A deterministic hybrid quantum/classical solver for a nanoscaled MOSFET device

Naoufel Ben Abdallah, María José Cáceres, José Antonio Carrillo, Francesco Vecil

Quantum systems and semiconductor device analysis, simulations, applications, 20-24/04

Outline

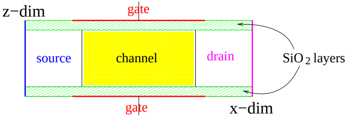
1. The model
   - Geometry
   - Mathematical model
2. Numerical methods for the Schrödinger-Poisson block
   - Iterative schemes
   - Solvers for Schrödinger and Poisson
3. Solvers for the BTE block
   - Adimensionalizations
   - Time discretization
   - Linear advection
   - PWENO interpolations
4. Experiments
   - Simplifying assumptions
   - Equilibria
   - Time-dependent simulations
   - Newton vs. Gummel
   - Plasma oscillations

Outline

1. The model
   - Geometry
   - Mathematical model
2. Numerical methods for the Schrödinger-Poisson block
   - Iterative schemes
   - Solvers for Schrödinger and Poisson
3. Solvers for the BTE block
   - Adimensionalizations
   - Time discretization
   - Linear advection
   - PWENO interpolations
4. Experiments
   - Simplifying assumptions
   - Equilibria
   - Time-dependent simulations
   - Newton vs. Gummel
   - Plasma oscillations

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.

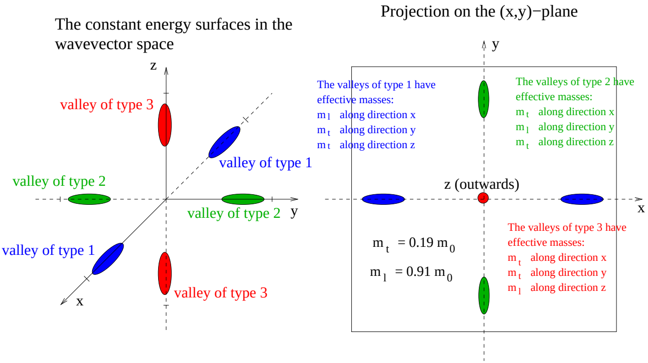
Coupling between dimensions

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 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.

Outline

1. The model
   - Geometry
   - Mathematical model
2. Numerical methods for the Schrödinger-Poisson block
   - Iterative schemes
   - Solvers for Schrödinger and Poisson
3. Solvers for the BTE block
   - Adimensionalizations
   - Time discretization
   - Linear advection
   - PWENO interpolations
4. Experiments
   - Simplifying assumptions
   - Equilibria
   - Time-dependent simulations
   - Newton vs. Gummel
   - Plasma oscillations

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

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

$$\begin{aligned}
&-\frac{\hbar^2}{2}\frac{d}{dz}\left[\frac{1}{m_{z,\nu}}\frac{d\chi_{\nu,p}[V]}{dz}\right] - q\left(V + V_c\right)\chi_{\nu,p}[V] = \epsilon_{\nu,p}^{pot}[V]\chi_{\nu,p}[V]\\
&\langle \chi_{\nu,p}[V], \chi_{\nu,p'}[V]\rangle = \delta_{p,p'}\\
&-\mathrm{div}\left[\varepsilon_R \nabla V\right] = -\frac{q}{\varepsilon_0}\left(N[V] - N_D\right)\\
&N[V] = \sum_{\nu,p} \rho_{\nu,p} \left|\chi_{\nu,p}[V]\right|^2
\end{aligned}$$

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

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

every S^s represents a different interaction.

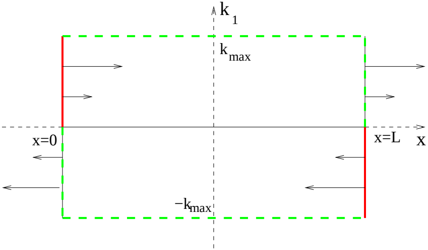
Structure of the S^s

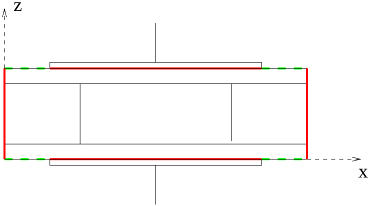
The missing dimension of the wave-vector $k \in \mathbb{R}^2$, instead of $k \in \mathbb{R}^3$, is replaced by an overlap integral $W_{(\nu,p),(\nu',p')}$:

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

$$\frac{1}{W_{(\nu,p),(\nu',p')}} = \int_{0}^{l_z} |\chi_{\nu,p}|^2 |\chi_{\nu',p'}|^2 dz, \qquad [W] = m.$$

Boundary conditions





━━━ = Dirichlet

- - - = homogeneous Neumann

━━━ = force the density to stay close to the equilibrium density

- - - = Homogeneous Neumann

Outline

1 The model
- Geometry
- Mathematical model

2 Numerical methods for the Schrödinger-Poisson block
- Iterative schemes
- Solvers for Schrödinger and Poisson

3 Solvers for the BTE block
- Adimensionalizations
- Time discretization
- Linear advection
- PWENO interpolations

4 Experiments
- Simplifying assumptions
- Equilibria
- Time-dependent simulations
- Newton vs. Gummel
- Plasma oscillations

The Newton 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 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

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

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

The Gummel scheme

The iteration

Solving the Schrödinger-Poisson block

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

Comparison with Newton

We here repeat the Newton iteration:

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

The Gummel scheme

The iteration

Solving the Schrödinger-Poisson block

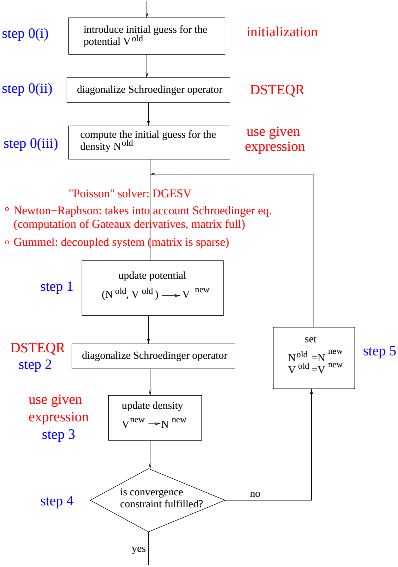
$$
\begin{aligned}
& -\mathrm{div}\left(\varepsilon_R \nabla V^{new}\right) + \frac{q}{\varepsilon_0} N[V^{old}] \frac{q}{k_B T_L} V^{new} \\
= \quad & -\frac{q}{\varepsilon_0}\left(N[V^{old}] - N_D\right) + \frac{q}{\varepsilon_0} N[V^{old}] \frac{q}{k_B T_L} V^{old},
\end{aligned}
$$

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
$V^{new} \rightarrow N^{new}$
step 4
is convergence constraint fulfilled?
no
yes
set
$N^{old} = N^{new}$
$V^{old} = V^{new}$
step 5

Outline

1. The model
   - Geometry
   - Mathematical model
2. Numerical methods for the Schrödinger-Poisson block
   - Iterative schemes
   - Solvers for Schrödinger and Poisson
3. Solvers for the BTE block
   - Adimensionalizations
   - Time discretization
   - Linear advection
   - PWENO interpolations
4. Experiments
   - Simplifying assumptions
   - Equilibria
   - Time-dependent simulations
   - Newton vs. Gummel
   - Plasma oscillations

Numerical methods

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.

Numerical methods

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.

Numerical methods

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.

Outline

1. The model
   - Geometry
   - Mathematical model
2. Numerical methods for the Schrödinger-Poisson block
   - Iterative schemes
   - Solvers for Schrödinger and Poisson
3. Solvers for the BTE block
   - Adimensionalizations
   - Time discretization
   - Linear advection
   - PWENO interpolations
4. Experiments
   - Simplifying assumptions
   - Equilibria
   - Time-dependent simulations
   - Newton vs. Gummel
   - Plasma oscillations

Wave-vector space

Two different adimensionalizations are proposed for the wave-vector space. Magnitudes with tilde are meant with dimension.

Cartesian coordinates

$$(\tilde{k}_x, \tilde{k}_y) = \frac{\sqrt{m_e \kappa_B T_L}}{\hbar} (k_x, k_y) .$$

Ellipsoidal coordinated

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

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

Wave-vector space

Two different adimensionalizations are proposed for the wave-vector space. Magnitudes with tilde are meant with dimension.

Cartesian coordinates

$$(\tilde{k}_x, \tilde{k}_y) = \frac{\sqrt{m_e \kappa_B T_L}}{\hbar} (k_x, k_y)\,.$$

Ellipsoidal coordinated

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

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

Wave-vector space

Two different adimensionalizations are proposed for the wave-vector space. Magnitudes with tilde are meant with dimension.

Cartesian coordinates

$$(\tilde{k}_x, \tilde{k}_y) = \frac{\sqrt{m_e \kappa_B T_L}}{\hbar} (k_x, k_y) \, .$$

Ellipsoidal coordinated

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 in cartesian coordinates

Let the flux coefficients

$$\begin{aligned} a^1_\nu(k) &= C^V v_{x,\nu}(k) \\ a^2_{\nu,p}(x) &= -C^V \frac{\partial \epsilon^{pot}_{x,\nu}}{\partial x}(x). \end{aligned}$$

Transport form

The BTE in transport form reads

$$\frac{\partial f_{\nu,p}}{\partial t} + a^1_\nu \frac{\partial f_{\nu,p}}{\partial x} + a^2_{\nu,p} \frac{\partial f_{\nu,p}}{\partial k} = \mathcal{Q}_{\nu,p}[f].$$

Conservation-law form

The BTE in conservation-law form reads

$$\frac{\partial f_{\nu,p}}{\partial t} + \frac{\partial}{\partial x}\left[a^1_\nu f_{\nu,p}\right] + \frac{\partial}{\partial k}\left[a^2_{\nu,p} f_{\nu,p}\right] = \mathcal{Q}_{\nu,p}[f].$$

BTE in cartesian coordinates

Let the flux coefficients

$$
\begin{aligned}
a^1_\nu(k) &= C^V v_{x,\nu}(k) \\
a^2_{\nu,p}(x) &= -C^V \frac{\partial \epsilon^{pot}_{x,\nu}}{\partial x}(x).
\end{aligned}
$$

Transport form

The BTE in transport form reads

$$\frac{\partial f_{\nu,p}}{\partial t} + a^1_\nu \frac{\partial f_{\nu,p}}{\partial x} + a^2_{\nu,p} \frac{\partial f_{\nu,p}}{\partial k} = \mathcal{Q}_{\nu,p}[f].$$

Conservation-law form

The BTE in conservation-law form reads

$$\frac{\partial f_{\nu,p}}{\partial t} + \frac{\partial}{\partial x}\left[a^1_\nu f_{\nu,p}\right] + \frac{\partial}{\partial k}\left[a^2_{\nu,p} f_{\nu,p}\right] = \mathcal{Q}_{\nu,p}[f].$$

BTE in cartesian coordinates

Let the flux coefficients

$$\begin{aligned} a^1_{\nu}(k) &= C^V v_{x,\nu}(k) \\ a^2_{\nu,p}(x) &= -C^V \frac{\partial \epsilon^{pot}_{x,\nu}}{\partial x}(x). \end{aligned}$$

Transport form

The BTE in transport form reads

$$\frac{\partial f_{\nu,p}}{\partial t} + a^1_{\nu} \frac{\partial f_{\nu,p}}{\partial x} + a^2_{\nu,p} \frac{\partial f_{\nu,p}}{\partial k} = \mathcal{Q}_{\nu,p}[f].$$

Conservation-law form

The BTE in conservation-law form reads

$$\frac{\partial f_{\nu,p}}{\partial t} + \frac{\partial}{\partial x}\left[a^1_{\nu} f_{\nu,p}\right] + \frac{\partial}{\partial k}\left[a^2_{\nu,p} f_{\nu,p}\right] = \mathcal{Q}_{\nu,p}[f].$$

BTE in ellipsoidal coordinates

Let the flux coefficients

$$
\begin{aligned}
a^1_\nu(w,\phi) &= C^V \frac{\sqrt{2w(1+\alpha_\nu w)}\cos(\phi)}{\sqrt{m_{x,\nu}}} \frac{1}{1+2\alpha_\nu w} \\
a^2_{\nu,p}(x,w,\phi) &= -C^V \frac{\partial \epsilon_{\nu,p}}{\partial x}(x) \frac{1}{1+2\alpha_\nu w} \frac{\sqrt{2w(1+\alpha_\nu w)}\cos(\phi)}{\sqrt{m_{x,\nu}}} \\
a^3_{\nu,p}(x,w,\phi) &= C^V \frac{\partial \epsilon_{\nu,p}}{\partial x}(x) \frac{\sin(\phi)}{\sqrt{m_{x,\nu}}\sqrt{2w(1+\alpha_\nu w)}}.
\end{aligned}
$$

Conservation-law form

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

BTE in ellipsoidal coordinates

Let the flux coefficients

$$
\begin{aligned}
a^1_\nu(w,\phi) &= C^V \frac{\sqrt{2w(1+\alpha_\nu w)}\cos(\phi)}{\sqrt{m_{x,\nu}}}\frac{1}{1+2\alpha_\nu w} \\
a^2_{\nu,p}(x,w,\phi) &= -C^V \frac{\partial \epsilon_{\nu,p}}{\partial x}(x)\frac{1}{1+2\alpha_\nu w}\frac{\sqrt{2w(1+\alpha_\nu w)}\cos(\phi)}{\sqrt{m_{x,\nu}}} \\
a^3_{\nu,p}(x,w,\phi) &= C^V \frac{\partial \epsilon_{\nu,p}}{\partial x}(x)\frac{\sin(\phi)}{\sqrt{m_{x,\nu}}\sqrt{2w(1+\alpha_\nu w)}}.
\end{aligned}
$$

Conservation-law form

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

Outline

1. The model
    - Geometry
    - Mathematical model
2. Numerical methods for the Schrödinger-Poisson block
    - Iterative schemes
    - Solvers for Schrödinger and Poisson
3. Solvers for the BTE block
    - Adimensionalizations
    - Time discretization
    - Linear advection
    - PWENO interpolations
4. Experiments
    - Simplifying assumptions
    - Equilibria
    - Time-dependent simulations
    - Newton vs. Gummel
    - Plasma oscillations

Runge-Kutta vs. splitting

We propose two discretizations for the time, following the choice between conservation-law form and transport form.

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

Runge-Kutta vs. splitting

We propose two discretizations for the time, following the choice between conservation-law form and transport form.

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

Splitting schemes

Time splitting

If the BTE is written in transport form, then we advance in time by time splitting schemes:

$$\begin{aligned} \frac{\partial f_{\nu,p}}{\partial t} + C^V\left\{\epsilon^{tot}, f_{\nu,p}\right\} &= 0 \\ \frac{\partial f_{\nu,p}}{\partial t} &= \mathcal{Q}_{\nu,p}[f]. \end{aligned}$$

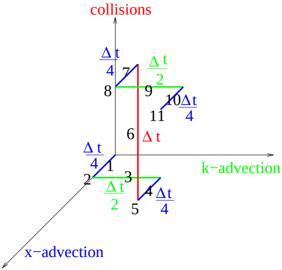
Splitting schemes

Dimensional splitting

Apart from time splitting, we also split the phase-space:

$$\begin{aligned}\frac{\partial f_{\nu,p}}{\partial t} + a^1_\nu \frac{\partial f_{\nu,p}}{\partial x} &= 0\\ \frac{\partial f_{\nu,p}}{\partial t} + a^2_{\nu,p} \frac{\partial f_{\nu,p}}{\partial k} &= 0\end{aligned}$$

The overall scheme is summarized in the following figure.



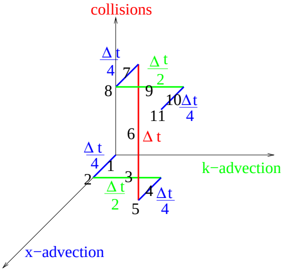
Splitting schemes

Dimensional splitting

Apart from time splitting, we also split the phase-space:

$$\begin{aligned} \frac{\partial f_{\nu,p}}{\partial t} + a^1_\nu \frac{\partial f_{\nu,p}}{\partial x} &= 0 \\ \frac{\partial f_{\nu,p}}{\partial t} + a^2_{\nu,p} \frac{\partial f_{\nu,p}}{\partial k} &= 0 \end{aligned}$$

The overall scheme is summarized in the following figure.



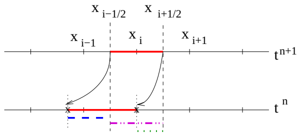
Outline

1. The model
   - Geometry
   - Mathematical model
2. Numerical methods for the Schrödinger-Poisson block
   - Iterative schemes
   - Solvers for Schrödinger and Poisson
3. Solvers for the BTE block
   - Adimensionalizations
   - Time discretization
   - Linear advection
   - PWENO interpolations
4. Experiments
   - Simplifying assumptions
   - Equilibria
   - Time-dependent simulations
   - Newton vs. Gummel
   - Plasma oscillations

Linear advection

Flux Balance Method:

Total mass conservation is forced. It is based on the idea of following backward the characteristics, but integral values are taken instead of point values:



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

FLUX BALANCE METHOD means evualuating the flux at time t^{n+1} from a balance of fluxes at previous time t^n :

-·-·-·- the average along the purple segment

- - - - plus the average along the blue segment

········· minus the average along the green segment

Outline

1. The model
   - Geometry
   - Mathematical model
2. Numerical methods for the Schrödinger-Poisson block
   - Iterative schemes
   - Solvers for Schrödinger and Poisson
3. Solvers for the BTE block
   - Adimensionalizations
   - Time discretization
   - Linear advection
   - PWENO interpolations
4. Experiments
   - Simplifying assumptions
   - Equilibria
   - Time-dependent simulations
   - Newton vs. Gummel
   - Plasma oscillations

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.

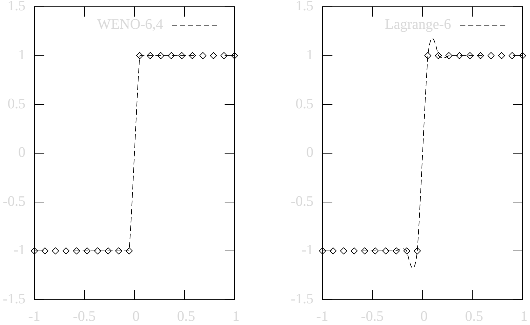


Figure: Left: PWENO interpolation. Right: Lagrange interpolation.

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.

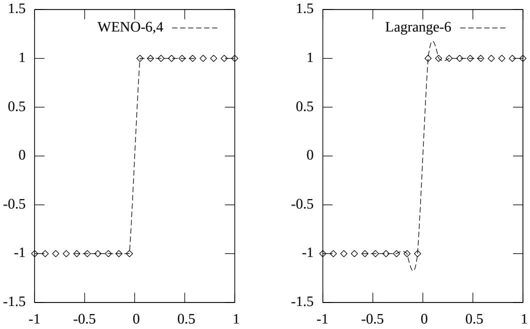
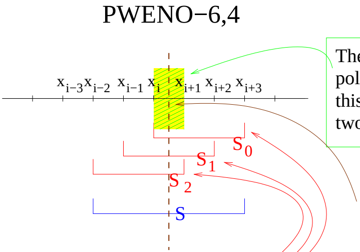


Figure: Left: PWENO interpolation. Right: Lagrange interpolation.

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:


PWENO−6,4
x_{i-3} x_{i-2} x_{i-1} x_i x_{i+1} x_{i+2} x_{i+3}
S_0
S_1
S_2
S

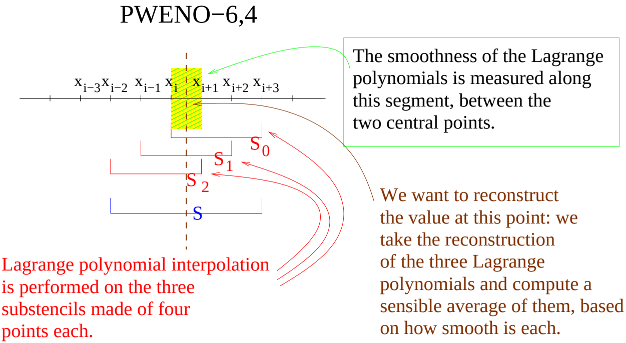
The smoothness of the Lagrange polynomials is measured along this segment, between the two central points.

We want to reconstruct the value at this point: we take the reconstruction of the three Lagrange polynomials and compute a sensible average of them, based on how smooth is each.

Lagrange polynomial interpolation is performed on the three substencils made of four points each.

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 smoothness of the Lagrange polynomials is measured along this segment, between the two central points.

We want to reconstruct the value at this point: we take the reconstruction of the three Lagrange polynomials and compute a sensible average of them, based on how smooth is each.

Lagrange polynomial interpolation is performed on the three substencils made of four points each.

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:

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

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

1. The model
   - Geometry
   - Mathematical model
2. Numerical methods for the Schrödinger-Poisson block
   - Iterative schemes
   - Solvers for Schrödinger and Poisson
3. Solvers for the BTE block
   - Adimensionalizations
   - Time discretization
   - Linear advection
   - PWENO interpolations
4. Experiments
   - Simplifying assumptions
   - Equilibria
   - Time-dependent simulations
   - Newton vs. Gummel
   - Plasma oscillations

Collision operator

Results are presented for the the DG MOSFET in the one-valley, parabolic-band approximation. Moreover, the complete collision operator is substituted by a simple relaxation-time operator:

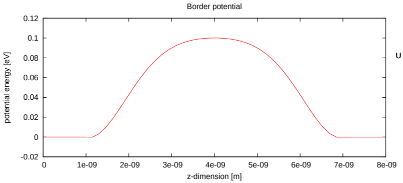
$$\mathcal{Q}_p f = \frac{1}{\tau}\left(\rho_p M - f_p\right).$$

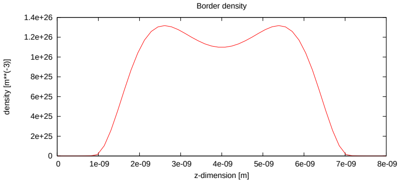
The goal of this work is just the setting up of numerical tools for a more profound and realistic simulation.
A parallel code in the most realistic case is being implemented.

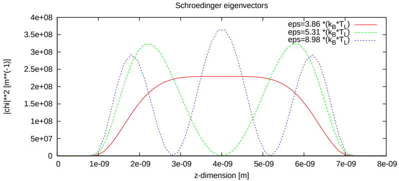
Outline

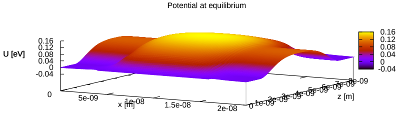
1. The model
   - Geometry
   - Mathematical model
2. Numerical methods for the Schrödinger-Poisson block
   - Iterative schemes
   - Solvers for Schrödinger and Poisson
3. Solvers for the BTE block
   - Adimensionalizations
   - Time discretization
   - Linear advection
   - PWENO interpolations
4. Experiments
   - Simplifying assumptions
   - Equilibria
   - Time-dependent simulations
   - Newton vs. Gummel
   - Plasma oscillations

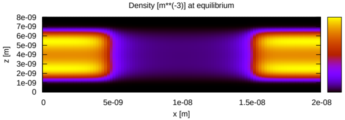
Thermodynamical equilibrium: one-valley case

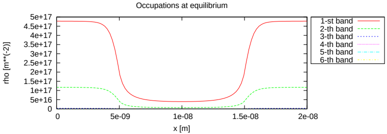


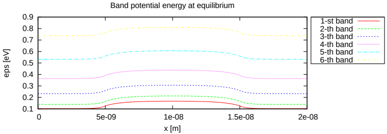












Outline

1. The model
   - Geometry
   - Mathematical model
2. Numerical methods for the Schrödinger-Poisson block
   - Iterative schemes
   - Solvers for Schrödinger and Poisson
3. Solvers for the BTE block
   - Adimensionalizations
   - Time discretization
   - Linear advection
   - PWENO interpolations
4. Experiments
   - Simplifying assumptions
   - Equilibria
   - Time-dependent simulations
   - Newton vs. Gummel
   - Plasma oscillations

Long-time behavior

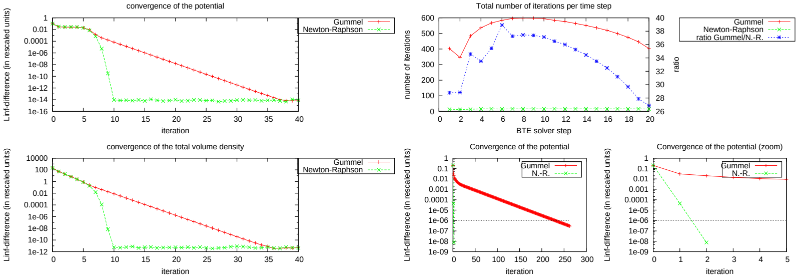
We propose now some results relative to the long-time behavior of the system.

Outline

1. The model
   - Geometry
   - Mathematical model
2. Numerical methods for the Schrödinger-Poisson block
   - Iterative schemes
   - Solvers for Schrödinger and Poisson
3. Solvers for the BTE block
   - Adimensionalizations
   - Time discretization
   - Linear advection
   - PWENO interpolations
4. Experiments
   - Simplifying assumptions
   - Equilibria
   - Time-dependent simulations
   - Newton vs. Gummel
   - Plasma oscillations

Number of iterations

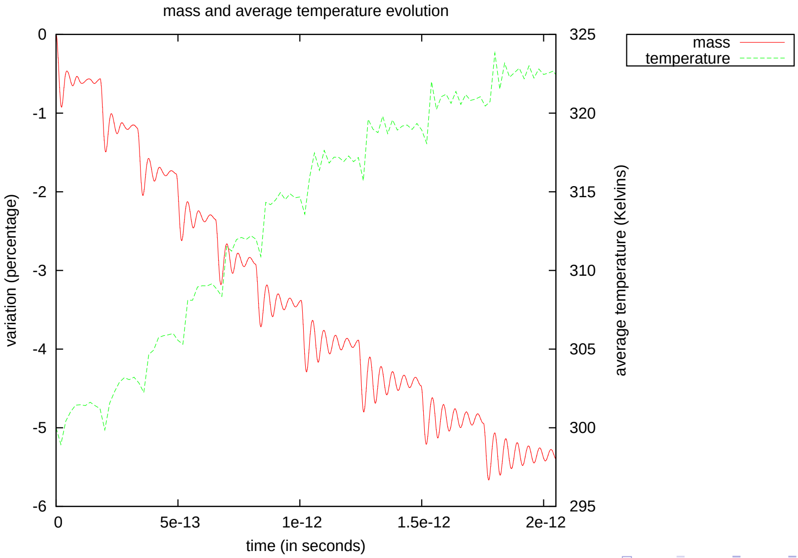
Newton schemes require much less iterations than Gummel in order to compute the thermodynamical equilibrium.



Outline

1. The model
   - Geometry
   - Mathematical model
2. Numerical methods for the Schrödinger-Poisson block
   - Iterative schemes
   - Solvers for Schrödinger and Poisson
3. Solvers for the BTE block
   - Adimensionalizations
   - Time discretization
   - Linear advection
   - PWENO interpolations
4. Experiments
   - Simplifying assumptions
   - Equilibria
   - Time-dependent simulations
   - Newton vs. Gummel
   - Plasma oscillations

Mass and temperature oscillations



Numerically-computed oscillations

The plasma frequency is given by

$$\omega_p = \sqrt{\frac{q^2 N_e}{\varepsilon_R \varepsilon_0 m_*}}.$$

N_D^{high} $(\times 10^{26} m^{-3})$	ε_R	$m_\star$	N_e $(\times 10^{26} m^{-3})$	ω_{num} $(\times 10^{14} s^{-1})$	ω_p $(\times 10^{14} s^{-1})$	Ratio $\frac{\omega_{num}}{\omega_{ref}}$	Expected Ratio
1	11.7	0.5	.400	$\omega_{ref} = 1.344$	1.475	1	/
2	11.7	0.5	.783	2.051	2.064	1.52	$\sqrt{2}$
4	11.7	0.5	1.544	2.813	2.899	2.09	2
1	5.85	0.5	.400	1.848	2.086	1.37	$\sqrt{2}$