kobra: a new Vlasov code for plasma wall modeling with AMR

S. Konewko[1], S. Van Loo[1]

[1] *Department of Applied Physics, Ghent Univeristy, Ghent, Belgium*

Abstract

The plasma-wall interactions in a fusion device can be modeled as a collisionless problem. When modeling large density gradients particle-in-cell codes suffer from statistical error originating from undersampling the low-density velocity space. Vlasov codes do not have this issue as they evolve the full distribution function. Here, we present a new finite-volume Vlasov code, *kobra*, equipped with adaptive-mesh refinement to reduce computational effort. Currently, the code solves the Vlasov-Poisson equations in full phase space. As a stepping stone towards more comprehensive problems, we present a use case of *kobra*: the 1d1v, electrostatic steady state plasma sheath.

Introduction

Fluid codes are used successfully to model the scrape-off layer (SOL) with relatively modest computing requirements. However, they cannot self-consistently treat the kinetic effects in this region [1]. Typically, a particle-in-cell (PIC) code is used however, this approach suffers from statistical noise when undersampling particles. In contrast, directly solving the Vlasov equation allows one to capture the full velocity distribution, eliminating statistical errors. This gain comes at a computational cost; the six dimensional phase space $(\mathbf{x},\mathbf{v})$ imposes an extraordinary need for memory and computation time. Here, we present a case study of *kobra*, a new finite volume Vlasov-Poisson code with adaptive-mesh refinement (AMR).

Having already benchmarked our code against classic kinetic problems [2], we move to model the plasma-wall interaction. First done in [3], a plasma impinges on an absorbing wall and subject to an oblique magnetic field. While this requires a 1d3v model, we first start with the computationally inexpensive electrostatic plasma sheath in 1d1v. First, we describe the numerical methods used in *kobra* and in the subsequent section, we present kobra's results. We finish with some concluding remarks.

kobra

To model a collisionless plasma kinetically, we need to solve the Vlasov equation. We reduce Maxwell's equations to Gauss's Law and rewrite this in terms of a scalar potential:

$$\frac{\partial f_s}{\partial t}+\mathbf{v}\frac{\partial f_s}{\partial \mathbf{x}}+\frac{q_s\mathbf{E}}{m_s}\frac{\partial f_s}{\partial \mathbf{v}}=0, \qquad \nabla^2\phi=-\sum_s q_s\, n_s(\mathbf{x}). \tag{1}$$

$f_s(\mathbf{x},\mathbf{v},t)$ is the distribution function in phase space, and the subscript s denotes the species. The electric field $\mathbf{E}$ is solved for by taking the gradient of the potential $\mathbf{E} = -\nabla\phi$ and the particle number densities are given by $n_s(\mathbf{x}) = \int f_s(\mathbf{x},\mathbf{v})\, d\mathbf{v}$.

For the Vlasov equation we use a second-order Runge-Kutta scheme for the temporal discretization, while the fluxes are calculated using a second order, upwind scheme with a slope limiter to avoid spurious oscillations in regions with large gradients [4, 5]. For the Poisson equation, we generate an auxiliary grid and implement a full approximation scheme (FAS) multigrid approach with a second-order lexicographic Gauss-Seidel (GS) relaxation scheme to solve for the potential $\phi(\mathbf{x})$ [6].

Furthermore, we implement an AMR scheme with refinement on a cell-by-cell basis [7] where we use the sum of the first two truncation error terms as the refinement criterion [8]. Thus, *kobra* has three modes of operation specified by the Poisson solver and the type of grid used: uniform grid with a GS solver, uniform grid with the FAS multigrid, adaptive grid with the FAS multigrid. For multigrid and AMR runs we use five levels. For the adaptive grid, we refine when a refinement tolerance of 1e-3 is exceeded.

On the core plasma side (left), we use an inflow boundary condition for $v > 0$ and free flow at $v < 0$. At the wall side (right) we assume an absorbing wall, i.e. free-flow, at $v > 0$ and a zero-inflow condition for $v < 0$. We use a floating boundary condition for the Poisson equation on both boundaries as the boundary charge evolves throughout the simulation. After every step, we recalibrate the potential by shifting the distribution setting the left boundary to zero.

Plasma Sheath

We normalize the Vlasov-Poisson equations so that length is scaled with the Debye length (λ_D), velocity with the thermal velocity of each species (v_{Te}, v_{Ti}) and time inversely with the plasma frequency (ω_p^{-1}). Therefore, $t = 2\pi$ corresponds to one plasma period.

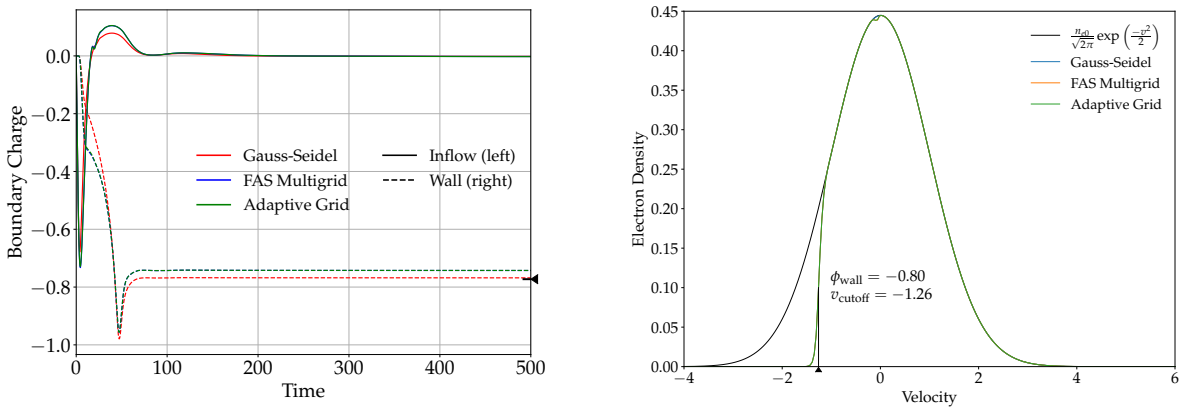
Following the setup given in [3], the distribution functions of the ions and electrons streaming in from a charge-neutral plasma side at $x = 0$ are set as fixed boundary conditions given by

$$f_e(0, v > 0) = \frac{n_{e0}}{\sqrt{2\pi}} \exp\left(-\frac{v^2}{2}\right), \qquad f_i(0, v > 0) = n_{i0}\ \delta(v - v_0) \tag{2}$$

The electrostatic Chodura setup, and consequently our work, represents a unique case of the sheaths discussed in [9]. Adapting these arguments, we find the steady state sheath is strictly a function of the ion velocity v_0. This velocity must satisfy the Bohm limit $v_0 \geq \sqrt{1/m_i} = 0.0233$ ($m_i = 1836$ in our normalized units) for a plasma sheath to form. We choose to set $v_0 = 0.2$ From this, we calculate the wall potential $\phi_{wall} = -0.8$, the electron

normalization $n_{e0} = 1.115$, the steady state wall charge $Q_{wall} = -0.77$, the cutoff velocity $v_{cut} = -1.26$, and the potential $\phi(x)$ at steady state.

We begin with an empty computational domain given by $(x,v) \in [0,10]$ x $[-7.5,7.5]$ with an effective resolution of $(128,384)$. For the ions we use $-0.75 \leq v \leq 0.75$ with the same resolution as for the electrons because the electric force produces a much smaller acceleration for the ions due to their higher mass. We run our simulations to $t = 500$ (~ 80 plasma periods).



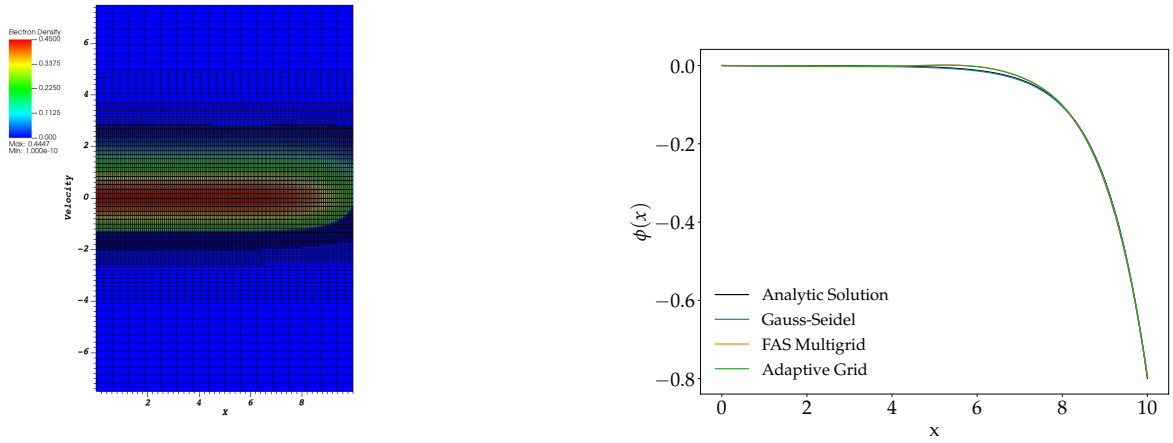
(a) (b)

Figure 1: Panel a) The evolution of the boundary charges. The theoretic Q_{wall} is displayed on the right. Panel b) slices of the electron distribution taken at $x = 0$. In black, an untruncated Maxwellian and v_{cut} for comparison.

As the plasma flows in from the plasma side, fast electrons reach the wall before the ions creating an initial potential difference. Eventually, the ions reach the wall, impart their positive charge, and thus a steady state is established (Figure 1a). The quasi-neutral state reached at the bulk plasma side (left) confirms our electron normalization n_{e0}. Fast electrons will be absorbed by the wall but a portion of the inflow electrons will not have enough kinetic energy to overcome ϕ_{wall}. These are recirculated back to the core plasma. This forms the truncated electron Maxwellian characterized by the cutoff velocity (Figure 1b).

Figure 2a shows the steady state plasma sheath in the full phase space with the adaptive grid overlayed. To the right, Figure 2b shows the converged potential distribution $\phi(x)$ for the different runs. We see that $\phi(x = 10)$ matches the theoretic ϕ_{wall} well.

The uniform grid, Gauss-Seidel run uses 80 MB of RAM and took 100 minutes to run. The AMR simulation ran for 70 minutes and at steady state (peak computational resources) used 50 MB with approximately half of the uniform grid's cells. Generally, the Poisson solver takes most of the computational effort. Previous benchmarks [2] have shown larger



(a) (b)

Figure 2: Panel a) the steady state electron distribution for the full phase space with the overlayed adaptive grid. Panel b) the analytic and simulated steady state potential distribution.

computational gain using the AMR techniques. In any case, we expect these gains to grow exponentially in higher dimensional simulations.

Conclusion

We have presented a use case for the new Vlasov code *kobra*: a steady state 1d1v electrostatic plasma sheath. Our simulations reproduce the analytics and our computational tools provide a test bed to model further plasma-wall interactions. Additionally, we show the AMR routine reduces both the required RAM and the wall time of the simulation.

Having verified the accuracy of the sheath results, we can move to the higher dimensional Chodura sheath. The computational gains provided by AMR will be fundamental for this 1d3v problem. To further these gains, one can develop more advanced *anisotropic AMR* techniques where the coordinate and velocity space are refined independently and where each refinement level uses its own timestep. While not presented in these proceedings, *kobra*'s parallel capabilities can further serve to shorten runtime.

As one looks to model more physically complex plasma-wall interactions, we can start introducing other features. While untested, we currently have the ability to add an external magnetic field. We can further develop collision operators, wall re-emission, wall currents, adding additional species, and finally the inclusion of the full Maxwell system.

References

[1] D. Tskhakaya et al. In: *Nuclear Materials and Energy* (2021).

[2] S. Konewko, N. Maestracci, and S. Van Loo. In: *Contributions to Plasma Physics* (2026).

[3] R. Chodura. In: *The Physics of Fluids* (Sept. 1982).

[4] S. A. E. G. Falle. In: *Monthly Notices of the Royal Astronomical Society* (Oct. 2003).

[5] Jingcheng Lu and Eitan Tadmor. 2023.

[6] Kengo Tomida and James M. Stone. In: 7 (May 2023).

[7] Van Loo, Sven and Falle, S. A. E. G. and Hartquist, T. W. eng. In: *Monthly Notices of the Royla Astronomical Society* (2006).

[8] J.A.F. Hittinger and J.W. Banks. In: *Journal of Computational Physics* (2013).

[9] L. A. Schwager and C. K. Birdsall. In: *Physics of Fluids B: Plasma Physics* (May 1990).