Some applications of kinetic equations

Francesco Vecil

Universitat de València

Universitat de València, 16/02/11

Outline

Outline

General idea

How do kinetic equations look like?

Kinetic equations are hyperbolic partial differential equations: the unknown function f is a probabilistic description of some magnitude which depends on the variables of the phase space, (x, v), (x, p) or (x, k): the choice of the problem may make more suitable the use of the velocity v instead of the impulsion p or the wave vector k.

Why do kinetic equations arise in physics?

Kinetic equations arise when the simulations of the individual-based models become unaffordable due to the huge amount of agents, like the electrons in an electronic device, the ions in a plasma or the fish in the sea.

Example.

An example is given by the following PDE

$$\begin{cases} \frac{d}{dt}x = v \\ \frac{d}{dt}v = F \end{cases} \leftrightarrow \frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} + F\frac{\partial f}{\partial v} = 0$$

which describes how a set of "particles" evolves under the free motion and the presence of a force field $F(t, x)$.

General idea

How do kinetic equations look like?

Kinetic equations are hyperbolic partial differential equations: the unknown function f is a probabilistic description of some magnitude which depends on the variables of the phase space, (x, v), (x, p) or (x, k): the choice of the problem may make more suitable the use of the velocity v instead of the impulsion p or the wave vector k.

Why do kinetic equations arise in physics?

Kinetic equations arise when the simulations of the individual-based models become unaffordable due to the huge amount of agents, like the electrons in an electronic device, the ions in a plasma or the fish in the sea.

Example.

An example is given by the following PDE

$$\begin{cases} \frac{d}{dt}x = v \\ \frac{d}{dt}v = F \end{cases} \leftrightarrow \frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} + F\frac{\partial f}{\partial v} = 0$$

which describes how a set of "particles" evolves under the free motion and the presence of a force field $F(t, x)$.

General idea

How do kinetic equations look like?

Kinetic equations are hyperbolic partial differential equations: the unknown function f is a probabilistic description of some magnitude which depends on the variables of the phase space, (x, v), (x, p) or (x, k): the choice of the problem may make more suitable the use of the velocity v instead of the impulsion p or the wave vector k.

Why do kinetic equations arise in physics?

Kinetic equations arise when the simulations of the individual-based models become unaffordable due to the huge amount of agents, like the electrons in an electronic device, the ions in a plasma or the fish in the sea.

Example.

An example is given by the following PDE

$$\begin{cases} \frac{d}{dt}x = v \\ \frac{d}{dt}v = F \end{cases} \leftrightarrow \frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} + F\frac{\partial f}{\partial v} = 0$$

which describes how a set of “particles” evolves under the free motion and the presence of a force field $F(t, x)$.

Outline

Aim

Introduction

Electronic devices are physical solid state devices which exploit the electronic properties of semiconductor materials (silicon) by manipulating their conductivity via the *doping*. The dynamics in semiconductor devices are due to the displacement of electric charges, which is described by kinetic equations.

The DG MOSFET

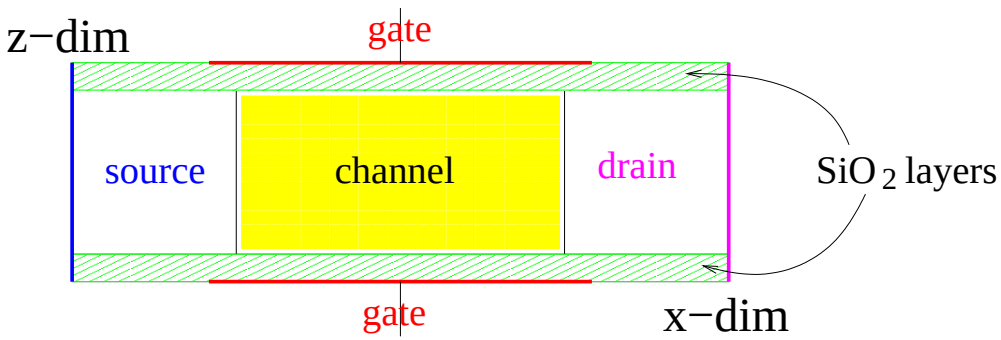


Figure: A **D**ouble-**G**ate **M**etal **O**xide **S**emiconductor **F**ield **E**ffect **T**ransistor.

Aim

Introduction

Electronic devices are physical solid state devices which exploit the electronic properties of semiconductor materials (silicon) by manipulating their conductivity via the *doping*. The dynamics in semiconductor devices are due to the displacement of electric charges, which is described by kinetic equations.

The DG MOSFET

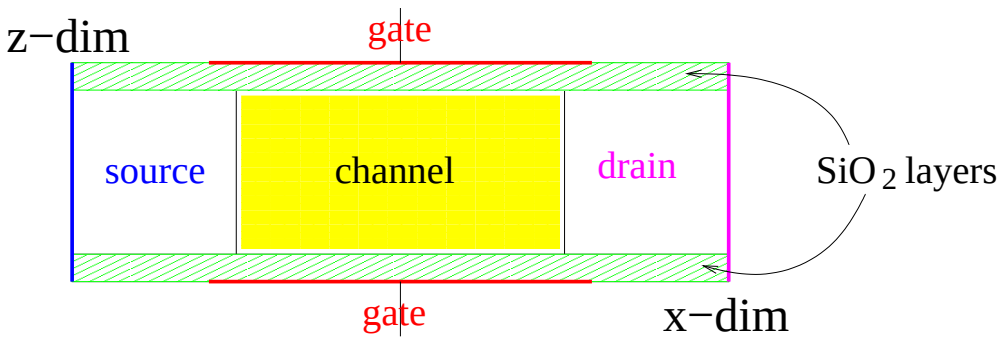


Figure: A **D**ouble-**G**ate **M**etal **O**xide **S**emiconductor **F**ield **E**ffect **T**ransistor.

Aim

About the scaling

In 1971, the Intel 4004 processor had 1000 transistors, whose channel length was 10000 nm. In 1974, the Intel 8008 processor had 6-7 thousand. In 2003 the Intel Pentium IV had 50 million. Nowadays processors may have 400 million transistors, whose channel is 28 nm long.

Why is it important?

Smaller MOSFETs allow for the construction of smaller devices with better performances; moreover, they allow silicon and energy saving, due to the lower source-drain potential drop needed to switch on or off the transistor.

Equations

Test case: the 1D Vlasov-Poisson system

Carriers move under the free motion and are driven by a self-consistent (i.e. created by the particles themselves) electric field:

$$\frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} - \frac{\partial \Phi}{\partial x}\frac{\partial f}{\partial v} = 0$$
$$\frac{\partial^2 \Phi}{\partial x^2} = 1 - \int_{\mathbb{R}} f dv.$$

Equations

The Boltzmann equation

Scatterings are taken into account, as well as the band-structure of silicon:

$$\frac{\partial f}{\partial t} + \frac{1}{\hbar}\nabla_k \varepsilon \cdot \nabla_x f - \frac{q}{\hbar} E \cdot \nabla_k f = \mathcal{Q}[f]$$

$$\Delta \Phi = \frac{q}{\epsilon_0}\left[\rho[f] - N_D\right], \qquad E = -\nabla_x \Phi$$

$$f_0(x,k) = N_D(x) M(k)$$

$$\varepsilon(k) = \frac{\hbar^2 |k|^2}{2 m_*}$$

$$\mathcal{Q}[f] = \int_{\mathbb{R}^3} \left[S(k',k) f(t,x,k') - S(k,k') f(t,x,k)\right] dk'.$$

Equations

The sub-bands model

Quantum effetcs are taken into account due to the confinement: The Boltzmann Transport Equation (one for each band) reads

$$\frac{\partial f_{\nu,p}}{\partial t} + \frac{1}{\hbar}\nabla_k \epsilon^{kin}_{\nu,p} \cdot \nabla_k f_{\nu,p} - \frac{1}{\hbar}\nabla_x \epsilon^{pot}_{\nu,p} \cdot \nabla_k f_{\nu,p} = \mathcal{Q}_{\nu,p}[f]$$

$$-\frac{\hbar^2}{2}\frac{d}{dz}\left[\frac{1}{m_{\nu,z}}\frac{d\chi_{\nu,p}}{dz}\right] - q\left(V + V_c\right)\chi_{\nu,p} = \epsilon^{pot}_{\nu,p}\chi_{\nu,p}$$

$$\{\chi_{\nu,p}\}_{\nu,p} \subseteq H^1_o(0, l_z) \text{ orthonormal basis}$$

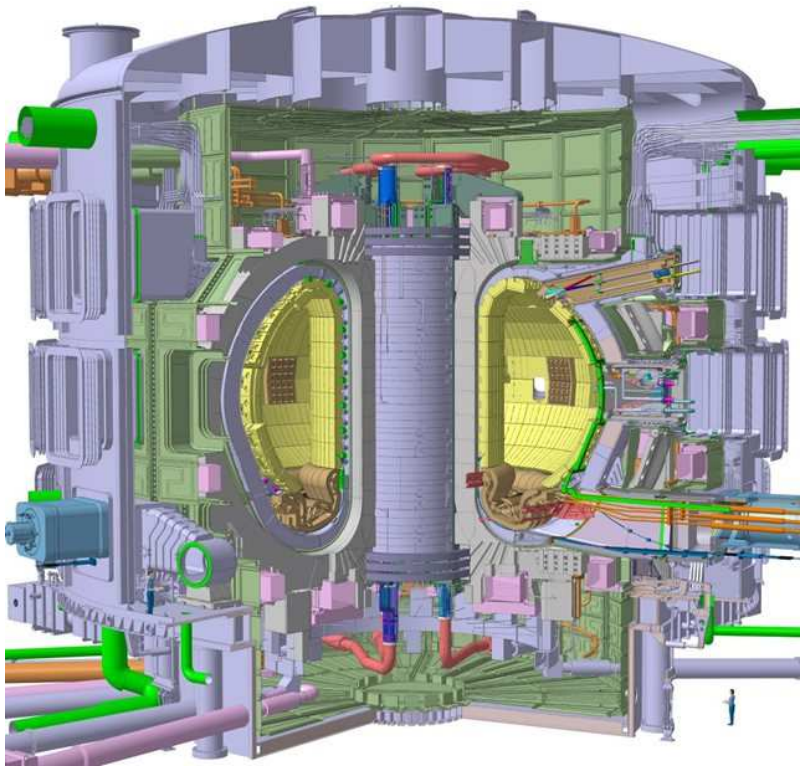
$$-\nabla_{x,z} \cdot [\varepsilon_R \nabla_{x,z} V] = -\frac{q}{\varepsilon_0}\left(N[V] - N_D\right)$$

$$N[V](x,z) = \sum_{\nu,p} \int_{\mathbb{R}^2} f(x,k)dk \left|\chi_{\nu,p}(x,z)\right|^2$$

Outline

Introduction

Plasmas are ionized gases: positive and negative charges dissociate. Plasma physics is of great interest in fusion energy research.



Laser-plasma interaction

The force field is given by the Lorentz force, i.e. by solving the Maxwell equations:

$$\frac{\partial f}{\partial t} + v(p) \cdot \nabla_x f - q\,(E + v \wedge B) \cdot \nabla_p f = 0$$

$$\mathrm{curl}(E) = -\frac{\partial B}{\partial t}$$

$$\mathrm{curl}(B) = \mu_0 \int_{\mathbb{R}} v(p) f dp + \mu_0 \varepsilon_0 \frac{\partial E}{\partial t}$$

$$\mathrm{div}(E) = \frac{\rho}{\varepsilon_0}$$

$$\mathrm{div}(B) = 0$$

$$B = \nabla \wedge A$$

$$E = -\frac{\partial A}{\partial t} - \nabla \Phi,$$

where v is the velocity given in terms of the impulsion p:

$$v(p) = \begin{cases} \dfrac{p}{m_e \sqrt{1 + \frac{|p|^2}{m_e^2 c^2}}} & \text{relativistic} \\ \dfrac{p}{m_e} & \text{non-relativistic.} \end{cases}$$

Outline

Introduction

In some contexts a large number of “agents” interacting through microscopic rules gives rise to macroscopically observable patterns without the presence of a leader. Examples: bird flocks, fish schools, stock exchanges, the evolution of languages in primitive societies.

Equations

Cucker-Smale model

The Cucker-Smale model is an alignment model: particles try to copy the velocity from the other ones. It reads

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f - \mathrm{div}_v \left[f \left(v *_v U_0^\phi *_x f \right) \right] = 0$$

$$U_0^\phi(x; v) = \frac{1}{(1 + |x|^2_{\mathbb{R}^d})^\gamma} \chi \left[\cos(x, v) \geq \cos(\phi) \right].$$

Attractive/repulsive models

Particles do not copy the velocity from the other ones, rather they try to stay neither too close not too far: this conditions aims to copy the animal behaviour. The model reads

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f + \mathrm{div}_v \left[(\alpha - \beta \|v\|^2) v f + (\nabla U * \rho) f \right] = 0.$$

Equations

Cucker-Smale model

The Cucker-Smale model is an alignment model: particles try to copy the velocity from the other ones. It reads

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f - \mathrm{div}_v \left[f \left(v *_v U_0^\phi *_x f \right) \right] = 0$$

$$U_0^\phi(x; v) = \frac{1}{(1 + |x|^2_{\mathbb{R}^d})^\gamma} \chi \left[\cos(x, v) \geq \cos(\phi) \right].$$

Attractive/repulsive models

Particles do not copy the velocity from the other ones, rather they try to stay neither too close not too far: this conditions aims to copy the animal behaviour. The model reads

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f + \mathrm{div}_v \left[(\alpha - \beta \|v\|^2) v f + (\nabla U * \rho) f \right] = 0.$$

Introduction

We shall propose three methods for the solution of transport problems.

Finite Differences methods

They are based on a Runge-Kutta discretization in time. The approximation of the PDE is realized via non-oscillatory techniques.

Semi-Lagrangian methods

They are based on following the characteristics backward. Need coupling to a reconstruction technique at the foot of characteristics. With respect to Finite Differences methods, they allow for larger time steps, but loose precision in the asymptotic behaviour.

Semi-Lagrangian Discontinuous Galerkin methods

They are based on a discontinuous representation of the distribution function, which can be well resolved by local refinement of the mesh, without constraining the time step.

Outline

Motivation

We need a **Pointwise** interpolation method which does not add spurious oscillations when high gradients appear, e.g. when a jump has to be transported.

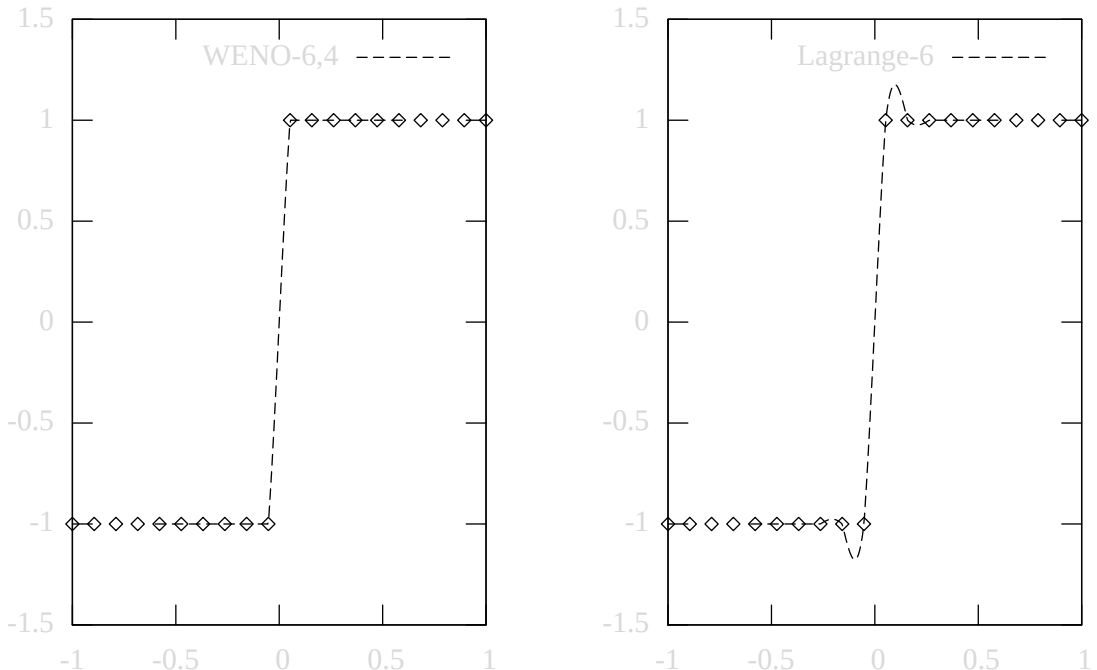


Figure: Left: PWENO interpolation. Right: Lagrange interpolation.

Motivation

We need a **Pointwise** interpolation method which does not add spurious oscillations when high gradients appear, e.g. when a jump has to be transported.

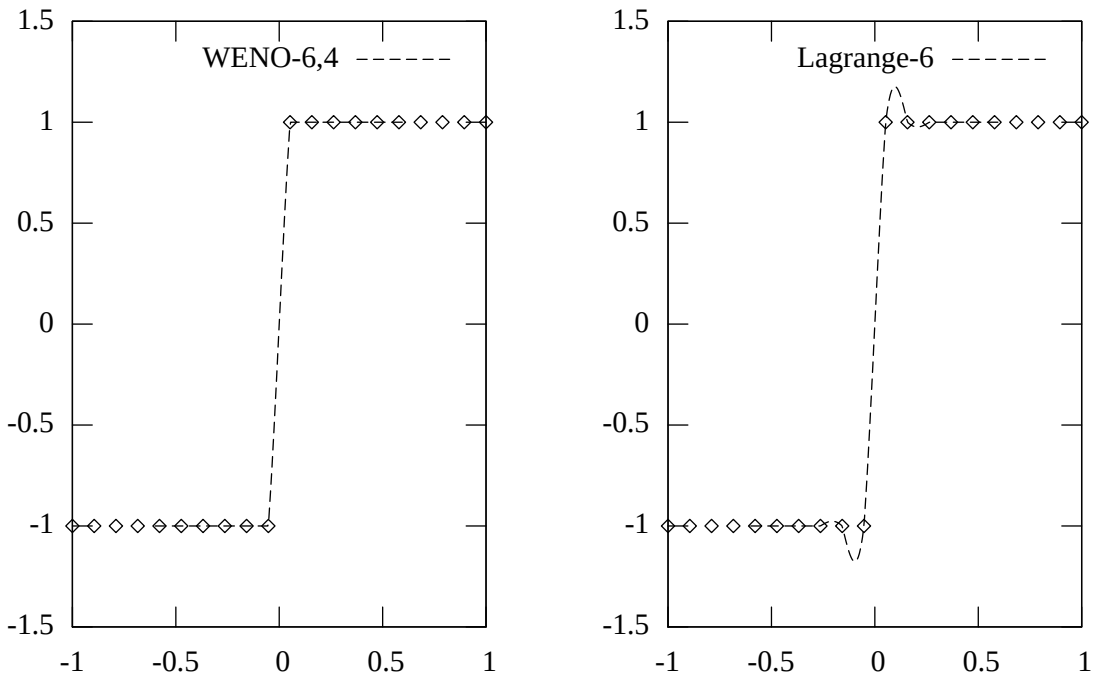
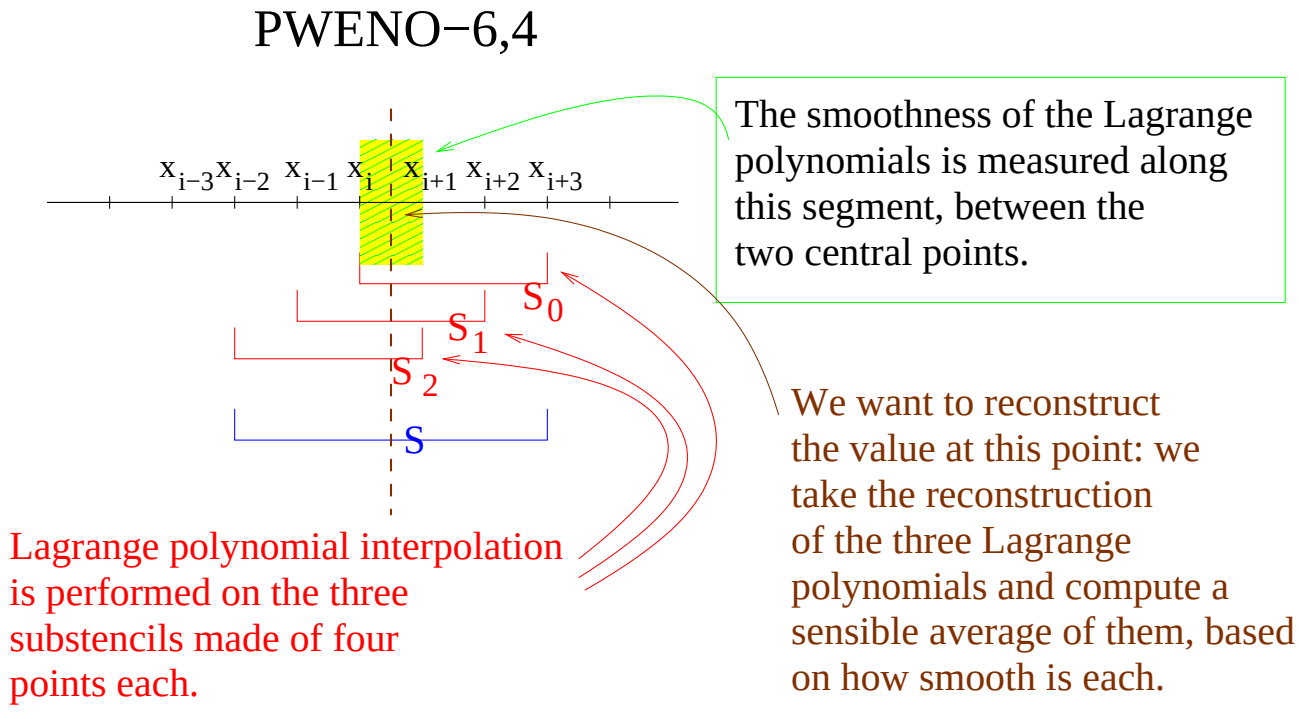


Figure: Left: PWENO interpolation. Right: Lagrange interpolation.

Non-oscillatory properties

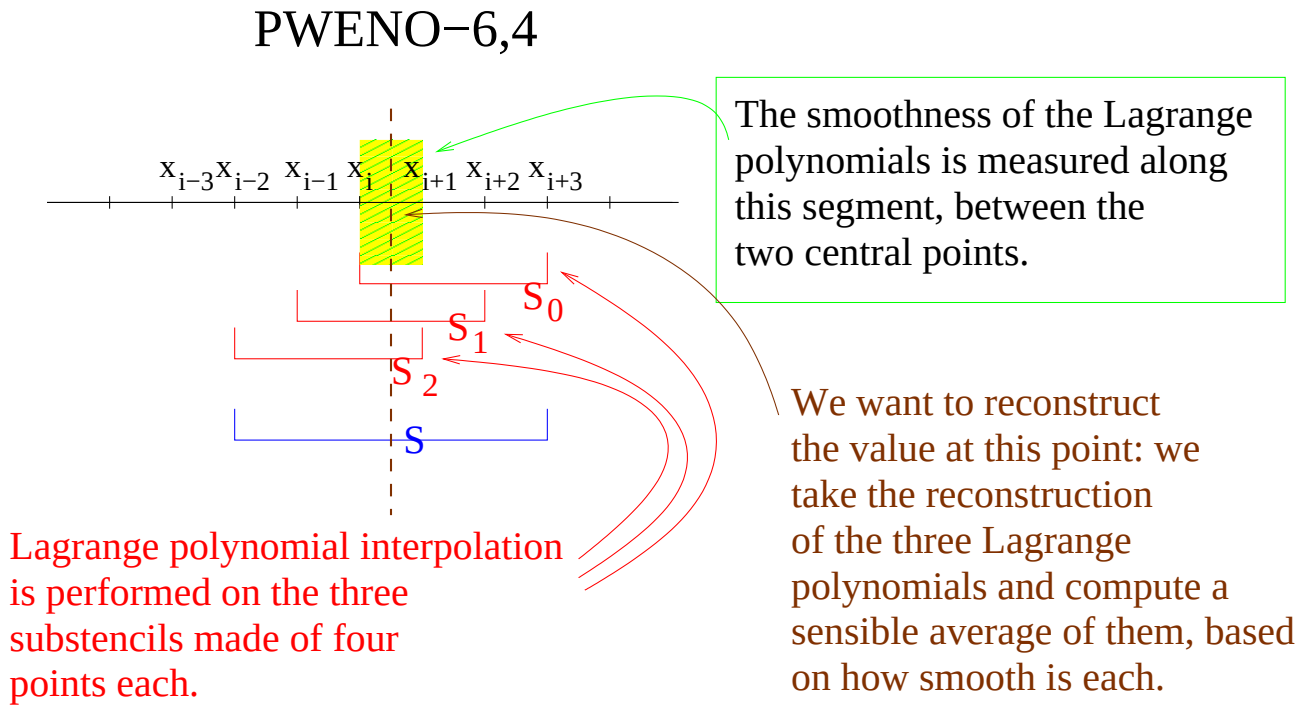
Essentially Non Oscillatory (ENO) methods are based on on a sensible average of Lagrange polynomial reconstructions.

We describe the case of PWENO-6,4: we take a stencil of six points and divide it into three substencils of four points:



Non-oscillatory properties

Essentially Non Oscillatory (ENO) methods are based on on a sensible average of Lagrange polynomial reconstructions.
We describe the case of PWENO-6,4: we take a stencil of six points and divide it into three substencils of four points:



The average

If we note $p_r(x)$ the Lagrange polynomials, PWENO reconstruction reads

$$p_{PWENO}(x) = \omega_0(x)p_0(x) + \omega_1(x)p_1(x) + \omega_2(x)p_2(x).$$

Convex combination.

The convex combination $\{\omega_r(x)\}_r$ must penalize the substencils $\mathcal{S}_r$ in which the $p_r(x)$ have high derivatives.

Smoothness indicators

In order to decide which substencils $\mathcal{S}_r$ are "regular" and which ones are not, we have to introduce the smoothness indicators: we use a weighted sum of the L^2-norms of the Lagrange polynomials $p_r(x)$ to measure their regularity close to the reconstruction point x. The following smoothness indicators have been proposed by Jiang and Shu:

$$\beta_r = \Delta x \left\| \frac{dp_r}{dx} \right\|_{L^2_{(x_i,x_{i+1})}} + \Delta x^3 \left\| \frac{d^2p_r}{dx^2} \right\|_{L^2_{(x_i,x_{i+1})}} + \Delta x^5 \left\| \frac{d^3p_r}{dx^3} \right\|_{L^2_{(x_i,x_{i+1})}}.$$

The average

If we note $p_r(x)$ the Lagrange polynomials, PWENO reconstruction reads

$$p_{PWENO}(x) = \omega_0(x)p_0(x) + \omega_1(x)p_1(x) + \omega_2(x)p_2(x).$$

Convex combination.

The convex combination $\{\omega_r(x)\}_r$ must penalize the substencils $\mathcal{S}_r$ in which the $p_r(x)$ have high derivatives.

Smoothness indicators

In order to decide which substencils $\mathcal{S}_r$ are "regular" and which ones are not, we have to introduce the smoothness indicators: we use a weighted sum of the L^2-norms of the Lagrange polynomials $p_r(x)$ to measure their regularity close to the reconstruction point x. The following smoothness indicators have been proposed by Jiang and Shu:

$$\beta_r = \Delta x \left\| \frac{dp_r}{dx} \right\|_{L^2_{(x_i,x_{i+1})}} + \Delta x^3 \left\| \frac{d^2p_r}{dx^2} \right\|_{L^2_{(x_i,x_{i+1})}} + \Delta x^5 \left\| \frac{d^3p_r}{dx^3} \right\|_{L^2_{(x_i,x_{i+1})}}.$$

The average

If we note $p_r(x)$ the Lagrange polynomials, PWENO reconstruction reads

$$p_{PWENO}(x) = \omega_0(x)p_0(x) + \omega_1(x)p_1(x) + \omega_2(x)p_2(x).$$

Convex combination.

The convex combination $\{\omega_r(x)\}_r$ must penalize the substencils $\mathcal{S}_r$ in which the $p_r(x)$ have high derivatives.

Smoothness indicators

In order to decide which substencils $\mathcal{S}_r$ are "regular" and which ones are not, we have to introduce the smoothness indicators: we use a weighted sum of the L^2-norms of the Lagrange polynomials $p_r(x)$ to measure their regularity close to the reconstruction point x. The following smoothness indicators have been proposed by Jiang and Shu:

$$\beta_r = \Delta x \left\| \frac{dp_r}{dx} \right\|_{L^2_{(x_i,x_{i+1})}} + \Delta x^3 \left\| \frac{d^2p_r}{dx^2} \right\|_{L^2_{(x_i,x_{i+1})}} + \Delta x^5 \left\| \frac{d^3p_r}{dx^3} \right\|_{L^2_{(x_i,x_{i+1})}}.$$

High order reconstruction

Admit for now that the convex combination is given by the normalization $\omega_r(x) = \frac{\tilde{\omega}_r(x)}{\sum_{s=0}^{2} \tilde{\omega}_s(x)}$ of the protoweights $\tilde{\omega}_r(x)$ defined this way:

$$\tilde{\omega}_r(x) = \frac{d_r(x)}{(\epsilon + \beta_r)^2}.$$

Regular reconstruction

Suppose that all the β_r are equal; then we have

$$\omega_r(x) = d_r(x).$$

The optimal order is achieved by Lagrange reconstruction $p_{Lagrange}(x)$ in the whole stencil $\mathcal{S}$, so if we define $d_r(x)$ to be the polynomials such that

$$p_{Lagrange}(x) = d_0(x)p_0(x) + d_1(x)p_1(x) + d_2(x)p_2(x),$$

then we have achieved the optimal order because $p_{PWENO}(x) = p_{Lagrange}(x)$.

High order reconstruction

Admit for now that the convex combination is given by the normalization $\omega_r(x) = \frac{\tilde{\omega}_r(x)}{\sum_{s=0}^{2} \tilde{\omega}_s(x)}$ of the protoweights $\tilde{\omega}_r(x)$ defined this way:

$$\tilde{\omega}_r(x) = \frac{d_r(x)}{(\epsilon + \beta_r)^2}.$$

Regular reconstruction

Suppose that all the β_r are equal; then we have

$$\omega_r(x) = d_r(x).$$

The optimal order is achieved by Lagrange reconstruction $p_{Lagrange}(x)$ in the whole stencil $\mathcal{S}$, so if we define $d_r(x)$ to be the polynomials such that

$$p_{Lagrange}(x) = d_0(x)p_0(x) + d_1(x)p_1(x) + d_2(x)p_2(x),$$

then we have achieved the optimal order because $p_{PWENO}(x) = p_{Lagrange}(x)$.

High order reconstruction

Admit for now that the convex combination is given by the normalization $\omega_r(x) = \frac{\tilde{\omega}_r(x)}{\sum_{s=0}^{2} \tilde{\omega}_s(x)}$ of the protoweights $\tilde{\omega}_r(x)$ defined this way:

$$\tilde{\omega}_r(x) = \frac{d_r(x)}{(\epsilon + \beta_r)^2}.$$

High gradients

Otherwise, suppose for instance that β_0 is high order than the other ones: in this case $\mathcal{S}_0$ is penalized and most of the reconstruction is carried by the other more "regular" substencils.

Outline

Motivation

In this work, splitting techniques are used at different levels, namely:

- to split the Boltzmann Transport Equation into the solution of the transport part and the collisional part for separate, i.e. the **Time Splitting**:

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

splits into

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f + F \cdot \nabla_v f = 0, \qquad \frac{\partial f}{\partial t} = \mathcal{Q}[f];$$

- to split the (x, v)-phase space (**Dimensional Splitting**):

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

splits into

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

Motivation

In this work, splitting techniques are used at different levels, namely:

- to split the Boltzmann Transport Equation into the solution of the transport part and the collisional part for separate, i.e. the **Time Splitting**:

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

splits into

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f + F \cdot \nabla_v f = 0, \qquad \frac{\partial f}{\partial t} = \mathcal{Q}[f];$$

- to split the (x, v)-phase space (**Dimensional Splitting**):

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

splits into

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

Motivation

In this work, splitting techniques are used at different levels, namely:

- to split the Boltzmann Transport Equation into the solution of the transport part and the collisional part for separate, i.e. the **Time Splitting**:

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

splits into

$$\frac{\partial f}{\partial t} + v \cdot \nabla_x f + F \cdot \nabla_v f = 0, \qquad \frac{\partial f}{\partial t} = \mathcal{Q}[f];$$

- to split the (x, v)-phase space (**Dimensional Splitting**):

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

splits into

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

General framework

The (formal) exact solution of the linear PDE

$$\frac{\partial f}{\partial t} = Lf, \qquad f(t=0) = f^0$$

is

$$f(t) = e^{Lt} f^0.$$

If we can write the linear operator L as the sum of two linear operators,

$$L = L_1 + L_2,$$

then we may approximate the exact solution by solving for separate

$$\frac{\partial f}{\partial t} = L_1 f \qquad \text{and} \qquad \frac{\partial f}{\partial t} = L_2 f.$$

Several schemes are proposed for reconstructing the solution of the original PDE from the solution of either blocks; a first order (in time) scheme is given by

$$\tilde{f}(t+\Delta t) = e^{L_2 \Delta t} e^{L_1 \Delta t} f(t),$$

while a second order (in time) scheme is given by

$$\tilde{f}(t+\Delta t) = e^{L_1 \frac{\Delta t}{2}} e^{L_2 \Delta t} e^{L_1 \frac{\Delta t}{2}} f(t).$$

General framework

The (formal) exact solution of the linear PDE

$$\frac{\partial f}{\partial t} = Lf, \qquad f(t=0) = f^0$$

is

$$f(t) = e^{Lt} f^0.$$

If we can write the linear operator L as the sum of two linear operators,

$$L = L_1 + L_2,$$

then we may approximate the exact solution by solving for separate

$$\frac{\partial f}{\partial t} = L_1 f \qquad \text{and} \qquad \frac{\partial f}{\partial t} = L_2 f.$$

Several schemes are proposed for reconstructing the solution of the original PDE from the solution of either blocks; a first order (in time) scheme is given by

$$\tilde{f}(t+\Delta t) = e^{L_2 \Delta t} e^{L_1 \Delta t} f(t),$$

while a second order (in time) scheme is given by

$$\tilde{f}(t+\Delta t) = e^{L_1 \frac{\Delta t}{2}} e^{L_2 \Delta t} e^{L_1 \frac{\Delta t}{2}} f(t).$$

General framework

The (formal) exact solution of the linear PDE

$$\frac{\partial f}{\partial t} = Lf, \qquad f(t=0) = f^0$$

is

$$f(t) = e^{Lt} f^0.$$

If we can write the linear operator L as the sum of two linear operators,

$$L = L_1 + L_2,$$

then we may approximate the exact solution by solving for separate

$$\frac{\partial f}{\partial t} = L_1 f \qquad \text{and} \qquad \frac{\partial f}{\partial t} = L_2 f.$$

Several schemes are proposed for reconstructing the solution of the original PDE from the solution of either blocks; a first order (in time) scheme is given by

$$\tilde{f}(t+\Delta t) = e^{L_2 \Delta t} e^{L_1 \Delta t} f(t),$$

while a second order (in time) scheme is given by

$$\tilde{f}(t+\Delta t) = e^{L_1 \frac{\Delta t}{2}} e^{L_2 \Delta t} e^{L_1 \frac{\Delta t}{2}} f(t).$$

General framework

The (formal) exact solution of the linear PDE

$$\frac{\partial f}{\partial t} = Lf, \qquad f(t=0) = f^0$$

is

$$f(t) = e^{Lt} f^0.$$

If we can write the linear operator L as the sum of two linear operators,

$$L = L_1 + L_2,$$

then we may approximate the exact solution by solving for separate

$$\frac{\partial f}{\partial t} = L_1 f \qquad \text{and} \qquad \frac{\partial f}{\partial t} = L_2 f.$$

Several schemes are proposed for reconstructing the solution of the original PDE from the solution of either blocks; a first order (in time) scheme is given by

$$\tilde{f}(t+\Delta t) = e^{L_2 \Delta t} e^{L_1 \Delta t} f(t),$$

while a second order (in time) scheme is given by

$$\tilde{f}(t+\Delta t) = e^{L_1 \frac{\Delta t}{2}} e^{L_2 \Delta t} e^{L_1 \frac{\Delta t}{2}} f(t).$$

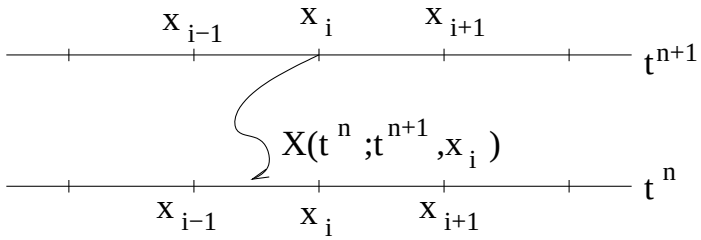
Outline

Linear advection

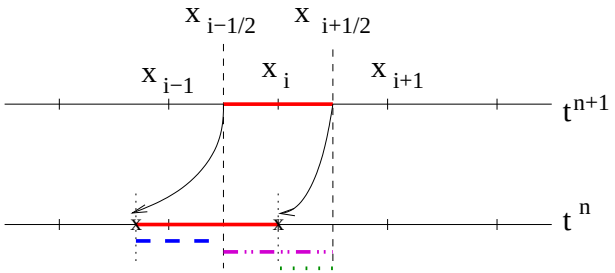
We propose two schemes for solving the 1D advection

$$\frac{\partial f}{\partial t} + \frac{\partial (af)}{\partial x} = 0 :$$

- **direct:** directly integrate backward in the characteristic $f^{n+1}(x) = f^n(\mathcal{X}(t^n; t^{n+1}, x))J(t^n; t^{n+1}, x)$:



- **Flux Balance Method:** total mass conservation is enforced:



The averages along the red segments are the same, because we have followed the characteristics backward.

FLUX BALANCE METHOD means evualuating

Outline

Discretization

Partition of the computational domain

The computational domain $\Omega = [0, 1]$ is partitioned into N cells of size Δx:

$$\Omega = \bigcup_{i=0}^{N-1} I_i, \qquad I_i = [x_{i-1/2}, x_{i+1/2}].$$

Discontinuous Galerkin space

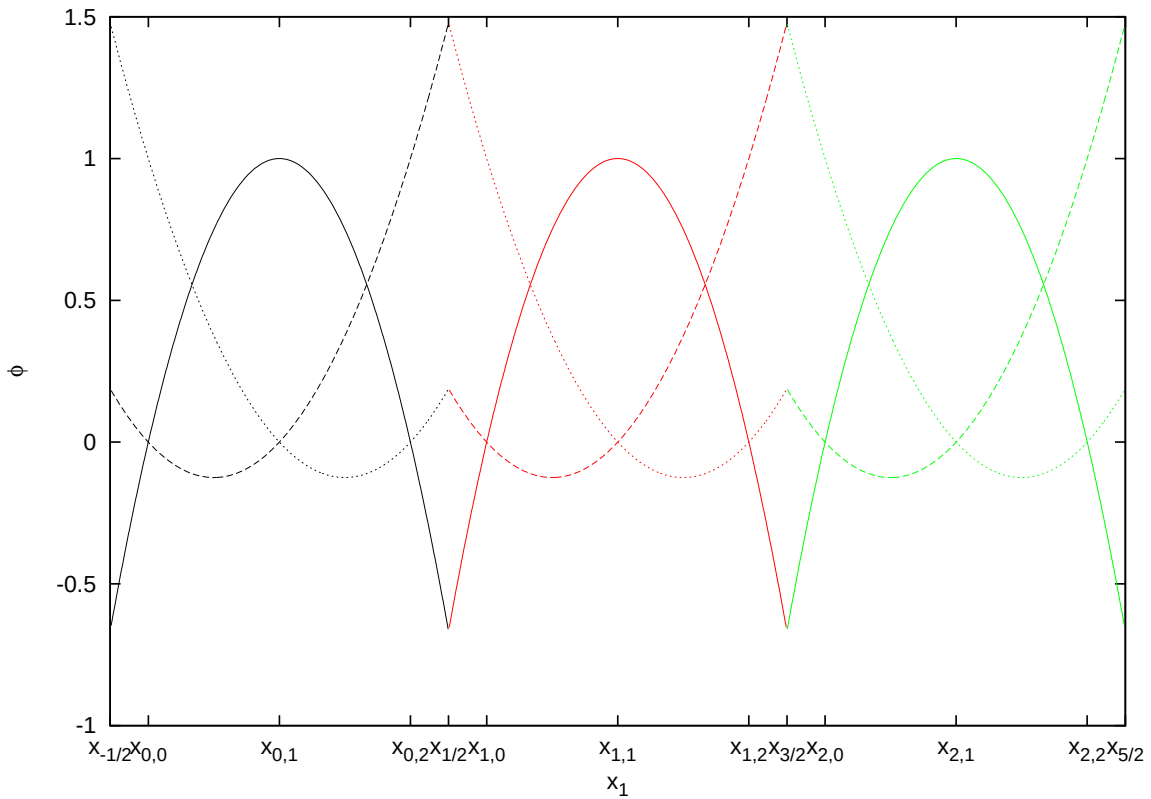
Let V^d the discontinuous finite elements space:

$$V^d = \left\{ \psi \in L^2(\Omega) : \psi \in \mathbb{R}_d[X](I_i), \qquad i = 0, ..., N-1 \right\}.$$

Choice of the basis

Lagrange polynomials

We choose to use the Lagrange polynomials at the Gauß points as basis.



Choice of the basis

The Gauß points on the interval $[-1, 1]$

The Gauß points $\{\alpha_r\}_{r=0}^d$ and the Gauß weights $\{\omega_r\}_{r=0}^d$ are quadrature points determined by imposing

$$\int_{-1}^{1} f(x)dx = \sum_{r=0}^{d} \omega_r f(\alpha_r)$$

for all polynomials $f \in \mathbb{R}_{2d+1}[X]$.

Distributing the Gauß points

We can now introduce the notation $x_{i,j}$ for the j-th Gauß point inside the interval I_i; more precisely

$$x_{i,j} = x_{i-1/2} + \frac{\Delta x}{2}\alpha_j.$$

Choice of the basis

Orthogonality of the basis

As the Lagrange polynomials at the Gauß points are defined by

$$\varphi_{i,j} = \prod_{l=0, l\neq j}^{d} \frac{x - x_{i,l}}{x_{i,j} - x_{i,l}},$$

it is easy to check that

$$\int_{I_i} \varphi_{i,j_1}(x)\varphi_{i,j_2}(x) = \frac{\Delta x}{2} \sum_{r=0}^{d} \omega_r \varphi_{i,j_1}(\alpha_r)\varphi_{i,j_2}(\alpha_r) = \frac{\Delta x}{2} \omega_{j_1} \delta_{j_1,j_2}.$$

Notation for the future

We shall denote by $\{\tilde{\varphi}^j\}_{j=0}^d$ and $\{\tilde{\alpha}_j\}_{j=0}^d$ the Lagrange polynomials and the Gauß points on the interval $[0, 1]$.

Characteristics-based method

The strategy follows that of the 1D linear advection.

Starting point

Test f^{n+1} over the interval I_i:

$$\int_{x_{i-1/2}}^{x_{i+1/2}} f^{n+1}(x)\varphi(x)dx$$

use the solution given by the characteristics

$$\int_{x_{i-1/2}}^{x_{i+1/2}} f^{n+1}(x)\varphi(x)dx = \int_{x_{i-1/2}}^{x_{i+1/2}} f^{n}(\mathcal{X}(t^n;t^{n+1},x))J(t^n;t^{n+1},x)\varphi(x)dx$$

change variables $x \to \mathcal{X}(t^n;t^{n+1},x)$

$$\int_{x_{i-1/2}}^{x_{i+1/2}} f(t^{n+1},x)\varphi(x)dx = \int_{\mathcal{X}(t^n;t^{n+1},x_{i-1/2})}^{\mathcal{X}(t^n;t^{n+1},x_{i+1/2})} f(t^n,x)\varphi(\mathcal{X}(t^{n+1};t^n,x))dx.$$

Characteristics-based method

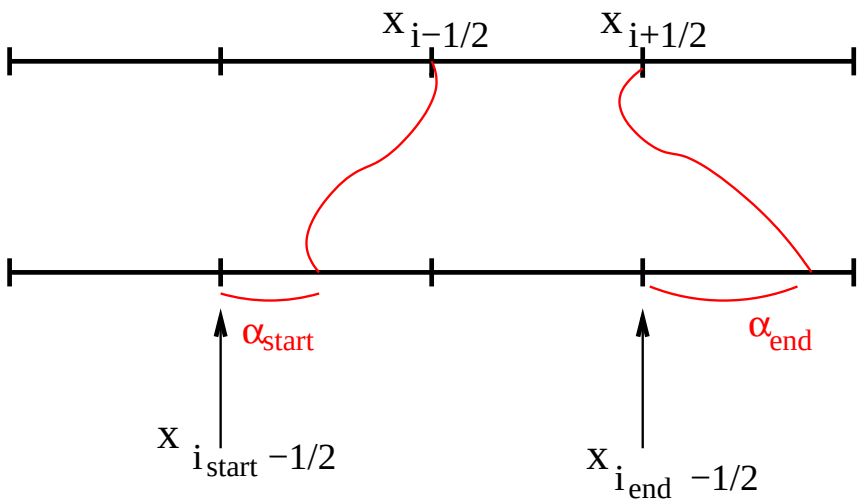
Developing the scheme

Inject the representation of $f(x)$ into the scheme and test on $\varphi_{i,j}(x)$:

$$\int_{x_{i-1/2}}^{x_{i+1/2}} f(t^{n+1},x)\varphi_{i,j}(x)dx = \sum_{i',j'} f^n_{i',j'} \int_{\mathcal{X}(t^n;t^{n+1},x_{i-1/2})}^{\mathcal{X}(t^n;t^{n+1},x_{i+1/2})} \varphi_{i',j'}(x)\varphi_{i,j}(\mathcal{X}(t^{n+1};t^n,x))dx.$$

Some notations

Let $i_{start} = i_{start}(i)$, $\alpha_{start} = \alpha_{start}(i) \in [0,1]$ and $i_{start} = i_{end}(i)$, $\alpha_{end} = \alpha_{end}(i) \in [0,1]$ such that



Characteristics-based method

Treating the right hand side

The integral is decomposed into three pieces:

$$
\begin{aligned}
f_{i,j}^{n+1}\omega_j\frac{\Delta x}{2} \quad = \quad & \sum_{i',j'} f_{i',j'}^{n} \int_{x_{i_{start}-1/2}+\alpha_{start}\Delta x}^{x_{i_{start}+1/2}} \varphi_{i',j'}(x)\varphi_{i,j}(\mathcal{X}(t^{n+1};t^n,x))dx \\
& + \sum_{i',j'} f_{i',j'}^{n} \sum_{i''=i_{start}+1}^{i_{end}-1} \int_{x_{i''-1/2}}^{x_{i''+1/2}} \varphi_{i',j'}(x)\varphi_{i,j}(\mathcal{X}(t^{n+1};t^n,x))dx \\
& + \sum_{i',j'} f_{i',j'}^{n} \int_{x_{i_{end}-1/2}}^{x_{i_{end}-1/2}+\alpha_{end}\Delta x} \varphi_{i',j'}(x)\varphi_{i,j}(\mathcal{X}(t^{n+1};t^n,x))dx.
\end{aligned}
$$

Characteristics-based method

The scheme

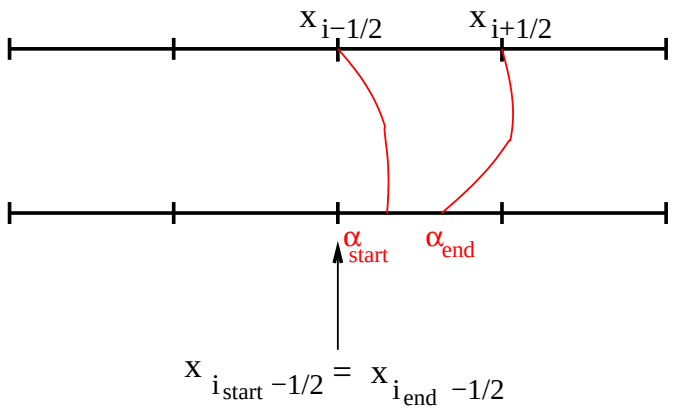
By changing variables to reduce to integrating on $[0,1]$ and using Gauß quadrature we are led to

$$
\begin{aligned}
f_{i,j}^{n+1} \quad = \quad & \frac{1}{\omega_j} \sum_{j'=0}^{d} f_{i_{start},j'}^{n} (1-\alpha_{start}) \sum_{r=0}^{d} \omega_r \tilde{\varphi}^{j'}(\alpha_{start} + \tilde{\alpha}_r(1-\alpha_{start})) \\
& \times \varphi_{i,j}(\mathcal{X}(t^{n+1}; t^n, x_{i_{start}-1/2} + (\alpha_{start} + \tilde{\alpha}_r(1-\alpha_{start}))\Delta x)) \\
+ \quad & \frac{1}{\omega_j} \sum_{i''=i_{start}+1}^{i_{end}-1} \sum_{j'=0}^{d} f_{i'',j'}^{n} \sum_{r=0}^{d} \omega_r \tilde{\varphi}^{j'}(\tilde{\alpha}_r) \varphi_{i,j}(\mathcal{X}(t^{n+1}; t^n, x_{i''-1/2} + \tilde{\alpha}_r \Delta x)) \\
+ \quad & \frac{1}{\omega_j} \sum_{j'=0}^{d} f_{i_{end},j'}^{n} \alpha_{end} \sum_{r=0}^{d} \omega_r \tilde{\varphi}^{j'}(\alpha_{end}\tilde{\alpha}_r) \varphi_{i,j}(\mathcal{X}(t^{n+1}; t^n, x_{i_{end}-1/2} + \alpha_{end}\tilde{\alpha}_r \Delta x)).
\end{aligned}
$$

Characteristics-based method

Case of compression

In case a compression should happen



then the formula reduces to just one integral:

$$f_{i,j}^{n+1} = \frac{1}{\omega_j}\sum_{j'=0}^{d} f_{i_{start},j'}^{n}(\alpha_{end}-\alpha_{start})\sum_{r=0}^{d}\omega_r\varphi^{j'}\left((\alpha_{end}-\alpha_{start})\tilde{\alpha}_r+\alpha_{start}\right)$$
$$\times\varphi_{i,j}(\mathcal{X}(t^{n+1};t^n,x_{i_{start}-1/2}+\Delta x((\alpha_{end}-\alpha_{start})\tilde{\alpha}_r+\alpha_{start}))).$$

Characteristics-based method

Solving the characteristics

In order to write the scheme, we still need to solve the characteristics, both forward and backward. In order to do this, we shall use an explicit formula if it is available, otherwise Runge-Kutta methods of order 1 to 4.

Outline

Order-in-space of the methods

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

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

Total variation control

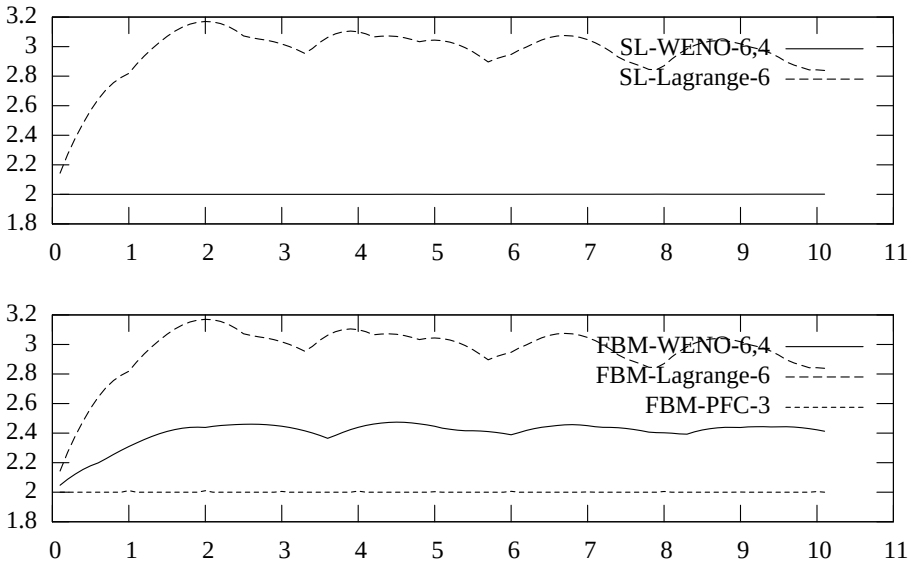


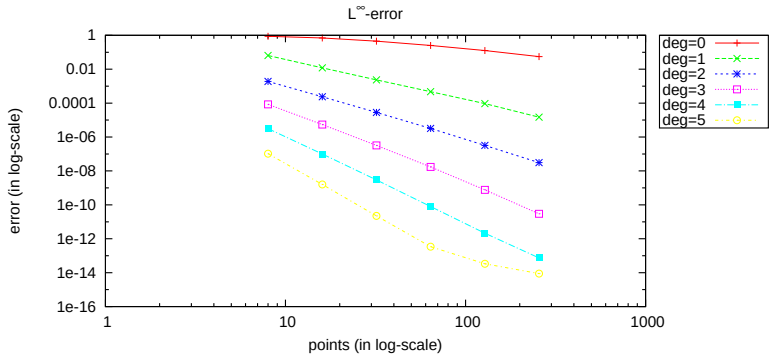
Figure: The evolution of Discrete Total Variation against time. In this test, $N = 100$, $x \in [-\pi, \pi]$, $\Delta t = 0.1$, $t_{max} = 10$, $f_0(x) = f^{step}(x)$.

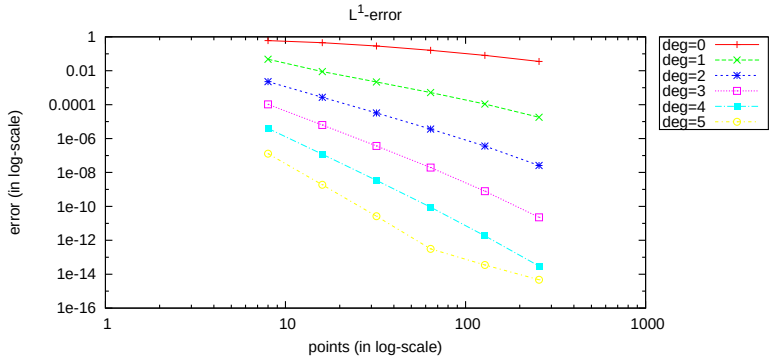
Two-stream instability

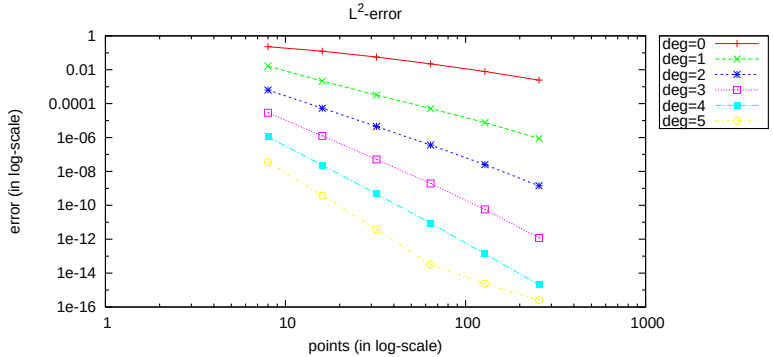
SL-WENO-6,4 behaves properly, while SL-WENO-5,3 does not.

Outline

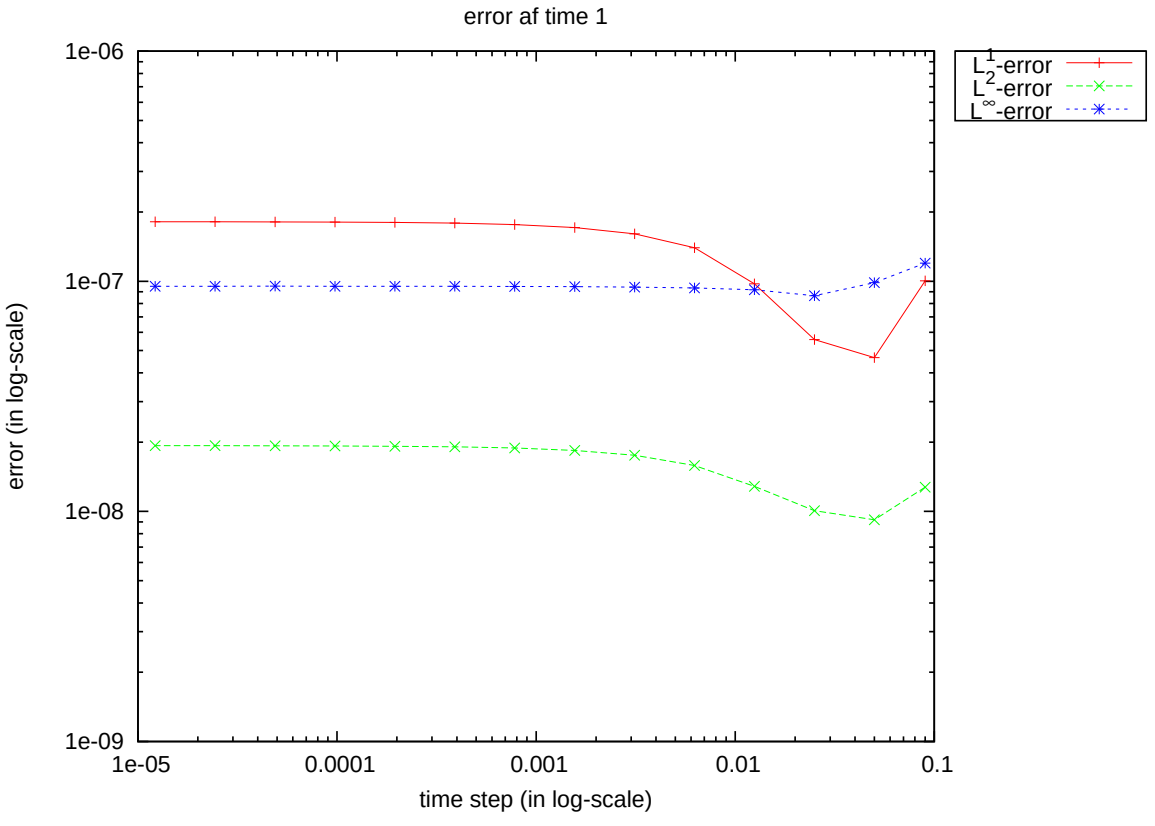
1D linear advection







1D linear advection



Landau damping

Period and decay

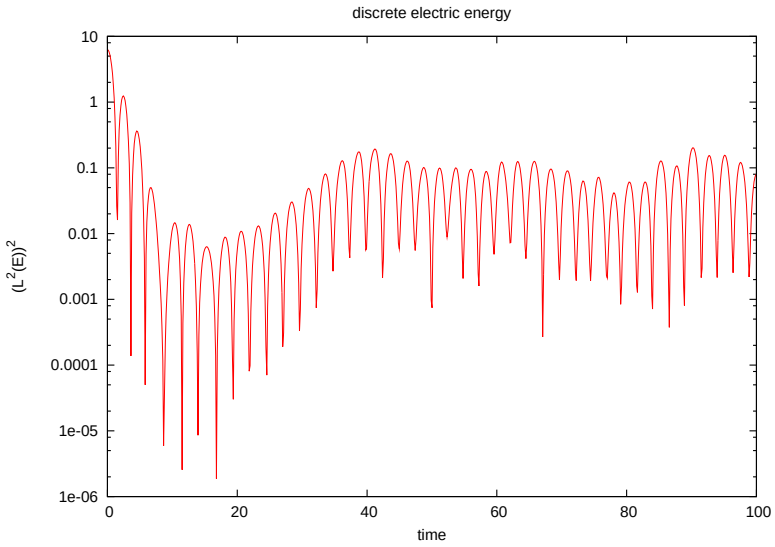
Set the problem $f_0(x,v) = \frac{1}{\sqrt{2\pi}} e^{-\frac{v^2}{2}} (1 + \alpha \cos(kx))$.

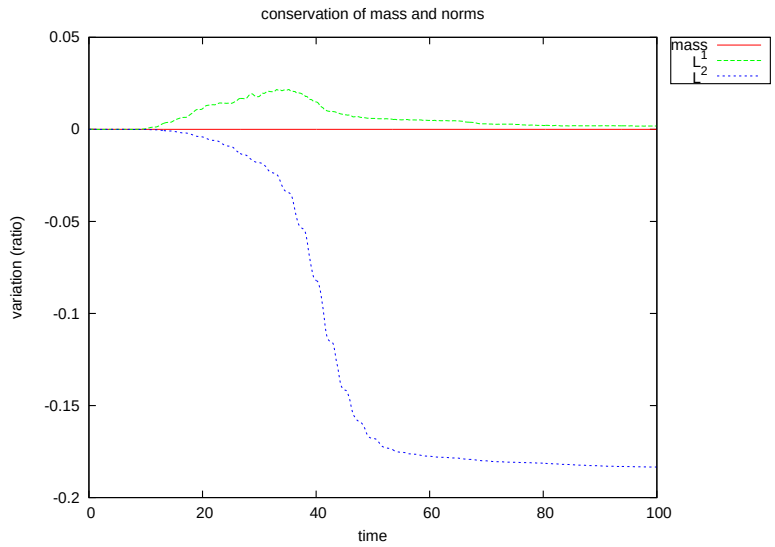
k	$\alpha = 0.001$ (linear)	$\alpha = 0.5$ (nonlinear)
0.2	$\pm 1.07154 + 6.81267 \times 10^{-5} i$ $(\pm 1.0640 - 5.51 \times 10^{-5} i)$	$\pm 1.09402 - 0.00107607 i$
0.3	$\pm 1.16209 - 0.0124224 i$ $(\pm 1.1598 - 0.0126 i)$	$\pm 1.30507 - 0.128511 i$
0.4	$\pm 1.28645 - 0.0659432 i$ $(\pm 1.2850 - 0.0661 i)$	$\pm 1.3581 - 0.205133 i$
0.5	$\pm 1.41696 - 0.152849 i$ $(\pm 1.4156 - 0.1533 i)$	$\pm 1.47343 - 0.279512 i$

Table: **1D Landau damping.** The decay rate and period of the oscillations of the electric field in the Landau damping problem. Here, $d = 4$, $N_x \times N_v = 30 \times 30$.

Landau damping

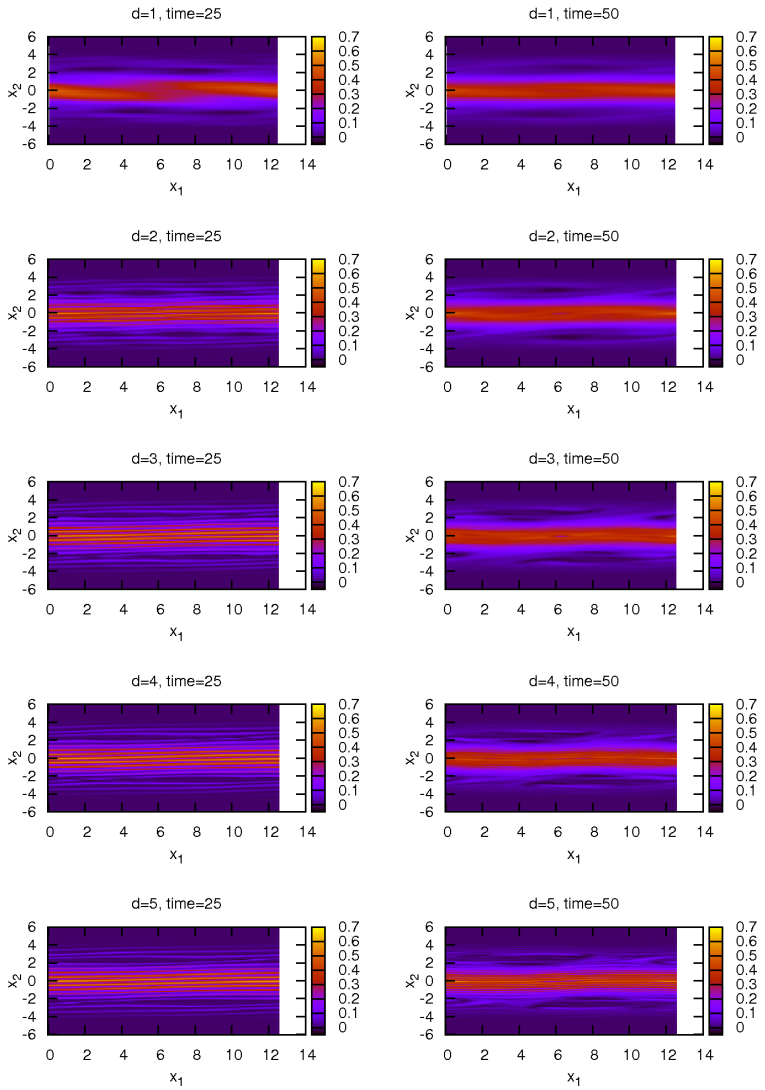
Nonlinear Landau damping





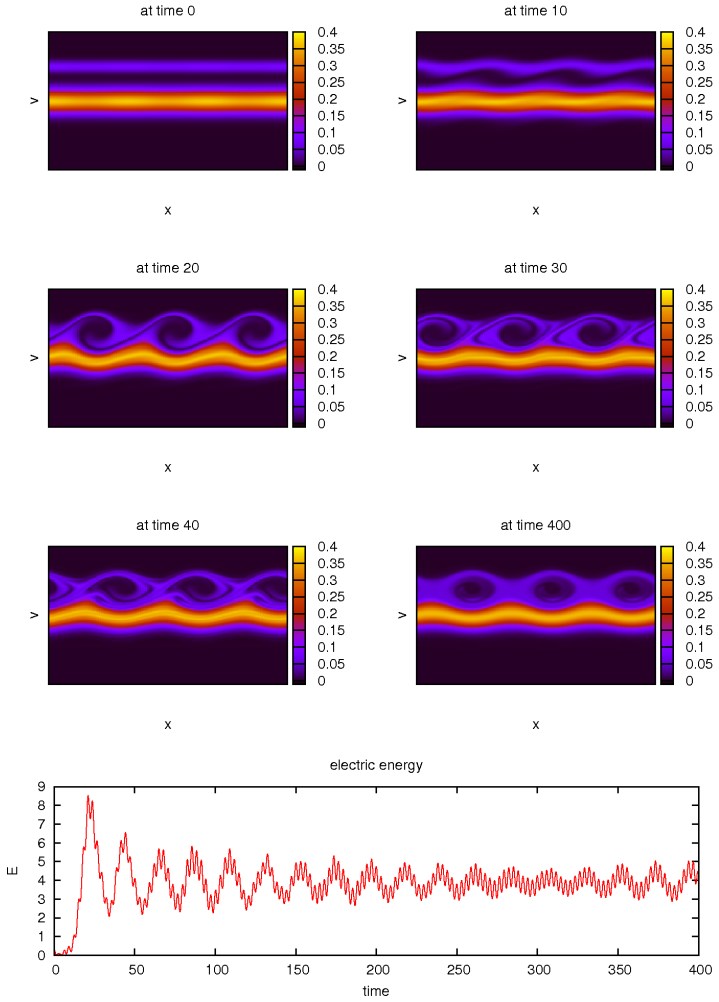
Landau damping

Filamentation of the phase-space



Bump-On-Tail

By using $f_0(x,v) = \frac{9}{10\sqrt{2\pi}} e^{-\frac{v^2}{2}} + \frac{2}{10\sqrt{2\pi}} e^{-2|v-4.5|^2}(1 + 0.03\cos(0.3x))$ as initial condition, we expect to observe some vortices.



1D nonlinear advection

We take as test case

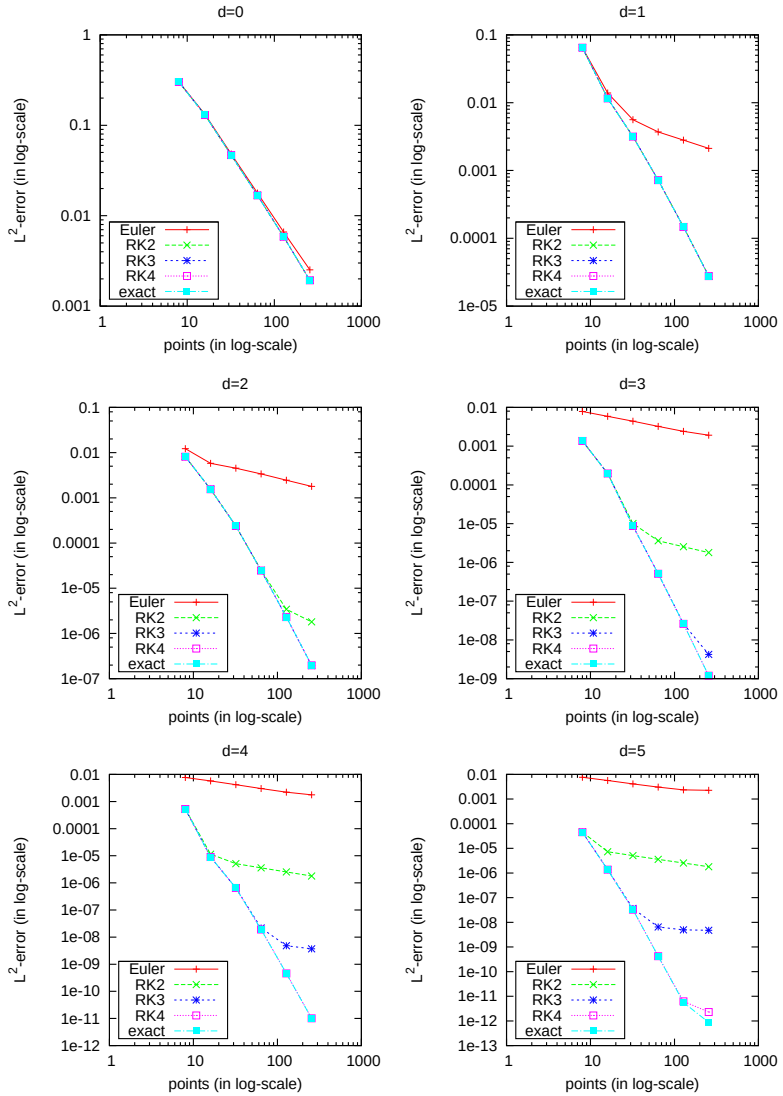
$$\frac{\partial f}{\partial t} + \frac{\partial (sin(x) f)}{\partial x} = 0,$$

which has explicit characteristics and solution:

$$\begin{aligned} \mathcal{X}(s;t,x) &= 2\arctan\left(\tan\left(\frac{x}{2}\right)e^{s-t}\right) + 2\pi\left\lfloor\frac{x+\pi}{2\pi}\right\rfloor \\ f(t,x) &= \frac{1}{1+\left(\tan\left(\frac{x}{2}\right)e^{-t}\right)^2}\frac{1}{cos^2\left(\frac{x}{2}\right)}e^{-t}. \end{aligned}$$

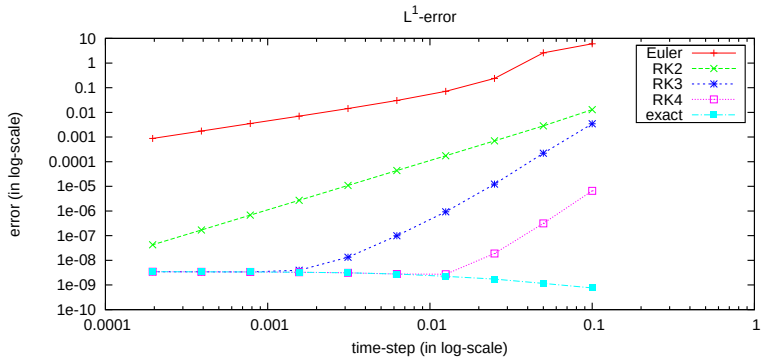
1D nonlinear advection

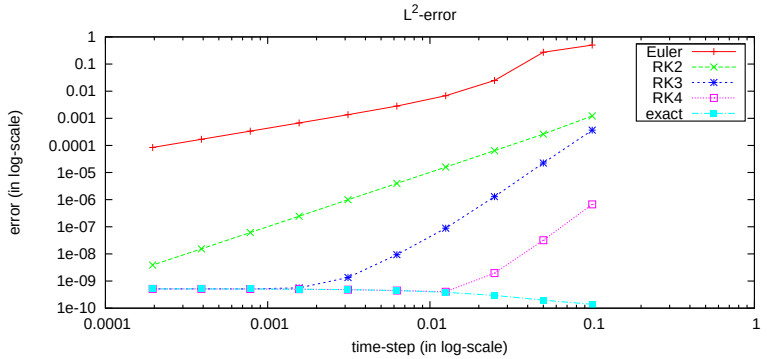
Order in space

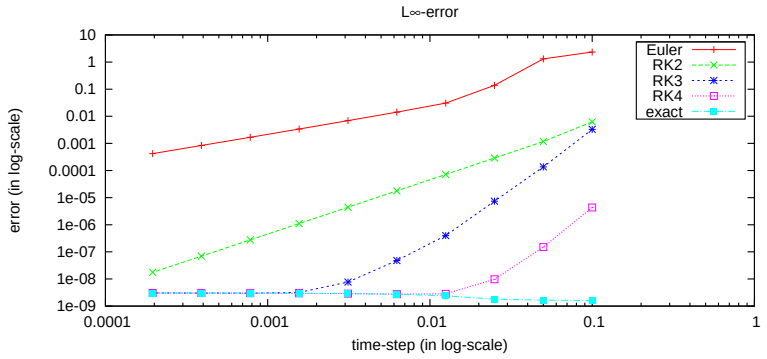


1D nonlinear advection

Order in time



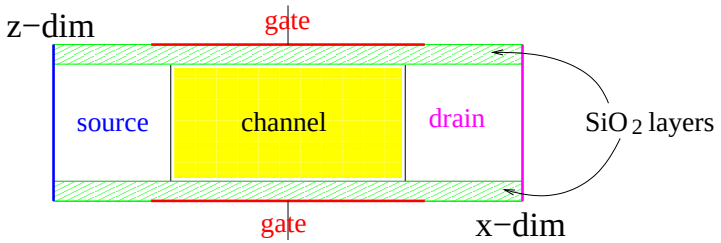




Outline

The model

We afford the simulation of a nanoscaled MOSFET.



Dimensional coupling

x-dimension is unconfined unlike *z*-dimension, therefore we adopt a different description:

- along *x*-dimension the electrons behave like particles, their movement being described by the Boltzmann Transport Equation;
- along *z*-dimension the electrons, confined in a potential well, behave like waves; the equilibrium being reached much faster than transport (quasi-static phenomenon), their state is given by the stationary-state Schrödinger equation.

The model

Subband decomposition

Due to the confinement, different *sub-bands* (another name for the eigenvalues of the Schrödinger equation) identify independent populations, which have to be transported for separate.

Coupling between dimensions

Dimensions and subbands are coupled in the Poisson equation for the computation of the electrostatic field in the expression of the total density.

The model

Subband decomposition

Due to the confinement, different *sub-bands* (another name for the eigenvalues of the Schrödinger equation) identify independent populations, which have to be transported for separate.

Coupling between dimensions

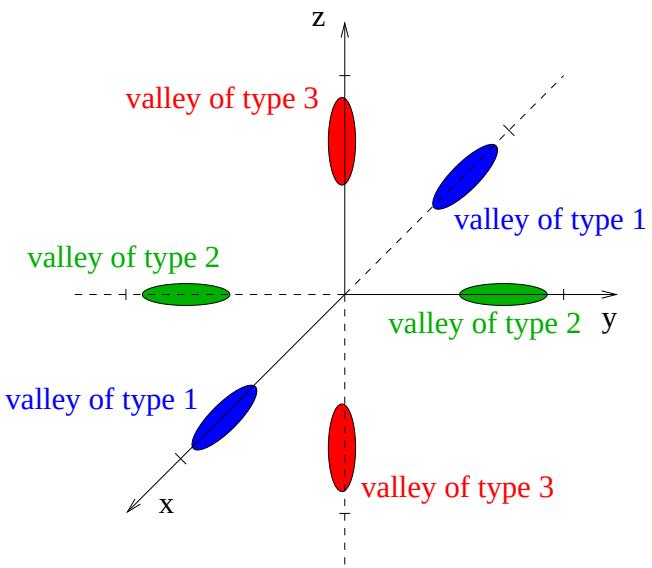
Dimensions and subbands are coupled in the Poisson equation for the computation of the electrostatic field in the expression of the total density.

Bandstructure

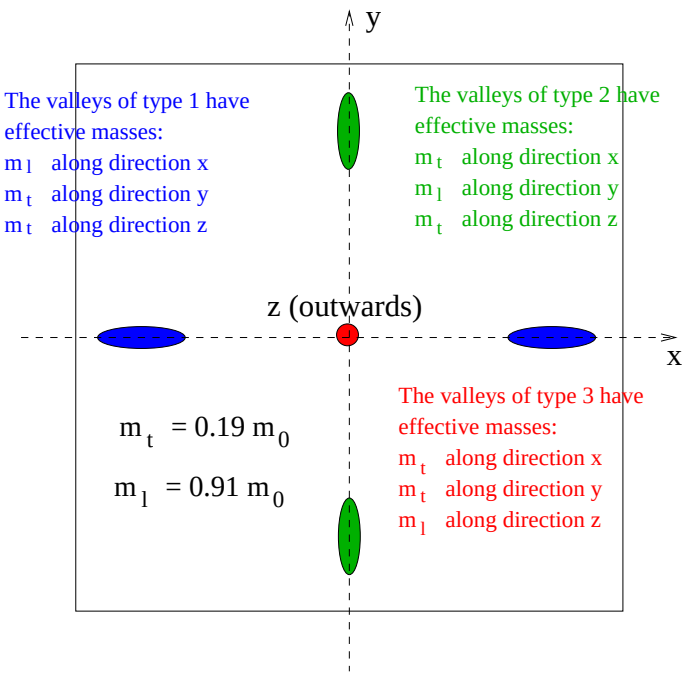
The three valleys

The Si bandstructure presents six minima in the first Brillouin zone:

The constant energy surfaces in the wavevector space



Projection on the (x,y)−plane



The axes of the ellipsoids are disposed along the *x*, *y* and *z* axes of the reciprocal lattice. The three minima have the same value, therefore there is no gap.

Bandstructure

Coupling between subbands and valleys

The sub-bands as well as the valleys are coupled by the Poisson equation in the expression of the total density and, if the case, by the scattering operator.

Non-parabolicity

The bandstructure around the three minima can be expanded following the Kane non-parabolic approximation (ν indexes the valley):

$$\epsilon_\nu^{kin} = \frac{\hbar^2}{1+\sqrt{1+2\tilde{\alpha}_\nu\hbar^2\left(\frac{k_x^2}{m_{x,\nu}}+\frac{k_y^2}{m_{y,\nu}}\right)}}\left(\frac{k_x^2}{m_{x,\nu}}+\frac{k_y^2}{m_{y,\nu}}\right),$$

where $m_{\{x,y,z\},\nu}$ are the axes of the ellispoids (called *effective masses*) of the ν^{th} valley along x, y and z directions, and the $\tilde{\alpha}_\nu$ are known as Kane dispersion factors.

Bandstructure

Coupling between subbands and valleys

The sub-bands as well as the valleys are coupled by the Poisson equation in the expression of the total density and, if the case, by the scattering operator.

Non-parabolicity

The bandstructure around the three minima can be expanded following the Kane non-parabolic approximation (ν indexes the valley):

$$\epsilon_\nu^{kin} = \frac{\hbar^2}{1+\sqrt{1+2\tilde{\alpha}_\nu\hbar^2\left(\frac{k_x^2}{m_{x,\nu}}+\frac{k_y^2}{m_{y,\nu}}\right)}}\left(\frac{k_x^2}{m_{x,\nu}}+\frac{k_y^2}{m_{y,\nu}}\right),$$

where $m_{\{x,y,z\},\nu}$ are the axes of the ellispoids (called *effective masses*) of the ν^{th} valley along x, y and z directions, and the $\tilde{\alpha}_\nu$ are known as Kane dispersion factors.

The model

BTE

The Boltzmann Transport Equation (one for each band and for each valley) reads

$$\frac{\partial f_{\nu,p}}{\partial t} + \frac{1}{\hbar}\nabla_k \epsilon_\nu^{kin} \cdot \nabla_x f_{\nu,p} - \frac{1}{\hbar}\nabla_x \epsilon_{\nu,p}^{pot} \cdot \nabla_k f_{\nu,p} = \mathcal{Q}_{\nu,p}[f], \qquad f_{\nu,p}(t=0) = \rho_{\nu,p}^{eq} M_\nu.$$

Schrödinger-Poisson block

$$-\frac{\hbar^2}{2}\frac{d}{dz}\left[\frac{1}{m_{z,\nu}}\frac{d\chi_{\nu,p}[V]}{dz}\right] - q\left(V + V_c\right)\chi_{\nu,p}[V] = \epsilon_{\nu,p}^{pot}[V]\chi_{\nu,p}[V]$$

$$\langle \chi_{\nu,p}[V], \chi_{\nu,p'}[V] \rangle = \delta_{p,p'}$$

$$-\text{div}\left[\varepsilon_R \nabla V\right] = -\frac{q}{\varepsilon_0}\left(N[V] - N_D\right)$$

$$N[V] = \sum_{\nu,p} \rho_{\nu,p} |\chi_{\nu,p}[V]|^2$$

These equations cannot be decoupled because we need the eigenfunctions to compute the potential (in the expression of the total density), and we need the potential to compute the eigenfunctions.

The model

BTE

The Boltzmann Transport Equation (one for each band and for each valley) reads

$$\frac{\partial f_{\nu,p}}{\partial t} + \frac{1}{\hbar}\nabla_k \epsilon_\nu^{kin} \cdot \nabla_x f_{\nu,p} - \frac{1}{\hbar}\nabla_x \epsilon_{\nu,p}^{pot} \cdot \nabla_k f_{\nu,p} = \mathcal{Q}_{\nu,p}[f], \qquad f_{\nu,p}(t=0) = \rho_{\nu,p}^{eq} M_\nu .$$

Schrödinger-Poisson block

$$-\frac{\hbar^2}{2}\frac{d}{dz}\left[\frac{1}{m_{z,\nu}}\frac{d\chi_{\nu,p}[V]}{dz}\right] - q\left(V + V_c\right)\chi_{\nu,p}[V] = \epsilon_{\nu,p}^{pot}[V]\chi_{\nu,p}[V]$$

$$\langle \chi_{\nu,p}[V], \chi_{\nu,p'}[V] \rangle = \delta_{p,p'}$$

$$-\mathrm{div}\left[\varepsilon_R \nabla V\right] = -\frac{q}{\varepsilon_0}\left(N[V] - N_D\right)$$

$$N[V] = \sum_{\nu,p} \rho_{\nu,p} |\chi_{\nu,p}[V]|^2$$

These equations cannot be decoupled because we need the eigenfunctions to compute the potential (in the expression of the total density), and we need the potential to compute the eigenfunctions.

The model

The collision operator

The collision operator takes into account the phonon scattering mechanism. It reads

$$\mathcal{Q}_{\nu,p}[f] = \sum_s \mathcal{Q}^s_{\nu,p}[f]$$

$$\mathcal{Q}^s_{\nu,p}[f] = \sum_{\nu',p'} \int_{\mathbb{R}^2} \left[S^s_{(\nu',p',k')\to(\nu,p,k)} f_{\nu',p'}(k') - S^s_{(\nu,p,k)\to(\nu',p',k')} f_{\nu,p}(k) \right] dk' :$$

every S^s represents a different interaction.

Structure of the S^s

The missing dimension of the wave-vector $k \in \mathbb{R}^2$, instead of $k \in \mathbb{R}^3$, is replaced by an overlap integral $W_{(\nu,p),(\nu',p')}$:

$$S^s_{(\nu,p,k)\to(\nu',p',k')} = C_{\nu\to\nu'} \frac{1}{W_{(\nu,p),(\nu',p')}} \delta\left(\epsilon^{tot}_{\nu',p'}(k') - \epsilon^{tot}_{\nu,p}(k) \pm \text{some energy}\right)$$

$$\frac{1}{W_{(\nu,p),(\nu',p')}} = \int_0^{l_z} |\chi_{\nu,p}|^2 |\chi_{\nu',p'}|^2 dz, \qquad [W] = m.$$

The model

The collision operator

The collision operator takes into account the phonon scattering mechanism. It reads

$$\mathcal{Q}_{\nu,p}[f] = \sum_s \mathcal{Q}^s_{\nu,p}[f]$$

$$\mathcal{Q}^s_{\nu,p}[f] = \sum_{\nu',p'} \int_{\mathbb{R}^2} \left[S^s_{(\nu',p',k')\to(\nu,p,k)} f_{\nu',p'}(k') - S^s_{(\nu,p,k)\to(\nu',p',k')} f_{\nu,p}(k) \right] dk' :$$

every S^s represents a different interaction.

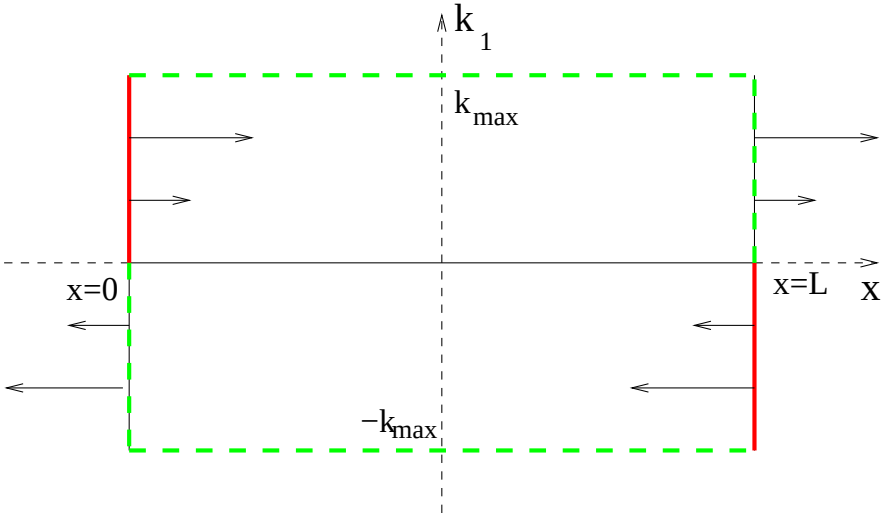
Structure of the S^s

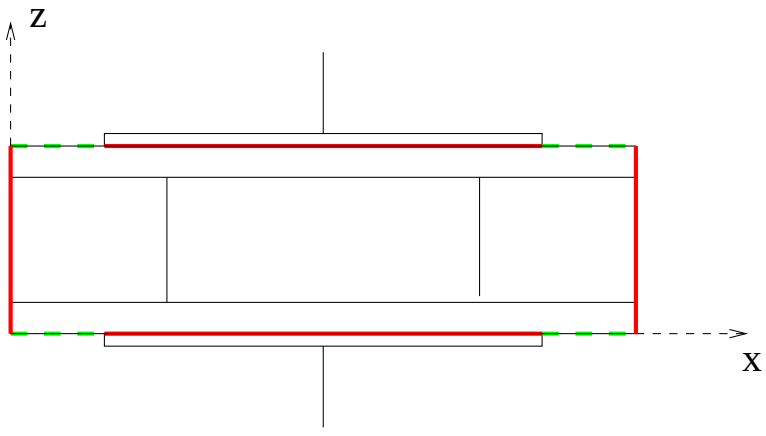
The missing dimension of the wave-vector $k \in \mathbb{R}^2$, instead of $k \in \mathbb{R}^3$, is replaced by an overlap integral $W_{(\nu,p),(\nu',p')}$:

$$S^s_{(\nu,p,k)\to(\nu',p',k')} = C_{\nu\to\nu'} \frac{1}{W_{(\nu,p),(\nu',p')}} \delta\left(\epsilon^{tot}_{\nu',p'}(k') - \epsilon^{tot}_{\nu,p}(k) \pm \text{some energy} \right)$$

$$\frac{1}{W_{(\nu,p),(\nu',p')}} = \int_0^{l_z} |\chi_{\nu,p}|^2 |\chi_{\nu',p'}|^2 dz, \qquad [W] = m.$$

Boundary conditions





= force the density to stay close to the equilibrium density

= Homogeneous Neumann

= Dirichlet

= homogeneous Neumann

Outline

The Newton-Raphson scheme

The functional

Solving the Schrödinger-Poisson block

$$-\frac{\hbar^2}{2}\frac{d}{dz}\left[\frac{1}{m_{z,\nu}}\frac{d\chi_{\nu,p}[V]}{dz}\right]-q\left(V+V_c\right)\chi_{\nu,p}[V]=\epsilon^{pot}_{\nu,p}[V]\chi_{\nu,p}[V]$$

$$-\mathrm{div}\left[\varepsilon_R\nabla V\right]=-\frac{q}{\varepsilon_0}\left(N[V]-N_D\right)$$

is equivalt to minimizing, under the constraints of the Schrödinger equation, the functional $P[V]$

$$P[V]=-\mathrm{div}\left(\varepsilon_R\nabla V\right)+\frac{q}{\varepsilon_0}\left(N[V]-N_D\right),$$

The scheme

which is achieved by means of a Newton-Raphson iterative scheme

$$dP(V^{old},V^{new}-V^{old})=-P[V^{old}].$$

The Newton-Raphson scheme

The functional

Solving the Schrödinger-Poisson block

$$-\frac{\hbar^2}{2}\frac{d}{dz}\left[\frac{1}{m_{z,\nu}}\frac{d\chi_{\nu,p}[V]}{dz}\right]-q\left(V+V_c\right)\chi_{\nu,p}[V]=\epsilon^{pot}_{\nu,p}[V]\chi_{\nu,p}[V]$$

$$-\mathrm{div}\left[\varepsilon_R\nabla V\right]=-\frac{q}{\varepsilon_0}\left(N[V]-N_D\right)$$

is equivalt to minimizing, under the constraints of the Schrödinger equation, the functional $P[V]$

$$P[V]=-\mathrm{div}\left(\varepsilon_R\nabla V\right)+\frac{q}{\varepsilon_0}\left(N[V]-N_D\right),$$

The scheme

which is achieved by means of a Newton-Raphson iterative scheme

$$dP(V^{old},V^{new}-V^{old})=-P[V^{old}].$$

The iterations

Derivatives

The Gâteaux-derivatives of the eigenproperties are needed:

$$
\begin{aligned}
d\epsilon_{\nu,p}(V,U) &= -q\int U(\zeta)|\chi_{\nu,p}[V](\zeta)|^2 d\zeta \\
d\chi_{\nu,p}(V,U) &= -q\sum_{p'\neq p}\frac{\int U(\zeta)\chi_{\nu,p}[V](\zeta)\chi_{\nu,p'}[V](\zeta)d\zeta}{\epsilon_{\nu,p}[V]-\epsilon_{\nu,p'}[V]}\chi_{\nu,p'}[V](z).
\end{aligned}
$$

Iterations

After computing the Gâteaux-derivative of the density and developping calculations, we are led to a Poisson-like equation

$$
\begin{aligned}
&-\mathrm{div}\left(\varepsilon_R\nabla V^{new}\right)+\int_0^{l_z}\mathcal{A}[V^{old}](z,\zeta)V^{new}(\zeta)d\zeta \\
&= -\frac{q}{\varepsilon_0}\left(N[V^{old}]-N_D\right)+\int_0^{l_z}\mathcal{A}[V^{old}](z,\zeta)V^{old}(\zeta)d\zeta,
\end{aligned}
$$

where $\mathcal{A}[V]$ is essentially the Gâteaux-derivative of the functional $P[V]$.

The iterations

Derivatives

The Gâteaux-derivatives of the eigenproperties are needed:

$$
\begin{aligned}
d\epsilon_{\nu,p}(V,U) &= -q\int U(\zeta)|\chi_{\nu,p}[V](\zeta)|^2 d\zeta \\
d\chi_{\nu,p}(V,U) &= -q\sum_{p'\neq p}\frac{\int U(\zeta)\chi_{\nu,p}[V](\zeta)\chi_{\nu,p'}[V](\zeta)d\zeta}{\epsilon_{\nu,p}[V]-\epsilon_{\nu,p'}[V]}\chi_{\nu,p'}[V](z).
\end{aligned}
$$

Iterations

After computing the Gâteaux-derivative of the density and developping calculations, we are led to a Poisson-like equation

$$
\begin{aligned}
&-\mathrm{div}\,(\varepsilon_R\nabla V^{new}) + \int_0^{l_z}\mathcal{A}[V^{old}](z,\zeta)V^{new}(\zeta)d\zeta \\
= &-\frac{q}{\varepsilon_0}\left(N[V^{old}]-N_D\right) + \int_0^{l_z}\mathcal{A}[V^{old}](z,\zeta)V^{old}(\zeta)d\zeta,
\end{aligned}
$$

where $\mathcal{A}[V]$ is essentially the Gâteaux-derivative of the functional $P[V]$.

The Gummel scheme

The iteration

Solving the Schrödinger-Poisson block

$$\begin{aligned} & -\mathrm{div}\left(\varepsilon_R \nabla V^{new}\right) + \frac{q}{\varepsilon_0} N[V^{old}] \frac{q}{k_B T_L} V^{new} \\ = \quad & -\frac{q}{\varepsilon_0}\left(N[V^{old}] - N_D\right) + \frac{q}{\varepsilon_0} N[V^{old}] \frac{q}{k_B T_L} V^{old}, \end{aligned}$$

Comparison with Newton

We here repeat the Newton iteration:

$$\begin{aligned} & -\mathrm{div}\left(\varepsilon_R \nabla V^{new}\right) + \int_0^{l_z} \mathcal{A}[V^{old}](z,\zeta) V^{new}(\zeta) d\zeta \\ = \quad & -\frac{q}{\varepsilon_0}\left(N[V^{old}] - N_D\right) + \int_0^{l_z} \mathcal{A}[V^{old}](z,\zeta) V^{old}(\zeta) d\zeta, \end{aligned}$$

The Gummel scheme

The iteration

Solving the Schrödinger-Poisson block

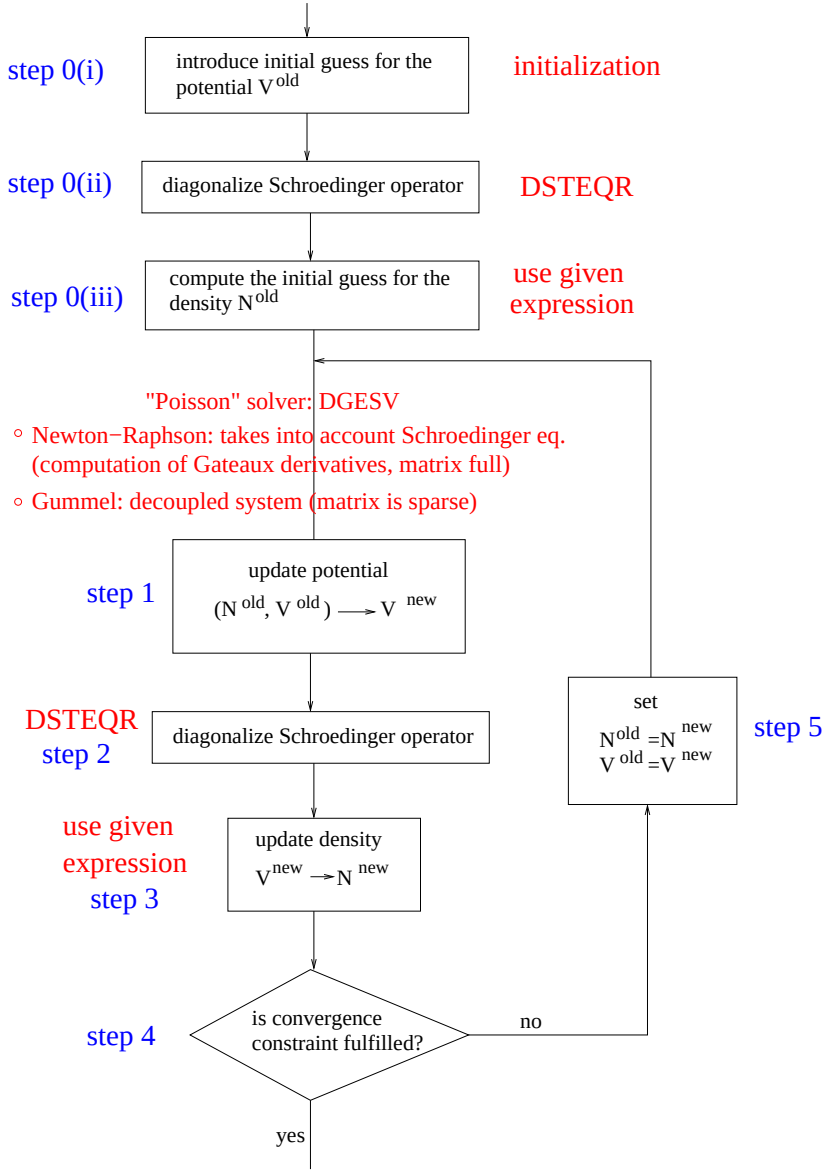
$$\begin{aligned} & -\mathrm{div}\left(\varepsilon_R \nabla V^{new}\right) + \frac{q}{\varepsilon_0} N[V^{old}] \frac{q}{k_B T_L} V^{new} \\ = \quad & -\frac{q}{\varepsilon_0}\left(N[V^{old}] - N_D\right) + \frac{q}{\varepsilon_0} N[V^{old}] \frac{q}{k_B T_L} V^{old}, \end{aligned}$$

Comparison with Newton

We here repeat the Newton iteration:

$$\begin{aligned} & -\mathrm{div}\left(\varepsilon_R \nabla V^{new}\right) + \int_0^{l_z} \mathcal{A}[V^{old}](z,\zeta) V^{new}(\zeta) d\zeta \\ = \quad & -\frac{q}{\varepsilon_0}\left(N[V^{old}] - N_D\right) + \int_0^{l_z} \mathcal{A}[V^{old}](z,\zeta) V^{old}(\zeta) d\zeta, \end{aligned}$$

Framework


step 0(i)
introduce initial guess for the potential V^{old}
initialization
step 0(ii)
diagonalize Schroedinger operator
DSTEQR
step 0(iii)
compute the initial guess for the density N^{old}
use given expression
"Poisson" solver: DGESV
Newton−Raphson: takes into account Schroedinger eq. (computation of Gateaux derivatives, matrix full)
Gummel: decoupled system (matrix is sparse)
step 1
update potential
$(N^{old}, V^{old}) \longrightarrow V^{new}$
DSTEQR
step 2
diagonalize Schroedinger operator
use given expression
step 3
update density
$V^{new} \longrightarrow N^{new}$
step 4
is convergence constraint fulfilled?
no
yes
set
$N^{old} = N^{new}$
$V^{old} = V^{new}$
step 5

Solver for the Schrödinger and the Poisson equations

We need to solve the Schrödinger eigenvalue problem and Poisson equations.

The Schrödinger equation

Equation

$$-\frac{\hbar^2}{2}\frac{d}{dz}\left[\frac{1}{m_{z,\nu}}\frac{d\chi_{\nu,p}}{dz}\right]-q\left(V+V_c\right)\chi_{\nu,p}=\epsilon_{\nu,p}\chi_{\nu,p}$$

is discretized by alternate finite differences for the derivatives then the symmetric matrix is diagonalized by a LAPACK routine called DSTEQR.

The Poisson equation

We need to solve equations like

$$-\mathrm{div}\left[\varepsilon_R\nabla V\right]+\int_0^{l_z}\mathcal{A}(z,\zeta)V(\zeta)d\zeta=\mathcal{B}(z).$$

The derivatives are discretized by finite differences in alternate directions, the integral is computed via trapezoid rule and the linear system (full) is solved by means of a LAPACK routine called DGESV.

Solver for the Schrödinger and the Poisson equations

We need to solve the Schrödinger eigenvalue problem and Poisson equations.

The Schrödinger equation

Equation

$$-\frac{\hbar^2}{2}\frac{d}{dz}\left[\frac{1}{m_{z,\nu}}\frac{d\chi_{\nu,p}}{dz}\right]-q\left(V+V_c\right)\chi_{\nu,p}=\epsilon_{\nu,p}\chi_{\nu,p}$$

is discretized by alternate finite differences for the derivatives then the symmetric matrix is diagonalized by a LAPACK routine called DSTEQR.

The Poisson equation

We need to solve equations like

$$-\mathrm{div}\left[\varepsilon_R\nabla V\right]+\int_0^{l_z}\mathcal{A}(z,\zeta)V(\zeta)d\zeta=\mathcal{B}(z).$$

The derivatives are discretized by finite differences in alternate directions, the integral is computed via trapezoid rule and the linear system (full) is solved by means of a LAPACK routine called DGESV.

Solver for the Schrödinger and the Poisson equations

We need to solve the Schrödinger eigenvalue problem and Poisson equations.

The Schrödinger equation

Equation

$$-\frac{\hbar^2}{2}\frac{d}{dz}\left[\frac{1}{m_{z,\nu}}\frac{d\chi_{\nu,p}}{dz}\right]-q\left(V+V_c\right)\chi_{\nu,p}=\epsilon_{\nu,p}\chi_{\nu,p}$$

is discretized by alternate finite differences for the derivatives then the symmetric matrix is diagonalized by a LAPACK routine called DSTEQR.

The Poisson equation

We need to solve equations like

$$-\mathrm{div}\left[\varepsilon_R\nabla V\right]+\int_0^{l_z}\mathcal{A}(z,\zeta)V(\zeta)d\zeta=\mathcal{B}(z).$$

The derivatives are discretized by finite differences in alternate directions, the integral is computed via trapezoid rule and the linear system (full) is solved by means of a LAPACK routine called DGESV.

Adimensionalization

Adimensionalization is needed in order to improve numerical precision and to exploit the structure of the collision operator.

Wave vector

The wave-vector for the ν^{th} valley reads:

$$(\tilde{k}_x, \tilde{k}_y) = \frac{\sqrt{m_e \kappa_B T_L}}{\hbar} \sqrt{2w(1+\alpha_\nu w)} \left(\sqrt{m_{x,\nu}} \cos(\phi), \sqrt{m_{y,\nu}} \sin(\phi) \right).$$

BTE

$$\frac{\partial \Phi_{\nu,p}}{\partial t} + \frac{\partial}{\partial x}\left[a^1_\nu \Phi_{\nu,p}\right] + \frac{\partial}{\partial w}\left[a^2_{\nu,p} \Phi_{\nu,p}\right] + \frac{\partial}{\partial \phi}\left[a^3_{\nu,p} \Phi_{\nu,p}\right] = \mathcal{Q}_{\nu,p}[\Phi] s(w)$$

Time discretizaion

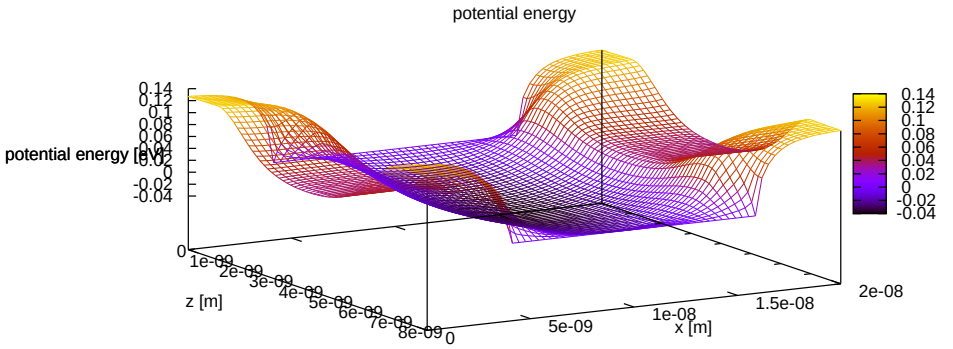
Runge-Kutta

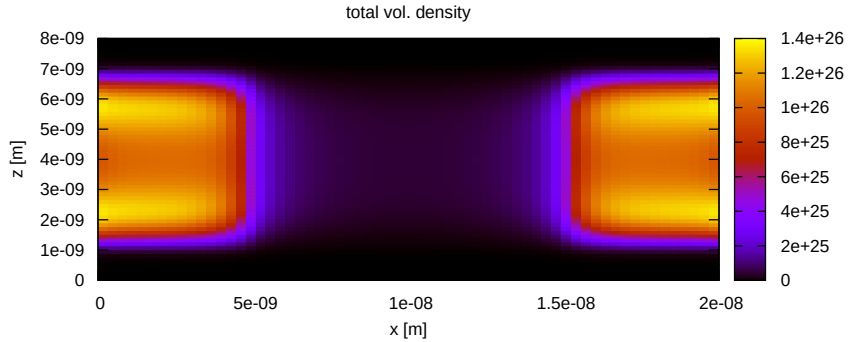
If the BTE is written in conservation-law form, then we advance in time by the third order Total Variation Diminishing Runge-Kutta scheme: if the evolution equation reads $\frac{df}{dt} = H(t, f)$, then

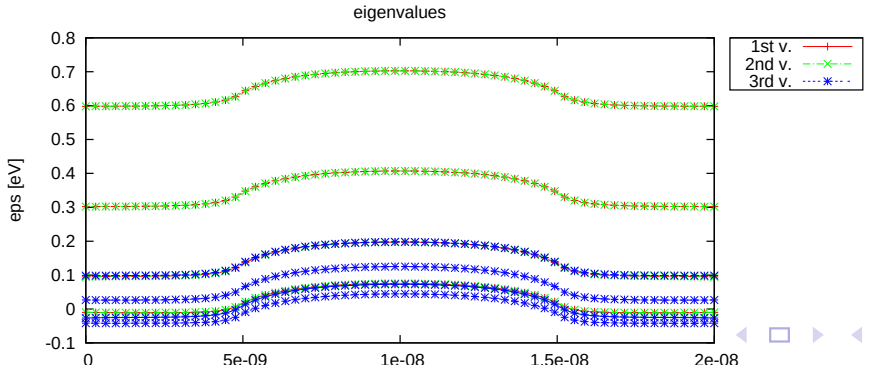
1. $f^{(1)} = \Delta t H^n(t^n, f^n)$
2. $f^{(2)} = \frac{3}{4} f^n + \frac{1}{4} f^{(1)} + \frac{1}{4} \Delta t H^{(1)}(t^n + \Delta t, f^{(1)})$
3. $f^{n+1} = \frac{1}{3} f^n + \frac{2}{3} f^{(2)} + \frac{2}{3} H^{(2)} \left(t^n + \frac{1}{2} \Delta t, f^{(2)} \right)$

Outline

Thermodynamical equilibrium

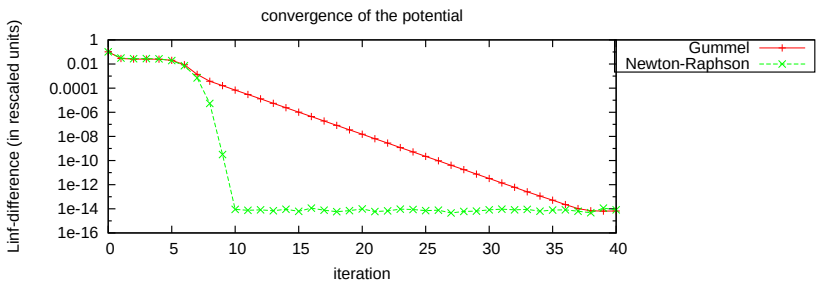


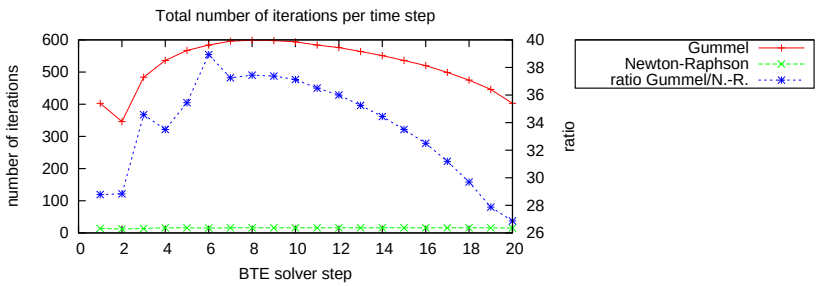


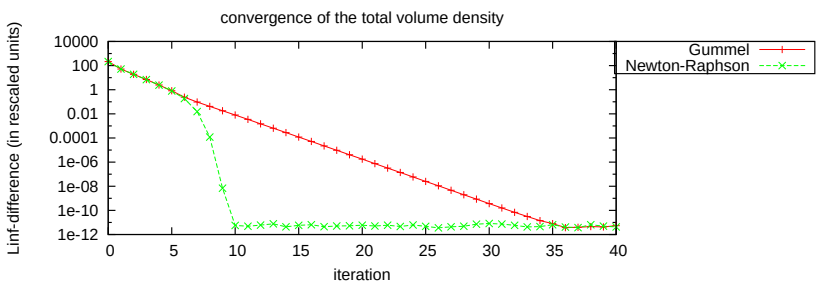


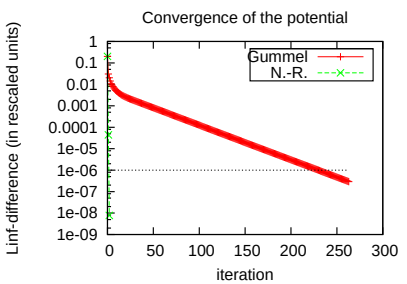
Newton vs. Gummel

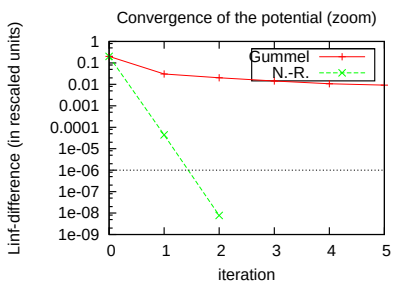
Newton schemes require much less iterations than Gummel in order to compute the thermodynamical equilibrium.











Plasma oscillations