PAUL SCHERRER INSTITUT

ETH zürich

WIR SCHAFFEN WISSEN – HEUTE FÜR MORGEN

Tobia Claglüna :: Midterm Presentation

The Langevin Approach to Discretize the Collision Operator

Introduction : Combining Vlasov and Fokker-Plank

Vlasov Equation with a Fokker Plack collisional term (Risken [1984]):

$$\frac{\partial f(\boldsymbol{r},\boldsymbol{v})}{\partial t} + \boldsymbol{v} \cdot \frac{\partial f}{\partial \boldsymbol{r}} + \frac{\boldsymbol{F}}{m} \cdot \frac{\partial f}{\partial \boldsymbol{v}} = \left(\frac{\partial f}{\partial t}\right)_{\text{coll}} \tag{1}$$

$$\left(\frac{\partial f(\boldsymbol{v})}{\partial t}\right)_{\text{coll}} = -\frac{\partial}{\partial \boldsymbol{v}} \cdot \left(f \underbrace{\langle \Delta \mathbf{v} \rangle}_{\boldsymbol{F_d}}\right) + \frac{1}{2} \frac{\partial^2}{\partial \mathbf{v} \partial \mathbf{v}} : \left(f \underbrace{\langle \Delta \mathbf{v} \Delta \mathbf{v}^{\mathsf{T}} \rangle}_{\boldsymbol{D}}\right) \tag{2}$$

$\boldsymbol{F_d}$ and $\boldsymbol{D}$ are approximated by Rosenbluth Potentials via following elliptic identities (Rosenbluth et al. [1957]):

$$\nabla^2_{\boldsymbol{v}} \nabla^2_{\boldsymbol{v}} G(\boldsymbol{v}) = -8\pi f(\boldsymbol{v}) \tag{3}$$

$$\nabla^2_{\boldsymbol{v}} G(\boldsymbol{v}) = H(\boldsymbol{v}) \tag{4}$$

Resulting Langevin Scheme and Problem Setting

Stochasticity is accounted for by $d\boldsymbol{W}_t$ following $\langle d\boldsymbol{W}_t \rangle = 0$ and $\langle d\boldsymbol{W}_t d\boldsymbol{W}_t^T \rangle = \boldsymbol{I} \cdot dt$.

$$\begin{cases} \dfrac{d\boldsymbol{r}}{dt} = \boldsymbol{v} \\ \dfrac{d\boldsymbol{v}}{dt} = \dfrac{\boldsymbol{F}}{m} + \boldsymbol{F}_d + \boldsymbol{Q} \cdot d\boldsymbol{W}_t \\ \boldsymbol{D} = \boldsymbol{Q}\boldsymbol{Q}^T \end{cases} \tag{5}$$

Disorder Induced Heating (DIH):

- Cold plasma beam evolves from a disordered state to an ordered one when emitted from a electron gun
- The beam is heated by the disorder in the plasma
- Resolving collisions is thus crucial for the simulation of the beam

Chainable Differential Stencils in 3D

- Allows concatentation of any 1D user defined stencils
- Can define one stencil type per dimension and operator

Generalized Hessian Operator for the calculation of the Diffusion tensor $\boldsymbol{D}$

- Want: FD / BD on system boundaries; centered differencing in the interior
- Can create separate operators for center, face, edge and slab subdomains of the 3D grid

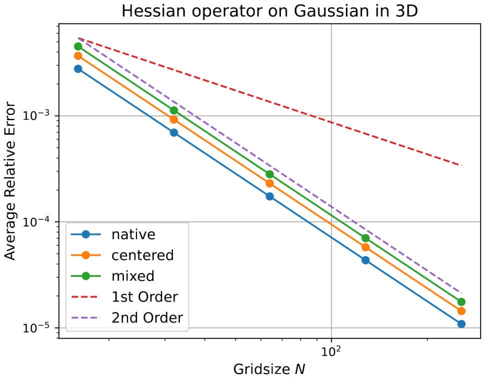

Figure: Error Convergence of concatenated stencils

Chainable Differential Stencils in 3D

Defining an operator for the second derivative: $\frac{\partial}{\partial x} \cdot \frac{\partial}{\partial x}$

Listing: Stencil Definition

```
template<Dim D, typename T, class Callable>
inline T centered_stencil(const T &hInv, const Callable &F, size_type i, j, k){
   return 0.5 * hInv * (- shiftedIdxApply<D>(F,-1,i,j,k) + shiftedIdxApply<D>(F,1,i,j,k));
}
```

Listing: Operator Definition

```
enum Dim {X, Y, Z};
DiffType DiffX = Centered;

DiffOpChain<Dim::X,DiffX,
            DiffOpChain<Dim::X,DiffX,FView_t> > diff_xx(FView_t F, Vector_t hInv);

// Call it on an index
std::cout << diff_xx(42,42,42) << std::endl;
```

Analysis of Current Solver (I)

(✓) Not parallelizeable

- Some parts had to be made GPU compatible
- Shared memory parallelism only for collisionless solver

✓ Consisted of two large files of convoluted code

- Improve general readability
- Refactoring
- Split into multiple files

✓ : fixed (✓) : partially accomplished ✗: still problematic

Analysis of Current Solver (II)

(✓) With collisions, crashed after a few timesteps

- Particles leave domain with FFT solver (open boundary conditions)

✗ Results suggest Drag/Diffusion forces are too strong

- Current workaround: Scale each term by a tunable factor

- Would be cheaper to compute collisions every *n*-th timestep (Stoel [2015])

✓ : fixed (✓) : partially accomplished ✗: still problematic

Analysis of Current Solver (III)

(✓) Results only stable for considered time frame (5 plasma periods)

- Apparent upward trend for longer time frames (see next slide)

✗ High memory consumption for 256^3 spatial grid (use Kokkos functionality for profiling)

- Some fields could be shared (i.e. currently storing 17 Scalar fields; not counting temp. fields of the 3 FFT solvers!)

✓ : fixed (✓) : partially accomplished ✗: still problematic

Comparison to P3M

Change in mesh size drastically improves results, though an upwards trend becomes apparent:

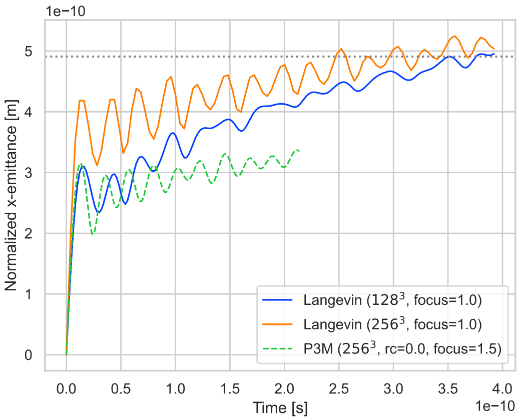

Figure: Normalized Emittance of a cold sphere w/o collisions

Comparison to P3M

Constant focusing factor is an important hyperparameter:

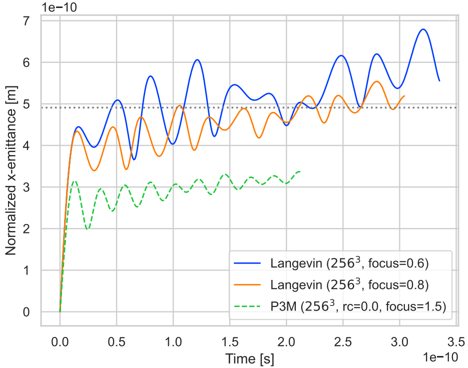

Figure: Decreased Constant Focusing

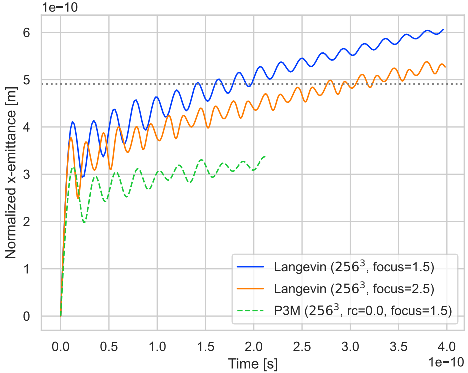

Figure: Increased Constant Focusing

Timeline in Restrospect

Date	Target Goals
30/01	Assist Severin and ensure correctness of current implementation
20/02	~~Find Order of convergence / accuracy~~ and compare whether really much better than P3M (even though it might still run only on single core)
20/02	Ensure Performance Portability (MPI, OpenMP and GPU)
06/03	Benchmarking of ~~accuracy~~, runtime and ~~scalability~~
27/03	Start improving most pressing bottlenecks

Table: Initial Timeline; Font-Encoding: partially accomplished, ~~not accomplished~~

Timeline Going Forward

Date	Target Goals
08/05	Find causes of divergence / phase shift in comparison to P3M
	Carry out rigorous profiling
	Analyse Friction / Diffusion coefficients
08/05	Implement algorithmic improvements and compare accuracy / performance to previous implementation
12/06	Start writing and code clean-up
03/07	Submission

Table: Timeline with approximate milestones

Conclusion

- Implementation of Generalized Hessian Operator
- Refactoring and verification of Severin's results
- Comparison to P3M (observation of unphysical upward trend)

Personal Take-Aways:

- Don't loose yourself in small details
- Actively seek discussions about the current problems with supervisors and students
- Keep an eye on the timeline, adjust if necessary

References I

Hannes Risken. Fokker-Planck Equation. Springer, 1984.

Marshall N. Rosenbluth, William M. MacDonald, and David L. Judd. Fokker-planck equation for an inverse-square force. Phys. Rev., 107:1–6, Jul 1957. doi: 10.1103/PhysRev.107.1. URL `https://link.aps.org/doi/10.1103/PhysRev.107.1`.

Linda Stoel. The numerical solution of the vlasov-poisson-fokker-planck equation in the context of accelerator physics. Master's thesis, Utrecht University, 2015.

James D. Callen. Plasma kinetic theory, 2018. Accessed: 2023-01-29.

Rosenbluth Potentials Rosenbluth et al. [1957]

Potentials on velocity space only Callen [2018]:

$$\Gamma = \frac{q^4 \ln(\Lambda)}{4\pi\epsilon_0^2 m^2} \tag{6}$$

$$H(\boldsymbol{v}) = 2\int d^3v' \, \frac{f(\boldsymbol{v'})}{|\boldsymbol{v} - \boldsymbol{v'}|} \tag{7}$$

$$G(\boldsymbol{v}) = \int d^3v' f(\boldsymbol{v'})|\boldsymbol{v} - \boldsymbol{v'}| \tag{8}$$

$$\langle \Delta \mathbf{v} \rangle = \Gamma \frac{\partial H}{\partial \boldsymbol{v}} = \boldsymbol{F_d} \tag{9}$$

$$\langle \Delta \mathbf{v} \Delta \mathbf{v}^T \rangle = \Gamma \frac{\partial^2 G}{\partial \boldsymbol{v} \partial \boldsymbol{v}} = \boldsymbol{D} \tag{10}$$

Resulting Elliptic Identities:

$$\nabla_{\boldsymbol{v}}^2 \nabla_{\boldsymbol{v}}^2 G(\boldsymbol{v}) = -8\pi f(\boldsymbol{v}) \tag{11}$$

$$\nabla_{\boldsymbol{v}}^2 G(\boldsymbol{v}) = H(\boldsymbol{v}) \tag{12}$$