

# VLASOV SIMULATION OF BEAMS AND HALO

E. Sonnendrücker, M. Gutnic, M. Haefele, G. Latu, Université Louis Pasteur, Strasbourg, France,  
and CALVI-INRIA Lorraine, J.-L. Lemaire, CEA, Bruyères-Le-Châtel, France

## Abstract

This paper is devoted to the development and optimization of an adaptive grid Vlasov solver on a reduced axisymmetric model where the canonical angular momentum invariant is assumed to be 0. Whereas in PIC simulations most computational effort is put where the particle density is highest, the number of grid points in our adaptive Vlasov solver is higher in zones where the phase-space distribution function  $f$  varies most. Therefore, we expect that they might be more efficient for the investigation of non linear halo formation.

## INTRODUCTION

Even though PIC simulations have proven to be an efficient tool for beam simulations for many years, they are subject to numerical noise which only decreases slowly (as  $1/\sqrt{N}$ ) when the number of particles  $N$  is increased. Therefore other methods should be investigated, when one is interested in accurate simulations of high intensity beams especially in the low density part of phase space.

We have been developing new methods based on the direct resolution of the Vlasov equation on a grid of phase space [1]. However, especially for high intensity beam simulations in periodic or alternating gradient focusing fields, where particles are localized in phase space, adaptive strategies are required to get computationally efficient codes. To this aim, we have been developing moving grid techniques [2] as well as fully adaptive techniques based on interpolating wavelets [3, 4, 5]. On our way to full 2D simulations of transverse phase space, already solved with uniform discretization [6], we introduce here a reduced model which will enable us to evaluate the code, optimize it and give us a good idea of its features.

The paper is organized as follows. We first derive the reduced model that we shall use to evaluate the method. We then recall the idea of hierarchical approximation and give the algorithm implemented in our code. Thereafter, we discuss optimization strategies for the adaptive method and conclude with representative simulations.

## A REDUCED AXISYMMETRIC MODEL

We consider an axisymmetric beam described by its distribution function  $f(z, r, v_r, v_\theta)$  transported with constant velocity  $v_b$  along the  $z$  axis. Its transverse motion can be described by the paraxial Vlasov-Poisson model in cylindrical coordinates:

$$\frac{\partial f}{\partial z} + \frac{v_r}{v_b} \frac{\partial f}{\partial r} + \left( \frac{q}{\gamma m v_b} [(1 - \beta^2) E_r + \frac{P_\theta}{r^3}] \right) \frac{\partial f}{\partial v_r} = 0, \quad (1)$$

where  $\beta = v_b/c$  and  $\gamma = (1 - \beta^2)^{-1/2}$ . The self electric field is  $E_r$  and  $v \times B_{self}$  in this model reduces to  $\beta^2 E_r$ . The external field  $B(z)$  is supposed to be periodic with period  $L$ . Hence this equation is coupled self-consistently with the Poisson equation

$$\frac{1}{r} \frac{d}{dr} (r E_r) = \frac{q}{\epsilon_0} n. \quad (2)$$

In the sequel, we will consider a reduced model, where the invariant canonical momentum is  $P_\theta = 0$ . Moreover, in order to avoid numerical round-off errors arising when working with quantities of different orders of magnitude, we shall use dimensionless coordinates. Thus, we define our dimensionless variables  $\tilde{z} = Lz$ ,  $\tilde{r} = Lr$ ,  $\tilde{v} = v_b v$ ,  $\tilde{E}_r = \tilde{E} E_r$ ,  $\tilde{B} = \tilde{B} B$ ,  $\tilde{n} = \tilde{n} n$ . Plugging those into the Vlasov equation (1), we get

$$\frac{\partial \tilde{f}}{\partial \tilde{z}} + \frac{\tilde{v}_r}{v_b} \frac{\partial \tilde{f}}{\partial \tilde{r}} + \left( \frac{q L \tilde{E}}{\gamma^3 m v_b^2} \tilde{E}_r - \left( \frac{q \tilde{B} L \tilde{B}(z)}{2 \gamma m v_b} \right)^2 \tilde{r} \right) \frac{\partial \tilde{f}}{\partial \tilde{v}_r} = 0,$$

$$\frac{1}{\tilde{r}} \frac{d}{d\tilde{r}} (\tilde{r} \tilde{E}_r) = \frac{q}{\epsilon_0} \tilde{n} \tilde{n}.$$

Now, the total number of particles  $N_0$  obeys  $2\pi \int L^2 \tilde{n} \tilde{r} d\tilde{r} = N_0$ . In order to normalize it to 1 in the dimensionless variables, we set  $2\pi \tilde{n} L^2 = N_0$ . Moreover, in order to eliminate the coefficients in Poisson's equation, we set  $\epsilon_0 \tilde{E} = q \tilde{n} L$ . We then find

$$\frac{q L \tilde{E}}{\gamma^3 m v_b^2} = \frac{q^2 N_0}{2\pi \epsilon_0 \gamma^3 m v_b^2} = K$$

which is the dimensionless perveance. Finally, we set

$$k_0^2(z) = \left( \frac{q \tilde{B} L \tilde{B}(z)}{2 \gamma m v_b} \right)^2.$$

Then dropping the tilde symbol our model reads:

$$\frac{\partial f}{\partial z} + v_r \frac{\partial f}{\partial r} + (K E_r - k_0^2(z) r) \frac{\partial f}{\partial v_r} = 0, \quad (3)$$

coupled self-consistently with the Poisson equation

$$\frac{1}{r} \frac{d}{dr} (r E_r) = n. \quad (4)$$

Although this model, being only one-dimensional in velocity does not support the KV solution as an exact solution, we can use the Courant-Snyder theory, assuming the

self field is linear in  $r$  to get an approximate matching condition by solving the envelope equation

$$R'' + k_0^2(z)R - \frac{K}{R} = \frac{\epsilon_{KV}^2}{R^3}. \quad (5)$$

The initial distribution  $f_0$  with the given  $N_0$  and  $v_b$  such that  $r_{RMS}$  and  $v_{rRMS}$  are equal to those computed from the matched envelope values yields an approximately matched beam.

## HIERARCHICAL APPROXIMATION

Our aim is to develop an efficient solver for the Vlasov equation, based on an adaptive grid of phase space. We shall use the semi-Lagrangian method [1], generalizing the split solver of Cheng and Knorr [7]. Using the conservation of the distribution function  $f$  along the characteristics, the method updates the values of  $f$  at the grid points in two steps: 1) Compute the origin of the trajectory ending at the grid point, 2) Interpolate the value at the origin from the known grid value.

Especially for beam transport in periodic and alternating gradient focusing fields, using a uniform grid of phase space is very inefficient, as lots of grid points bring very little enhancement in terms of precision. Therefore, we want to use a method finding out automatically, depending on the function  $f$  being interpolated, which grid points from an underlying uniform grid are really needed to get the desired precision. This can be done with non linear approximation [8]. Let us briefly explain the idea. Assume we want to approximate a given function by its linear interpolant on a given set of uniformly distributed grid points. Then  $f$  is approximated by  $f_h = \sum_i f(x_i)\varphi_i(x)$  where  $\varphi_i$  is the hat function centered at grid point  $x_i$  (see Fig. 1). In order to get a better approximation, the usual method is e.g. to use twice as many grid points and hat functions half as large (bottom right part of Fig. 1). Another way, which gives exactly the same approximate  $f_h$  on the finer grid, is to keep the hat functions from the coarser grid, and add finer new ones at the new grid points (bottom left part of Fig. 1). This is called hierarchical approximation. The difference is that in the latter case the coefficient in front of the hat function at a new grid point is not anymore the value of  $f$  at that grid point, but the difference between  $f$  and its interpolant at the coarser level. Hence the coefficients are small when the coarse approximation is good. Non linear approximation consists in keeping the  $N$  highest coefficients in a hierarchical decomposition. It is non linear because the coefficients that are kept depend on  $f$ . Using a hierarchical approximation of  $f$ , only the grid points where  $f$  varies most are kept. And only those need to be handled by our time advance algorithm. This gives an accurate representation of  $f$  with relatively few grid points. The grid points used to approximate  $f$  can be compared to the approximation of the beam by macro-particles in PIC codes. Fig. 2 gives the description of  $f$  in the case of semi-Gaussian beam by PIC macro-particles (left) and by a non

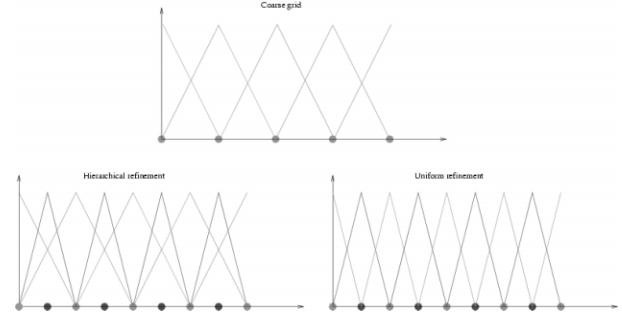


Figure 1: Hierarchical (left) and uniform (right) refinement of a coarse grid.

linear approximation on a grid (right). In the usual particle description, particles are concentrated where  $f$  is highest. In the non linear grid approximation, kept grid points are concentrated where  $f$  varies most. Therefore one expects that non linear grid approximation will be more effective at describing variations of  $f$  in zones of weak density, typically where halos are formed.

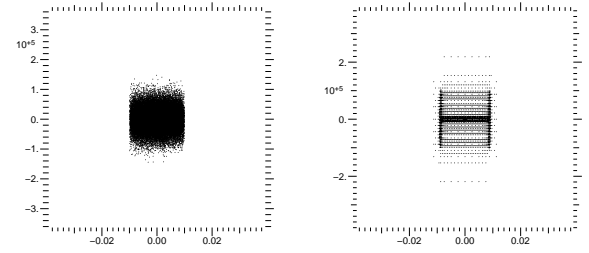


Figure 2: Description of  $f$  with a PIC code (left) and with a non linear approximation (right).

## NUMERICAL ALGORITHM

Hierarchical decomposition is closely related to wavelet theory. In the wavelet terminology, the approximating function at the coarse level is called scaling function, and the approximating functions at the finer levels (which can be the same as in our example) are called wavelets. We will consider only scaling functions (basis functions) constructed using Lagrange interpolating polynomials and wavelets which are identical scaled versions of the scaling function (see [3] for more details). The wavelets can be modified as presented in [4] in order to exactly conserve moments of  $f$  if needed, but we shall not do this here.

The 2D description of  $f$  is done using a tensor product representation. The distribution function is approximated, in a hierarchical approximation on  $J$  levels, by

$$f(r, v_r) = \sum_{k_1, k_2} c_{k_1, k_2} \varphi_{k_1}(r) \varphi_{k_2}(v_r) + \sum_{j=0}^J \sum_{k_1, k_2} \sum_{\beta=1}^{2^j-1} d_{k_1, k_2}^{j, \beta} \phi_{k_1, k_2}^{j, \beta}(r, v_r), \quad (6)$$

where  $\phi_{k_1, k_2}^{j, \beta}(r, v_r) = \varphi_{k_1}^{j, \beta/2}(r) \varphi_{k_2}^{j, \beta \% 2}(v_r)$ . Here  $\beta/2$  is the integer division of  $\beta$  by 2 and represents the first term in the binary expansion of  $\beta$  and  $\beta \% 2$  is the remainder of the integer division of  $\beta$  by 2 and represents the last term in the binary expansion of  $\beta$ . Using the same principle of binary expansion this formula can be extended to arbitrary dimension by replacing the  $2^2$  in the sum over  $\beta$  in 2D by  $2^n$  in nD. Moreover, the scaling function at level  $j$  is  $\varphi_{k_1}^{j, 0}(x) = \varphi(2^j x - k_1)$  and the wavelet at level  $j$  is  $\varphi_{k_1}^{j, 1}(x) = \varphi(2^j(2x - 1) - k_1)$ . Here  $\varphi$  denotes the scaling function (or basis function) at the coarsest level. Using the principle of non linear approximation, only the largest terms in this sum will be kept. The charge and current densities  $\rho(r, t) = q \int f dv_r$  and  $j(r, t) = q \int f v_r dv_r$  can be easily computed by integrating (6) with respect to  $v_r$ , making the adequate change of variables and noticing that  $\varphi$  is constructed such that  $\int \varphi(x) dx = 1$  and  $\int \varphi(x)x dx = 0$ . Computing  $\rho$  and  $j$  in this way only takes into account the terms in expression (6) which are kept in the non linear approximation and thus reduces considerably the number of operations needed.

In order to perform the time advance, knowing  $f^n$  at time  $t_n$  by its non linear approximation, we need to predict the terms that will appear in the non linear approximation of  $f^{n+1}$ . Therefore, we predict a set of grid points by advancing the grid points kept for  $f_n$  and including in the predicted grid, points around the advanced points. Indeed, the predicted points are all the grid points around the advanced points, in a square of width the size of the scaling function at this level, at the given level and one level finer. The predicted grid is at the end enhanced so that it becomes a well formed tree needed to perform the wavelet decomposition. This is an additional step compared to the usual semi-Lagrangian method. We can now give the algorithm, that can be implemented both using a splitting between  $r$  and  $v_r$  coordinates, as proposed by Cheng and Knorr in [7], or using a non split algorithm as proposed in [2]. The split algorithm is given in Fig. 3 and the non-split algorithm in Fig. 4.

## OPTIMIZATION

**Data structure.** Thanks to non linear approximation, the distribution function can be approximated almost as well using a lot fewer grid points. Hence, in order to save memory as well as computing time, we do not want to store anymore the approximation points in an array containing all the points of the fine grid. In the first version of the code, a hash table was used for storing the points. This was very good for memory use, but far from optimal in terms of computing time. Hence we changed to a new data structure, trading some additional memory use for a much faster computing time. We now use a sparse data structure based on two levels of dense arrays. The first array contains all the grid points up to some intermediate level. The second, which is allocated where needed, contains all the grid points from this intermediate up to the finest level. Hence

1. Read input parameters;
2. Compute initial conditions: electrostatic field, wavelet decomposition of the distribution function;
3. For all time steps  $t$  required;
  - 3A. Splitting in  $v$ -direction;
    - 3A.1 Prediction step in  $v$ -direction;
    - 3A.2 Make a well formed tree; backward advection in  $v$ -direction;
    - 3A.3 Adaptive wavelet transform; → Compute new nonlinear approximation
  - 3B. Splitting in  $x$ -direction;
    - 3B.1 Prediction step in  $x$ -direction;
    - 3B.2 Make a well formed tree; backward advection in  $x$ -direction.
    - 3B.3 Adaptive wavelet transform; → Compute new nonlinear approximation
  - 3C. Compute charge density  
Solve Poisson.
- End for

Figure 3: Split algorithm

1. Read input parameters;
2. Compute initial conditions:  $\rho, j, E_r, E_r^{pred}$   
wavelet decomposition of the distribution function;
3. For all time steps  $t$  required;
  - 3.1 Global prediction step. Grid points are predicted using explicit Euler scheme.
  - 3.2 Make a well formed tree;
  - 3.3 Backward advection based on following;
    - Compute  $j^n$ , then  $E_r^{pred}$  from Ampere's law;
    - Do until  $|E_r - E_r^{pred}| < \epsilon$ ;
    - 3.3.1  $V_{r,j}^{n+\frac{1}{2}} = v_{r,j} - \frac{\Delta t}{2} E_r^{pred}(r_i)$
    - 3.3.2  $R_i^n = r_i - \Delta t V_{r,j}^{n+\frac{1}{2}}$
    - 3.3.3  $V_{r,j}^n = V_{r,j}^{n+\frac{1}{2}} - \frac{\Delta t}{2} E_r^n(R_i^n)$
    - 3.3.4. Compute charge density;  
Solve Poisson. Compute new  $E_r^{pred}$ .
  - 3.4 Adaptive wavelet transform;
- End for

Figure 4: Non split algorithm

all grid points can be accessed with at most one indirection pointer. This new data structure enabled us to reduce the execution time by a factor 3.

**Parallelization.** There are basically two kinds of data locality in the different steps of the algorithms. Some need to access points being physically close in phase space, and others need to access points level by level. At coarser levels, points are not physically very close. Hence one sin-

gle domain decomposition, as we wanted to use, implies complex data shape access. For this reason the code was parallelized using OpenMP targeting shared memory computers to avoid calling communication subroutines. For a large grid, the efficiency of the parallelization was .89 on 16 processors, .79 on 32 processors and .66 on 64 processors of an SGI Origin 3800 at 500 MHz (an efficiency of 1 correspond to a speed-up equal to the number of processors). The parallelization strategy and experimental results are described in [9].

## SIMULATION RESULTS

We consider the evolution of a non stationary semi-Gaussian beam defined by the initial distribution in the dimensionless units

$$f_0(r, v_r) = \frac{1}{\sqrt{2\pi}v_{th}} e^{-\frac{v_r^2}{2v_{th}^2}}, \quad \text{for } r < a,$$

where  $a$  and  $v_{th}$  are given by the matching condition for the envelope equation (5). We consider a  $L$ -periodic focusing field of the form  $B(z) = \alpha(1 + \cos(2\pi z/L)^2)$ , with  $\alpha = 1.12 T$ . In our simulations a 5 MeV proton beam is transported over 60 lattice periods. For a beam current of 1.9 A and a non-normalized emittance of  $10^{-5} \pi m. rad$ . This yields a perveance  $K = 10^{-4}$ , an undepressed phase advance  $\sigma_0 = 2.3 rad$  per period, and  $\sigma = 0.45 rad$  per period, so that the tune depression for the envelope equation is high and approximately 0.2. The underlying fine grid used for the simulation consists of  $512 \times 512$  grid points and we take 50 time steps per lattice period.

On Fig. 5 and 6 we display snapshots of the phase space distribution and the charge density in the matched case. The little islands around the beam represent values of  $10^{-4}$  which are in the range of numerical errors. We do not consider them physically relevant.

On Fig. 8 and 9 we display snapshots of the phase space distribution and the charge density in the case where  $a$ ,  $v_{th}$  and  $K$  are all increased by a factor of 50 % compared to the matched case.

The Vlasov solver follows very well the evolution of filamentation. Shoulders appear. Notice however that our reduced model ignores voluntarily azimuthal motion of particles. We compared the uniform and adaptive grid solvers in the matched case. For the adaptive solver there is an additional relative error of  $10^{-4}$ , but it runs about 5 times faster than the uniform grid solver. Fig. 7 shows the grid points that are kept by the adaptive solver at two different time steps.

## CONCLUSIONS

Grid based Vlasov solvers are a valuable tool to have in one's simulation toolbox. They are devoid of numerical noise, which makes them interesting for applications involving physics in low density regions of phase space.

We have shown here that by using an adaptive grid strategy they can also become computationally efficient. The reduced model that has been presented here gives a good overview of the features of such a code. Indeed they can be used as a cross-check with PIC codes. We are now working on a full 2D solver based on the same technique. It should confirm the PIC simulation results and give new insight in high tune depression regime for any beam distribution where strong nonlinear effects dominate.

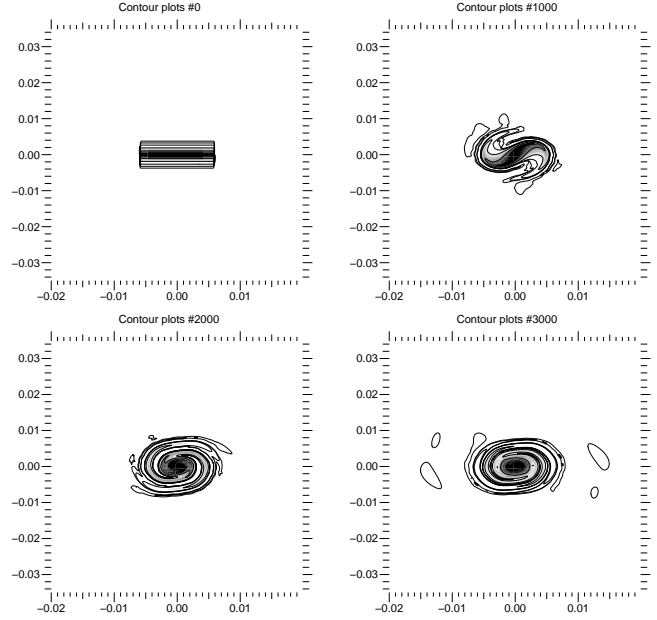


Figure 5: Periodic focusing, matched semi-Gaussian distribution: Snapshots of the distribution function in the  $r - r'$  phase space, initial condition (top left), after 20 (top right), 40 (bottom left), 60 (bottom right) periods. The black contour lines correspond respectively to densities  $0.9, 0.5, 0.1, 10^{-2}, 10^{-3}, 10^{-4}$  times the beam maximal density.

## REFERENCES

- [1] E. Sonnendrücker, J. Roche, P. Bertrand, A. Ghizzo, "The Semi-Lagrangian Method for the Numerical Resolution of Vlasov Equations", J. Comput. Phys. 149 (1998), pp. 201-220.
- [2] E. Sonnendrücker, F. Filbet, A. Friedman, E. Oudet, J.-L. Vay, "Vlasov simulations of beams with a moving grid", Computer Physics Communications 164 (1-3) (2004), pp. 390-395.
- [3] M. Gutnic, M. Haeefe, I. Paun, E. Sonnendrücker, "Vlasov simulations on an adaptive phase-space grid", Computer Physics Communications 164 (1-3) (2004), pp. 214-219.
- [4] M. Gutnic, M. Haeefe, E. Sonnendrücker, "An adaptive semi-Lagrangian method for the Vlasov equation based on multiresolution analysis with moments conservation", to appear in proceedings of ICAP 2004 conference, Saint-Petersburg, Russia.

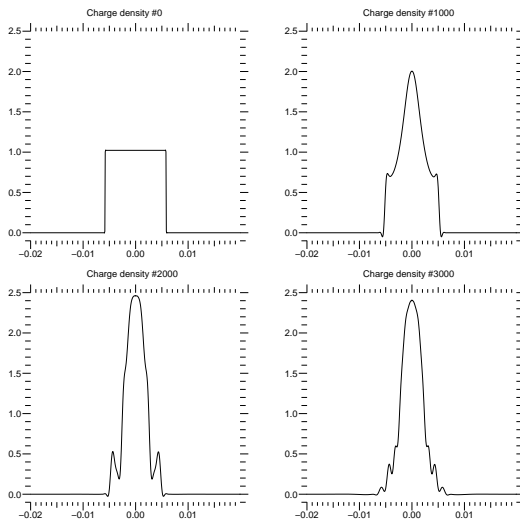


Figure 6: Periodic focusing, matched semi-Gaussian distribution: Snapshots of charge density, initial condition (top left), after 20 (top right), 40 (bottom left), 60 (bottom right) periods.

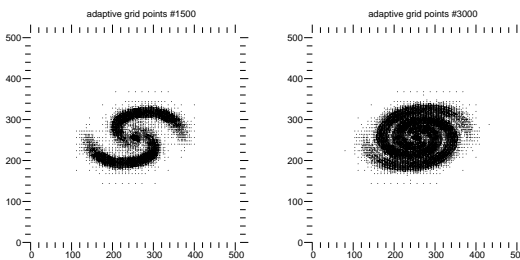


Figure 7: Periodic focusing, mismatched semi-Gaussian distribution: Snapshots of adaptive grid points after 30 (left) and 60 (right) periods.

- [5] E. Sonnendrücker, M. Gutnic, M. Haefele, J.-L. Lemaire, "Adaptive Vlasov simulations of intense beams", to appear in proceedings of 33rd advanced beam dynamics workshop on high intensity and high brightness hadron beams, Bensheim, Germany (2004).
- [6] J.-L. Lemaire, E. Sonnendrücker, "Beam dynamics comparisons between semi-Lagrangian and PIC techniques for simulation of the propagation of intense charged particle beams in 2D channels", to appear in proceedings of 33rd advanced beam dynamics workshop on high intensity and high brightness hadron beams, Bensheim, Germany (2004).
- [7] C.Z. Cheng, G. Knorr, "The integration of the Vlasov equation in configuration space", *J. Comput. Phys.* 22 (1976), pp. 330–348.
- [8] R.A. De Vore, "Nonlinear approximation", Acta Numerica 7 (1998), pp. 51-150, Cambridge. Univ. Press.
- [9] M. Haefele, G. Latu, M. Gutnic, "A Parallel Vlasov Solver using a Wavelet based Adaptive Mesh Refinement", to appear in proceedings of the 7th Workshop on High Performance Scientific and Engineering Computing (ICPP-HPSEC-05).

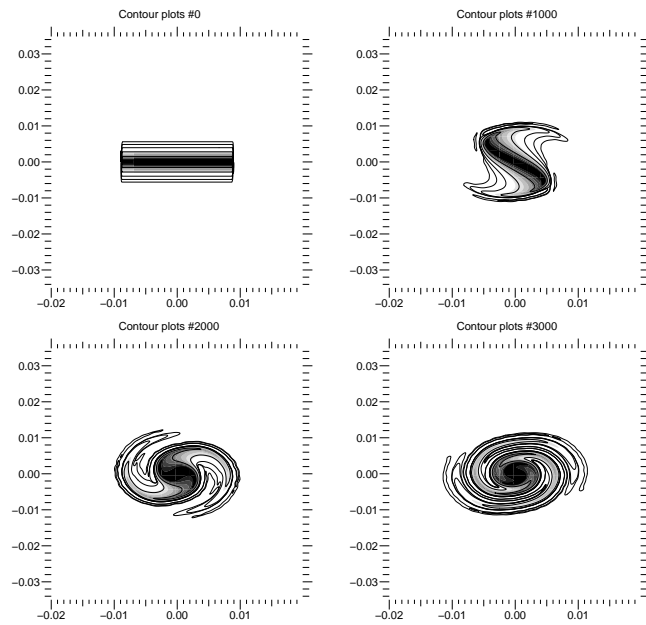


Figure 8: Periodic focusing, mismatched semi-Gaussian distribution: Snapshots of the distribution function in the  $r - r'$  phase space. See Fig 5 for details.

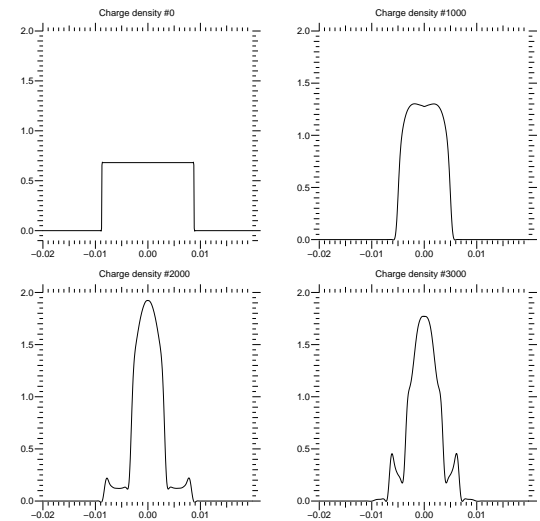


Figure 9: Periodic focusing, mismatched semi-Gaussian distribution: Snapshots of charge density at time, initial condition (top left), after 20 (top right), 40 (bottom left), 60 (bottom right) periods.