Plasma Simulations with Basilisk

Vladimir Kolobov, Robert Arslanbekov,
Dmitry Levko

CFD Research Corporation
The University of Alabama in Huntsville

Basilisk/Gerris Users' Meeting 2019

Paris, France

June 18, 2019

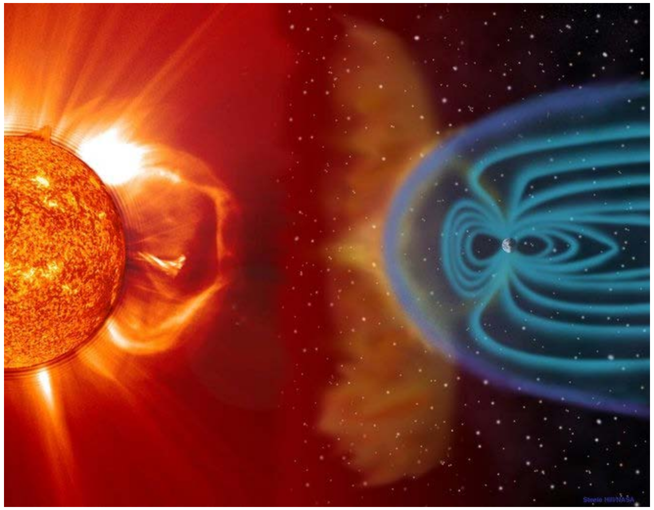
Center for Predictive Control of Plasma Kinetics: Multi-Phase and Bounded Systems

Department of Energy Plasma Science Center

DOE Plasma Science Center
Control of Plasma Kinetics

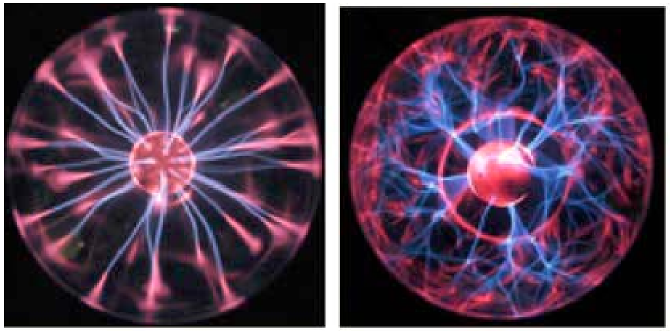
What is Plasma ?

The Greek word plasma means “formed or molded”. In 19th century, physiologist Jan Evangelista Purkinje introduced use of plasma to denote the clear fluid which remains after removal of all the corpuscular material in blood. In 1922, physicist Irving Langmuir proposed that the electrons, ions and neutrals in an ionized gas could similarly be considered as corpuscular material entrained in some kind of fluid medium and called this entraining medium plasma. However it turned out that there is no “fluid medium” entraining the electrons, ions, and neutrals in an ionized gas. Ever since, plasma scientists have to explain that they were not studying blood! – *after Paul M. Bellan, Fundamentals of Plasma Physics*


Steele Hill/NASA

- **Plasma Is a quasi-neutral mixture of electrons, ions and neutrals**
- **Key properties of plasma are due to large difference of electron and ion mass**

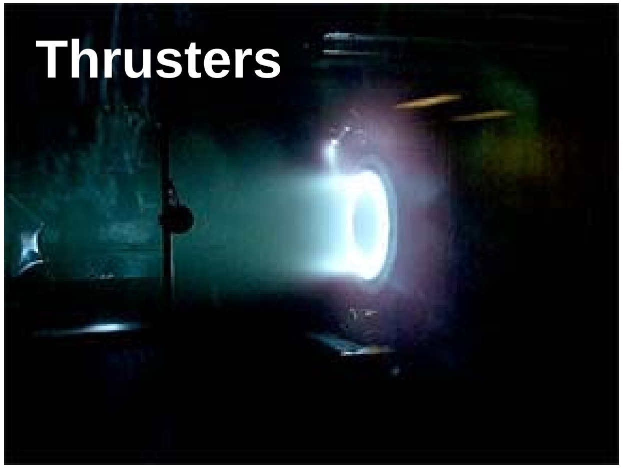
- **Due to its non-equilibrium and nonlinear nature, plasma is prone to instabilities self-organization**



The motion of filaments in a plasma ball - Campanell et al, Phys Plasmas **17**, 053507 (2010)

Plasma Technologies

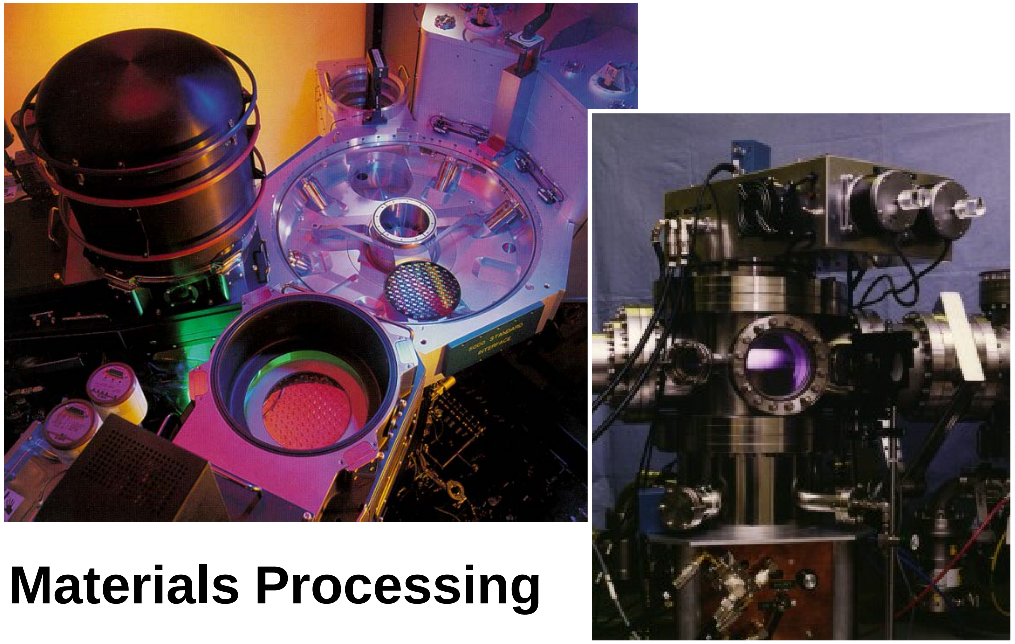
"The XXI century will be not only the century of informatics, biology and space exploration, but also the century of plasma technologies" A.I.Morozov 2006



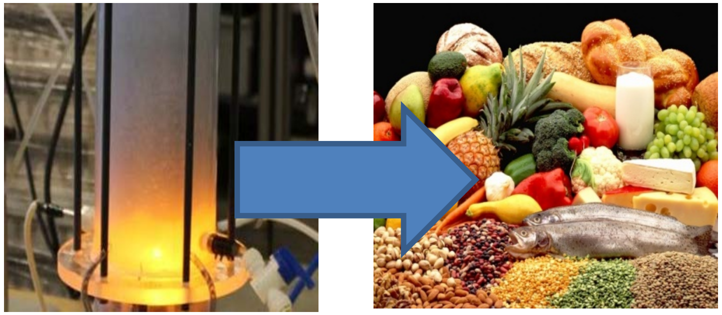
Thrusters

Lighting

Mecicine



Materials Processing



Plasma-activated liquids (water & solutions) are the key for large volume plasma agriculture.

Kinetic and Hydrodynamic Models

Hydrodynamic

Particles are described by five characteristics:

1. Density $n(\mathbf{r},t)$

2. Mean directed velocity, $\mathbf{v}(\mathbf{r},t)$

3. Temperature, $T(\mathbf{r},t)$

They depend on 4 scalar arguments – 3 spatial coordinates and time.

Kinetic

The only characteristic is the velocity distribution function (VDF) – the particle density in phase space

$$f(\mathbf{v},\mathbf{r},t)$$

It depends on 7 scalar arguments – 3 spatial coordinates, 3 velocity components and time.

Kinetic description is far more detailed and inevitably much more complicated!

Many problems in gas discharge physics can only be understood at kinetic level

Challenges of Plasma Simulations

Four major categories:

- **high spatial resolution is required in certain parts of the system (sheaths, shock layers, etc.) – using spatially uniform mesh results in a very large problem size**
- **kinetic solvers (both particle and grid based) are expensive computationally and require huge computer memory or large number of particles**
- **disparate time scales for electron and ions require long simulation time if both the electron and ion time scales need to be resolved**
- **coupling electromagnetics with charge transport results in highly non-linear problems (especially when ionization processes are important).**

Discrete Velocity Method

$$\frac{\partial f}{\partial t} + \nabla_{\mathbf{r}} \cdot (\xi f) = -\nu f + \Phi$$

$$\nu = \int_{R^3} \int_{S^2} \sigma(g,\Theta) g f(\xi'_\beta) d\xi_\beta d\Omega$$

$$\Phi = \int_{R^3} \int_{S^2} \sigma(g,\Theta) g f(\xi') f(\xi'_\beta) d\xi_\beta d\Omega$$

σ- differential collision cross-section,

$g = |\xi - \xi_\beta|$ relative velocity

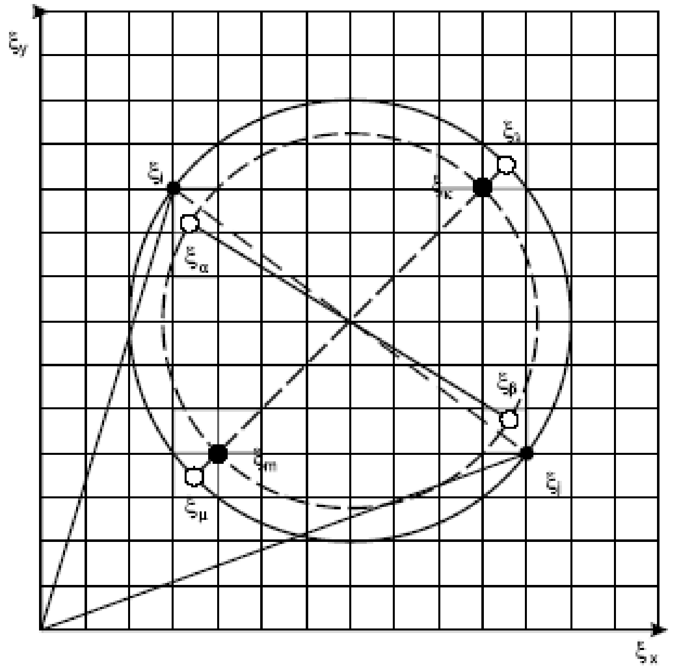
$\mathbf{\Omega} = \xi / \xi$ is a unit vector on the sphere

Cartesian mesh in velocity space

Conservative Projection Method

Boltzmann Collision Integral

Collisional Sphere in Velocity Space



conservation of momentum and energy in each collision

$$\xi + \xi_1 = \xi' + \xi'_1,$$

$$|\xi|^2 + |\xi_1|^2 = |\xi'|^2 + |\xi'_1|^2$$

Vlasov & Fokker-Planck Kinetic Equations

Vlasov Equation for Collisionless Plasma

$$\frac{\partial f}{\partial t} + \nabla_6 \cdot (\dot{\mathbf{z}} f) = 0$$

Conservation of phase-space volume

$$\mathbf{z} = (\mathbf{x}, \mathbf{v})$$

$$\dot{\mathbf{z}} = (\mathbf{v}, \mathbf{a})$$

$$\nabla_6 \cdot (\dot{\mathbf{z}}) = 0$$

- Non-Magnetized Plasma
- Magnetized Plasma (6D->5D)

$$\mathbf{a} = \frac{e}{m}\left(\mathbf{E} + \mathbf{v} \times \mathbf{B}\right)$$

Fokker-Planck Collisions:

- **Small change of energy and momentum**:
  - Coulomb interactions among charged particles
  - (quasi)-elastic collisions of electrons with neutrals
  - Excitation of rotational (and vibrational ?) levels of molecules
  - Continuum energy loss for fast electrons
- **Wave-Particle Interactions (quasi-linear theory)**

$$\frac{\partial f}{\partial t} + \nabla_{\mathbf{r}} \cdot (\mathbf{v} f) + \nabla_{\mathbf{v}} \cdot \left[\mathbf{A} f + \mathbf{D} \nabla f\right] = 0$$

To Split or not to Split (Phase Space) ?

- **Non-splitting**: 5B points for 6D phase space (Deriaz & Peirani, Six-Dimensional Adaptive Simulation of the Vlasov Equations using a Hierarchical Basis (2018)
- **Full Splitting**: Six one-dimensional advection for each dimension in phase space (6R^1): 128^{3}x128^3 cells (Tanaka et al, Multidimensional Vlasov–Poisson Simulations with High-order Monotonicity- and Positivity-preserving Schemes (Astrophysical J. (2017))

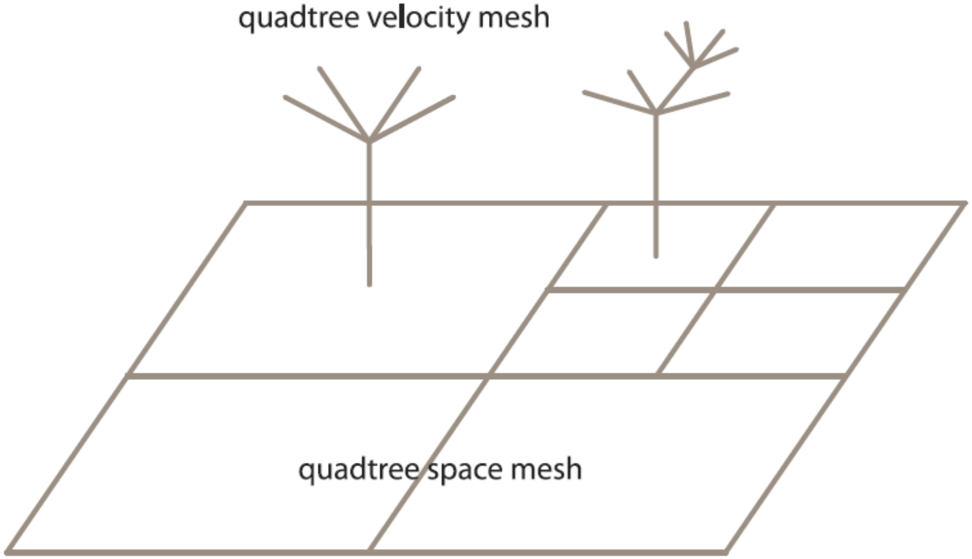
$$\begin{aligned} f(\boldsymbol{x}, \boldsymbol{v}, t^{n+1}) = {} & T_{v_z}(\Delta t/2) T_{v_y}(\Delta t/2) T_{v_x}(\Delta t/2) \\ & \times T_x(\Delta t) T_y(\Delta t) T_z(\Delta t) \\ & \times T_{v_z}(\Delta t/2) T_{v_y}(\Delta t/2) T_{v_x}(\Delta t/2) f(\boldsymbol{x}, \boldsymbol{v}, t^n). \end{aligned}$$

- **Splitting Velocity & Configuration Spaces:**

$$\frac{\partial f^*}{\partial t} + \mathrm{div}_{\mathbf{r}}\left(\xi f^*\right) = 0 \qquad (1)$$
$$f^*(0,\mathbf{r},\xi) = f^n(\mathbf{r},\xi)$$

$$\frac{\partial f^{**}}{\partial t} + \mathrm{div}_{\xi}\left(\mathbf{a} f^{**}\right) = I(f^*, f^*) \qquad (2)$$
$$f^{**}(0,\mathbf{r},\xi) = f^*(\Delta t,\mathbf{r},\xi)$$

Tree-of-Trees Data Structure in Gerris



Tree-based *ξ* meshes “grown” in *r* cells, representing the concept of a tree-of-trees (ToT) data structure for a 2D2V case.

VDFs are stored in *r*- and ξ-cell centers. Advection in *r*-space requires calculating normal fluxes across cell faces of neighboring *r*-cells

R R Arslanbekov, V I Kolobov, and A A Frolova, Kinetic solvers with adaptive mesh in phase space, Phys. Rev. E 88 (2013) 063301

Cartesian Mesh in Velocity Space is Suitable for Bi-Linear Boltzmann Collision Integrals

It is possible to have different topology ξ-grids in each r-space cells: velocity space of different sizes and positions

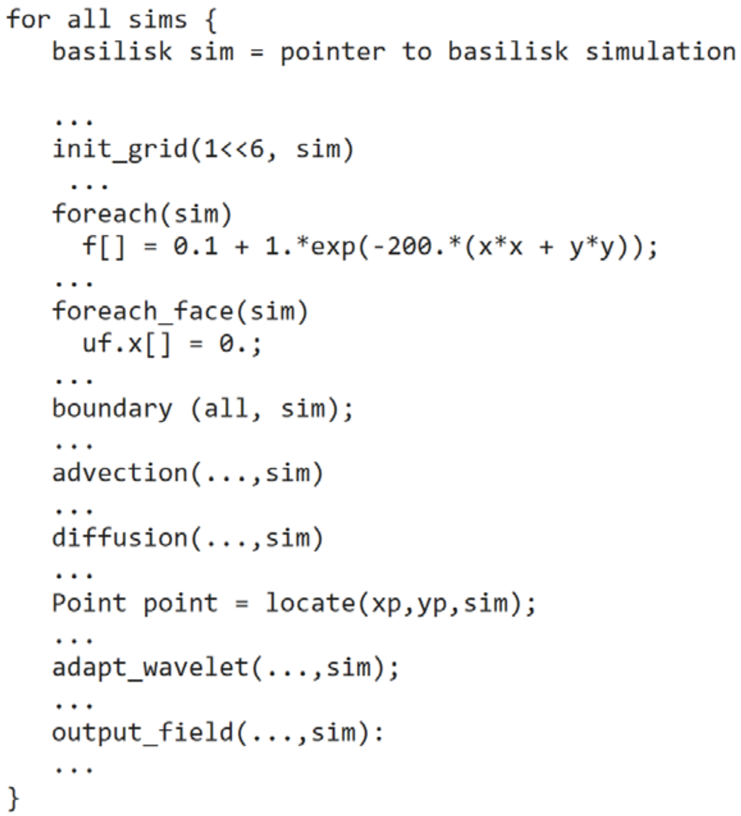
For efficient implementation of the advection operator, it is desirable to have similar topology ξ-grids so that each ξ-space cell in r-space cell can find a corresponding leaf, parent or children cell in neighboring r-cells

Such implementation allows improved conservation of (at least to a second degree of accuracy) when advecting VDF from one r-space cell to another

How to run multiple instances of basilisk ?

For a given set of N cells in external application (e.g., 1D Poisson solver), we would like to create N "TREE" grids. Each such TREE grid should be accessible with a pointer (like sim pointer in gerris) and have its own set of objects and events (scalars, face_vectors, init_grid, advection, diffusion, etc.).

A sample simulation flow :



```
for all sims {
   basilisk sim = pointer to basilisk simulation

   ...
   init_grid(1<<6, sim)
    ...
   foreach(sim)
     f[] = 0.1 + 1.*exp(-200.*(x*x + y*y));
   ...
   foreach_face(sim)
     uf.x[] = 0.;
   ...
   boundary (all, sim);
   ...
   advection(...,sim)
   ...
   diffusion(...,sim)
   ...
   Point point = locate(xp,yp,sim);
   ...
   adapt_wavelet(...,sim);
   ...
   output_field(...,sim):
   ...
}
```

All new sim objects will have the same set of events (run, outputs), boundary conditions, tracers, etc. and can be created by cloning one basilisk setup (the same we did in gerris).

However, the grids and solutions will be different in each sim object.

The sim objects should be able to be destroyed and created on the fly (when N changes).

Is it doable in the current version of basilisk ?

Spherical Coordinates in Velocity Space

In velocity space, collision operators are close to diagonal in spherical coordinate system, the force term is nearly diagonal in cylindrical coordinate system.

$$\frac{\partial f}{\partial t} + v\mu\frac{\partial f}{\partial x} - \frac{eE}{m}\left(\mu\frac{\partial f}{\partial v} + \frac{1-\mu^2}{v}\frac{\partial f}{\partial \mu}\right) = \frac{1}{v^2}\frac{\partial}{\partial v}\left(v^2 Ff\right) + D_\mu \frac{\partial}{\partial \mu}\left(\left(1-\mu^2\right)\frac{\partial f}{\partial \mu}\right) + I_e$$

- Fokker-Planck-type for the fast electrons (small-angle scattering and continuous energy loss in collisions)
- Boltzmann-type collision operator for elastic scattering of slow electrons on atoms

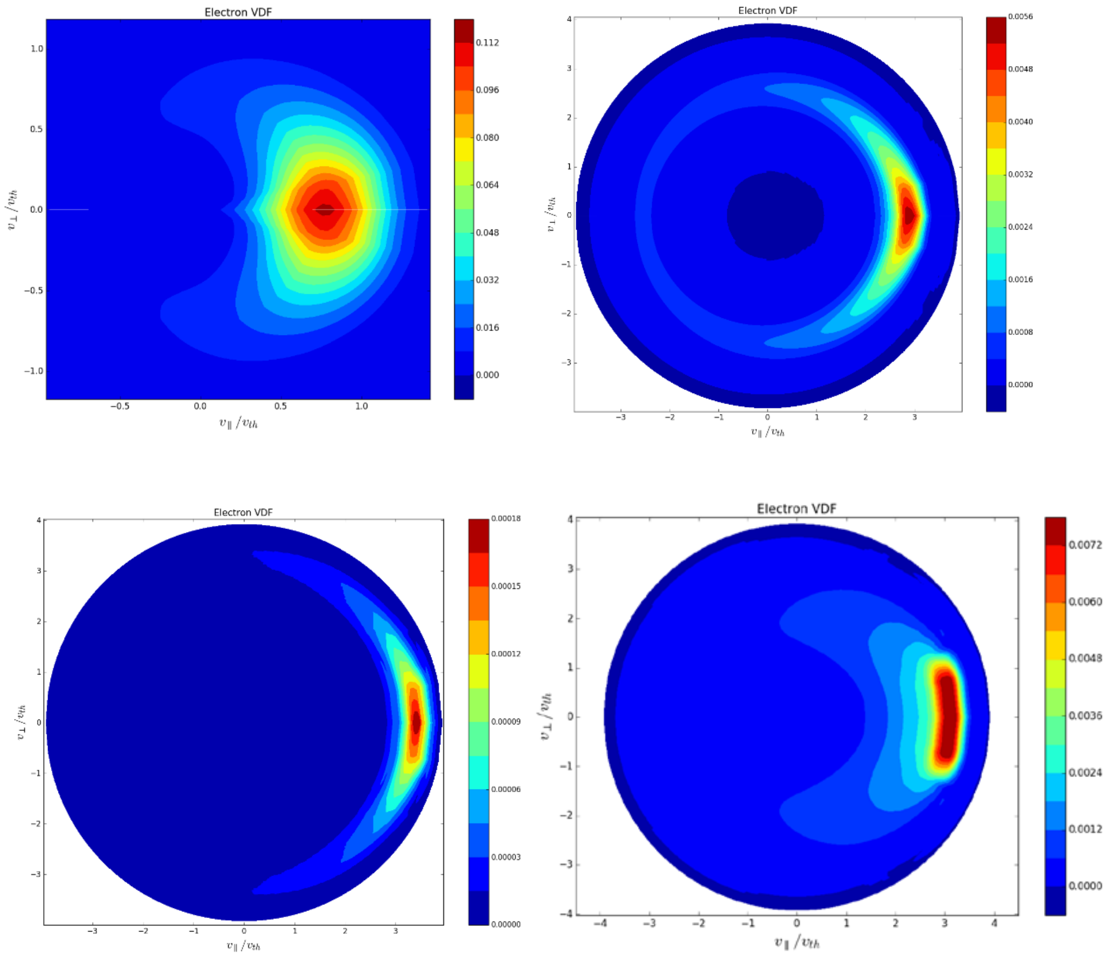
$$I_e(v,\mu) = \nu_e(\mathrm{v})\left[\int_{-1}^{1} p(\mu,\mu')f(\mu')d\mu' - f(v,\mu)\right]$$

$$p(\mu,\mu') = \int_0^{2\pi} p(\cos\Theta)d\varphi$$ is the azimuthally integrated phase function

Two-stream model (1d velocity space): Lorentz operator corresponds to “forward-backward” model

$$I = \nu(v)\cdot[f(v) - f(-v)]$$

Acceleration, Scattering, and Energy Loss



1d2v problem of electron acceleration in spatially inhomogeneous electric field, elastic scattering in collisions and energy loss in inelastic collisions.

$$eE / F_0 = 5$$

Deceleration length

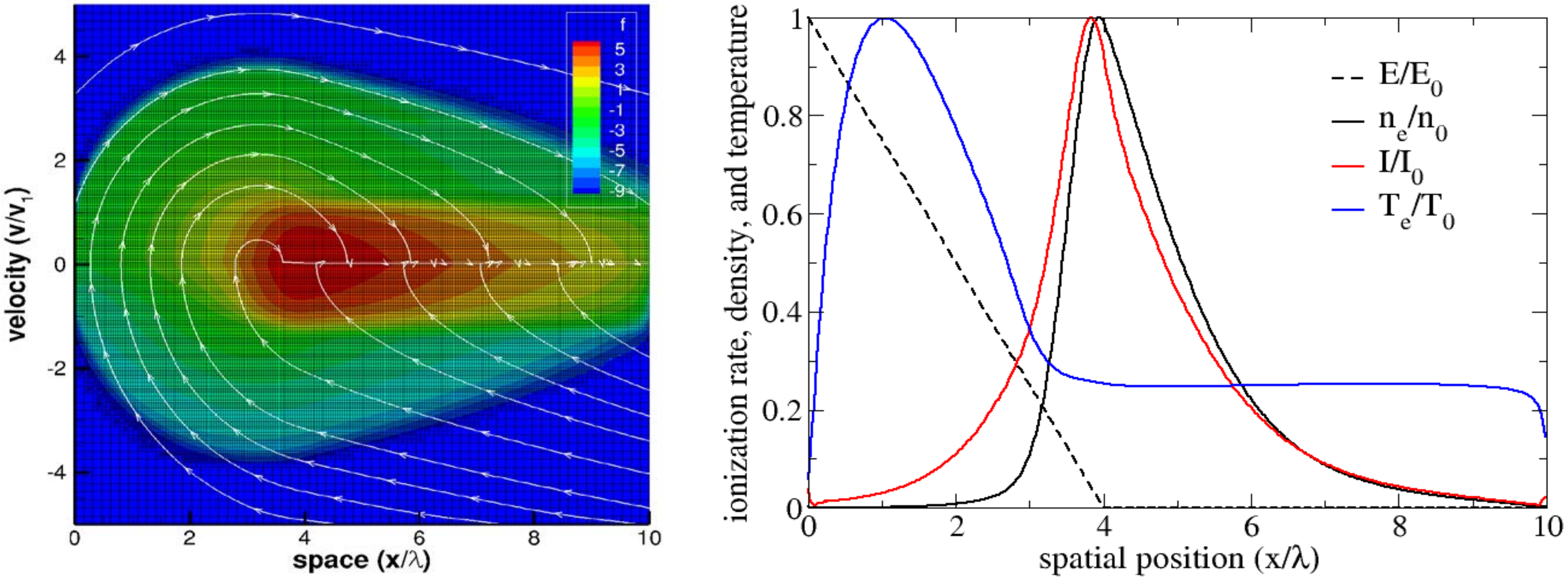
$$\Lambda = U_c / F_0$$

Potential drop U_c

Electron VDF $Y = v^2 f$ at different spatial positions, x/L = 0.1, 0.2. 0.4. 0.8, for d/L =0.4 and $\lambda = 0.01L$

Two-Stream Model: Electron Runaway

Two stream model: acceleration, scattering, energy loss, and ionization.
Electrons are injected at the left boundary (cathode) with a Maxwellian distribution, move along characteristics (streamlines) and get absorbed at the anode.



Streamlines (characteristics), adapted computational mesh, and calculated VDF (colour map in log scale) in the 1d1v phase space (left) and spatial distributions of normalized macro-parameters (right).

- Asymmetry between positive and negative v: slow electrons have near-isotropic VDF, whereas fast (runaway) electrons have strongly anisotropic VDF .
- Axial distributions of plasma parameters correspond to the typical distributions observed in the cathode region of short glow discharges without positive column.

Tensor Diffusion

In Basilisk, it is easy to set up an anisotropic diffusion when the diffusion tensor is diagonal.

Basilisk can use larger stencils compared to gerris, so it might be able to solve anisotropic diffusion problems.

Does anybody has an example of solving a tensor diffusion problem ?

We are looking for a sample of a symmetric diffusion tensor with non-zero off-diagonal components, when cross-derivatives are important.

The question is how to make use of the extended 5x5 stencil.
Or it can be done on the standard 3x3 stencil ?
The 5x5 stencil is limited (does not work for face vectors).

The new scheme with the tensor D has to remain implicit (multigrid implicit)

Fluid Model for Collision-Dominated Plasma

- A set of equations for the electron and ion densities, and Poisson equation for electrostatic potential
- Ion transport is included
- Multiple species ions are allowed by introducing species specific mass, diffusion coefficients, and ionization rates
- Local field approximation for ionization rate
- Electron Energy Transport with account for electron thermal conductivity
- Ionization rate a function of electron temperature

$$\frac{\partial n_e}{\partial t} + div\left(\vec{\Gamma}_e\right) = \nu n_e$$

$$\vec{\Gamma}_e = -\mu_e \vec{E} n_e - D_e \nabla n_e$$

$$\Delta\varphi = 4\pi\left(n_e - n_i\right)$$

$$\frac{\partial n_i}{\partial t} + div(\Gamma_i) = \nu n_e$$

$$\vec{\Gamma}_i = \mu_i \vec{E} n_i - D_i \nabla n_i$$

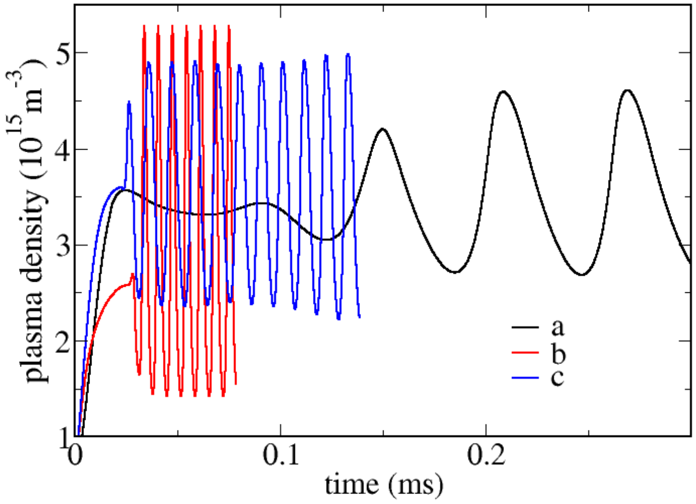
$$\nu = \nu_0 \exp\left[-\frac{E_0}{E}\right]$$

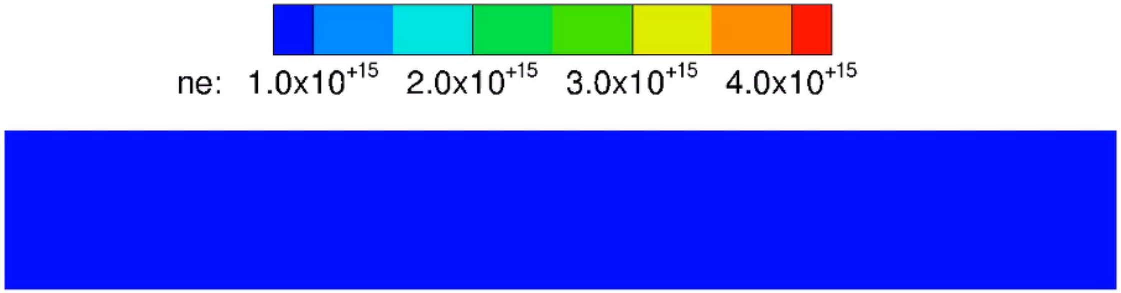
$$\frac{\partial n_\varepsilon}{\partial t} + \nabla\cdot\vec{\Gamma}_\varepsilon = S_\varepsilon \qquad n_\varepsilon = n_e\overline{\varepsilon}$$

$$\vec{\Gamma}_\varepsilon = -\frac{5}{3}\mu_e \vec{E} n_\varepsilon - \frac{5}{3} D_e \nabla n_\varepsilon$$

Plasma Stratification

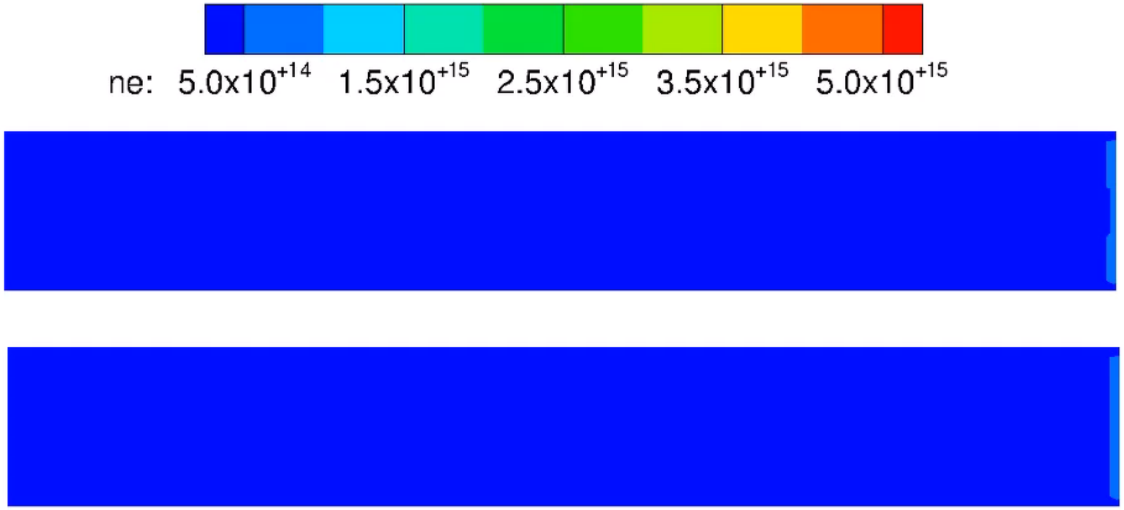
Plasma stratification remains a great challenge for our understanding of electron kinetics and physics of gas discharges





No volume recombination

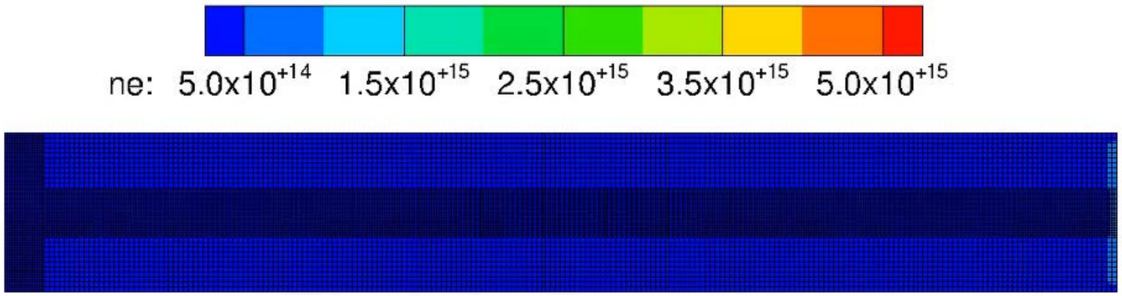
Self-organization at the kinetic level



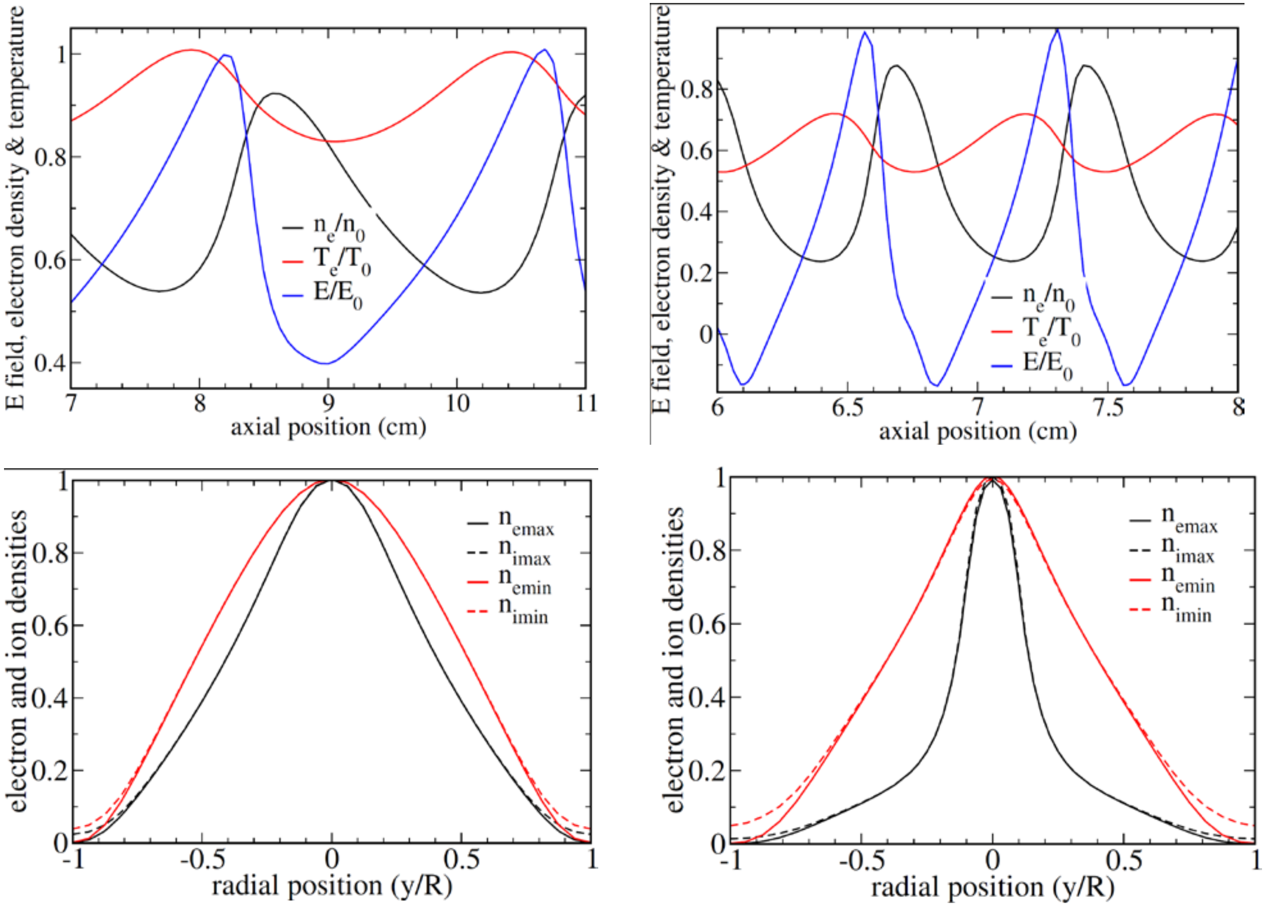
Large volume recombination

Intermediate case

Two types of striations



Dynamically adaptive Cartesian mesh



New implicit coupled solver using fluid plasma model makes these simulations possible

Implicit Solver

How to implement an implicit advection scheme that will match the existing explicit Bell-Collela-Glaz,1989 scheme (2nd order time-space or better).

Given a face advection vector, for each given cell, we will need stencil of weights (approximating the advection operator) from all neighboring cells (of different refinement levels) which will be used to form a global matrix of cell-centered unknowns.

The main goal is to build a proper matrix so that the scheme remains 2nd order accurate in space-time (as the current explicit scheme). This matrix can then be fed into a linear matrix solver (PETSc, Trilinos, etc.)

How to access this information in basilisk (in gerris, we used to use the _stencil routines).

Any suggestions what types of matrix solvers (e.g., direct, iterative, multigrid) are best suited for such problems on TREE grids ?

Has somebody tried it ? A working sample ?