ILMATIETEEN LAITOS
METEOROLOGISKA INSTITUTET
FINNISH METEOROLOGICAL INSTITUTE

Comparison between semi-Lagrangian and FVM Vlasov solvers

QuESpace science club 23.4.2010

L. Daldorff & A. Sandroos

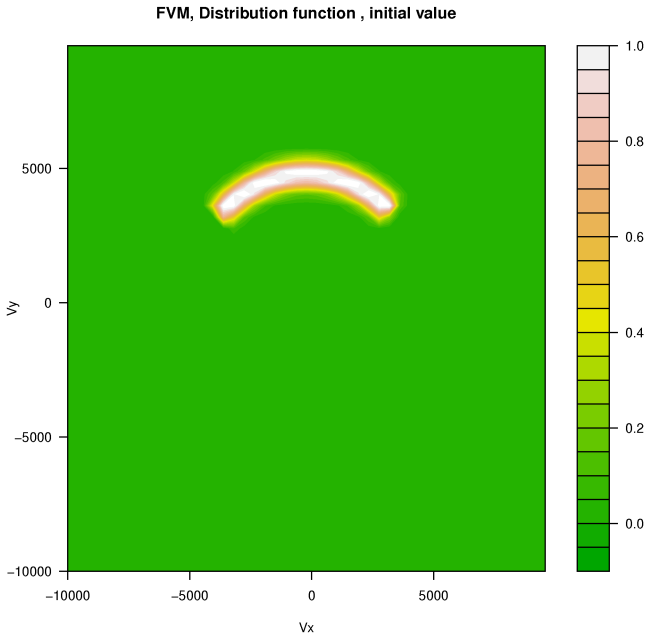
Contents

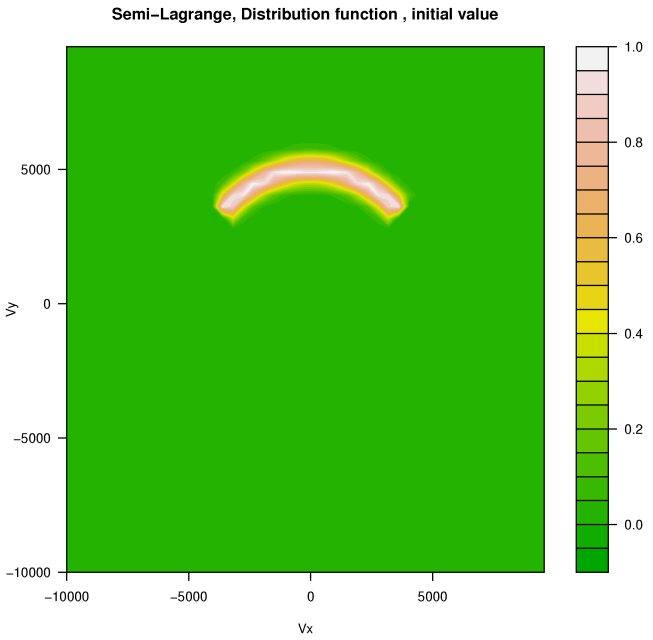
- Test case
- Results:
  - Mass Conservation
  - Min/Max Amplitudes
  - Kinetic energy
- Scalability issues (FVM)

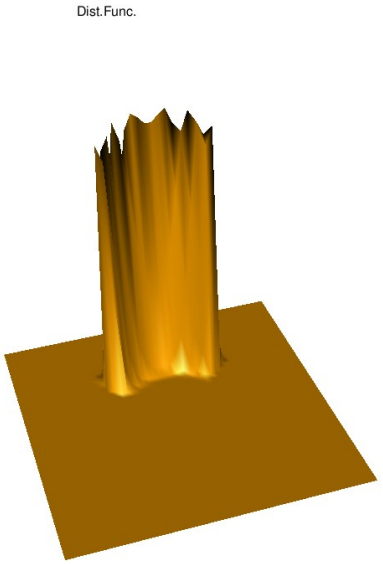
Test Case

We are limited to 2D geometry. Gyration of protons in 1 nT **B**.

- Velocity grid 45x45, -10km/s to +10km/s, Δv=425.532m/s.
- Gyro period is 65.62 s, 10 full periods simulated.
- Initial state below (Vx vs. Vy), rotation is clockwise.







Test Case

Quantities calculated from the simulations (normalized to initial values):

- Minimum and maximum value of $f(\boldsymbol{v},t)$
- Total density (conservation of mass)
- Kinetic energy
- Pressure (proxy for diffusion)

$$n(t)=m\int f(\boldsymbol{v},t)\,d^2v$$

$$K(t)=\frac{1}{2}m\int v^2\left|f(\boldsymbol{v},t)\right|d^2v$$

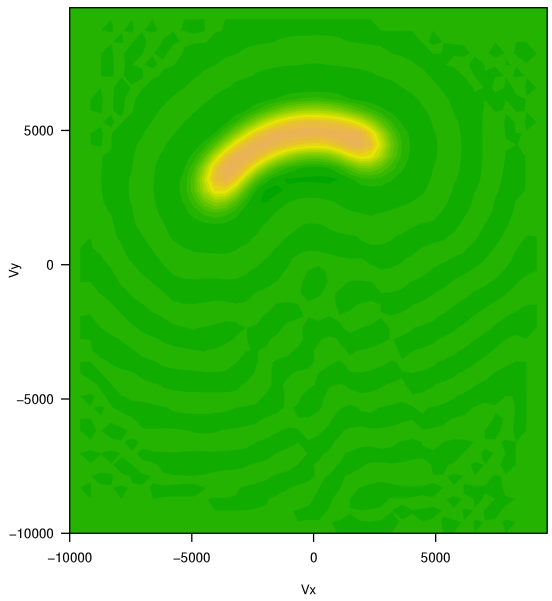
$$p(t)=\frac{1}{3}m\int (\boldsymbol{v}-\boldsymbol{U})^2\left|f(\boldsymbol{v},t)\right|d^2v$$

Results

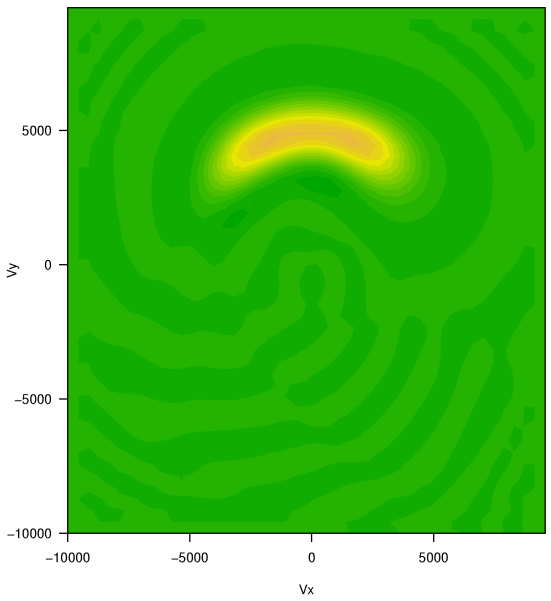
Below are the final states from sims. SL has maintained the correct shape, however no. steps are ~100 and ~10 times less than in FVM.

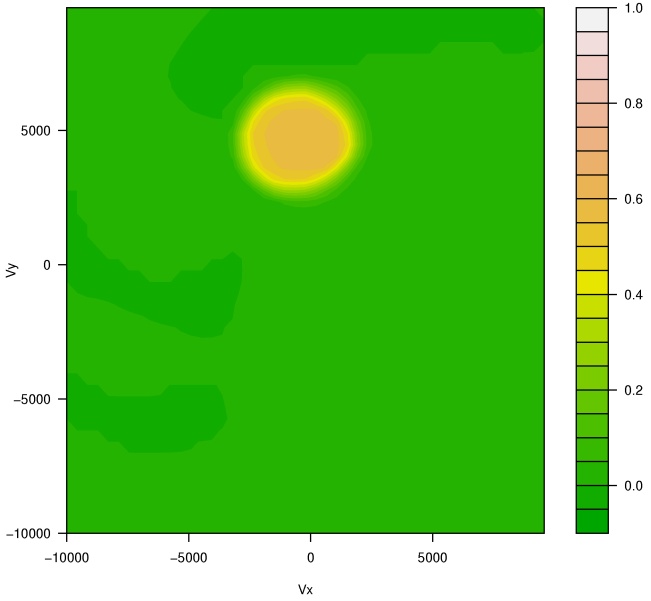
Phase error in CFL=1 SL is likely due to Boris algorithm.

CFL=1



CFL=0.1





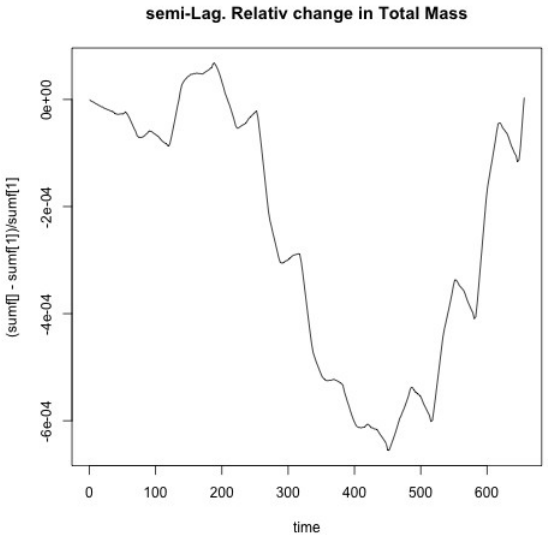
Results: Mass Conservation

SL shows small fluctuations in total mass $\sim 10^{-4} - 10^{-3}$.

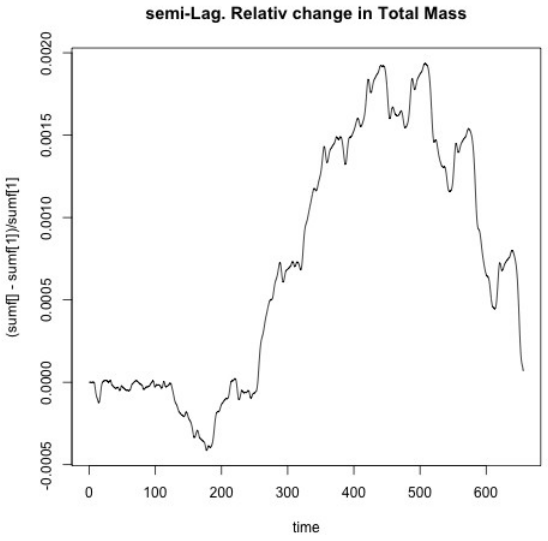
Larger fluctuations with smaller CFL.

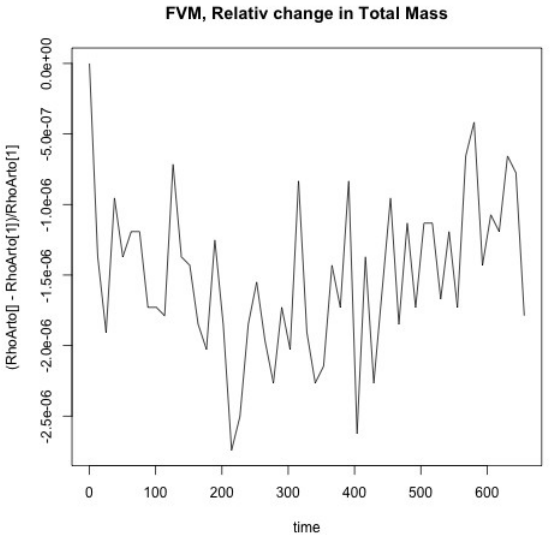
FVM rather constant after initial transition.

CFL=1

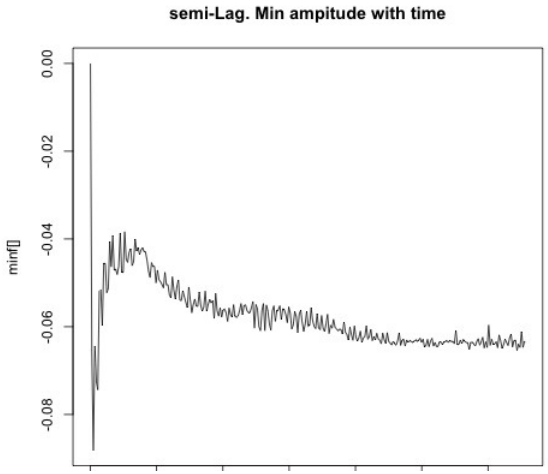


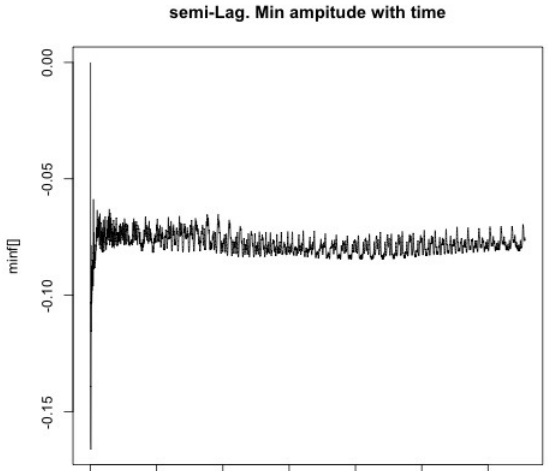
CFL=0.1

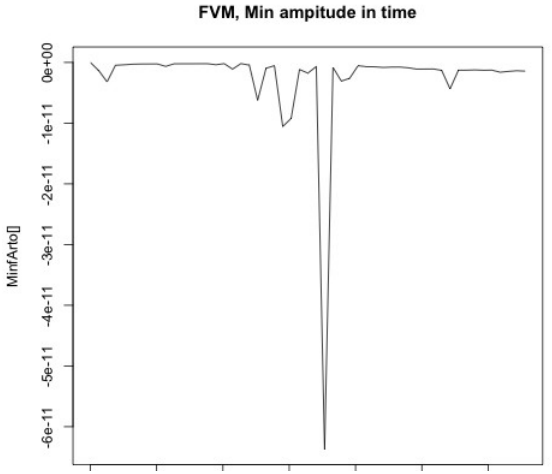


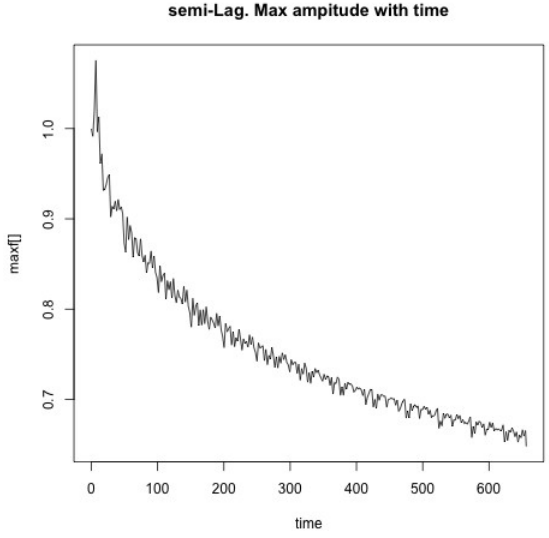


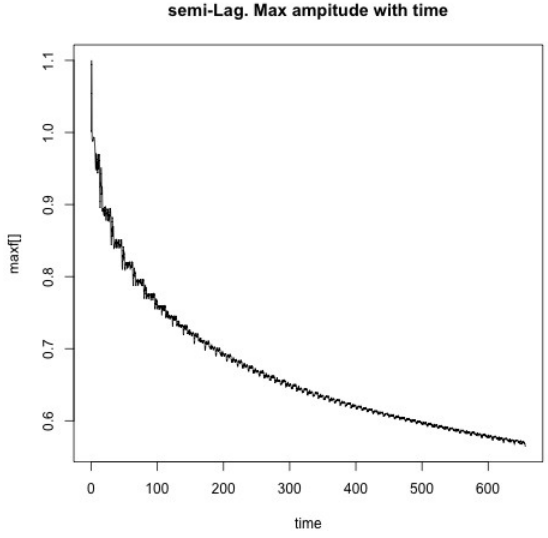
Results: Min/Max Amplitudes

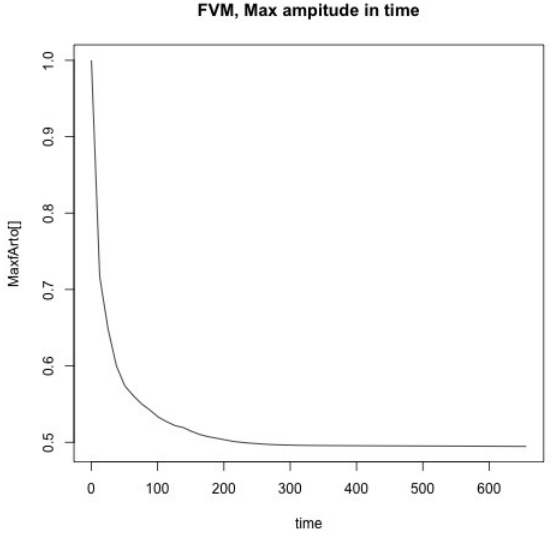










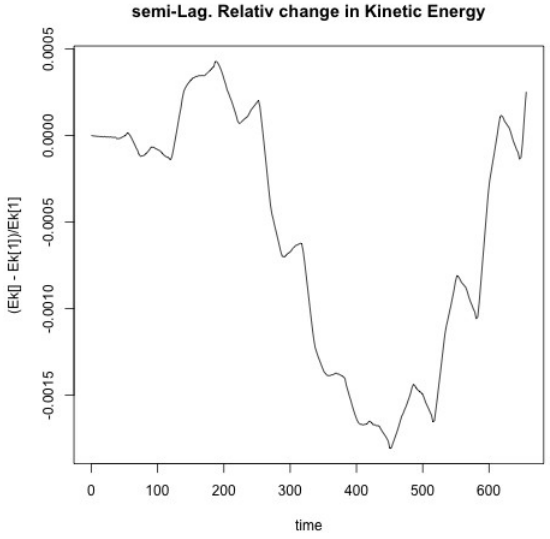


Results: Kinetic Energy

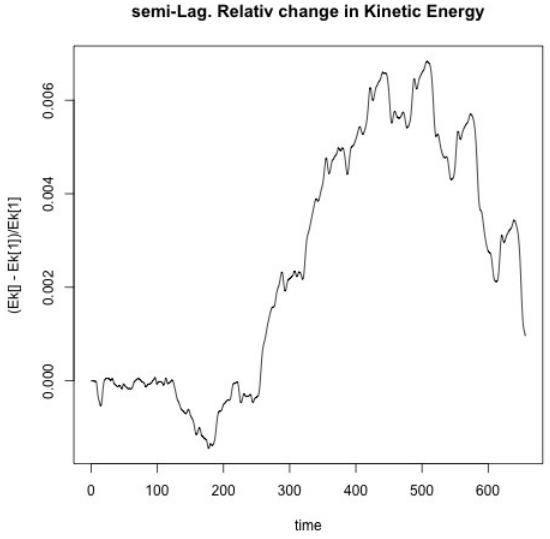
SL kin. energy tied to mass conservation.

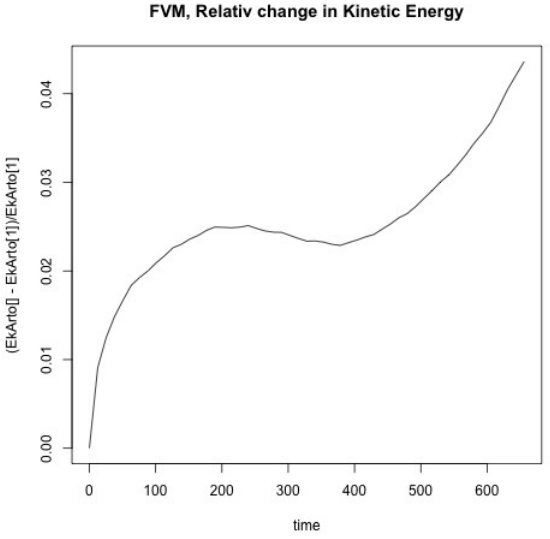
FVM kin. energy increases, jump after plateau due to reshaping of the distribution function.

CFL=1



CFL=0.1

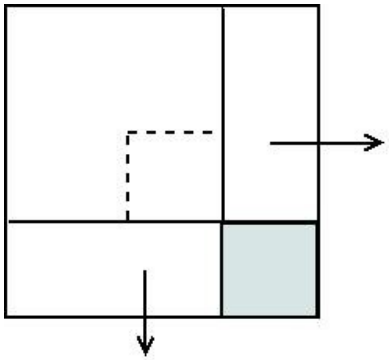




Scalability Issues (FVM)

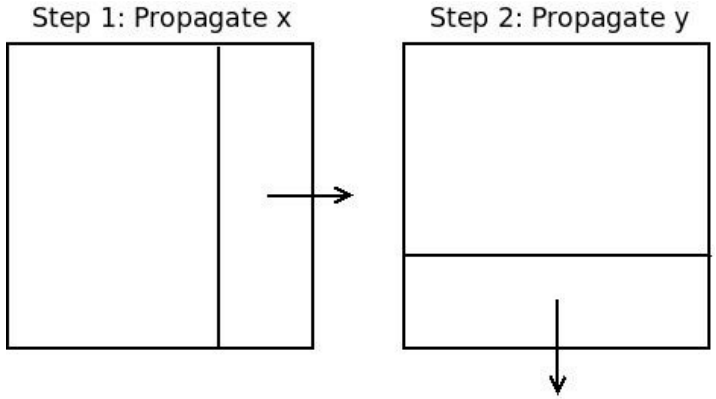
FVM time step: multidimensional system, consider dimensionally split and unsplit schemes.

Unsplit: propagate x&y simultaneously.



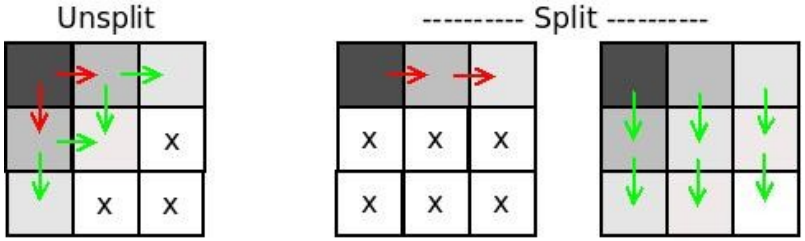
Danger of emptying the cell, small dt.

Split: First propagete x, then y.



Can use large dt.

Split scheme is unfortunately too diffusive



Scalability Issues (FVM)

Unsplit needs cell avg + 1 flux/dim + rec. params./dim. For 2nd order this is 2N+1 floats, where N is the number of dimensions. Grid processed 3 times.

Split scheme needs only avg + flux + rec. params, i.e. 3 floats. dt can be much larger, but grid needs to be processed 3N times.

Furthermore, 2nd order Runge-Kutta needs another copy of the grid.

Memory consumption: Split ~D, where D is the number or cells. Unsplit: D cells, but size of cell =2N+1 grows with N.

Time consumption: Both solvers itself should ~D. Unsplit has larger data packets, split sends more (but smaller) data packets.