1.1 Semi-Lagrangian Vlasov Codes for the Transport of Intense Particle Beams in the 4D transverse phase-space

Nicolas Crouseilles[1], Guillaume Latu[2], Jean-Louis Lemaire[3], Eric Sonnendrücker[4]
mail to: crouseil@math.u-strasbg.fr
[1]IECN and CALVI-INRIA Lorraine, Nancy
[2]LABRI - INRIA Futurs, Bordeaux
[3]CEA/DAM/DIF, Bruyères-Le-Châtel
[4]IRMA and CALVI-INRIA Lorraine, Strasbourg

1.1.1 Introduction

Particle In Cell (PIC) simulations have proven very efficient for the simulation of particle beams in accelerators, in particular for low intensity beams. However, for very intense beams as those needed e.g. for heavy ion fusion, their inherent noise and slow convergence when the number of particles increases might not make them the most efficient tool. This is why we have been investigating for several years now, direct Vlasov solvers using a grid of phase space, based on the semi-Lagrangian method. This method consists in following the particle trajectories backward from each grid point and interpolating at the origin to update the particle distribution on the phase space grid [4]. Other types of eulerian Vlasov solvers have been investigated in [5]. Simulation of the whole six dimensional phase space is not accessible yet with this approach. However a lot of interesting physics can be obtained using the paraxial model [3,7], to study the evolution of a slice in 4D transverse phase space. The particle distribution function $f(t, \mathbf{x}, \mathbf{v})$, depending on time t, transverse position $\mathbf{x}=(x, y)$ and transverse velocity $\mathbf{v} = (v_x, v_y)$, then obeys the paraxial Vlasov equation

$$\frac{\partial f}{\partial z} + \frac{v}{v_b} \cdot \nabla_x f + \frac{q}{\gamma_b m v_b} (E + v \times B) \cdot \nabla_v f = 0, \qquad (1)$$

where $\gamma_b = (1 - (v_b/c)^2)^{-1/2}$, v_b is the velocity of the beam, c is the velocity of light in free space, m the mass of the considered particles and q their charge. Moreover the force term F is given by

$$F = E + (\mathbf{v}, v_b)^T \times B,$$

where $E=(E_x,E_y)$ solves the 2D Poisson equation

$$E = -\nabla \phi, \qquad -\Delta_x \phi = \frac{q}{\varepsilon_0} \int f(z, \mathbf{x}, \mathbf{v}) d\mathbf{v},$$

and B is the magnetic focusing field, of the form $\mathbf{B} = (-\frac{1}{2} B'(z)x, -\frac{1}{2} B'(z)y, B(z))$ for periodic solenoidal focusing and of the form $\mathbf{B} = (\kappa(z)y, \kappa(z)x, 0)$ for magnetic quadrupole focusing.

Recently we have been improving the parallel efficiency of our previous code based on a uniform grid [10] and extending our 1D axisymmetric adaptive solver [8,9] to 2D (i.e. 4D phase space). This letter is devoted to the assessment of these new solvers on a relevant test case. In the next section, we shall recall the features of these solvers and then we shall assess their performance for the transport of a beam in a magnetic quadrupole channel.

1.1.2 Description of the codes

Eulerian methods have proven their efficiency on uniform meshes in two dimensional phase space, but when the dimensionality increases, since a minimum number of points per direction is required to accurately describe the physics, the total number of points on a grid becomes very important. Hence one issue to overcome this problem can be the efficient use of parallel computing. On the other side, for inhomogeneous systems, many of the grid points are wasted (where no particles are present). This is especially the case for beam simulations, where the beam moves rapidly through phase space (due to alternating-gradient focusing forces, for example). For this reason, in order to tackle the 4D phase space problems we are interested in, new methodologies had to be developed. First for codes based on a uniform grid, a parallel method with good scalability on hundreds of processors was developed and then in order to obtain en efficient adaptive solver a specific effort on sparse data structures was needed.

Our previous Eulerian Vlasov Solvers where based either on global cubic spline interpolation [4], or on flux exchanges with polynomial reconstruction [11]. The latter is more dissipative and thus less adequate for beam transport over many lattice periods. On the other hand the cubic spline solver has the drawback of needing a full transpose of the distribution function at each split step. Even with specific implementation efforts like the overlapping of computation and communications as much as possible, this does not scale well over 100 processors. For this reason we developed a new version of the code based on a novel local spline interpolation designed to give results very close to those of the global spline interpolation [2]. Indeed, the phase space computational domain is decomposed into patches; each patch is devoted to a processor and computes its own cubic spline coefficients by solving reduced linear systems. Finally, some boundary conditions are imposed at the boundary of the patches to get a C^1 global solution. This new code, LOSS (LOcal Spline Simulator), shows very good scalability properties as expected.

Another aspect needed to handle a 4D phase space with Eulerian methods is grid adaptivity. To this aim we developed an adaptive method based on wavelet interpolation for 2D phase space [8,9]. The distribution function is decomposed on a wavelet basis at different levels such that the coefficients called details are small where the function does not vary a lot. Then, according to a prediction and a thresholding procedure, only the significant details are computed. The distribution function can then be determined on the corresponding adaptive grid. Even though the mathematical method can be generalized to arbitrarily high dimensions, already for 4 dimensions, the amount of data that need to be handled requires adequate optimized data structures [12].

The sequel is devoted to benchmarking and comparisons between the two new Vlasov solvers: the uniform and massively parallel solver LOSS [2] and the adaptive solver based on interpolating wavelets OBIWAN [12].

1.1.3 The benchmark test

We consider the transport of a transverse slice of a semi-Gaussian proton beam in a magnetic quadrupole lattice using the paraxial Vlasov equation described in the introduction. The magnetic focusing field is of the form $\mathbf{B} = (\kappa(z)y, \kappa(z)x, 0)$. The initial distribution function for the semi-Gaussian beam is

$$f_0(x,v) = e^{-(v_x^2+v_y^2)/2} \text{ if } x^2 + y^2 < 1 \text{ and 0 elsewhere.}$$

The beam parameters for all the simulations are the following: the energy of the beam is equal to *6.7 MeV*, the beam current *I* is *0.3 A*, the emittance is $3\ 10^6\ \pi$ *mm mrad*. The

length of the period is equal to *0.4196 m*. These values give a tune depression of *0.56*. The beam is exactly matched to the focusing channel.

1.1.4 Simulation results

1.1.4.1 *Code based on uniform mesh*

We ran the LOSS code on a 64^2x128^2 mesh for 26 periods and on a 128^4 mesh for 6 periods. RMS values as well as phase spaces plots are displayed in Figures 1 to 5. The numerical diffusion which amount to a loss of particles at the boundaries is kept to less than 1 % over 26 periods as shown in Figure 1. This can be still improved when the grid is refined. The phase spaces snapshots come from LOSS but look very similar for the OBIWAN code.

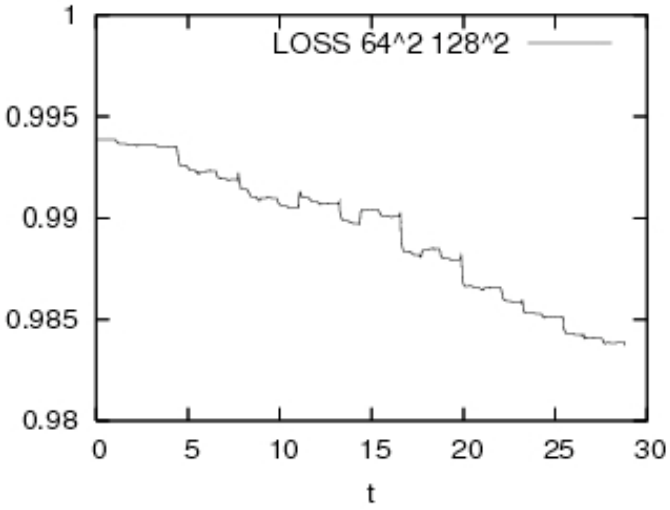


Figure 1: Evolution of the normalized total number of particles

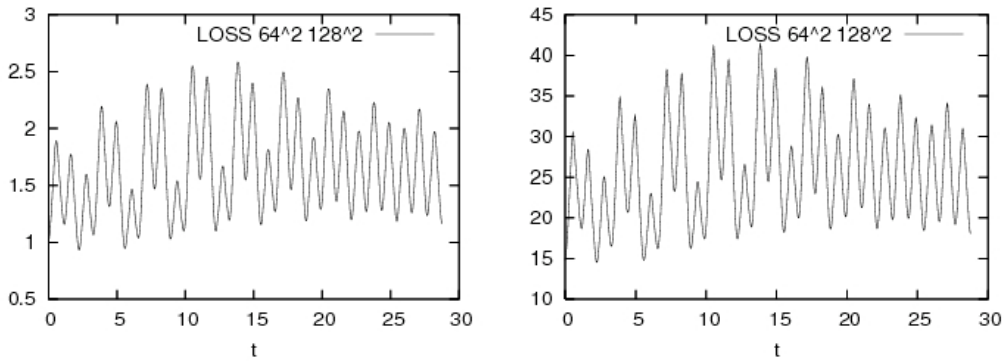


Figure 2: Evolution of x_{rms}. and ε_{rms}.

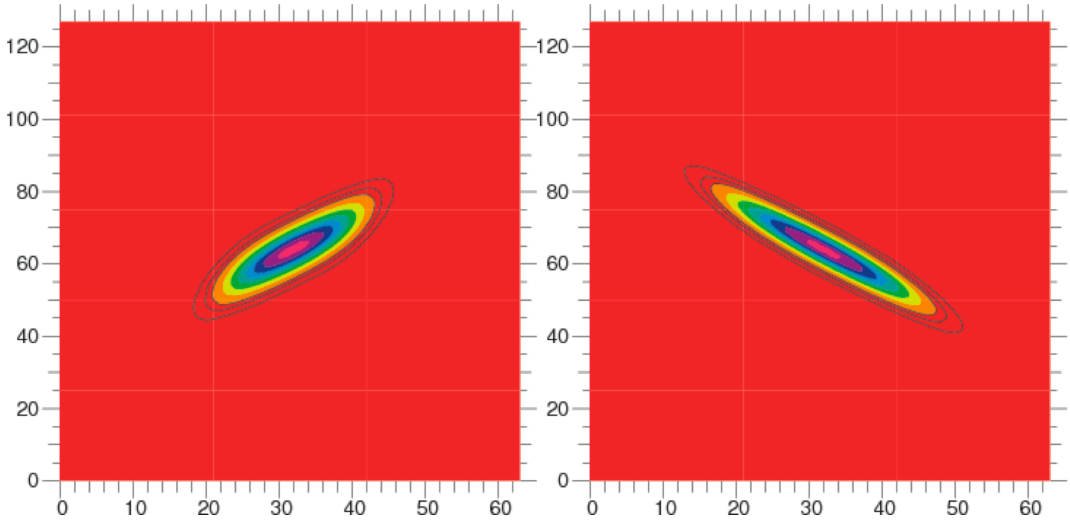


Figure 3: x-x' phase space in the 25th period on 64^2x128^2 grid

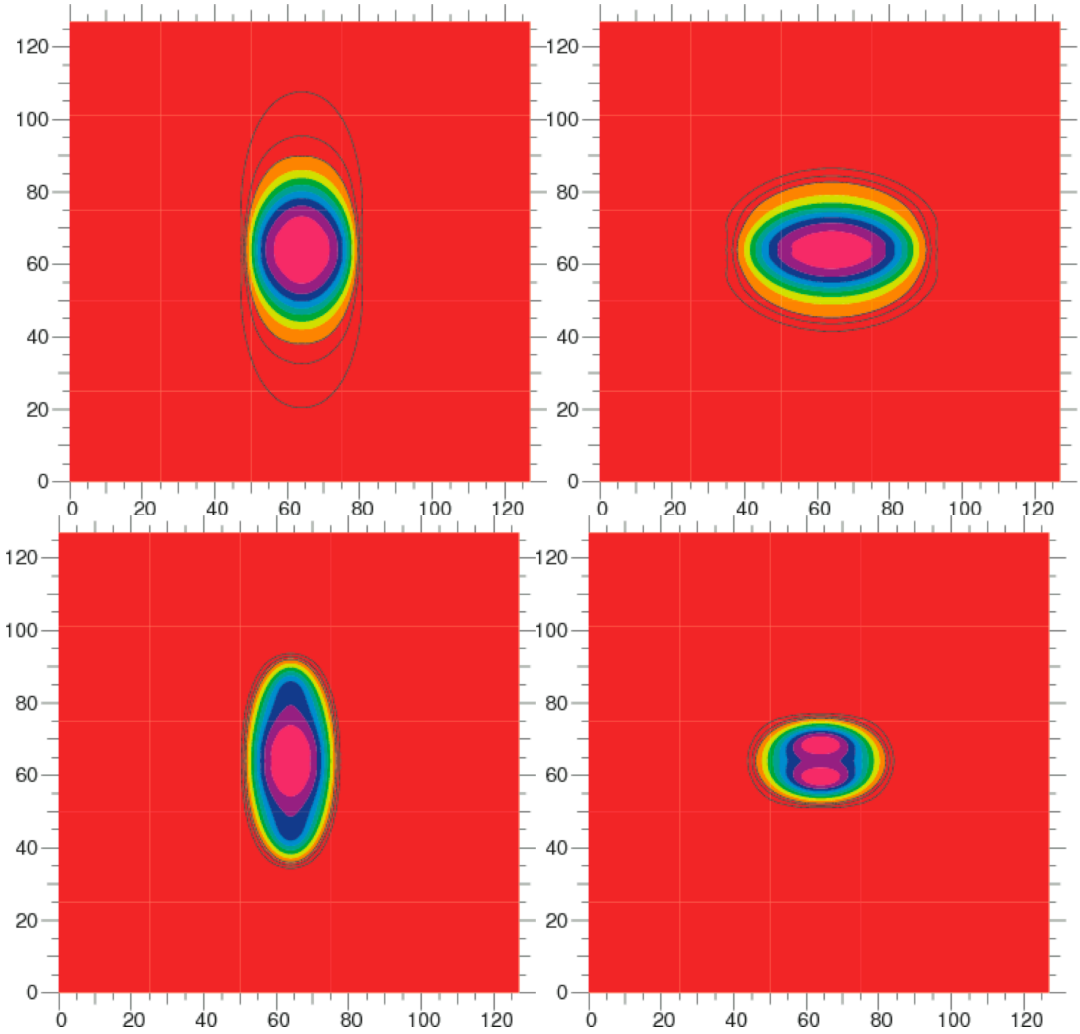


Figure 4: x-y phase space on 128^4 grid, after 4, 4.5, 5 and 5.5 periods

Jean-Louis, la dernière image me paraît curieuse qu'en penses-tu ?

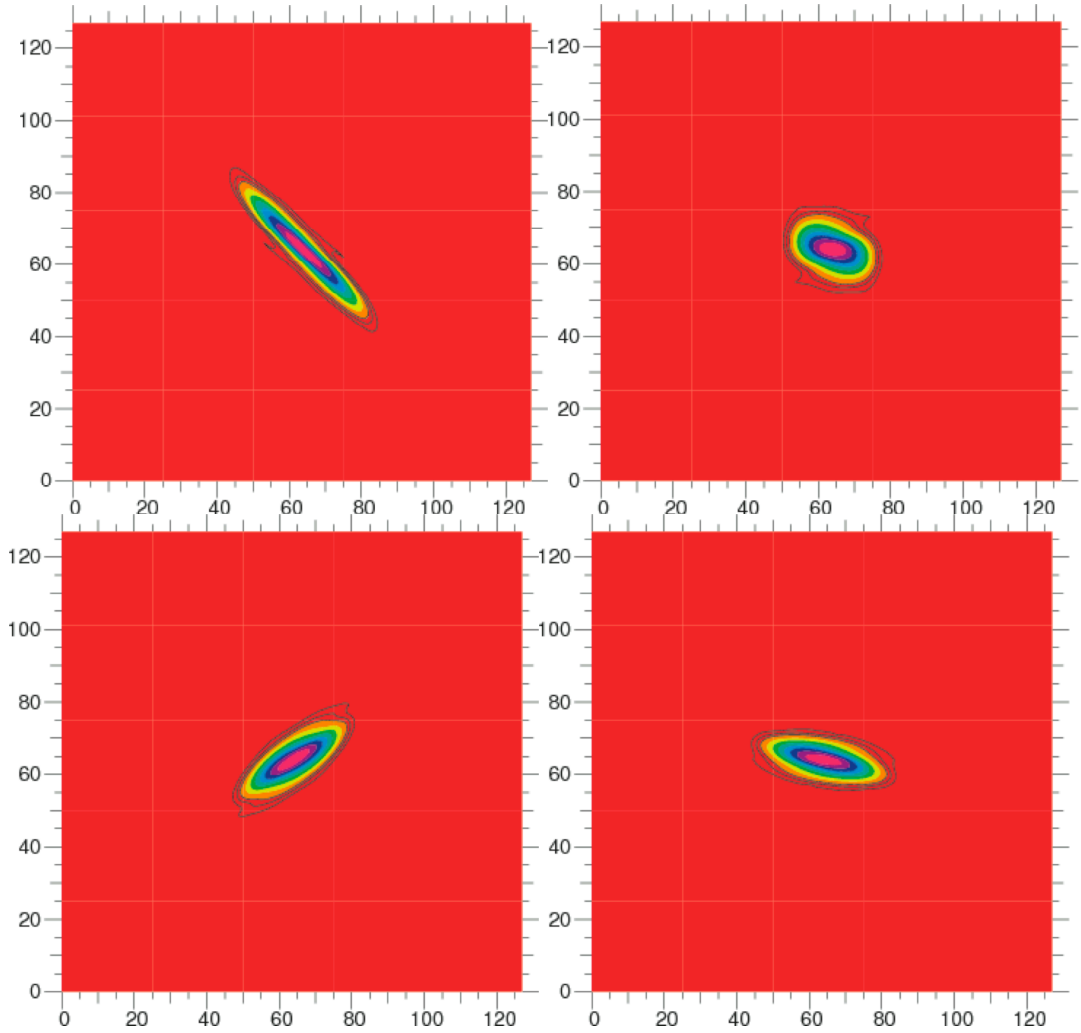


Figure 5: x-x' phase space on 128^4 grid, after 4.75, 5, 5.25 and 5.5 periods

1.1.4.2 *Comparison of uniform grid code and adaptive code*

The adaptive method we are using does not insure the exact conservation of the number of particles. This feature can be added for a slightly higher computational cost. However, as shown in Figure 6, the lack of conservation is very small and can be made even smaller by setting a lower threshold for discarding the grid points.
The evolution of the x_{RMS} for the different codes is given in Figure 7 and compared to this of a KV beam showing the influence of the nonlinear effects. All the codes give pretty much the same value and converge as the number of grid points is refined.

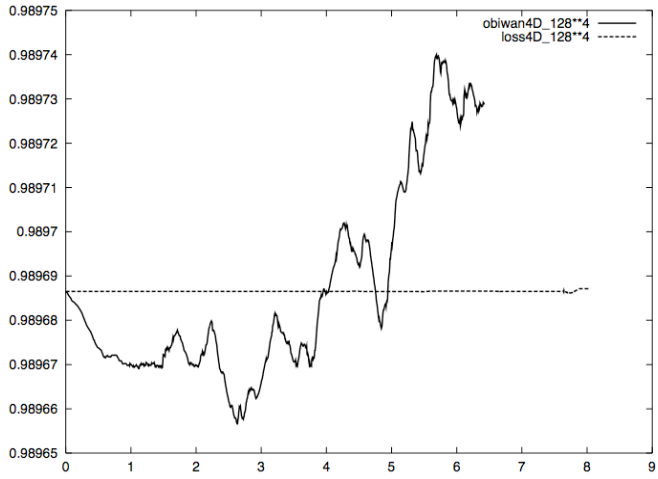
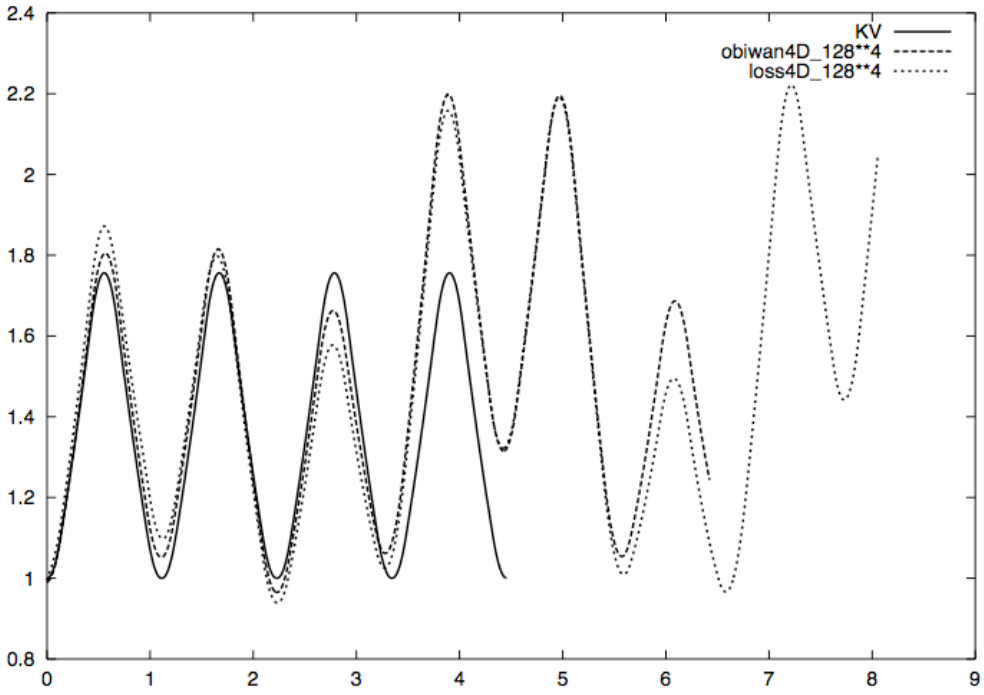


Figure 6: Evolution of normalized total number of particles with uniform and adaptive codes.



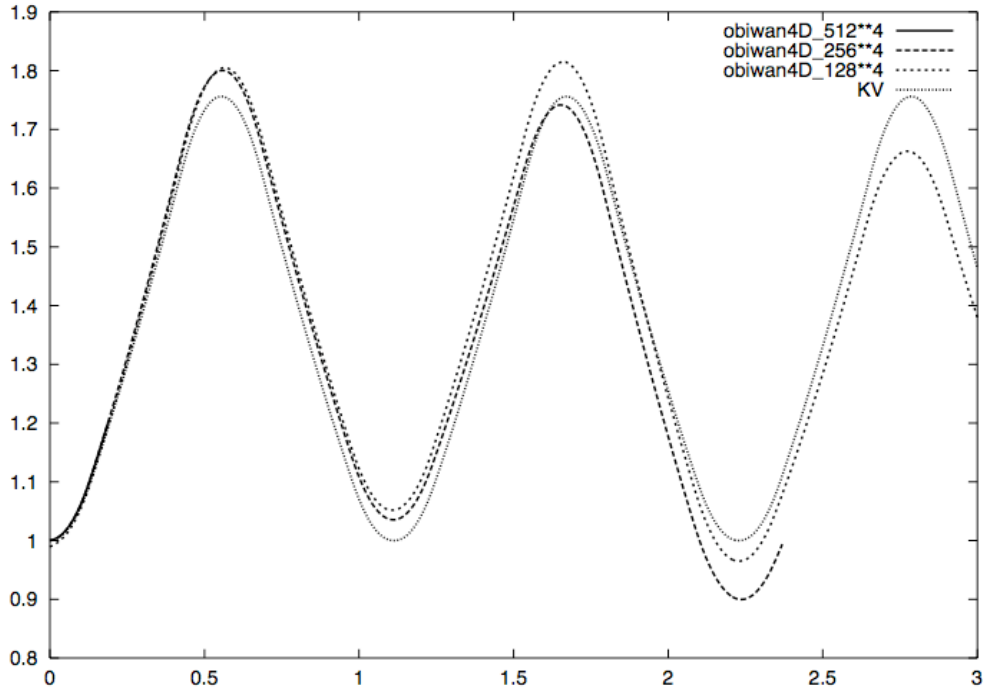


Figure 7: Evolution of x_{rms} at different resolutions with uniform and adaptive codes.

1.1.4.3 *Computational and memory cost*

Table 1 below gives the computation time for one time step and the speedup for the two methods on a 64^4 grid. On the one hand, we notice that the computational cost of the two solvers are comparable with a slight advantage for OBIWAN. On the other hand, the speedup of the two methods is quite good, since communication-computation overlapping is performed in the two codes.

Table 1: Computation cost and speedup for uniform and adaptive codes.

Numbers of processors	1	2	4	8
Time (in s.) LOSS/OBI	433/408	226/206	111/105	63/55
Speedup LOSS/OBI	1/1	1.92/1.98	3.9/3.88	6.87/7.41

Table 2 gives the time taken by the different steps of the algorithm for LOSS and in different cases for OBIWAN, on a 128^4 grid with a different number of remaining grid points (which evolves during the simulation). Figure 8 displays the evolution of the number of remaining grid points during the simulation. This number is very small throughout the simulation for our test case.

Table 2: Time in seconds for one iteration in 4D simulators on 4 processors (using IBM 16-way nodes of power 5 processors)

	Velocity Advections	**Space Advections**	**Field Computation**	**Diagnostics**	**Total**
LOSS (128^4)	38.1	84.9	1.0	1.0	125.1
OBIWAN (128^4), non-zero=2%, it=1, threshold=10^{-4}	12.4	16.4	0.3	0.4	29.5
OBIWAN (128^4), non-zero=9%, it=750, threshold=10^{-4}	44.1	57.8	0.3	1.0	103.2
OBIWAN (256^4), non-zero=0.75%, threshold=10^{-4}	49.8	70.1	1.1	2.0	123.0

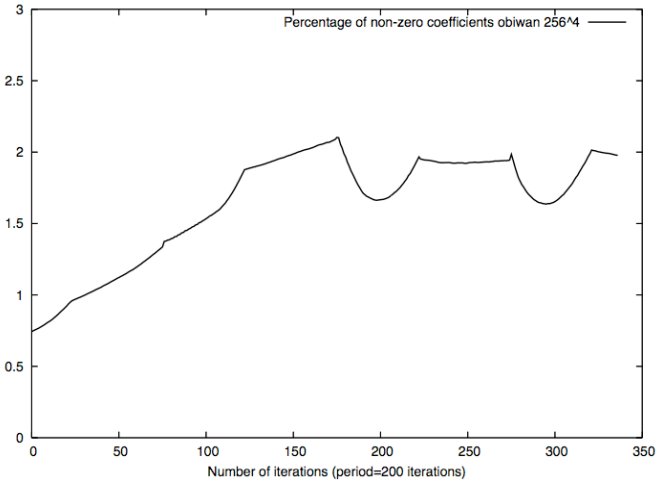


Figure 8: Percentage of remaining grid points in a 256^4 adaptive run.

The memory consumption is reduced considerably with the OBIWAN code versus the LOSS code. For the test case with a 128^4 grid, OBIWAN achieves to use only 0.9 GO memory, wheras LOSS uses 2.8 GO. A grid as large as 256^4 requires 45 GO of available memory for the LOSS code, but only 2.7 GO for OBIWAN.

1.1.5 References

1. C.Z. Cheng, G. Knorr, “The integration of the Vlasov equation in configuration space”, J. Comput. Phys., 22, p. 330, (1976)
2. N. Crouseilles, G. Latu, E. Sonnendrücker, "Hermite spline interpolation on patches for a parallel solving of the Vlasov-Poisson equation", Research report INRIA Lorraine 5926: http://hal.inria.fr/inria-00078455.
3. F. Filbet, E. Sonnendrücker, “Modeling and numerical simulation of space charge dominated beams in the paraxial approximation”, Math. Models Meth. App. Sc., 16, pp.1-29, (2006).
4. E. Sonnendrücker, J. Roche, P. Bertrand, A. Ghizzo, “The semi-Lagrangian method for the

numerical resolution of the Vlasov equations", J. Comput. Phys., 149, pp. 201-220, (1999).
5. F. Filbet, E. Sonnendrücker, "Comparison of Eulerian Vlasov solvers", Comput. Phys. Comm., 151, pp. 247-266, (2003).
6. M. Gutnic, M. Haefele, I. Paun, E. Sonnendrücker, "Vlasov simulation on an adaptive phase space grid" , Comput. Phys. Comm., 164, pp. 214-219, (2004).
7. R.C. Davidson, H. Qin, "Physics of Charged Particle Beams in High Energy Accelerators", Imperial College Press, World Scientific, 2001.
8. E. Sonnendrücker, M. Gutnic, M.Haefele, G. Latu, J.-L. Lemaire, " Vlasov simulation of beams and halo", Proc. 21st biennial Particle Accelerator Conference, PAC05, Knoxville TN, 2005.
9. E. Sonnendrücker, M. Gutnic, M.Haefele, J.-L. Lemaire, "Adaptive Vlasov simulations of intense beams", 33rd ICFA Advanced Beam Dynamics Workshop on High Intensity and High Brightness Hadron Beams, 18-22 october 2004, AIP Conference Proceedings 773, pp 154-157 (2005).
10. J.-L. Lemaire, E. Sonnendrücker, "Beam Dynamics Comparisons between Semi-Lagrangian and PIC Techniques for Simulation ot the Propagation of Intense Charged Particle Beams in 2D Channels ", 33rd ICFA Advanced Beam Dynamics Workshop on High Intensity and High Brightness Hadron Beams, 18-22 october 2004, in AIP Conference Proceedings 773, pp 152-154 (2005).
11. F. Filbet, E. Sonnendrücker, P. Bertrand, "Conservative Numerical Schemes for the Vlasov Equation", J. Comput. Phys. 172, no. 1, pp 1-22, (2001).
12. M. Gutnic, G. Latu, E. Sonnendrücker , "Adaptive two-dimensional Vlasov simulation of heavy ion beams". Proceedings 16th International Symposium on Heavy Ion Fusion, Saint-Malo France, to appear in Nucl. Inst. Meth. Phys. Res. A.