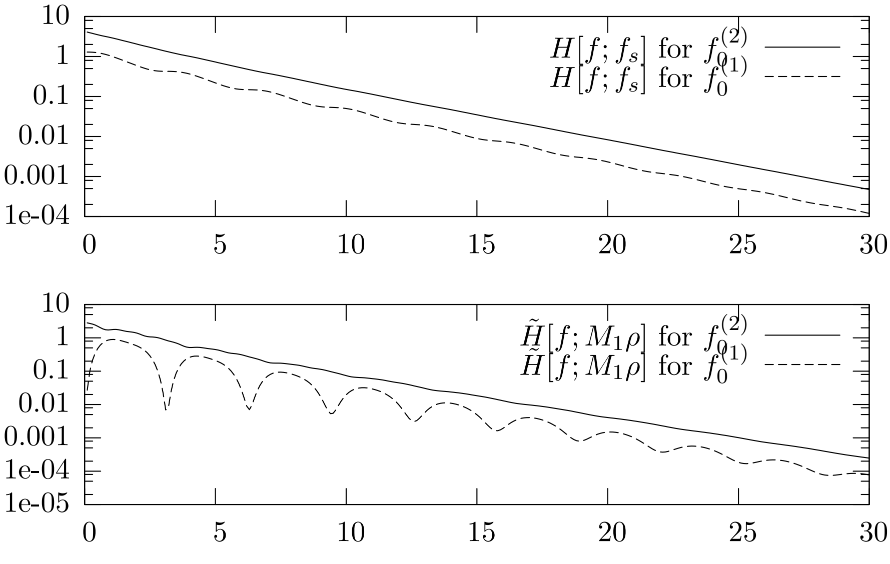


Figure 5.4: From this comparison, it is quite clear that the oscillation rate is the same for both initial functions, even if the behaviour is different. compH

Cauchy problem by neglecting the distribution function f outside a suitable domain chosen in such a way that the values of f near the border are almost negligible.

In Figure 5.4 we compare the decay of global and relative entropies for two different initial functions. Even if the amplitude of the oscillation is different, it seems evident that the oscillation rate and the decay slope correspond.

The system seems to "hesitate" between states where it is close to a local equilibrium ρM_1 and the convergence to the global equilibrium f_s. In [12] similar oscillations have been reported in the case of the full non-linear Boltzmann equation for rarefied gases in a box with periodic boundary conditions. A numerical approximation of the slope γ and the frequency ω of the decaying oscillations towards global equilibrium gives:

$f_0(x)$	L	ω	γ
$f_0^{(1)}(x)$	4π	3.15	-0.298368
$f_0^{(1)}(x)$	6π	3.15	-0.298872
$f_0^{(2)}(x)$	4π	3.125	-0.304400
$f_0^{(2)}(x)$	6π	3.125	-0.304858

A refinement study has been perfomed to check that the oscillation frequency and the decay slope do not depend on the dimensions of the domain, nor on the initial datum we choose: they are determined by the system itself.

5.2 1D Landau damping

Landau theory has been developed in order to study the propagation of small amplitude waves, with wave number k, in a uniform plasma (with no magnetic field and no collisions); the positively charged ions are supposed not to move since they are much heavier than electrons, and the only transport is due to the electrons. The mathematical study is based on a linearization of the Vlasov-Poisson equation. The total energy is conserved (according to the Vlasov-Poisson equation theory), but the interesting phenomenon is that the Landau theory conjectures an interchange of energy between the electric field and resonant particles driven by the wave. Resonant particles are the ones which get trapped by the potential, as in Figure 5.5. The decay of the electric field is oscillating, with a period similar to the bounce time[1] $\tau = 2\pi\left(\frac{m_e}{ekE_0}\right)^{\frac{1}{2}}$ (E_0 is the amplitude of the electric field), where the bounce time is the time particles spend to be reflected inside the potential well, that is the time they spend to be bounced from point x_1 to point x_2 in Figure 5.5. The oscillations are due to the exchange of energy between the particles

[1] The time for the electron to shift its position relative to the wave.

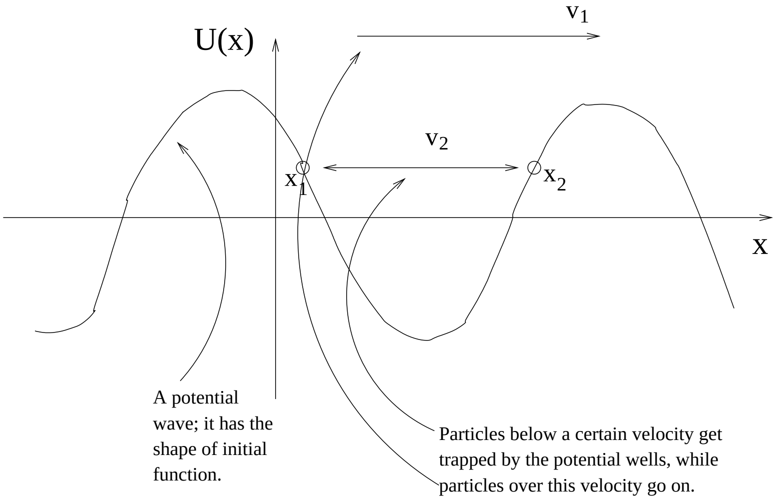

U(x)
v1
v2
x1
x2
x
A potential wave; it has the shape of initial function.
Particles below a certain velocity get trapped by the potential wells, while particles over this velocity go on.

Figure 5.5: The phenomenon of particle trapping. landau

(kinetic energy) and the wave (potential energy) during these reflections. In the particular case of an initial Maxwellian distribution, it has been proven there is a net transfer of energy from the electric field to the particles, and the electric energy is almost linearly damped, without oscillations, because no phenomenon of particle trapping appears.

It is interesting to check how precise is the reconstruction of the electric energy[2] performed by the different methods. It is expected to decay, but when approaching the recurrence time $T_R = \frac{2\pi}{k\Delta v}$ the reconstructions may be less accurate and eventually, after this time, it becomes completely wrong. Remark that this recurrence time is a numerical and non physical effect.

We shall set as initial function

$$f_0(x,v) = \frac{1}{\sqrt{2\pi}} e^{-\frac{v^2}{2}} \left[1 + \alpha \cos(kx)\right] \tag{5.3}$$

initlandau

Parameter k shall be chosen in order to get a periodic function, i.e. according to the length of the device. In our tests, $x \in [0, 4\pi]$ and $k = 0.5$; in this way we get 4 periods in x-direction. Parameter α is (of course) most influent on the results, its meaning is a sort of non-linearity rate in Landau phenomena. It does not influence the oscillation rate of the decay of the electric field, but it does influence the amplitude of these oscillations, and, most of all, as α grows larger, the decay rate of the electric energy is damped. So, to make

[2] We recall electric energy is defined as

$$\int |E(t,x)|^2 dx$$

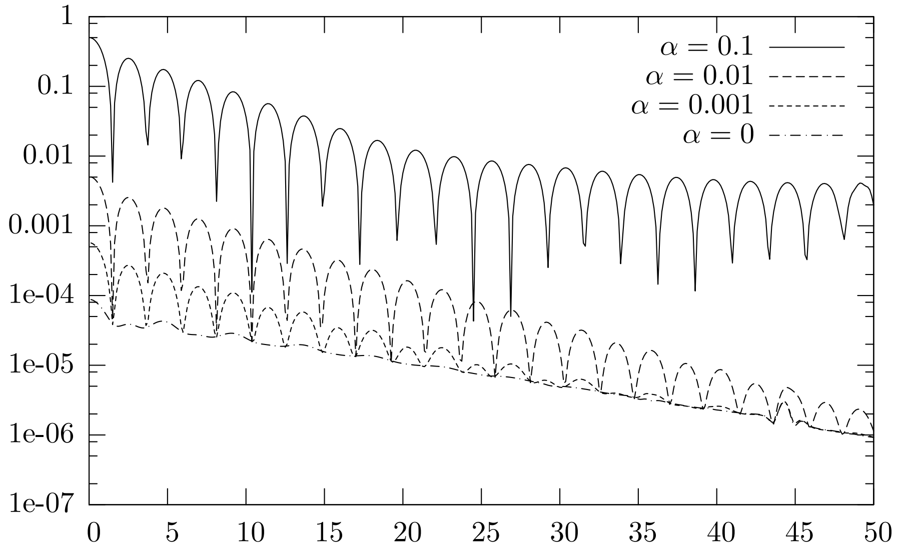


Figure 5.6: The decay of electric field for different α.

compalpha

a resumé (look at Figure compalpha 5.6): if we set $\alpha = 0$, the electric energy is linearly (in logarithmic scale) damped; if we set a small α, like $\alpha = 0.01$ in our example, we can observe that the decay of the damping rate is negligible, and we can observe the phenomenon of particle trapping: the electric energy decays with oscillations; if we set a large α, like $\alpha = 0.5$, we can of course observe the phenomenon of particle trapping in the oscillations (and note that the oscillation rate is always the same) and we can also observe that initially the electric energy decays linearly (in log. scale) but after some time the decay rate is killed and the electric energy starts oscillating around some equilibrium point.

An interesting point is to observe whether the total energy is conserved or not. In the case of Vlasov-Poisson equation, the total energy is the sum of the kinetic and the potential energy as it appears in (totalenergyvlasovpoisson 1.3).

5.2.1 Linear $1D$ Landau damping

Let us take $\alpha = 0.01$ in (initlandau 5.3), this case is called linear Landau damping since the behavior is expected to be led by the linearization of the Vlasov-Poisson equation around the maxwellian distribution. This case does not really provide problems in numerical computations. No strong oscillations occur in either phase. Reconstructions given by both methods with any interpolation method are nearly the same.

If we take the solutions given by Lagrange-19 as exact (there is no difference in using SL or FBM), here is a table with L^∞ and L^1 errors of

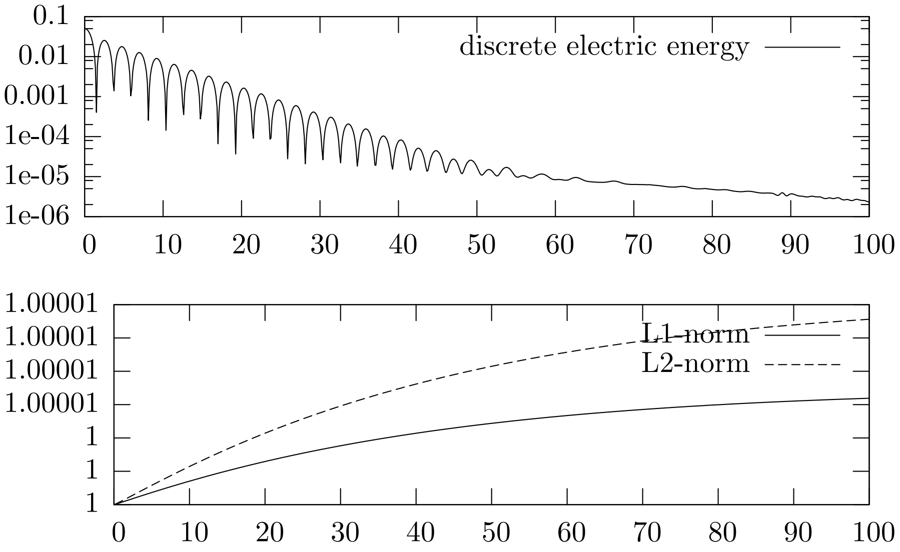


Figure 5.7: Evolution of discrete electric energy and the L^1 and L^2 norms in linear Landau damping, for $N_x = 256$, $N_v = 256$, $\Delta t = 0.125$, $t_{max} = 100$, Lagrange-19 interpolation in SL method.

`lld_lagr19`

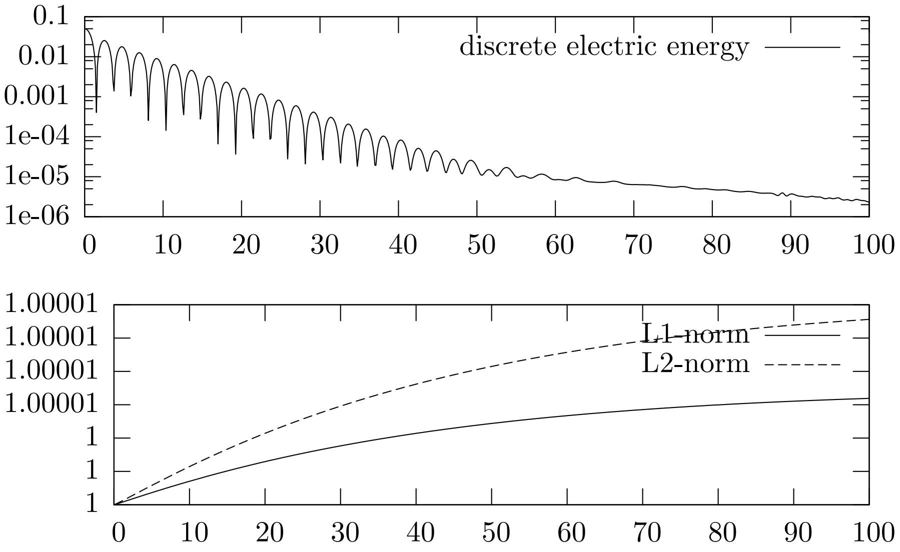


Figure 5.8: Evolution of discrete electric energy and the L^1 and L^2 norms in linear Landau damping, for $N_x = 256$, $N_v = 256$, $\Delta t = 0.125$, $t_{max} = 100$, Lagrange-19 interpolation in FB method.

`lld_fbm_lagr19`

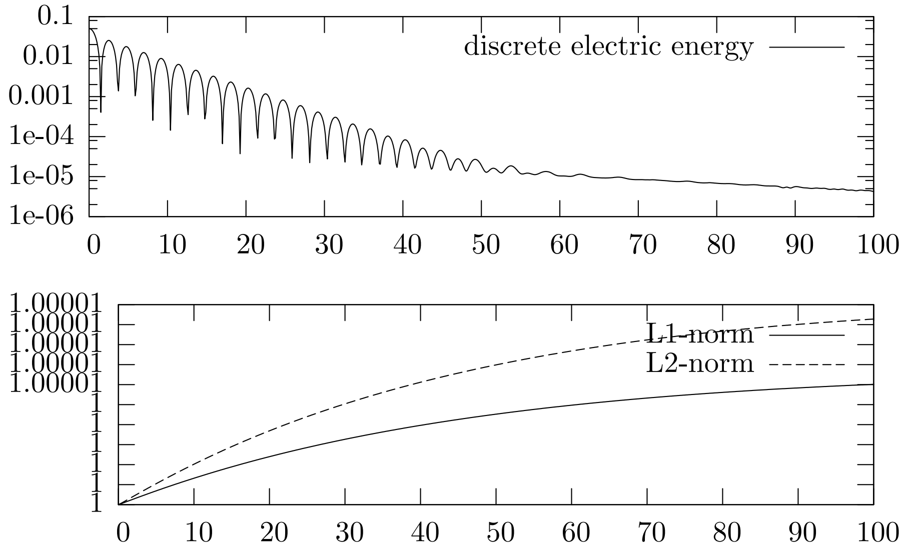


Figure 5.9: Evolution of discrete electric energy and the L^1 and L^2 norms in linear Landau damping, for $N_x = 256$, $N_v = 256$, $\Delta t = 0.125$, $t_{max} = 100$, WENO-6,4 interpolation in SL method.

lld_sl_64

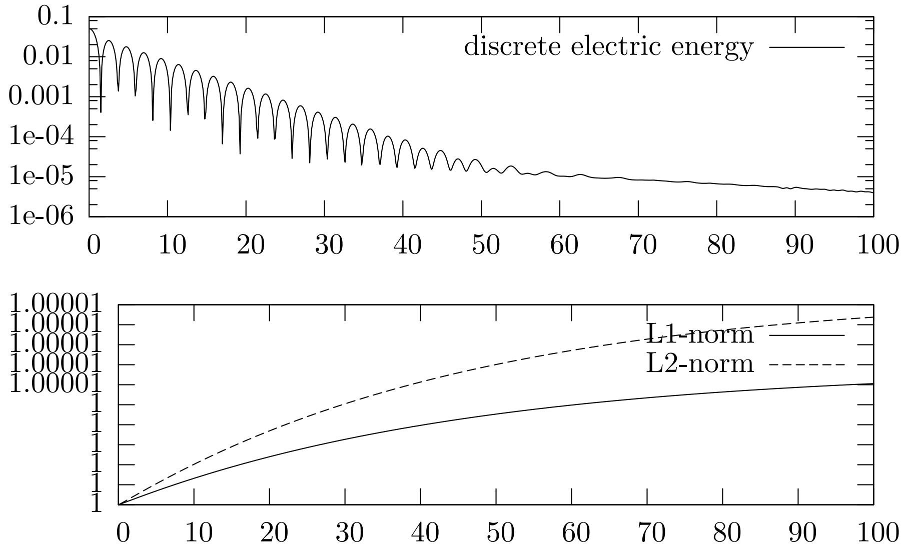


Figure 5.10: Evolution of discrete electric energy and the L^1 and L^2 norms in linear Landau damping, for $N_x = 256$, $N_v = 256$, $\Delta t = 0.125$, $t_{max} = 100$, WENO-6,4 interpolation in FB method.

lld_fbm_64

the macroscopic density given by the the other methods with respect to Lagrange-19.

used method	L^∞ – error	L^1 – error
SL-WENO-6,4	0.000044	0.000019
FBM-WENO-6,4	0.000045	0.000019
SL-WENO-7,4	0.000004	0.000003
FBM-WENO-7,4	0.000003	0.000003
FBM-WENO-9,5	0.000002	0.000002

We can see that most of all the order of the method has influence on the accuracy of the result since no great oscillations are produced.

5.2.2 Non-linear $1D$ Landau damping

Set $\alpha = 0.5$ in (initlandau 5.3), this case is called non-linear Landau damping since the behavior obtained from the linearisation is expected not to hold. In non-linear Landau damping the most problematic phenomenon is the filamentation. In the macroscopic density (v-integration of distribution function f) strong oscillations are produced, and the method must be able to control them properly. Lagrange reconstruction is expected to work poorly due to the appearance of undesired oscillations. In fact, these strong oscillatory behavior is equivalent to the appearance of strong gradients, and numerically speaking, they produce the same problems as shocks although we deal here with very smooth functions.

Semilagrangian method coupled with WENO-6,4 does not provide a very good reconstruction: at a certain time a strong mass loss appears, which influences the amplitude of the oscillations. If you compare Figures nlld_sl_6_4 5.15 and nlld_sl_lagr19 5.11, strong oscillations occur in the macroscopic density, and WENO tends to control them; as semi-lagrangian method was not developed in order to converve mass, this regularization is what induces the mass loss. In FBM, (Figure nlld_fbm_6_4 5.17) the oscillations are controlled and the system behaves better because we do not loose mass. This method works very well in this case.

As for the conservation of the total energy, Flux Balance method works better than Semi-Lagrangian: in FBM with WENO-6,4 the oscillation is about 0.04%, while in SL with WENO-6,4 it is about 0.25%.

In this case, Lagrange-19 reconstructions give non reliable results since the increase of the total variation of the solution is so large that most of the reported oscillations are spurious. This is even clearer when we take the solutions given by Lagrange-19 and compute a table with L^∞ and L^1 differences of the macroscopic density given by the the other methods with

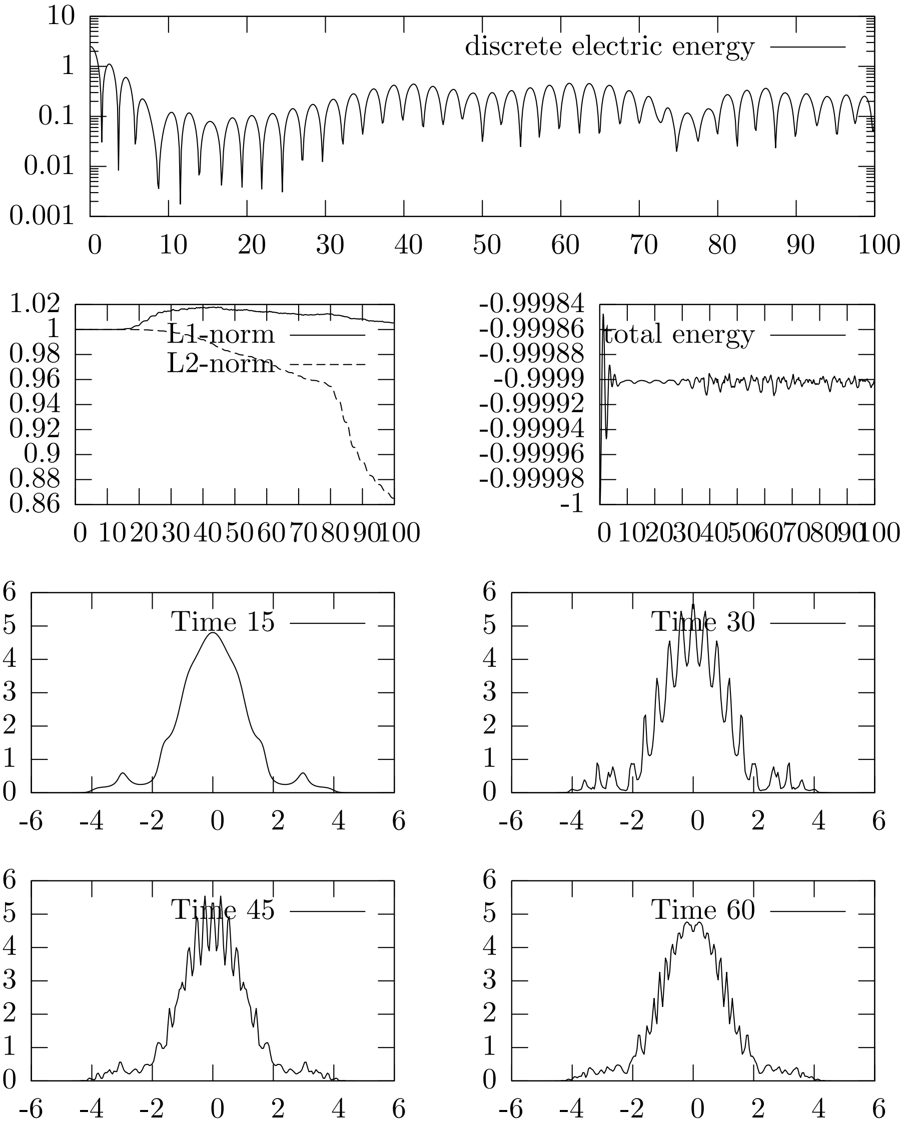


Figure 5.11: Non-linear Landau damping. $N_x = 256$, $N_v = 256$, $\Delta t = 0.125$,Lagrange-19 interpolation in SL method. Upside: evolution of discrete electric energy, evolution of L^1 and L^2 norms, evolution of the total energy. Downside: evolution of v-integrated distribution function.

nlld_sl_lagr19

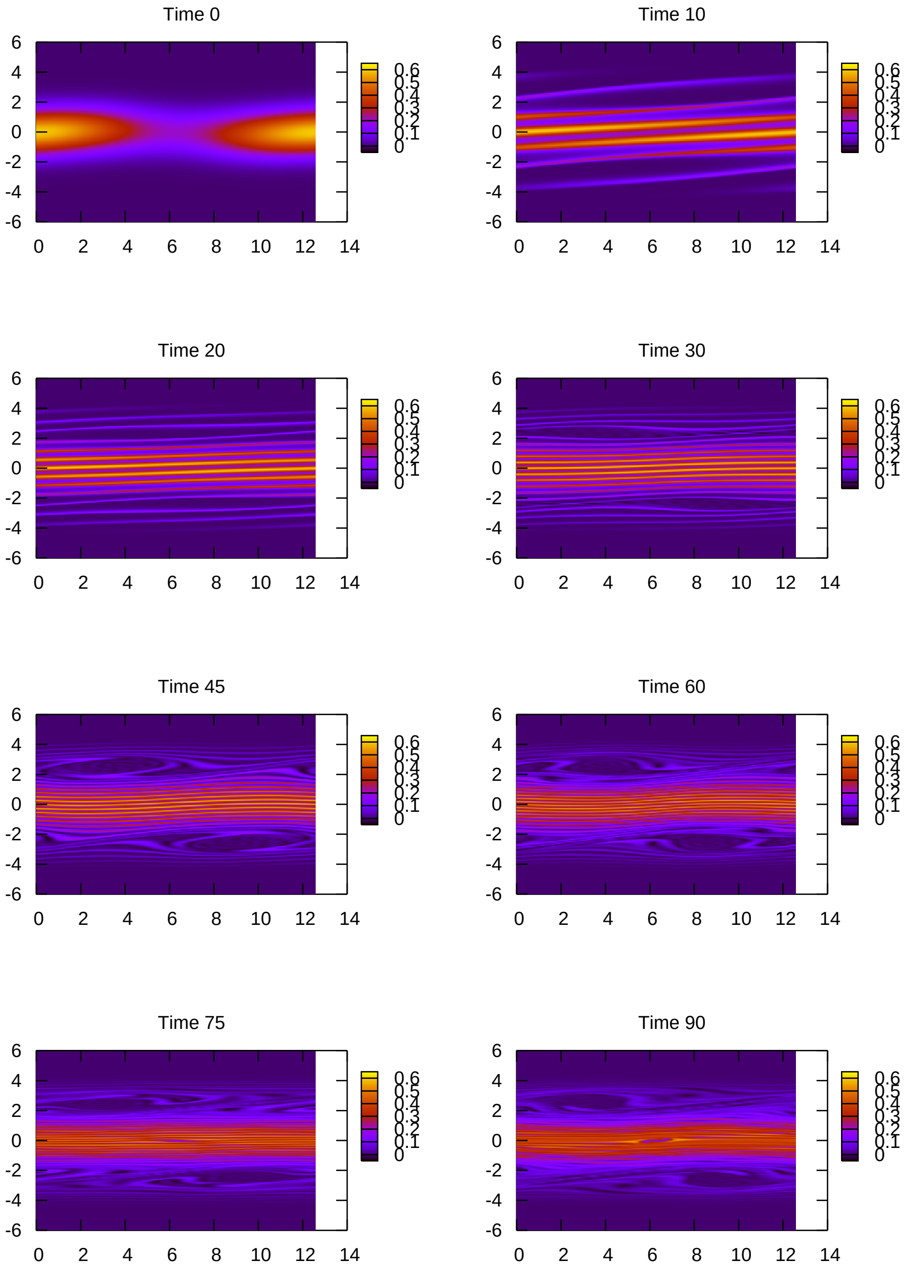


Figure 5.12: Evolution of level curves in non-linear Landau damping, for 256×256 points, $\Delta t = 0.125$, with SL-Lagrange-19 method. Levels are: 0.05, 0.3 and 0.6.

sl_lagr19_lc_1

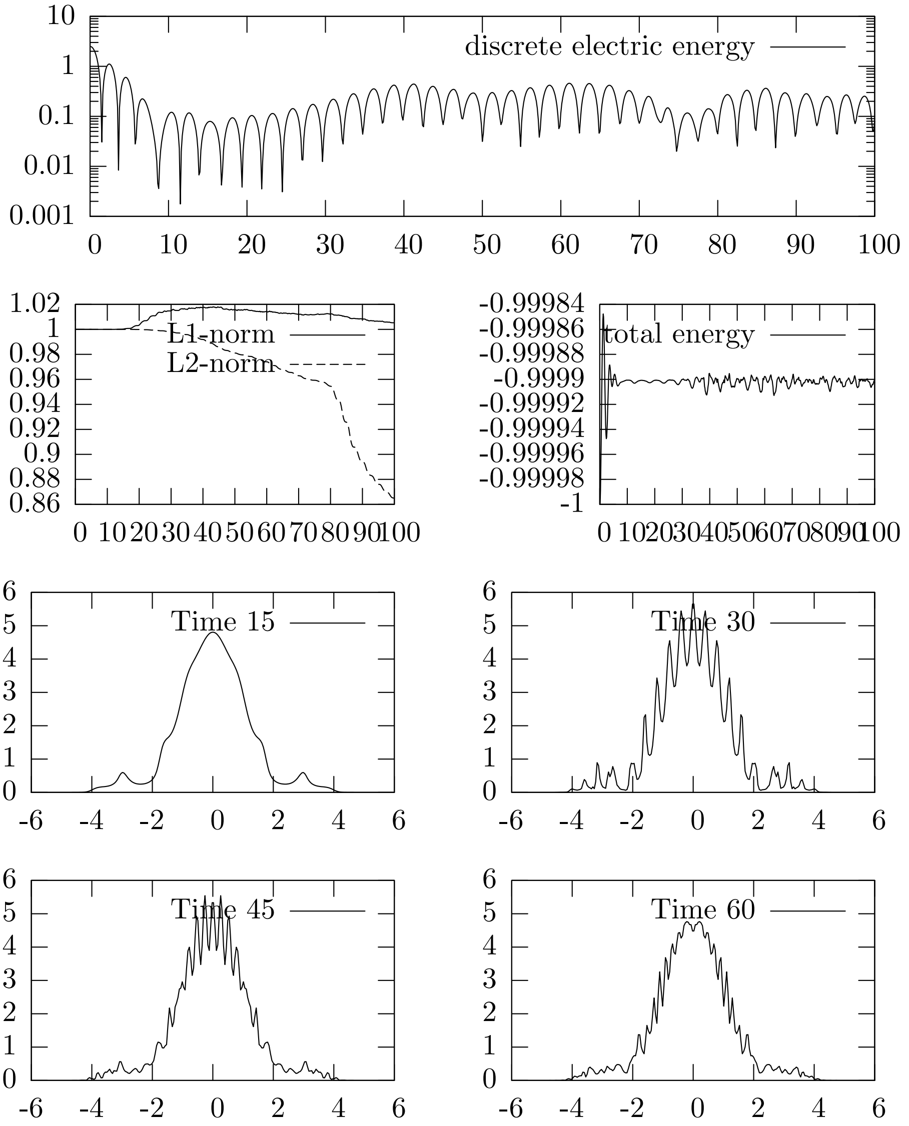


Figure 5.13: Non-linear Landau damping. $N_x = 256$, $N_v = 256$, $\Delta t = 0.125$,Lagrange-19 interpolation in FBM. Upside: evolution of discrete electric energy, evolution of L^1 and L^2 norms, evolution of the total energy. Downside: evolution of v-integrated distribution function.

`nlld_fbm_lagr19`

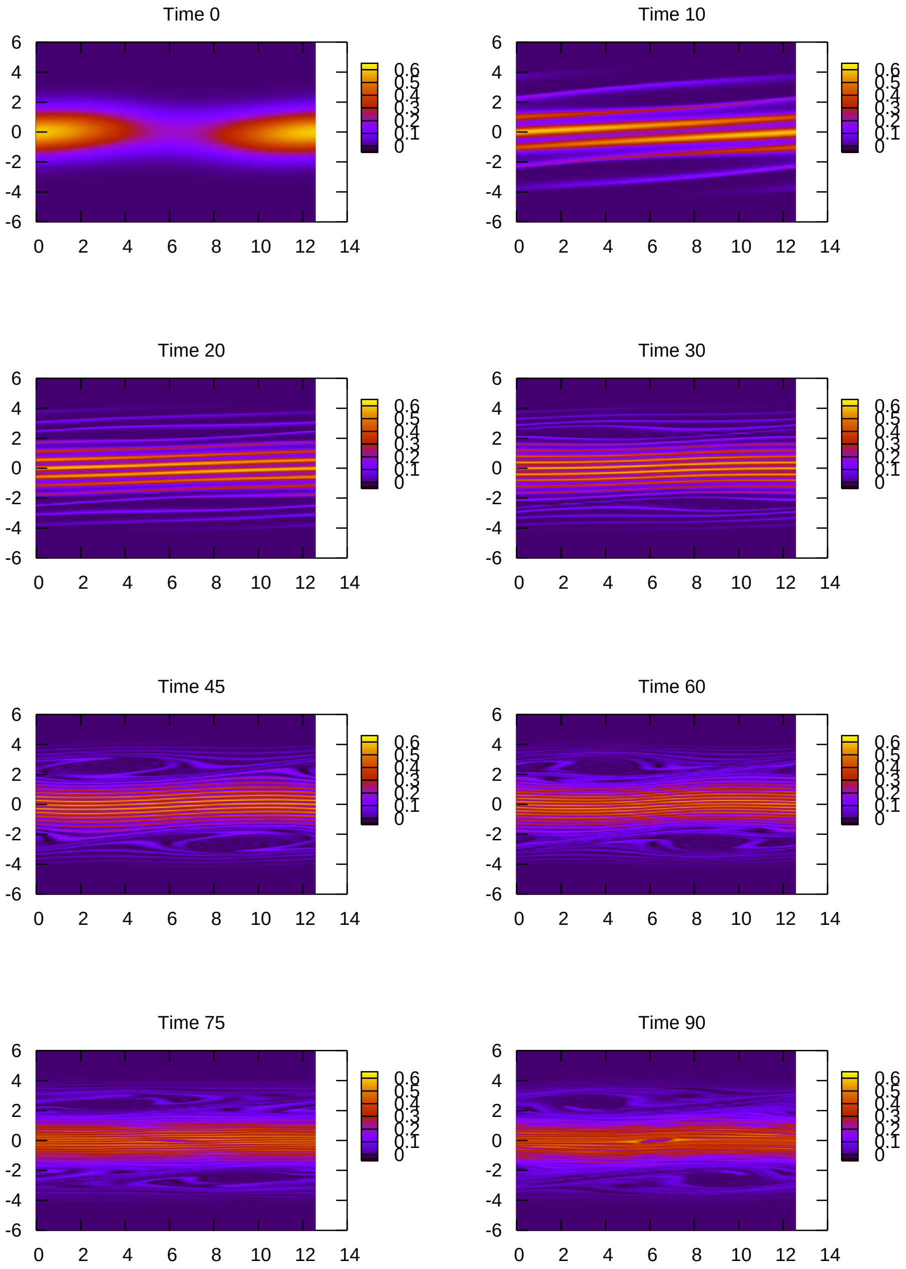


Figure 5.14: Evolution of level curves in non-linear Landau damping, for 256×256 points, $\Delta t = 0.125$, with FBM-Lagrange-19 method. Levels are: 0.05, 0.3 and 0.6.

fbm_lagr19_lc_1

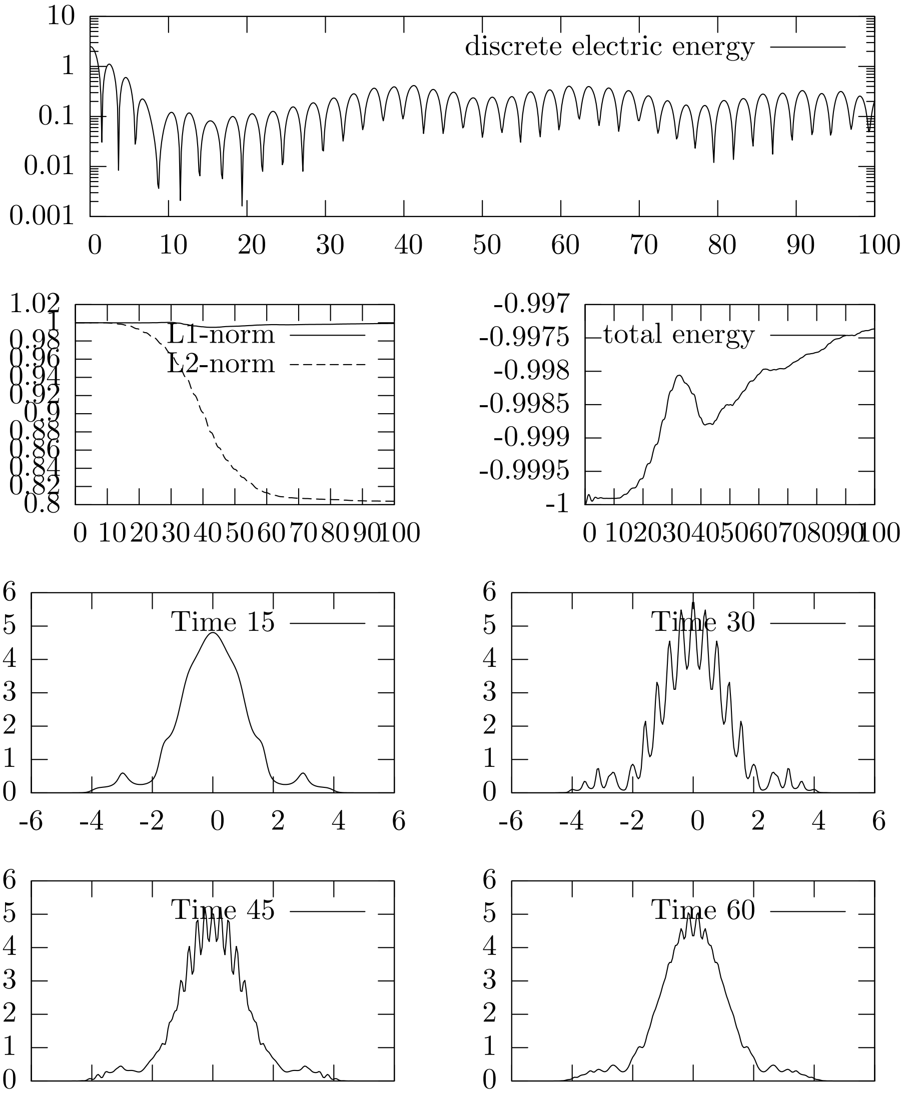


Figure 5.15: Non linear Landau damping. $N_x = 256$, $N_v = 256$, $\Delta t = 0.125$, WENO-6,4 interpolation in SL. Upside: evolution of discrete electric energy, evolution of L^1 and L^2 norms, evolution of the total energy. Downside: evolution of v-integrated distribution function.

`nlld_sl_6_4`

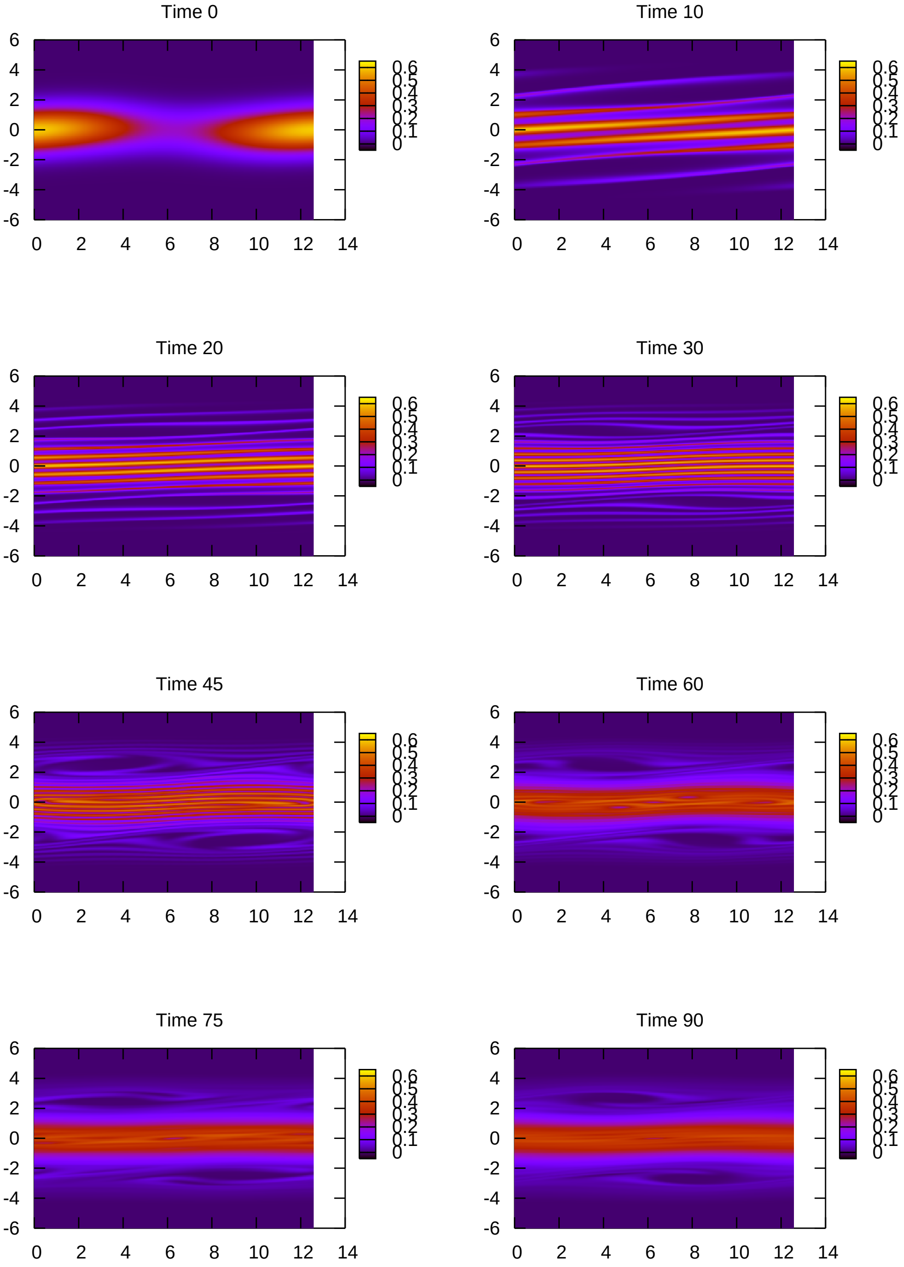


Figure 5.16: Evolution of level curves in non-linear Landau damping, for 256×256 points, $\Delta t = 0.125$, with SL-WENO-6,4 method. Levels are: 0.05, 0.3 and 0.6.

sl64_lc_1

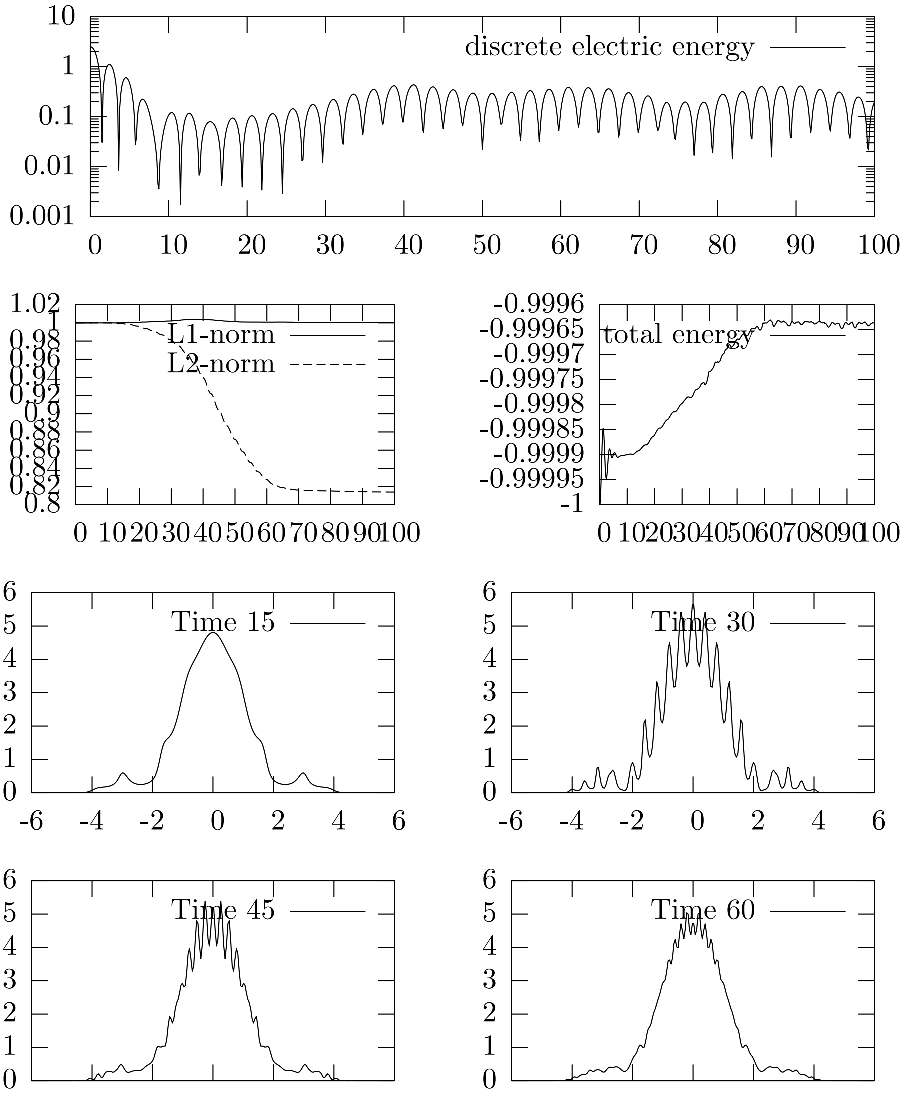


Figure 5.17: Non lineal Landau damping. $N_x = 256$, $N_v = 256$, $\Delta t = 0.125$, WENO-6,4 interpolation in FBM. Upside: evolution of discrete electric energy, evolution of L^1 and L^2 norms, evolution of the total energy. Downside: evolution of v-integrated distribution function.

nlld_fbm_6_4

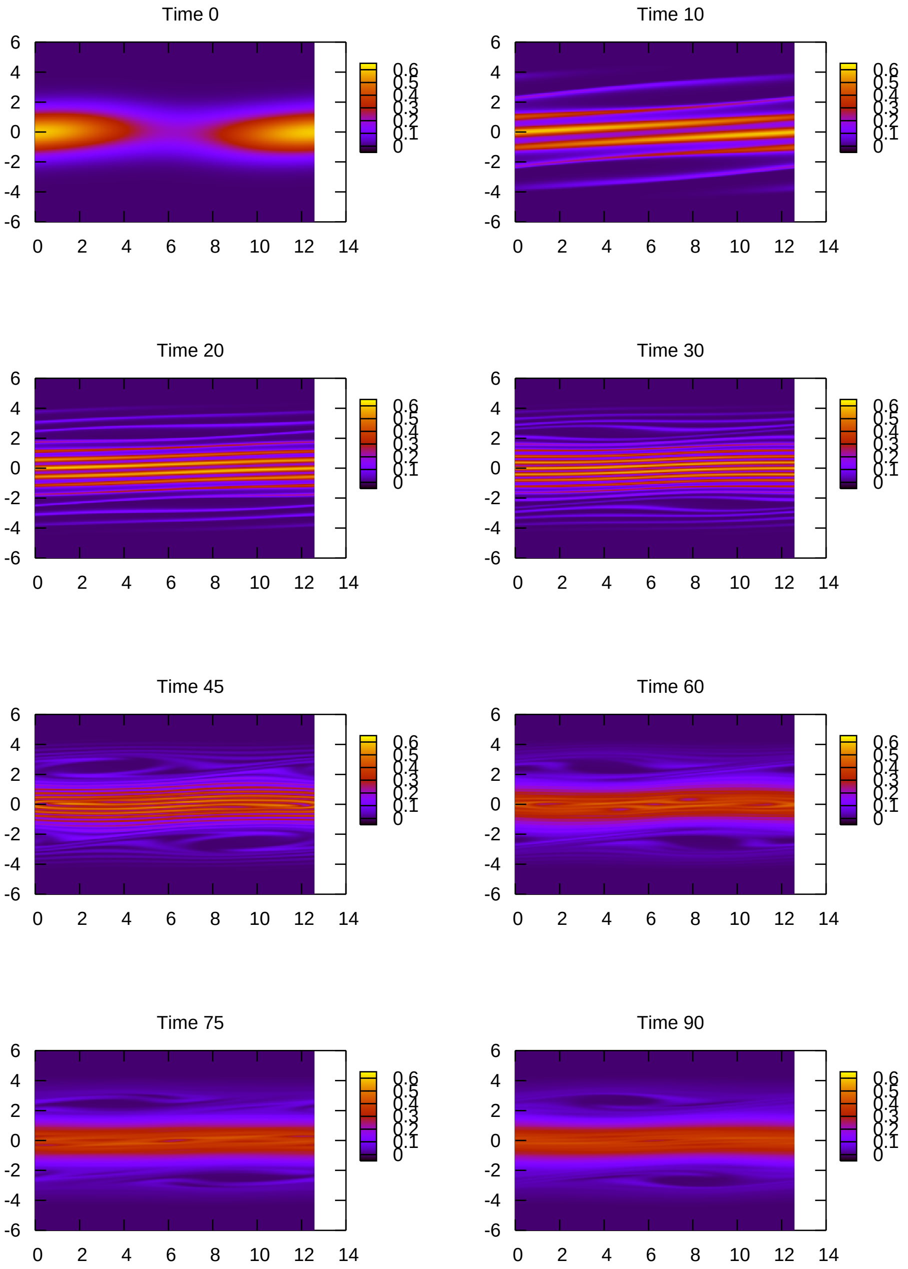


Figure 5.18: Evolution of level curves in non-linear Landau damping, for 256 × 256 points, $\Delta t = 0.125$, with FBM-WENO-6,4 method. Levels are: 0.05, 0.3 and 0.6.

fbm64_lc_1

respect to Lagrange-19.

used method	L^∞ – difference	L^1 – difference
SL-WENO-6,4	0.441668	0.953818
SL-WENO-7,4	0.619164	1.366983
FBM-WENO-6,4	0.481264	0.839781
FBM-WENO-7,4	0.520260	0.837280
FBM-PFC-3	0.493379	1.203918

In this case, we repeat that Lagrange-19 results are completely wrong while methods based on PWENO interpolation do control the total variation of the function. In fact, it is worthy to remark that interpolation methods like WENO-7,4, which does not control oscillation, is worse than lower order methods like WENO-6,4 or PFC-3.

5.3 Vlasov-Maxwell

Let us solve the simplified Vlasov-Maxwell system modelling the laser-plasma interaction. The theoretical derivation of the quasi-relativistic Vlasov-Maxwell model is explained in chapter 1, and the numerical schemes which we shall use are explained in chapter 4. We still need to initialize the variables A, $\mathcal{E}$ and B, to be able to start the numerical method, and, of course, we have to give the initial distribution $f_0(x,p)$.

Initialization of A, $\mathcal{E}$ and B

We need to initialize $A(t,x)$, $\mathcal{E}(t,x)$ and $B(t,x)$. In fact, we just need to give an initialization for $A(t,x)$, because of relations (4.1) and (4.2). In the tests it will be set

$$A(t,x) = A_0 \cos(\omega t - kx).$$

Then, $\mathcal{E}(t,x) = -A_0\omega \sin(\omega t - kx)$ and $B(t,x) = -A_0 k \sin(\omega t - kx)$. The proper initializations shall be

$$\begin{cases} \mathcal{E}_i^0 = A_0 \omega \sin(kx_i) \\ A_i^{-\frac{1}{2}} = A_0 \cos\left(kx_i + \omega\frac{\Delta t}{2}\right) \\ B_{i+\frac{1}{2}}^{-\frac{1}{2}} = A_0 k \sin\left[\omega\frac{\Delta t}{2} + k\left(x_i + \frac{\Delta x}{2}\right)\right]. \end{cases}$$

Initial distribution

We shall choose as initial data a two-temperature initial distribution. We want to define a two-temperatures maxwellian distribution: in our scale we shall center it around velocities (thermic and hot) $v_{th} = \sqrt{\frac{15}{511}}$ and $v_{hot} =$

$\sqrt{\frac{100}{511}}$: the first part will be a classical maxwellian and the second part a relativistic maxwellian, to take into account the difference between velocity and momentum p.

$$G(p) = \alpha \frac{1}{\sqrt{2\pi} v_{th}} e^{-\frac{p^2}{2v_{th}^2}} + \frac{1-\alpha}{z} e^{-\frac{\sqrt{1+p^2}-1}{k_B T_{hot}}}$$

where z is just a normalizing factor, α gives the ratio of thermic electrons (we shall set $\alpha = 95\%$), k_B is the Boltzmann's constant in our scale and T_{hot} is the high temperature for carriers, corresponding to v_{hot}: it is $T_{hot} = 100(eV)$. Initial function f_0 will be set

$$f_0(x,p) = (1 + \epsilon \cos(k_e x))\, G\left(p - \epsilon v_{th} \cos(k_e x)\right),$$

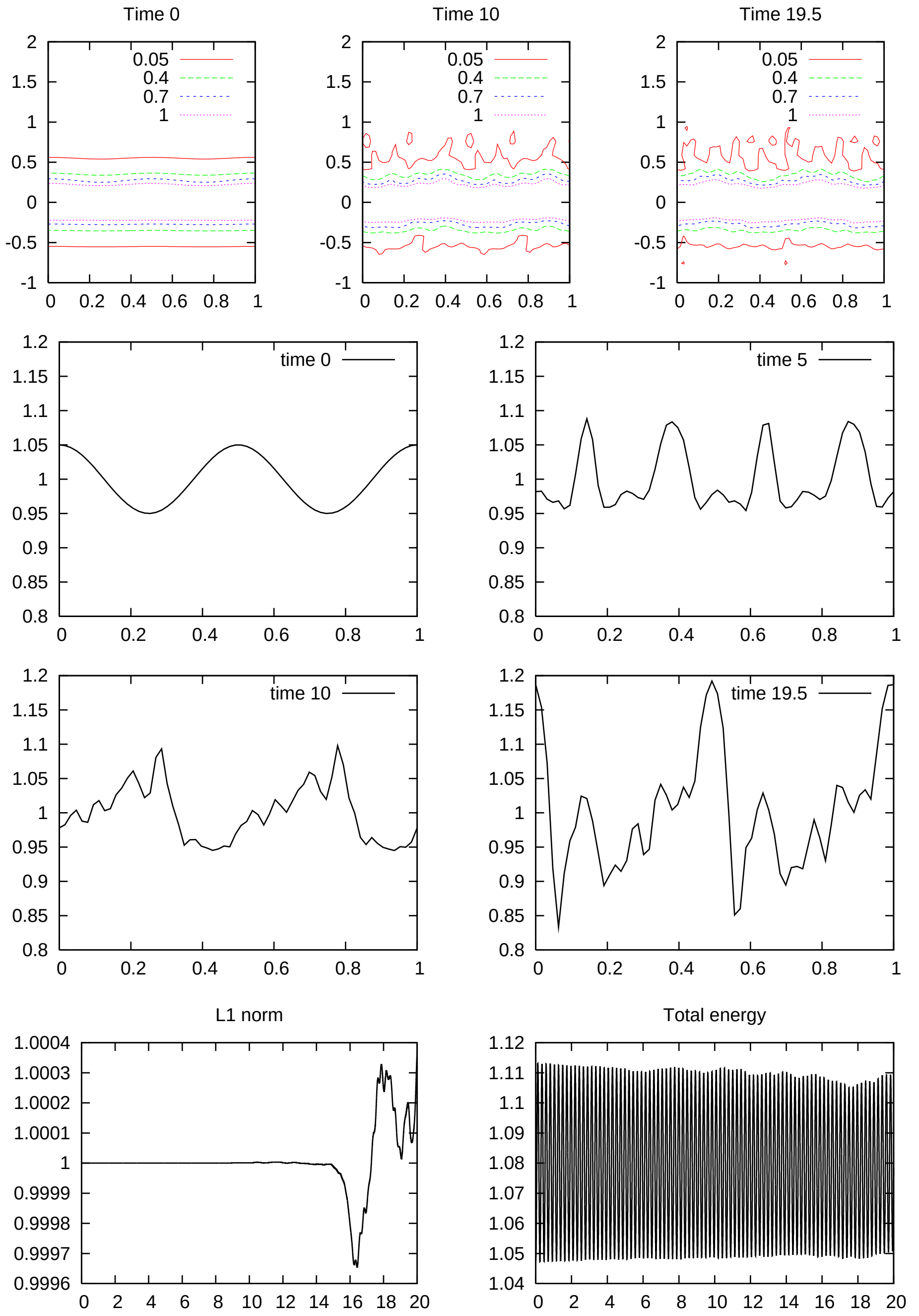
where $\epsilon = 0.05$.

Results

The computations have been performed for 64 × 64 points, and a time step of $\Delta t = 0.001$. Simulating up to time $t = 20$ this way has taken almost one day.

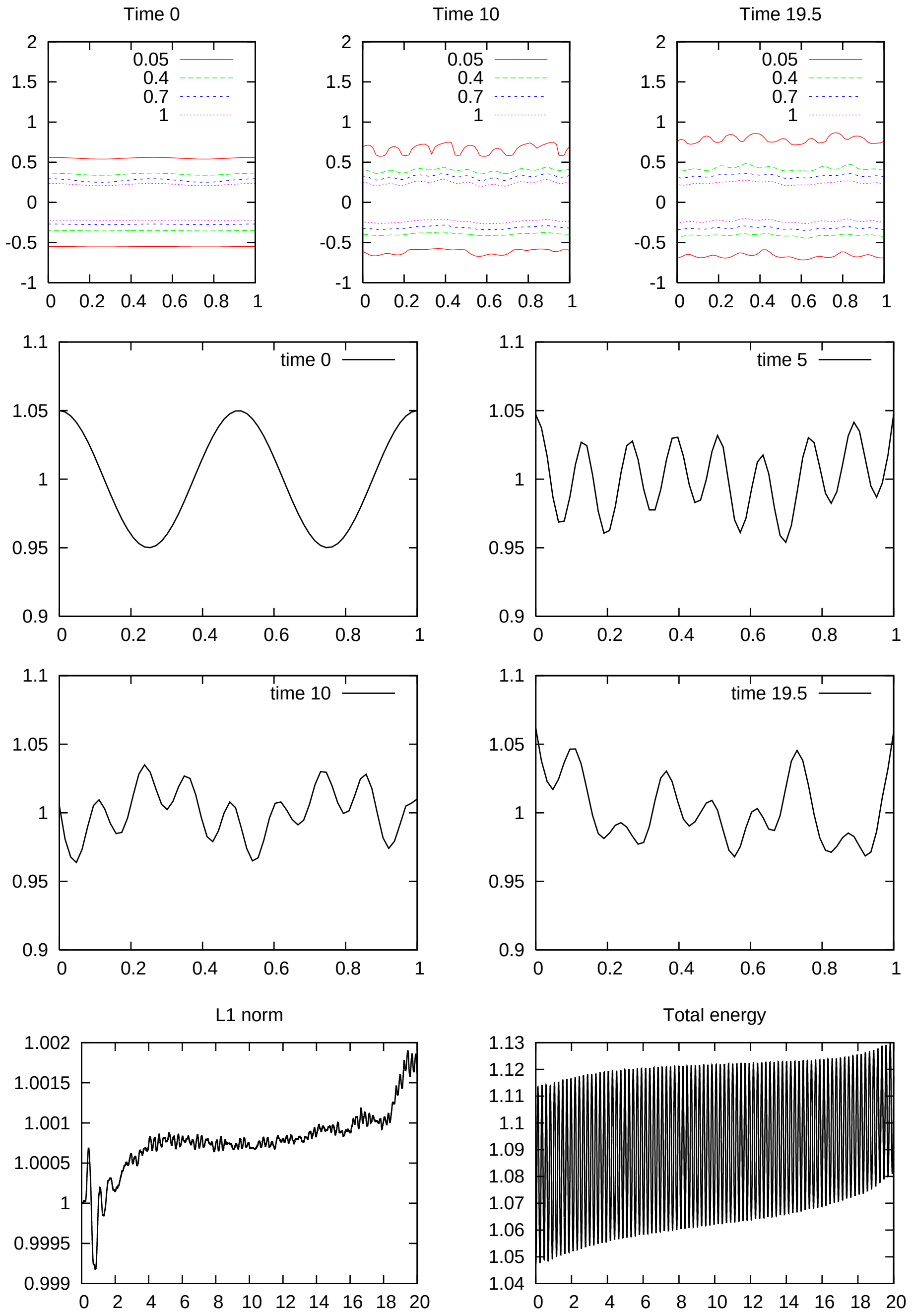
As you can see from the evolution of level curves and the density in Figure vm_sl_lagr19 5.3, Lagrange-19 method produces spurious oscillations, even if the mass is quite well conserved (varies about 0.4%).

As for WENO-6,4 the results given by Flux Balance method have probably to be considered as more reliable: the mass is conserved, the total energy has an oscillation about 6.5%, and, most of all, no spurious oscillations seem to appear during the simulation.



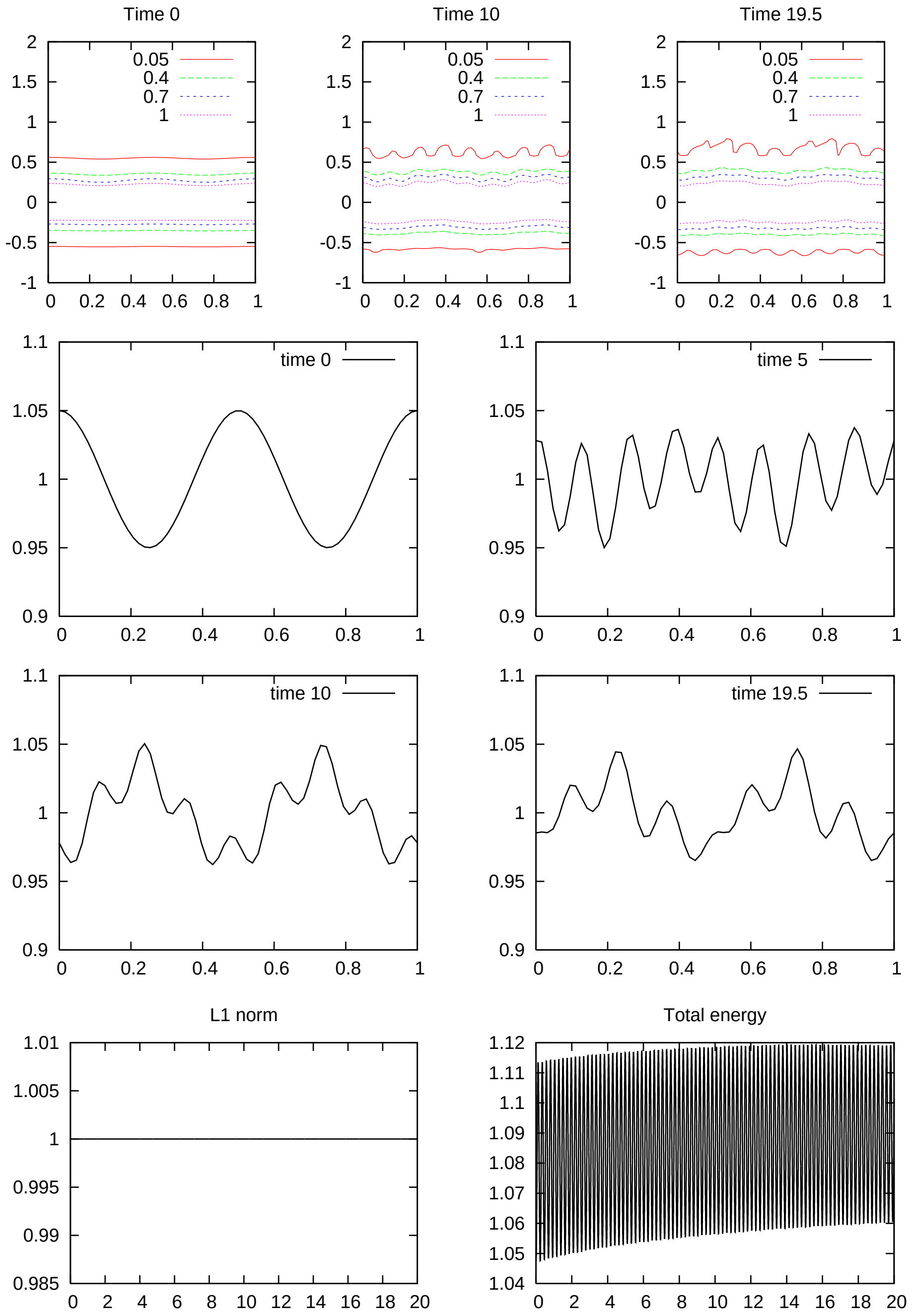
`vm_sl_lagr19`

Figure 5.19: Vlasov-Maxwell simulation performed with 64 × 64 points, $\Delta t = 0.001$, Semi-Lagrangian method, Lagrange-19. Upside: The evolution of level curves. Levels are 0.05, 0.4, 0.7, 1. Middle: The evolution of the macroscopic density. downside: The evolution of the L^1-norm and of the total energy.



vm_sl_64

Figure 5.20: Vlasov-Maxwell simulation performed with 64 × 64 points, $\Delta t = 0.001$, Semi-Lagrangian method, WENO-6,4. Upside: The evolution of level curves. Levels are 0.05, 0.4, 0.7, 1. Middle: The evolution of the macroscopic density. downside: The evolution of the L^1-norm and of the total energy.



vm_fbm_64

Figure 5.21: Vlasov-Maxwell simulation performed with 64 × 64 points, $\Delta t = 0.001$, FBM, WENO-6,4. Upside: The evolution of level curves. Levels are 0.05, 0.4, 0.7, 1. Middle: The evolution of the macroscopic density. downside: The evolution of the L^1-norm and of the total energy.

Appendix A

Physical constants

Here is the table with all the physical constants:

$e = 1.602 \times 10^{-19}\ C$	elementary electric charge
$m_e = 9.10938188 \times 10^{-31}\ Kg$	electron mass
$m^* = 1.7307825572 \times 10^{-31}\ Kg = 0.19 \times m_e$	electron effective mass
$\epsilon = 1.03594 \times 10^{-10}\ \frac{C^2}{Jm}$	Silicon dielectric permittivity
$h = 6.626068 \times 10^{-34}\ \frac{m^2 Kg}{s}$	Planck's constant
$c = 299792458\ \frac{m}{s}$	light speed

Bibliography

arsenev [1] A. A. Arsen'ev, *Global Existence of a Weak Solution of Vlasov's System of Equations*, Zh. vychisl. Mat. i Mat. Fiz. **15**, pp. 136-147 (1975)

bouchut1991 [2] F. Bouchut, *Global Weak Solutions of the Vlasov-Poisson System for Small Electron Mass*, Comm. Partial Diff. Equations **16**, pp. 1337-1365 (1991)

bouchut [3] F. Bouchut, F. Golse, M. Pulvirenti, *Kinetic Equations and Asymptotic Theory*, Series in Applied Mathematics (Paris) No. 4(Gauthiers-Villars, 2000)

mjcaceres [4] M. J. Cáceres, J. A. Carrillo, J. Dolbeault, *Nonlinear Stability in L^p for a Confined System of Charged Particles*, SIAM J. Math. Anal. **34**, no. 2, pp. 478-494 (2002)

llocaceresgoudon [5] J. A. Carrillo, M. J. Cáceres, T. Goudon, *Equilibration Rate for the Linear Inhomogeneous Relaxation-Time Boltzmann Equation for Charged Particles*, Communications in Partial Differential Equations **28** nos. 5 & 6, pp. 969-989 (2003)

laserplasma [6] J. A. Carrillo, S. Labrunie, *Global Solutions for the One-Dimensional Vlasov-Maxwell System for Laser-Plasma Interaction*, HYKE preprint no. 2004-171, to appear in Math. Models Meth. Appl. Sci.

timesplitting [7] Cheng, G. Knorr, *The Integration of the Vlasov Equation in Configuration Space*, Journal of Computational Physics **22** , pp. 330-348 (1976)

degondbardos [8] P. Degond and C. Bardos, *Global Existence for Vlasov-Poisson Equation in* 3 *Space Variables with Small Initial Data.*, Ann. Inst. Henri Poincaré, Anal. Non Linéaire **2**, pp. 101-118 (1985)

dipernamaxwell [9] R. J. DiPerna, P. L. Lions, *Global Existence of Weak Solutions for the Vlasov-Maxwell System*, Comm. Pure Applied Math **42**, pp. 729-757 (1989) diperna [10] R. J. DiPerna, P. L. Lions, *Solutions Globales d'Équations du Type Vlasov-Poisson*, C. R. Acad. Sci. Paris Sér. I Math. **307**, pp. 306-329 (1988)

esplittingvlasovboltzmann [11] F. Filbet, G. Russo, *Accurate Numerical Methods for the Boltzmann Equation*, Model. Simul. Sci. Eng. Technol., pp. 117-145, Birkhäuser Boston (2004)

oscillations [12] F. Filbet, C. Mouhot, L. Pareschi, *Solving the Boltzmann Equation in* $N \log N$, submitted to SIAM J. Scientific Computing

filbet [13] F. Filbet, *Thèse Doctorale: Constribution à l'Analyse et a la Simulation Numérique de l'Equation de Vlasov*, Université Henri Poincaré (Nancy,02/07/2001)

glassey3 [14] R. Glassey, *The Cauchy Problem in Kinetic Theory*, SIAM, Philadelphia (1996)

glassey2 [15] R. Glassey, W. Strauss, *Large Velocities in the Relativistic Vlasov-Maxwell Equations*, J. Fac. Sci. Univ. Tokyo Sect. IA Math. **36**, pp. 615-627 (1989)

glassey1 [16] R. Glassey, W. Strauss, *Singularity Formation in a Collisionless Plasma could only occur at High Velocities*, Arch. Ration. Mech. Anal. **92**, pp. 59-90 (1986)

glassey4 [17] R. Glassey, J. Schaeffer, *The Two and One-Half Dimensional Relativistic Vlasov-Maxwell System*, Comm. Math. Phys. **185**, pp. 257-284 (1997)

herau [18] F. Hérau, *Hypercoercitivity and Exponential Time Decay for the Linear Inhomogeneous Relaxation Boltzmann Equation*, preprint, Université de Reims (2005)

horst [19] E. Horst, *On the Classical Solutions of the Initial Value Problem for the Unmodified Non-Linear Vlasov Equation*, Math. Methods Appl. Sci. **3**, 229-248 (1981)

33 [20] E. Hörst, *Global Strong Solutions of Vlasov Equations, Necessary and Sufficient Conditions for their Existence*, Partial Differential Equations **19**, Banach Center Publ., Warsaw, pp. 143-153 (1987)

34 [21] E. Hörst, R. Hunze, *Weak Solutions of the Initial Value Problem for the Unmodified Nonlinear Vlasov Equation*, Math. Methods Appl. Sci. **6**, pp. 262-279 (1984)

iordanskii [22] S. V. Iordanskii, *The Cauchy Problem for the Kinetic Equation of Plasma*, Trudy Mat. Inst. Steklov. **60**, pp. 181-194 (1961) smooth [23] G. S. Jiang and C. W. Shu, *Efficient Implementation of Weighted ENO Schemes*, Journal of Computational Physics **126**, no.1, pp. 202-228 (1996)

cheney [24] D. Kincaid, W. Cheney, *Análisis Numérico*, Addison-Wesley Iberoamericana (1994)

landaudamping [25] S. Labrunie, J. A. Carrillo, P. Bertrand, *Numerical study on hydrodynamic and quasi-neutral approximations for collisionless two-species plasmas*, Journal of Computational Physics **200**, pp. 267-298 (2004)

lambert [26] J. D. Lambert, *Computational Methods in Ordinary Differential Equations. Introductory Mathematics for Scientists and Engineers*, Jonh Wiley and Sons, London-New York-Sydney (1973)

36 [27] P. L. Lions, B. Perthame, *Propagation of Moments and Regularity for the Vlasov-Poisson System*, Invent. Math. **105**, no. 2, pp. 415-430 (1991)

magnus [28] A. Magnus, *Analyse Numérique 2*, Uninersité Catholique de Louvain (2000)

markowich [29] P. A. Markowich, C. A. Ringhofer, C. Schmeiser, *Semiconductor Equations*, Springer-Verlag Wien (1990)

calculonumerico1 [30] *Cálculo Numérico I*, UNED (1989)

41 [31] K. Pfaffelmoser, *Global Classical Solutions of the Vlasov-Poisson System in three Dimensions for General Initial Data*, J. Differential Equations **95**, pp. 281-303 (1992)

shu-osher [32] C. W. Shu and S. Osher, *Efficient Implementation of Essentially Non-Oscillatory Shock capturing Schemes*, Journal of Computational Physics **77**, no. 2, pp. 439-471 (1988)

shu-osher-2 [33] C. W. Shu and S. Osher, *Efficient Implementation of Essentially Non-Oscillatory Shock capturing Schemes, II*, Journal of Computational Physics **83**, no. 1, pp. 32-78 (1989)

schaeffer [34] J. Schaeffer, *Global Existence of Smooth Solutions to the Vlasov-Poisson System in three Dimensions*, Comm. Partial Differential Equations **16**, pp. 1313-1335 (1991)

chiwangshu [35] C.W. Shu, *Essentially Non-Oscillatory and Weighted Essentially Non-Oscillatory Schemes for Hyperbolic Conservation Laws*, Lecture Notes in Mathematics **1697**, pp. 325-432 (1997)

stoer [36] J. Stoer, R. Bulirsch, *Introduction to Numerical Analysis*, Springer-Verlag, New York (1993) strang [37] G. Strang, *On the Construction and Comparison of Difference Schemes*, SIAM J. Numer. Anal. **5**, pp. 506-517 (1968)

ukaiokabe [38] S. Ukai and T. Okabe, *On Classical Solutions in the Large in Time of Two-Dimensional Vlasov's Equation*, J. Osaka J. Math **15**, pp. 245-261 (1978)