

# Vlasov versus $N$ -body: the Hénon sphere

S. Colombi,<sup>1\*</sup> S. Peirani,<sup>1</sup> G. Plum,<sup>1</sup> T. Soubie<sup>1</sup> and Y. Suto<sup>2</sup>

<sup>1</sup>*Institut d’Astrophysique de Paris, CNRS UMR 7095 and UPMC, 98bis, bd Arago, F-75014 Paris, France*

<sup>2</sup>*Department of Physics, The University of Tokyo, Tokyo 113-0033, Japan*

15 August 2014

## ABSTRACT

We perform a detailed comparison of the phase-space density traced by the particle distribution in **Gadget** simulations to the result obtained with a spherical Vlasov solver using the splitting algorithm. The systems considered are apodized Hénon spheres with two values of the virial ratio,  $R \simeq 0.1$  and  $0.5$ . After checking that spherical symmetry is well preserved by the  $N$ -body simulations, visual and quantitative comparisons are performed. In particular we introduce new statistics, correlators and entropic estimators, based on the likelihood of whether  $N$ -body simulations actually trace randomly the Vlasov phase-space density. When taking into account the limits of both the  $N$ -body and the Vlasov codes, namely collective effects due to the particle shot noise in the first case and diffusion and possible nonlinear instabilities due to finite resolution of the phase-space grid in the second case, we find a spectacular agreement between both methods, even in regions of phase-space where nontrivial physical instabilities develop. However, in the colder case,  $R = 0.1$ , it was not possible to prove actual numerical convergence of the  $N$ -body results after a number of dynamical times, even with  $N = 10^8$  particles.

**Key words:** gravitation – methods: numerical – galaxies: kinematics and dynamics – dark matter

## 1 INTRODUCTION

Stars in galaxies and dark matter in the Universe can be modeled in phase-space as self-gravitating collisionless fluids obeying Vlasov-Poisson equations

$$\frac{\partial f}{\partial t} + \mathbf{u} \cdot \nabla_{\mathbf{r}} f - \nabla_{\mathbf{r}} \phi \cdot \nabla_{\mathbf{v}} f = 0, \quad (1)$$

$$\Delta_{\mathbf{r}} \phi = 4\pi G \rho = 4\pi G \int f(\mathbf{r}, \mathbf{u}, t) \, d\mathbf{u}, \quad (2)$$

where  $f(\mathbf{r}, \mathbf{u}, t)$  represents the phase-space density at position  $\mathbf{r}$ , velocity  $\mathbf{v}$ , and  $\phi$  is the gravitational potential. In general, these equations do not have simple analytical solutions. They are therefore often solved numerically. The most widely used numerical scheme is the  $N$ -body approach and there exist many ways to implement it, which mainly differ from each other by the way Poisson equation is solved (see, e.g., Bertschinger 1998; Colombi 2001; Dolag et al. 2008; Dehnen & Read 2011, for reviews on the subject). The  $N$ -body method consists of sampling the phase-space density with particles, that is an ensemble of Dirac functions in phase-space interacting with each other through gravitational force. To avoid numerical artefacts due to the  $1/r^2$

divergence of the force at small distances, the gravitational potential is usually replaced by an effective one, where the force is smoothed at scales smaller than a softening parameter  $\epsilon$ . This is also equivalent to assuming that the particles are clouds of size  $\epsilon$  interacting with each other.

Approximating the phase-space density with macro-particles is not free of biases. Among numerical artefacts, one can naturally cite close  $N$ -body encounters, but also more subtle collective effects induced by the discrete nature of the distribution of the particles (see, e.g. Aarseth, Lin, & Papaloizou 1988; Binney 2004). The time integration scheme and the way Poisson equation is resolved numerically can of course be also sources of errors.

There are some clear example where  $N$ -body results can be questioned, sometimes because of underestimating strong numerical artefacts, particularly in the cold case where the initial velocity dispersion is null (see, e.g., Melott et al. 1997; Melott 2007), but also in cases where the  $N$ -body method is perfectly controlled, because of long term nonlinear resonant modes induced by the discrete nature of the representation (see, e.g., Colombi & Touma 2014). In addition, despite numerous thorough convergences studies of the  $N$ -body approach (see, e.g., Moore et al. 1998; Power et al. 2003; Springel et al. 2008; Stadel et al. 2009), it is not yet

\* E-mail: colombi@iap.fr

obvious that the fine inner structure of dark matter halos is completely understood from the numerical point of view.

It is therefore interesting to consider alternative to the traditional  $N$ -body approach.

In the cold case, relevant to the current paradigm of large-scale structure formation, the phase-space distribution function is supported by a three-dimensional sheet evolving in six-dimensional phase-space, which be partitioned in a continuous way with an ensemble of tetrahedra as proposed in recent works (see, e.g., Shandarin, Habib, & Heitmann 2012; Hahn, Abel, & Kaehler 2013). Of course, because of the increasing complexity of the structure of the system in phase-space, it is needed to add sampling elements during time and the cost might become prohibitive after a number of dynamical times.

In the warm case that we consider in this article, the phase-space distribution function has to be sampled on 6-dimensional grid, which is again potentially very costly even with the most powerful supercomputers available nowadays, but might become feasible in the next few years according to Moore’s law. However, one can restrict to systems with lower-dimensionality or with high level of symmetry, such as spherical systems considered in this work.

There exist many methods to solve Vlasov-Poisson equations in the warm case, most of them coming from plasma physics. The most famous solver is the so-called splitting algorithm of Cheng & Knorr (1976) and its numerous extensions (see, e.g. Shoucri & Gagne 1978; Sonnendrücker et al. 1999; Filbet, Sonnendrücker, & Bertrand 2001; Besse & Sonnendrücker 2003; Alard & Colombi 2005; Umeda 2008; Besse et al. 2008; Crouseilles, Mehrenberger, & Sonnendrücker 2010; Campos Pinto 2011; Rossmanith & Seal 2011; Güclü, Christlieb, & Hitchon 2014, but this list is far from being exhaustive). This algorithm, that we shall use here, exploits directly the Liouville theorem, namely that the phase-space density is conserved along motion: the equations of the dynamics during a time step are separated in “drift” and “kick” parts in terms of Hamiltonian dynamics and are resolved backwards,

$$f^*(\mathbf{r}, \mathbf{u}) = f(\mathbf{r} - \mathbf{v}\Delta t/2, \mathbf{v}, t), \quad \text{Drift,} \quad (3)$$

$$f^{**}(\mathbf{r}, \mathbf{u}) = f^*(\mathbf{r}, \mathbf{v} + \nabla_{\mathbf{r}}\phi\Delta t), \quad \text{Kick,} \quad (4)$$

$$f(\mathbf{r}, \mathbf{v}, t + \Delta t) = f^{**}(\mathbf{r} - \mathbf{v}\Delta t/2, \mathbf{v}), \quad \text{Drift,} \quad (5)$$

where  $\nabla_{\mathbf{r}}\phi$  is computed from  $f^*$ . In practice the phase-space distribution function is sampled on a mesh, and each step is performed by using test particles coinciding with mesh sites and following the equations of motion split as above. Resampling of  $f^*$ ,  $f^{**}$  and finally the phase-space distribution function at next time step is performed using some interpolation procedure relying for instance on spline functions. In astrophysics, the splitting scheme was applied in the eighties to one dimensional systems (Fujiwara 1981), galactic disks (Watanabe et al. 1981; Nishida et al. 1981) and spherical systems (Fujiwara 1983) but was seldom used since then, although one can mention a recent preliminary investigation of the algorithm in full 6-dimensional phase-space (Yoshikawa, Yoshida, & Umemura 2013).

As mentioned above, resolving the general problem in fully six-dimensional phase-space with sufficient accuracy remains still prohibitive. In this article, we therefore restrict our investigations to spherical systems, where phase-space is

only three dimensional: the three coordinates of interest are the radial position, the radial velocity and the angular momentum. We perform a comparison between a  $N$ -body code, namely the treecode **Gadget** (Springel, Yoshida, & White 2001; Springel 2005), and an improved version of the implementation by Fujiwara (1983) of the splitting algorithm. The goal is to check how well the particle distribution in **Gadget** traces the phase-space density obtained from the Vlasov code and how the results depend on various parameters of the simulations, in particular the number of particles in the  $N$ -body simulations and the spatial resolution in the Vlasov code. While visual inspection is primordial, we shall also perform a detailed quantitative comparison. To do so, we shall introduce correlators and entropic estimators based on a likelihood approach, the question being whether the  $N$ -body simulations can be considered as local Poisson realizations of the Vlasov code phase-space density. Note that the purpose of this article is not by any mean to find out which method is the best in terms of computational cost, although we shall discuss briefly about these issues.

Because of our restrictive choice of the geometry of the system, it is important to consider spherical configurations stable against small anisotropic perturbations induced by the shot noise of the particles. Indeed, we shall use the public treecode **Gadget**, without any specific modification to enforce spherical dynamics. The Hénon sphere (Hénon 1964) is particularly adapted to our problem since it is known to preserve well its spherical nature during the course of dynamics when being simulated with a  $N$ -body technique and, in particular, it is not prone to radial orbit instability (see, e.g., van Albada 1982; Roy & Perez 2004; Barnes, Lanzel, & Williams 2009). In this configuration, the initial phase-space distribution function is simply given by

$$f_H(\mathbf{r}, \mathbf{v}) = \frac{\rho_0}{(2\pi\sigma_v^2)^{3/2}} \exp\left(-\frac{1}{2} \frac{v^2}{\sigma_v^2}\right), \quad r \leq R_H, \quad (6)$$

with  $(4\pi/3)\rho_0 R_H^3 = M$ , the total mass of the system. Here we shall consider “warm” and a “cold” settings, corresponding to two values  $R \simeq 0.5$  and  $R \simeq 0.1$  of the initial virial ratio  $R = |2T/W| = 10R_H\sigma_v^2/(3M)$ , where  $T$  and  $W$  are respectively the total kinetic and potential energy of the system. In the simulations discussed in this article, the initial radius of the Hénon sphere and its total mass are chosen to be

$$M = 1, \quad R_H = 2, \quad (7)$$

which fixes  $\sigma_v$  in equation (6) once the virial ratio is given.

This article is thus organized as follows. In § 2 we describe our Vlasov solver, denoted by **Vlasolve**, and the simulations performed with it. § 3 provides information about the  $N$ -body runs and the parameters used in **Gadget**. In § 4, we check that the  $N$ -body simulations stay indeed spherical during evolution. In § 5, we perform a visual inspection of the phase-space density, which is followed by a quantitative statistical analysis in § 6. Finally § 7 summarizes and discusses the results.

## 2 THE VLASOV CODE

In spherical symmetry, Vlasov equation reads

$$\frac{\partial f}{\partial t} + v \frac{\partial f}{\partial r} + \left( \frac{j^2}{r^3} - \frac{GM_r}{r^2} \right) \frac{\partial f}{\partial v} = 0, \quad (8)$$

where  $v$  is the radial component of the velocity,  $j$  is the angular momentum,  $M_r = M(r' < r)$  the mass inside a sphere of radius  $r$  and  $G$  the gravitational constant. The choice of units in the remaining of this paper is such that  $G = 1$ .

In order to solve equation (8) numerically with the splitting algorithm, our code **Vlasolve** follows closely Fujiwara (1983). Phase space is discretized into a rectangular mesh of size  $(N_r, N_v, N_j)$  which spans the domain of interest as follows: the position  $R_{\min} \leq r \leq R_{\max}$  is sampled in a logarithmic fashion, the velocity  $-v_{\max} \leq v \leq v_{\max}$  linearly, and the angular momentum  $0 \leq j \leq J_{\max}$  is such that the position of the  $k^{\text{th}}$  angular momentum slice is  $j_k = J_{\max}(k - 1/2)^2/N_j^2$ , corresponding formally to the interval  $[J_{\max}(k - 1)^2/N_j^2, J_{\max}k^2/N_j^2]$ .

The splitting algorithm procedure is modified to exploit the property that angular momentum is an invariant of the dynamics. In particular, each slice corresponding to a given value of  $j$  can be treated independently, except for the obvious coupling from Poisson equation. In addition, the inertial component of the force,  $j^2/r^3$ , is included in the “drift” step (equations 3 and 5), while the “kick” step (equation 4) corresponds solely to gravitational force. Because the angular momentum contribution bends trajectories,<sup>1</sup> the drift step requires two-dimensional interpolation of the phase-space distribution function in  $(r, v)$  space, while the kick step, which only modifies the velocities, needs only a one dimensional interpolation. These interpolations are carried out using third order splines, exactly as in Fujiwara (1983). Hence, positivity of the phase-space distribution function is not warranted and aliasing effects can be expected, as well as diffusion, when details of the phase-space distribution become of the order of or smaller than the mesh element size.

Because of these defects, we decided to apodize the edges of the Hénon sphere as follows,

$$f_H \rightarrow f_H \times \frac{1}{2} \left[ 1 + \text{erf} \left( \frac{R_H - r}{\Delta} \right) \right], \quad (9)$$

with  $\Delta = 1/2$ , and to recompute  $\rho_0$  in equation (6) such that the total mass remains unity. This apodization slightly changes the actual values of the virial ratio to  $R \simeq 0.55$  and 0.11, although we shall still denote them by 0.5 and 0.1. This corresponds to a rather significant change of the original Hénon sphere and might completely modify the long term properties of the system compared to what is expected, which by itself justifies the fact that we check again in § 4 how spherical the  $N$ -body simulations remain.

Using a logarithmic scale along the radial direction proves particularly advantageous for tracing tiny features close to the center of the system that subsequently expand to larger scales as they get away from it. One downside of this procedure is that a small sphere of radius  $R_{\min}$  is necessarily missing from the computing domain. This shortcoming is usually dealt with by considering the central part as a small reflecting sphere (see, e.g., Gott 1973; Fujiwara 1983). Although simple, this method presents the major drawback of

$N_r$	$N_v$	$N_j$	$\Delta t$
1024	1024	512	$5 \times 10^{-4}$
512	512	512	$10^{-3}$
2048	2048	32	$2.5 \times 10^{-4}$
1024	1024	32	$5 \times 10^{-4}$

**Table 1.** The parameters used for the **Vlasolve** simulations.

introducing a systematic lag between orbits: particles reaching the reflective kernel boundary instantly travel the  $2R_{\min}$  distance through the central region, while they should actually take a finite time depending on their radial velocity and angular momentum.

In **Vlasolve**, we improve the reflecting sphere method by taking into account the actual time spent by particles travelling inside the region  $r \leq R_{\min}$ , which is made easily possible by neglecting the gravitational force. Technical details about the implementation are provided in Appendix A1.

As a final algorithmic detail, Appendix A2 discusses details about the hybrid parallelization of **Vlasolve** with OpenMP and MPI libraries.

For the analyses of this article, we performed 4 simulations with various resolutions for each value of the virial ratio considered, as listed in Table 1. To cover the dynamical range of interest, the computing mesh defined above was such that  $R_{\min} = 0.01$ ,  $R_{\max} = 25$ ,  $J_{\max} = 1.6$ . The maximum amplitude of the velocity was taken to be  $v_{\max} = 2$  and 4 for the virial ratios  $R = 0.5$  and 0.1, respectively. The time step  $\Delta t$  in each of the simulations was constant and resolutely small to stay on the safe side, which significantly increased the computational cost.

In appendix A3, a comparison between all simulations is performed for  $R = 0.1$ . It brings out diffusion and aliasing effects. These latter are significant, but do not seem to affect the dynamical properties of the system. It is tempting to undersample angular momentum space since  $j$  is an invariant of the dynamics. It is shown as well in this appendix that it is not wise to do so, because it can provoke nonlinear instabilities after a few dynamical times.

### 3 THE $N$ -BODY SIMULATIONS

To perform the  $N$ -body simulations, we used the latest version of the **Gadget-2** code (Springel 2005). Only the treecode part of this “treePM” algorithm was used to simulate isolated Hénon spheres. The simulations involved, for both values of the virial ratio, from  $N = 10^4$  to  $10^7$  particles. An additional simulation with  $N = 10^8$  particles was performed for  $R = 0.1$ . Note that these simulations, except for the 100 millions particles one, were much cheaper in terms of computational time than our Vlasov fiducial runs with  $(N_r, N_v, N_j) = (1024, 1024, 512)$ .

The main parameters used to perform the simulations were chosen as follows:

- The softening length of the force was given by  $\epsilon = 0.002 \times (N/10^6)^{-1/3}$ , that is about 1/16 of the initial mean interparticle distance.
- In **Gadget**, each particle has its individual time step

<sup>1</sup> In a way which can be computed analytically, see e.g., Colombi & Touma (2008).

bounded by  $dt = \min[dt_{\max}, (2\eta\epsilon/|\mathbf{a}|)^{1/2}]$ , where  $\mathbf{a}$  is the acceleration of the particle and  $\eta$  is a control parameter. Our choice was  $\eta = 0.025$  and  $\Delta t_{\max} = 0.01$ .

- The tolerance parameter  $\alpha_F = 0.005$  controlling the accuracy of the relative cell-opening criterion (parameter designed by `ErrTolForceAcc` in the documentation of `Gadget`, see equation 18 of Springel 2005).

In appendix B, we inspect visually the effects of changing these parameters on the phase-space distribution function for simulations with  $N = 10^6$  particles and a virial ratio of  $R = 0.1$ . These analyses, performed at  $t = 15$ , confirm that the parameters used for the simulations of this paper are reasonable. Interestingly, changing the softening length by large factors does not influence much the results, as already noticed previously in the literature (see, e.g. Barnes, Lanzel, & Williams 2009), as long as it is kept small enough.

#### 4 CONSISTENCY CHECK: SPHERICITY OF THE $N$ -BODY SIMULATIONS

Before performing any comparison between `Gadget` and `Vlasolve`, it is fundamental to check a crucial assumption of this paper, namely that the system remains spherical in the `Gadget` simulations. Figure 1 shows, for different values of the number of particles  $N$ , the evolution with time of the ratios  $b/a$  and  $b/c$ , where  $a \leq b \leq c$  are the eigenvalues of the inertia tensor of the particle distribution.

The dashed regions correspond to the one sigma zone obtained from an ensemble of 100 local Poisson realizations of the spherical density  $\rho(r)$ . This latter was interpolated over spherical shells from the `Gadget` particles. Particle shot noise introduces deviations from spherical symmetry which can be quantified as follows:

$$\left\langle \frac{b}{a} \right\rangle - 1 \simeq 1 - \left\langle \frac{b}{c} \right\rangle \simeq 2\sigma_{b/a} \simeq 2\sigma_{b/c} \sim \frac{1}{\sqrt{N}}, \quad (10)$$

where  $\sigma_{b/a}^2$  is the variance of  $b/a$  obtained from the dispersion over the 100 realizations, and similarly for  $\sigma_{b/c}^2$ . Note that equation (10) is only indicative. The asphericity due to discreteness is indeed expected to depend on the details of the shape of the profile  $\rho(r)$ , as can be seen on Fig. 1. Note that it would be possible to compute in a perturbative way the quantities in equation (10) from statistical analysis of the inertia tensor by assuming  $N \gg 1$  and using error propagation formulae, but this is a cumbersome exercise far beyond the scope of this paper.

It is also important to be aware of another possible source of errors related to the determination of the position of the center of the system. Indeed, a bad determination of the position of the center of the system obviously worsens apparent agreement with spherical symmetry. In the measurements presented in Fig. 1, the inertia matrix is not computed with respect to the center of gravity of the particle distribution, which can be affected by the fact that some particles can get far away from the system through  $N$ -body relaxation. Instead, we determine the center of the system using an iterative procedure trying to optimize the match of the phase-space distribution function with that of the Vlasov code, as detailed in § 6.1. This procedure is obviously not free of errors that can contribute to the fluctuations observed on the curves of Fig. 1.

Inspection of Fig. 1 shows that the measured ratios  $b/a$  and  $b/c$  behave differently in the  $R = 0.5$  and  $R = 0.1$  simulations. In the  $R = 0.5$  case, the agreement of the measurements with the Poisson prediction is in general good, with a slight trend to ellipticity, except for the top red curve and the bottom green curve where the deviation from spherical symmetry is definitely off the Poisson expectation. Still, for  $R = 0.5$ , we believe it can be fairly claimed that the system remains to a very good approximation spherical whatever  $N$  given expected deviations due to pure statistical noise.

For  $R = 0.1$ , the curves representing the axis ratios are much more steady than for  $R = 0.5$ , which is slightly puzzling. Deviations from spherical symmetry are clearly more significant compared to local Poisson expectations as soon as  $t \gtrsim 3$ , which corresponds roughly to collapse time. However, they remain small for a particle number larger than  $10^5$ , while  $N = 10^4$ , with a deviation larger than 10 percent, is clearly a matter of concern.

As a final remark, we also invested deviations of spherical symmetry for subsets of particles in excursions corresponding to  $f \geq f_{\text{th}}$  where  $f$  is the phase-space distribution function measured in the  $1024 \times 1024 \times 512$  `Vlasolve` simulations. For each value of the virial ratio, two thresholds  $f_{\text{th}}$  were chosen such that the excursions contained initially about 90 and 60 percent of the total mass (see bottom panels of Fig. 6 below). Given the uncertainties in the measurements, the conclusions derived above still stand: the properties of the deviations from spherical symmetry, that we do not show here, for simplicity, do not indeed depend significantly on radius. We only noticed a slight improvement in the  $R = 0.5$  case when considering particles in the excursions.

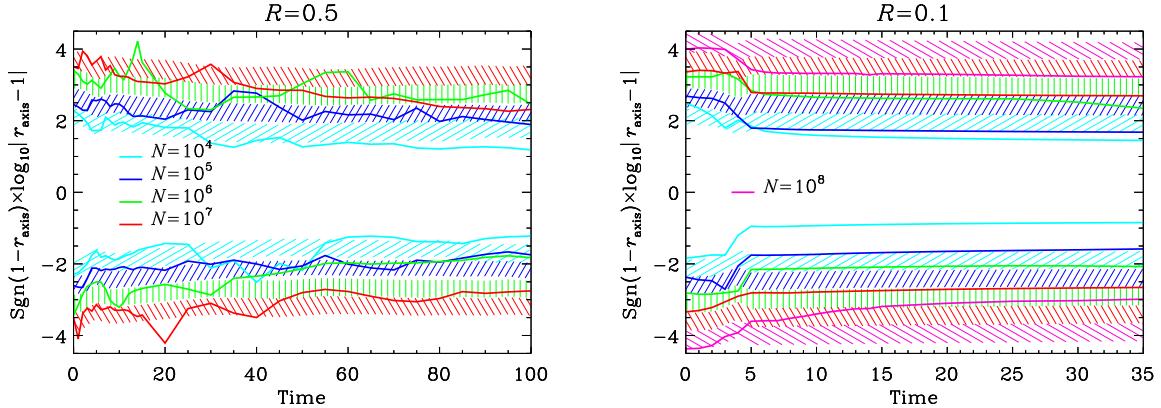
#### 5 PHASE-SPACE DENSITY: VISUAL INSPECTION

Figure 2 shows  $f(r, v, j \simeq 0.244)$  at various times for the `Vlasolve` and `Gadget` simulations corresponding to the warm case,  $R = 0.5$ . Figure 3 is analogous to Fig. 2, except that the quantity displayed is the integral of  $f$  over angular momentum space:

$$f_{\text{summed}}(r, v) = \int f(r, v, j) 2\pi j dj. \quad (11)$$

This allows one to distinguish better the details in the  $N$ -body simulations with less particles.

The level of agreement between the  $N$ -body and the semi-Lagrangian simulations is striking, even in regions where instabilities develop. Of course, this statement has to be tempered with the fact that the `Vlasolve` simulations are subject to significant diffusion, which smears out fine details of the phase-space distribution function. This diffusion effect is clearly visible at  $t = 50$ , when comparing the outer filamentary structures observed in the Vlasov simulations to the  $N$ -body result. Putting aside this coarse graining effect, the structures are exactly similar in both the  $N$ -body and Vlasov simulations at  $t \leq 50$ , including small gaps in the phase-space distribution function related to nonlinear instabilities that start building up. These instabilities grow furthermore at later time. They are considerably smeared out in the  $(1024, 1024, 512)$  `Vlasolve` simulation but unquestion-



**Figure 1.** Deviations from spherical symmetry: ratios of the eigenvalues of the inertia tensor of the system as functions of time in the **Gadget** simulations. To emphasize the differences from unity, which are expected to be small, the quantity  $\text{sgn}(1 - r_{\text{axis}}) \log_{10} |1 - r_{\text{axis}}|$  is plotted as a function of time, where  $r_{\text{axis}} = b/a$  (upper curves on each panel) or  $b/c$  (lower curves) and  $a \leq b \leq c$  are the eigenvalues of the inertia tensor of the **Gadget** particle distribution. Each color corresponds to a given value of the number  $N$  of particles as indicated the panels. In addition, there are dashed regions which correspond to the one sigma confidence level zone expected for a particle distribution locally Poisson sampling the spherically symmetrical projected density profile  $\rho(r, t)$ , where  $\rho(r, t)$  is estimated from interpolation of the **Gadget** particle distribution in spherical shells. To calculate the average of  $r_{\text{axis}}$  and the associated one sigma error contours, 100 local Poisson realizations have been performed for each snapshot and value of  $N$  considered, except for  $N = 10^7$  and  $N = 10^8$  (on right panel only for the latter). In the last cases, the dashes correspond to an extrapolation of the results obtained from  $N = 10^6$ .

ably present. Adding resolution in  $(r, v)$  space (at the cost of resolution in  $j$ ) improves the results, which confirms that the instabilities observed in the **Gadget** simulations are physical and not of numerical nature.

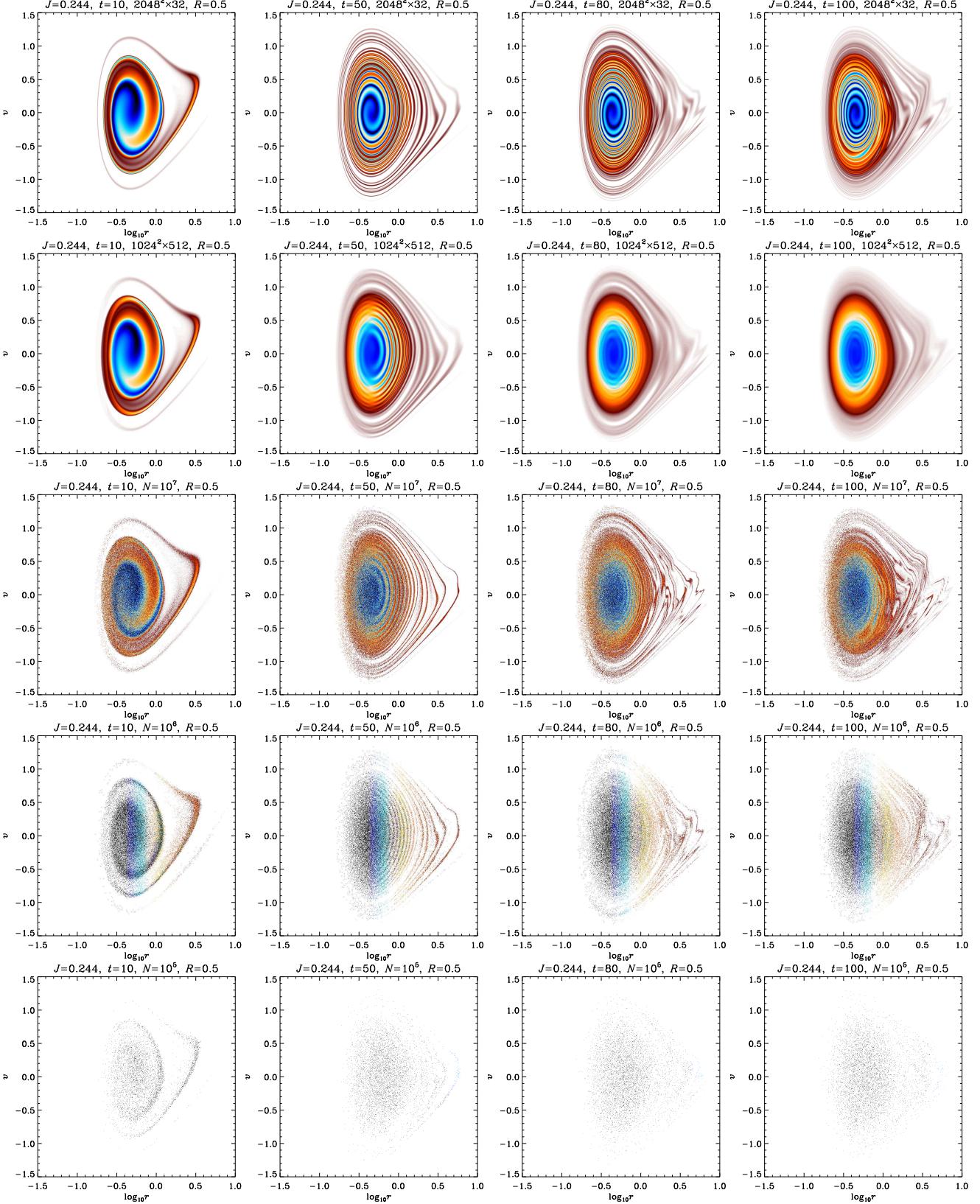
It is worth noticing, when examining Fig. 2, that lowering the number of particles seems, at least from the visual point of view, to have a dynamical effect somewhat similar to coarse graining, with a loss of fine details but not of the global shape of the phase-space density features. Note also that using a small number of slices in  $j$  in the Vlasov solver does not seem to alter the dynamical properties of the system despite the considerable level of aliasing it introduces.

Things get more complicated when considering a colder initial configuration with  $R = 0.1$ , as can be observed on Figs. 4 and 5. Indeed, while, at early times,  $t \lesssim 10$ , the conclusions obtained for  $R = 0.5$  still stand, some instabilities appear as soon as  $t \gtrsim 10$  in the **Gadget** simulations with  $N \leq 10^6$  particles as well as the (2048, 2048, 32) Vlasov run. At this point of time, the  $N \geq 10^7$  particles and the (1024, 1024, 512) simulations agree perfectly with each other (modulo the smearing effects already discussed above) and present a smooth phase-space density without any sign of instability, at variance with the other simulations, where the phase-space density is slightly irregular. This is best exemplified by Fig. 4, although not yet obvious in the (2048, 2048, 32) Vlasov simulation. These irregularities appear as well in the  $N \geq 10^7$  simulations but at later times and then develop in a dramatic way. A careful inspection of successive snapshots of the simulations indeed shows that the moment of appearance of these irregular patterns indeed increases with  $N$ .

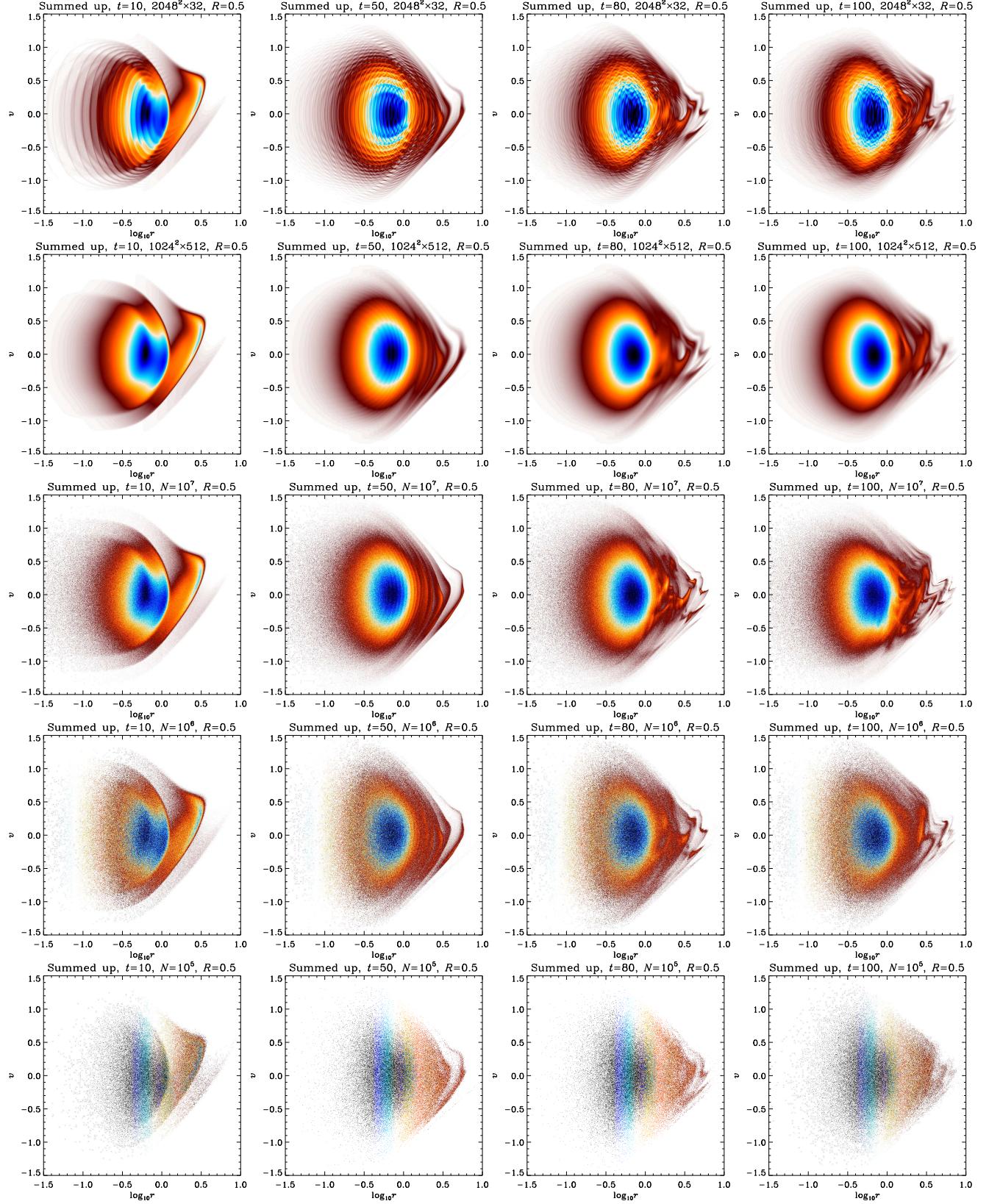
As discussed in appendices B and A3, these instabilities are due the discrete nature of the system in the  $N$ -body case or the aliasing effects introduced by sparse-sampling the angular momentum space in the Vlasov code. The numerical nature of these instabilities is also suggested by the fact that

their pattern changes completely from one simulation to another, unlike the  $R = 0.5$  case studied previously. As shown in Appendix B, they are rather insensitive to the choice of softening, time step or parameters controlling treecode force accuracy calculation in **Gadget**. They can therefore be reduced only by respectively increasing the number of particles and the resolution in the **Gadget** and **Vlasolve** simulations.

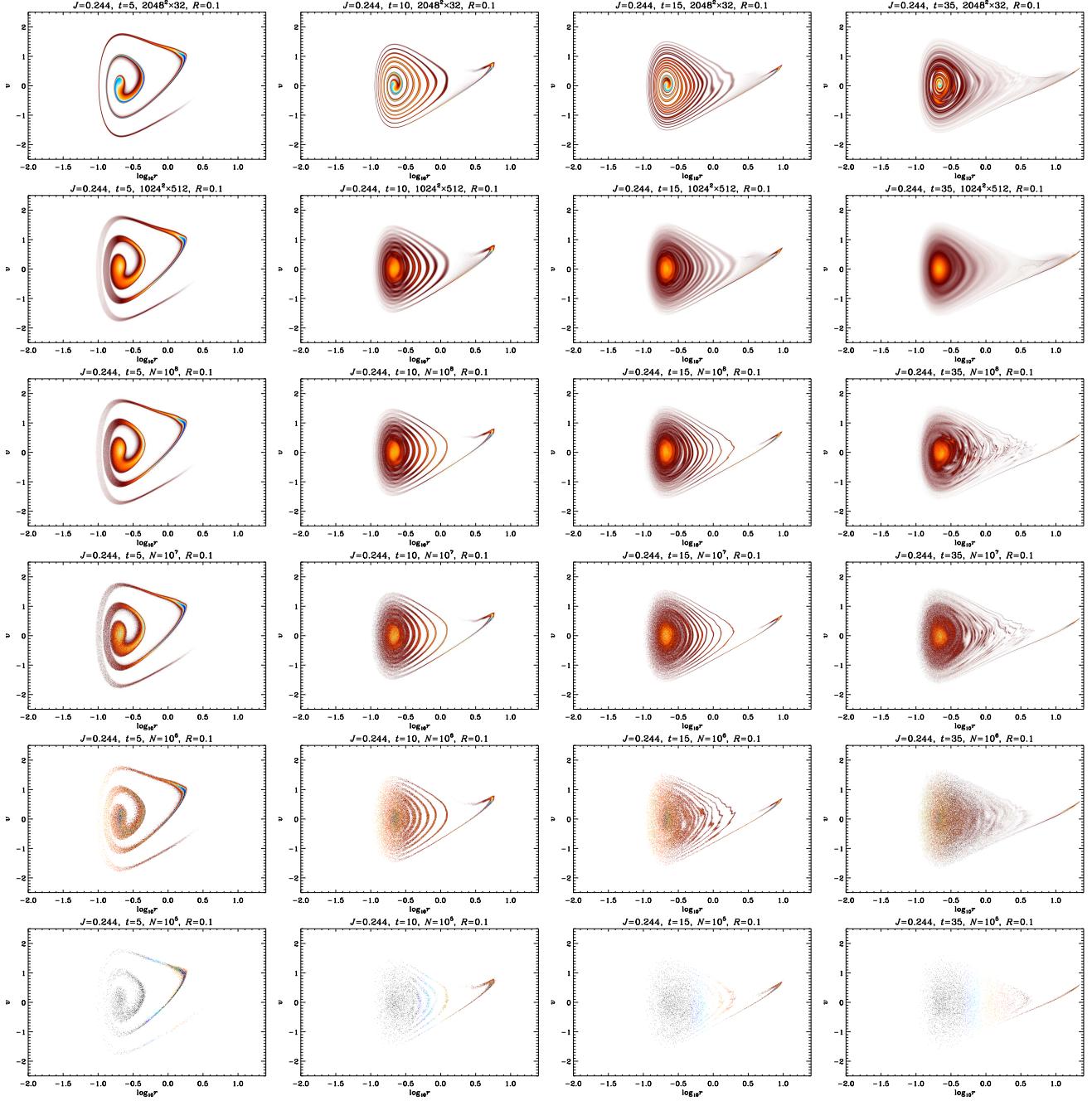
It is important to notice that even  $10^8$  particles might be insufficient to describe properly the system at late times. In the (1024, 1024, 512) Vlasov simulation, the phase-space distribution function seems to be rather smooth at all times and the system free of instability, contrarily to the  $R = 0.5$  case. However, it is difficult at this point to know if actual physical instabilities build up at late times in the  $R = 0.1$  case, because diffusion in the Vlasov simulation might prevent the appearance of some unstable modes. While the instabilities observed in Figs. 4 and 5 are definitely of numerical nature, the very fact that they appear so easily suggests that the system is prone to react nonlinearly to small perturbations. Uneven gaps between the filaments of the phase-space density clearly appear at  $t = 15$  (third column of Fig. 4), even in the (1024, 1024, 512) Vlasov simulation, and one might expect that they correspond to seeds of instabilities. In this respect, the system might actually develop, at some point, physical unstable modes. Yet, even with our 100 millions particles simulation, it is not clear whether these unstable mode dominate over collective effects due to discreteness. Only an appropriate convergence study showing that the pattern produced by the instability does not change with the number of particles, similarly as obtained for the  $R = 0.5$  case, would be conclusive.



**Figure 2.** Vlasolve versus Gadget in phase-space: phase-space density for  $R = 0.5$  and averaged over  $j \in I_J = [0.225, 0.264]$ . Each column of panels corresponds to a given value of time  $t$ , increasing from left to right. The first two lines of panels display  $f(r, v, j)$  for Vlasolve simulations with  $(N_r, N_v, N_j) = (2048, 2048, 32)$  and  $(1024, 1024, 512)$  respectively, while the three bottom lines correspond to the  $N$ -body simulations, with various values of the number of particles  $N$  as indicated on each panel. Note that the Vlasolve simulation with  $(N_r, N_v, N_j) = (2048, 2048, 32)$  has only one angular momentum slice,  $J = 0.244$ , in the interval  $I_J$ , so there is no blurring of the filamentary details of  $f(r, v, j)$  on the left side of the peak of the distribution function contrarily to the other cases. In the  $N$ -body case,  $f(r, v, j)$  was computed on the same mesh as the  $(1024, 1024, 32)$  Vlasolve simulation using nearest grid point interpolation, which explains the artefacts on the color pattern in the last two lines of panels.



**Figure 3.** Same as in Fig. 2, but the phase-space distribution function has now been summed up over the whole available range of values of  $j \in [0, J_{\max} = 1.6]$ , where  $J_{\max}$  is the maximum sampled value of  $j$  for the `Vlasolve` simulations.



**Figure 4.** Same as in Fig. 2, but for a colder initial configuration with virial ratio  $R = 0.1$ . There is also an additional line of panels corresponding to the `Gadget` simulation with  $N = 10^8$  particles.

## 6 STATISTICAL ANALYSIS

### 6.1 Correlators and entropic estimators: definitions and concepts

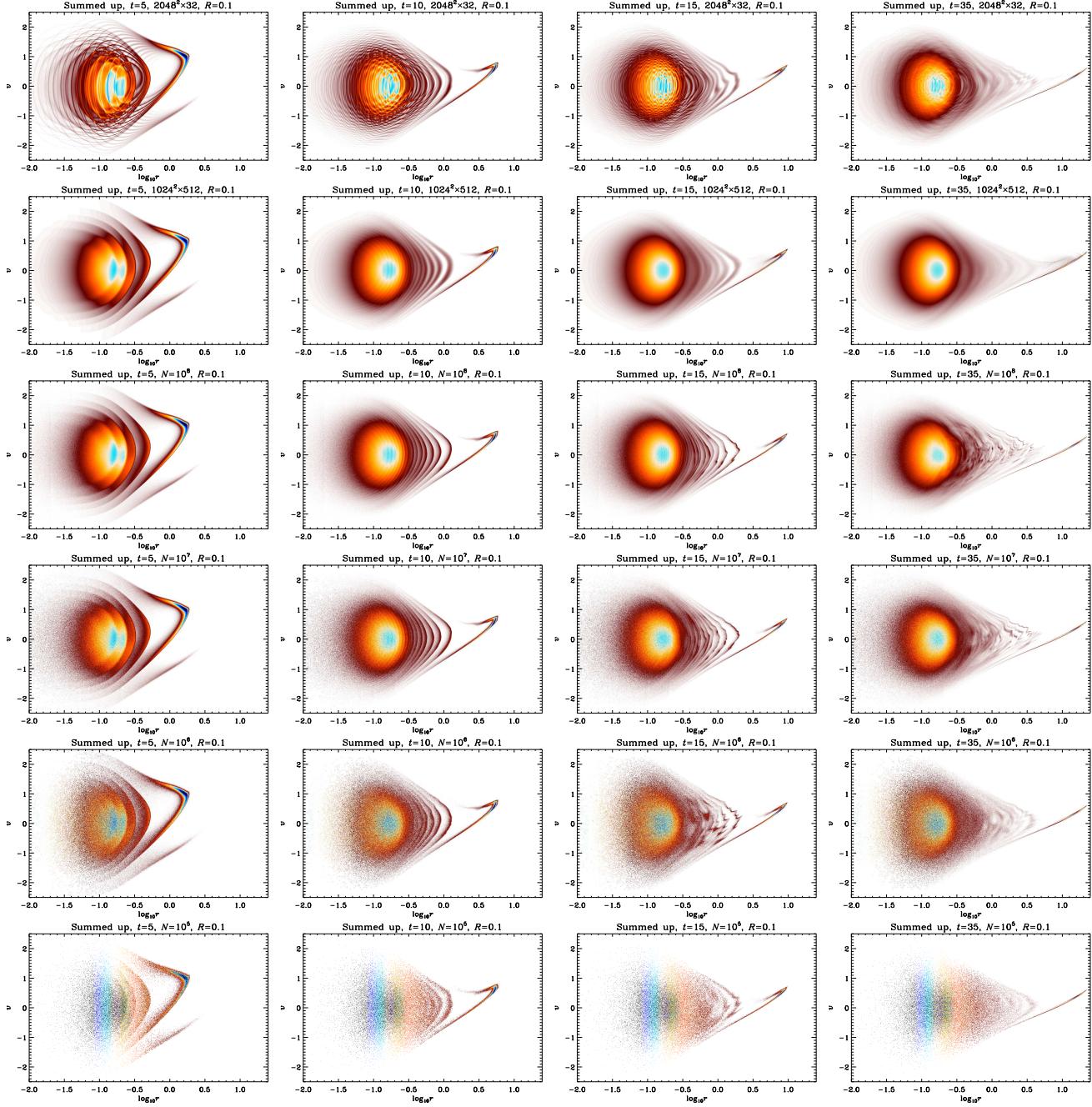
To perform a more accurate analysis, one can try to quantify to which extent the particle distribution in the  $N$ -body simulations can be considered as a local Poisson process of the phase-space density calculated in the semi-Lagrangian code. To do so, we use, in addition to entropic measurements described further, the following correlators,

$$C_k \equiv \frac{\mu_k}{\kappa_k}, \quad (12)$$

$$\mu_k = \frac{M}{N} \sum_{i=1}^N [f(\Omega_i)]^k, \quad (13)$$

$$\kappa_k = \int [f(\Omega)]^{k+1} d\Omega, \quad (14)$$

where  $k$  is a positive integer,  $f$  the `Vlasolve` phase-space density,  $M$  the total mass,  $d\Omega \equiv 2\pi dr \times dv \times jdj$  and  $\Omega_i = (r_i, v_i, j_i)$  where  $r_i$ ,  $v_i$  and  $j_i$  are respectively the radial position, radial velocity and angular momentum of each particle of the `Gadget` simulation. For a point set randomly sampling a smooth density distribution  $g$ , the probability density  $p(\Omega)$  of having a given particle at phase-space posi-



**Figure 5.** Same as in Fig. 3, but for  $R = 0.1$  and with the additional  $N$ -body simulation with  $N = 10^8$ .

tion  $\Omega$  is independent from the rest of the particle distribution and is simply proportional to  $g(\Omega)$ :

$$p(\Omega)d\Omega = \frac{g(\Omega)}{M}d\Omega. \quad (15)$$

The density probability of having  $N$  particles at respective positions  $\Omega_1, \Omega_2, \dots, \Omega_N$  is given by

$$\mathcal{P}(\Omega_1, \dots, \Omega_N) = \prod_{i=1}^N p(\Omega_i). \quad (16)$$

Ensemble averaging of  $\mu_k$  under the law  $g$  then reads

$$\langle \mu_k \rangle_g = \frac{M}{N} \int \sum_{i=1}^N [f(\Omega_i)]^k \mathcal{P}(\Omega_1, \dots, \Omega_N) d\Omega_1 \cdots d\Omega_N, \quad (17)$$

$$= \frac{M}{N} \sum_{i=1}^N \int_{\Omega} \frac{1}{M} [f(\Omega_i)]^k g(\Omega_i) d\Omega_i, \\ \times \prod_{j \neq i} \int_{\Omega_j} \frac{g(\Omega_j)}{M} d\Omega_j \quad (18)$$

$$= \int [f(\Omega)]^k g(\Omega) d\Omega, \quad (19)$$

and

$$\langle C_k \rangle_g = \frac{\int [f(\Omega)]^k g(\Omega) d\Omega}{\int [f(\Omega)]^{k+1} d\Omega}. \quad (20)$$

Hence, if the distributions  $g$  and  $f$  coincide, i.e., in our case, if **Gadget** actually Poisson samples the **Vlasolve** phase-space density, one obtains  $\langle C_k \rangle_{g=f} = 1$  after ensemble averaging.

When increasing  $k$ , more weight is given to regions in phase-space corresponding to larger values of  $f$ . For a point process totally anticorrelated with  $f$ ,  $C_k$  cancels, while its largest possible value is given by  $C_k = (M \max f^k)/\kappa_k > 1$ , when all the points stay in the region where  $f$  is maximal.

An important issue is to compute properly the origin of coordinates in the **Gadget** simulations, which do not necessarily conserve total momentum. In order to do this, we find the origin of coordinates which *maximizes*  $C_1$ , even though the result of such a procedure can potentially lead to  $C_1 > 1$ , to optimize the match between concentrations of particles and local extrema of  $f$ .

The variance of  $C_k$  can also be calculated in an analogous way to  $\langle \mu_k \rangle_g$ :

$$\Delta C_k^2 \equiv \langle C_k^2 \rangle_g - \langle C_k \rangle_g^2 \quad (21)$$

$$= \frac{1}{\kappa_k^2} \left[ \frac{M}{N} \langle \mu_{2k} \rangle_g - \frac{1}{N} \langle \mu_k \rangle_g^2 \right], \quad (22)$$

which reduces to  $\Delta C_k^2 = (M/N)(\kappa_{2k}/\kappa_k^2) - 1/N$  when  $f$  and  $g$  coincide. In practice, we shall use the following estimator for this statistical error:

$$\Delta C_k^2 \simeq \frac{1}{\kappa_k^2} \left[ \frac{M}{N} \mu_{2k} - \frac{1}{N} \mu_k^2 \right], \quad (23)$$

where  $\mu_{2k}$  and  $\mu_k$  are directly estimated from the  $N$ -body simulation.

The log-likelihood that the **Gadget** particle distribution locally Poisson samples the **Vlasolve** phase-space density  $f$  can be written, following the reasoning that lead to equation (16),

$$\ln \mathcal{L} = \sum_{i=1}^N \ln \left[ \frac{f(\Omega_i)}{M} \right]. \quad (24)$$

However, the region  $\mathcal{D}$  where  $f > 0$  being of finite extent, one expects  $\ln \mathcal{L} = -\infty$  as soon as a particle escapes  $\mathcal{D}$ , which is very likely, due for instance to  $N$ -body relaxation. Furthermore, the Vlasov solver does not guaranty the positivity of  $f$ . To take into account in a fair way both the defects of the  $N$ -body and the Vlasov simulations, it is better to restrict to a region  $\mathcal{D}_{\text{th}}$  where  $f$  is strictly positive:

$$\mathcal{D}_{\text{th}} \equiv \{\Omega \text{ such that } f(\Omega) \geq f_{\text{th}}, f_{\text{th}} > 0\}. \quad (25)$$

The log-likelihood of having  $Q \leq N$  particles in the region  $\mathcal{D}_{\text{th}}$  and the rest outside it (leaving the freedom of the remaining particles to span all the space outside  $\mathcal{D}_{\text{th}}$ ) is given by a binomial law

$$\ln \mathcal{L}_b(Q, \nu) = \ln \left[ \frac{N!}{(N-Q)!Q!} \nu^M (1-\nu)^{N-Q} \right], \quad (26)$$

where  $\nu$  is the fractional mass inside  $\mathcal{D}_{\text{th}}$  in the **Vlasolve** simulation. Hence, equation (24) simply becomes

$$\ln \mathcal{L} = \sum_{\Omega_i \in \mathcal{D}_{\text{th}}} \ln \left[ \frac{f(\Omega_i)}{M_{\text{th}}} \right] + \ln \mathcal{L}_b(Q_{\text{th}}, \nu), \quad (27)$$

where  $Q_{\text{th}}$  is the number of particles of the **Gadget** simulation inside  $\mathcal{D}_{\text{th}}$  and  $M_{\text{th}} = \int_{\mathcal{D}_{\text{th}}} d\Omega f(\Omega)$ .

Note that the distribution of particles which maximizes the first term in equation (27) corresponds again to the case where all the particles of  $\mathcal{D}_{\text{th}}$  stay in the region where  $f$  is maximal, similarly to the case when the correlator  $C_k$  is equal to its maximum possible value. Clearly, this situation is not typical, but it is in fact the most likely to consider when it can take place: this is why we maximize  $C_1$  to estimate the center of the  $N$ -body system, even though it might turn to be larger than unity.

The expectation value of the log-likelihood  $\ln \mathcal{L}$  under the law  $f$  can be obtained by ensemble averaging:

$$S(f_{\text{th}}) \equiv -\frac{1}{\nu N} \langle \ln \mathcal{L} \rangle_f = S_f(f_{\text{th}}) + S_b(f_{\text{th}}), \quad (28)$$

$$S_f(f_{\text{th}}) \equiv - \int_{\mathcal{D}_{\text{th}}} \frac{f(\Omega)}{M_{\text{th}}} \ln \left[ \frac{f(\Omega)}{M_{\text{th}}} \right] d\Omega, \quad (29)$$

$$S_b(f_{\text{th}}) \equiv -\frac{1}{\nu N} \sum_{Q=0}^N \mathcal{L}_b(Q, \nu) \ln \mathcal{L}_b(Q, \nu). \quad (30)$$

In the limit  $f_{\text{th}} \rightarrow 0$ , the quantity  $S_f(f_{\text{th}})$  reduces to the Gibbs entropy of the system, which explains the choice of notations. Moreover, if  $N \gg 1$  and if the fractional mass  $\nu$  inside the domain of interest  $\mathcal{D}_{\text{th}}$  is of order of unity, which is the case for our analyses, the term  $S_b(f_{\text{th}})$  is in practice negligible compared to  $S_f(f_{\text{th}})$ , so  $S(f_{\text{th}})$  depends only weakly on the total number of particles, as expected.

The variance of  $\ln \mathcal{L}$  can be calculated likewise

$$\sigma_L^2 \equiv \frac{1}{(\nu N)^2} [\langle \ln \mathcal{L}^2 \rangle_f - \langle \ln \mathcal{L} \rangle_f^2] \quad (31)$$

$$\simeq \frac{1}{\nu N} \left\{ \int_{\mathcal{D}_{\text{th}}} \frac{f(\Omega)}{M_{\text{th}}} \ln^2 \left[ \frac{f(\Omega)}{M_{\text{th}}} \right] d\Omega - \nu [S(f_{\text{th}})]^2 \right\}, \quad (32)$$

where we have neglected, following the arguments developed earlier, the contributions of  $S_b$  to the error.

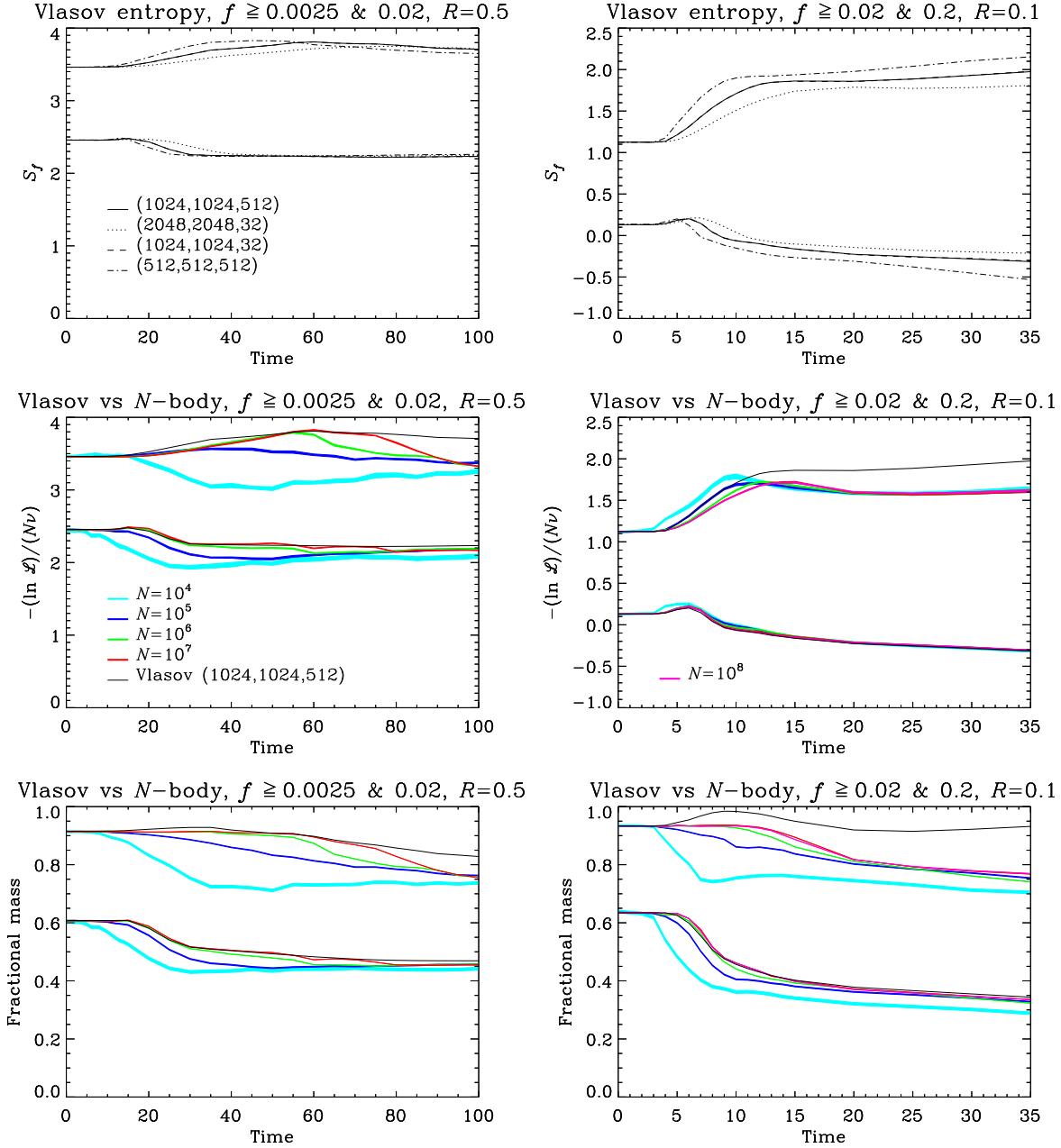
To understand better the interest of using the statistics given by equation (27), one can introduce the difference between the measured value of the log-likelihood and its expectation under the law  $f$ :

$$\delta S = \frac{1}{\nu N} [\langle \ln \mathcal{L} \rangle_f - \ln \mathcal{L}], \quad (33)$$

where  $\mathcal{L}$  is given by expression (27) calculated for  $\Omega_i$  extracted from a **Gadget** simulation. The quantity  $\delta S$  estimates the magnitude of the difference between the underlying smooth phase-space density  $g$  sampled by **Gadget** and the **Vlasolve** phase space density,  $f$ : its ensemble average other many **Gadget** realizations indeed reads, when neglecting the binomial term in equation (27),

$$\langle \delta S \rangle_g \simeq \int_{f \geq f_{\text{th}}} \frac{1}{M_{\text{th}}} [g(\Omega) - f(\Omega)] \ln \left[ \frac{f(\Omega)}{M_{\text{th}}} \right] d\Omega. \quad (34)$$

Under the assumption that the  $N$ -body simulation Poisson samples the distribution  $f$ , the magnitude of  $\delta S$  should be of the same order of  $\sigma_L$ .



**Figure 6.** Entropic measurements: effects of **Vlasolve** resolution (top two panels) and **Gadget** number of particles (four bottom panels). The left and right panels correspond respectively to  $R = 0.5$  and  $0.1$ . On the *top panels*, the quantity  $S_f(f_{\text{th}})$  given by equation (29) is plotted as a function of time for the Vlasov simulations and for two values of  $f_{\text{th}}$  indicated on each panel corresponding to approximately initially keeping 90 and 60 percent of the mass inside the excursion. Each curve corresponds to a given resolution as indicated on each panel (the dashes are nearly superposed to the solid line). The top/bottom group of four curves correspond to a smaller/larger value of  $f_{\text{th}}$ . On the *middle panels*, the solid line is the same as on the top panels, while the colored curves display, for each value of the particle number  $N$  in the **Gadget** simulations, the quantity  $-\ln \mathcal{L}/(N\nu)$  as a function of time, where  $\ln \mathcal{L}$  is given by equation (27). If the  $N$ -body simulations would Poisson sample the **Vlasolve** phase-space density, the ensemble average of this quantity over many **Gadget** realizations should match the solid line (except for a negligible correction due to the  $S_b$  term in equation 29). Finally, the *bottom panels* show the fractional mass as a function of time for the two values of  $f_{\text{th}}$  considered. On the two bottom right panels, there is an additional purple curve nearly indistinguishable from the red one, corresponding to the additional simulation with 100 millions particles we performed for  $R = 0.1$ . In the four bottom panels, the thickness of each colored curve takes into account statistical errors (equation 32 for  $\ln \mathcal{L}$ ). In addition, for the middle panels, systematic errors due to the interpolation of the phase-space distribution function in the **Vlasolve** simulations also contribute to the estimated errors. In the latter case, we compute  $f(\Omega_i)$  both using nearest grid point and linear interpolation from the values of  $f$  on the computational mesh. The difference between the two interpolating methods adds to the thickness of the curves. Note that we use the (1024,1024,512) **Vlasolve** simulation to perform the comparison to  $N$ -body results, to minimize the effects of interpolation.

## 6.2 Correlators and entropic estimators: measurements

Top panels of Fig. 6 show the quantity  $S_f(f_{\text{th}})$  as a function of time for the various **Vlasolve** simulations we performed and two values of  $f_{\text{th}}$  chosen such that approximately 90 percent and 60 percent of the total mass is initially inside the excursion  $D_{\text{th}}$ , respectively. The quantity  $S_f(f_{\text{th}})$  is a Casimir invariant –that is an integral over a function of  $f$ – and should thus be conserved during runtime if the code was perfect. This is not the case because of diffusion and aliasing effects in  $(r, v)$  space: deviation from conservation of  $S_f$  happens shortly after collapse time. Then there is a strong mixing phase during which  $S_f$  increases, then possibly decreases, according to the value of  $f_{\text{th}}$ , and finally reaches an approximate plateau. Deviation from conservation of  $S_f$  naturally happens sooner when resolution in  $(r, v)$  space is smaller. Resolution in  $j$  space does not have much influence on  $S_f$  because angular momentum is an invariant of the dynamics. However, as clearly shown in previous section for  $R = 0.1$ , we already know that sparse sampling in  $j$  space is not recommended since it can introduce some instabilities in the dynamics, even though this effect does not affect much our likelihood measurements.

Middle panels of Fig. 6 show the quantity  $-\ln \mathcal{L}/(N\nu)$  measured in **Gadget** from the particles belonging to the excursion  $D_{\text{th}}$  as a function of time, where  $\ln \mathcal{L}$  is given by equation (27). For a given value of the threshold  $f_{\text{th}}$ , if the **Gadget** simulations would actually behave like Poisson realizations of the **Vlasolve** ones, all the colored curves should be close to the solid line, which corresponds to  $S_f$ . This is clearly not the case for small  $f_{\text{th}}$  (upper group of curves), except at early times. Increasing the number of particles in the  $N$ -body simulation improves the agreement with the Vlasov code in the  $R = 0.5$  case, except at the latest times, but does not seem to have a convincing impact in the  $R = 0.1$  case: for  $f_{\text{th}} = 0.02$ , all the  $N$ -body simulations converge to the same plateau somewhat below the Vlasov code result. On the contrary, for  $f_{\text{th}} = 0.2$ , the agreement between **Gadget** and **Vlasolve** is striking at all times, except may be for the  $N = 10^4$  simulation during the strong mixing phase.

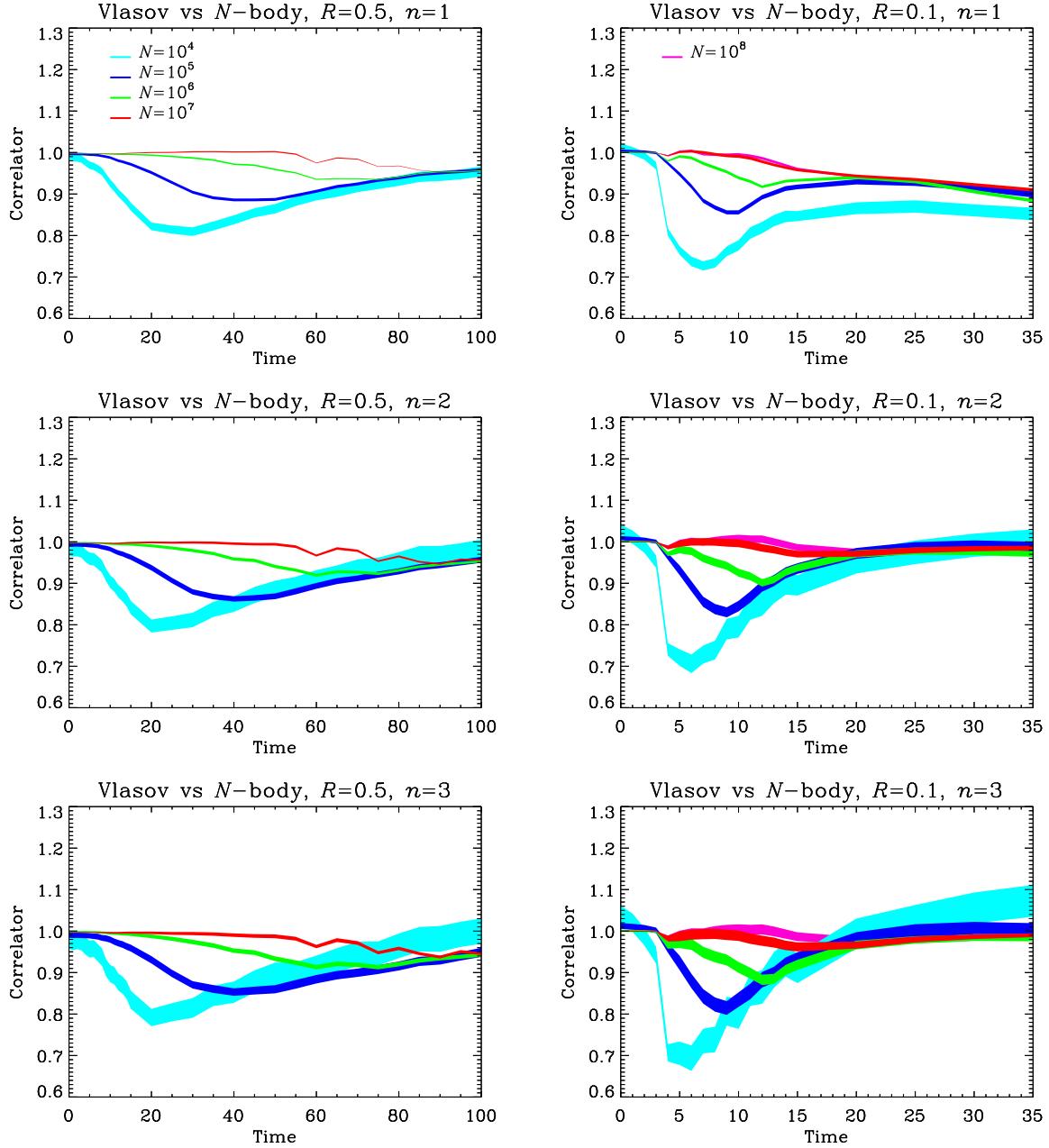
To complete the analyses and understand better the results obtained for the log-likelihood, the fractional mass inside the excursions  $f \geq f_{\text{th}}$  is shown in bottom panels of Fig. 6. Again, this quantity is a Casimir, so it should not change with time in the idealistic case. In practice, aliasing might artificially increase or decrease the **Vlasolve** mass inside  $D_{\text{th}}$ , while diffusion effects decrease it, especially by dilution of filamentary structures that build up during the course of dynamics. In the  $R = 0.5$  case, most of the disagreement between **Gadget** likelihood and its expectation given by **Vlasolve** can be understood in terms fractional mass: defects related to the discrete nature of the  $N$ -body simulations seem to eject particles outside  $D_{\text{th}}$ . However, even though they seem spectacular in terms of fractional mass in the excursion, these defects are in practice subtle because they do not affect significantly the projected density  $\rho(r)$  (that we do not show here, for simplicity), even for  $N = 10^4$ . Remind however that we noticed, during visual inspection of Fig. 3 some possible smearing of the dynamics in the  $N$ -body simulations similar to diffusion effects in the Vlasov code.

In the  $R = 0.1$  case, the interpretation of the results is slightly more complicated. For  $f_{\text{th}} = 0.2$ , the **Gadget** fractional mass inside the excursion  $D_{\text{th}}$ , which is simply proportional to the number of particles inside it, behaves similarly as in the  $R = 0.5$  case as a function of particle number. On the other hand, when examining the quantity  $-\ln \mathcal{L}/(N\nu)$ , which is a sum of terms proportional to  $\ln f(\Omega_i)$  over particle positions  $\Omega_i \in D_{\text{th}}$ , the  $N$ -body measurements converge with each other and with **Vlasolve** much better, especially after relaxation. This means that particles inside the excursion tend to concentrate in regions where  $f$  is larger to compensate for particles send outside the excursion, a possible signature of collisional relaxation. But it could also be attributable to non trivial collective effects related to particle Poisson noise already discussed in § 5. Remind however, that in the  $N = 10^4$  case, deviation from spherical symmetry is significant and certainly contributes to the mass loss inside the excursion. For  $f_{\text{th}} = 0.02$ , even the  $N = 10^8$  **Gadget** sample disagrees with the **Vlasolve** simulation. Clearly, the Vlasov simulation becomes quickly defective in regions where  $f$  is small. However, convergence of the **Gadget** simulations at late times might be misleading, as we noticed from visual inspection of Figs. 4 and 5 that some instabilities appeared in all of them as soon as  $t \gtrsim 15$ , although later when  $N$  is larger. Interestingly, the measurements in the  $N = 10^7$  and  $N = 10^8$  simulations are nearly indistinguishable from each other, which is a sign that we are close to numerical convergence.

Entropic measurements of Fig. 6 are confirmed, at least partly, by Fig. 7. In particular, a depression of which the depth depends on the number of particles in the  $N$ -body simulation appears on all the curves. When increasing  $N$ , the amplitude of the depression decreases and the occurrence of its maximum amplitude is delayed, independently of the actual dynamical state of the physical system. Again, it can certainly be attributed to collisional relaxation or to collective effects due to Poisson noise. Overall agreement between  $N$ -body and Vlasov codes improves when increasing the number of particles in the  $N$ -body simulation. For  $R = 0.5$ , this is rather independent of  $k$  in equation (12), i.e. of the fact of putting more or less weight to overdense regions in phase-space. In the  $R = 0.1$  case, putting aside the depression of which the depth depends on the number of particles, the correlator  $C_1$  starts to decrease with time at  $t \sim 10$ . This can be mainly attributed to defects in the Vlasov simulation in underdense regions as discussed earlier. For  $k \geq 2$ , which gives more weight to higher values of the phase-space density, the correlator stays steady as a function of time (again putting aside the  $N$ -dependent depression). However, one notices for  $k = 3$  a net increase with time of the correlator for the simulation with  $N = 10^4$  particles, suggesting possible effects of collisional relaxation at late times, but recall that this simulation presents significant deviations from spherical symmetry.

## 7 CONCLUSION

In this paper we have compared the phase-space distribution function traced by the particle distribution in **Gadget** simulations to the results obtained with a Vlasov code relying on an improved version of the splitting algorithm of Fujiwara



**Figure 7.** Correlators as functions of time between **Vlasolve** and **Gadget**. These quantities, defined in equations (12), (13) and (14), are plotted for  $k = 1, 2, 3$  increasing from top to bottom, while left and right panels correspond to  $R = 0.5$  and  $0.1$ , respectively. The thickness of the curves, analogously to Fig. 6, takes into account statistical errors (according to eq. 23, using the measured value of  $\nu_{2k}$  and  $\nu_k$ ) and systematic errors due to the interpolation of the phase-space density in the **Vlasolve** samples. Note that there is an additional purple curve on each panel of the right column corresponding to the 100 millions particles simulation.

(1983). To conduct the comparison, we used apodized Hénon spheres, which are known to be insensitive to radial orbit instability and in particular to preserve the spherical nature of the system even when simulated with three-dimensional  $N$ -body codes. Two values of the initial virial ratio  $R = 0.5$  and  $R = 0.1$  were considered, corresponding respectively to a “warm” and a “cold” configuration.

We performed a detailed visual inspection of the phase-space distribution function as well as a quantitative analysis by introducing two new statistical tools. The first one is of

entropic nature and corresponds to the log-likelihood quantifying to which extent the  $N$ -body results represent a local Poisson sampling of the Vlasov phase-space density. The second tool is a correlator of order  $k$ , proportional to the integral over phase-space of the product between the Vlasov phase-space density raised to the power  $k$  and the particle distribution function.

The main results of our paper are as follows:

- Visual inspection of the phase-space distribution shows a remarkable agreement between the Vlasov and the  $N$ -body

codes. In the warm case,  $R = 0.5$ , even details related to nonlinear instabilities developing in phase-space during the course of dynamics match. In the colder case,  $R = 0.1$ , although still good at the coarse level, the agreement between **Vlasolve** and **Gadget** at small scales worsens after a few dynamical times. This is mainly due to collective effects induced by the shot noise of the particles in the  $N$ -body simulations (and not to close particle encounters). Even with  $N = 10^8$  particles, we are not sure to have proved convergence of the  $N$ -body results. This proof is however rendered difficult by the fact that the Vlasov code is significantly diffusive, which might prevent the development of a variety of physical unstable modes.

- While the statistical tools do not provide, in fact, as rich information as visual inspection, they can evidence subtle effects not visible by eye. In particular, when taking into account general trends due to diffusion in the Vlasov code, significant for  $R = 0.1$ , we notice that the match between **Gadget** and **Vlasolve** worsens with time, then improves. The amplitude of the mismatch and the time of its appearance respectively augment and decrease when reducing the number of particles in the  $N$ -body simulation. Again, this could be imputed to collective effects due to the shot noise of the particles, however close encounters could in this case also play a role. Yet, the nearly perfect match between the **Gadget** simulations with  $N = 10^7$  and  $N = 10^8$  particles suggests that convergence is nearly reached in terms of number of particles and information theory, even if it is not fully proven.

It is worth mentioning again that the collective effects mentioned above are not related to  $N$ -body relaxation, but rather result from random Poisson fluctuations. This can be formulated as follows (Aarseth, Lin, & Papaloizou 1988): a given particle at some distance  $r$  from the center of the system feels a force proportional to the number  $N_{\text{in}}$  of particles inside the sphere of radius  $r$ . Poisson fluctuations imply thus that there is a relative error of order of  $1/\sqrt{N_{\text{in}}}$  on this force. Importantly, the inner number of particles  $N_{\text{in}}$  changes with time with random fluctuations around the mean behavior: these fluctuations can be considered as a (correlated) random walk. Indeed, because of the finite velocity dispersion, particles cross both inwards and outwards the frontier of the sphere of radius  $r$ .

Therefore, a coherent flow of particles will tend to fuzz radially because of the fluctuations of the force. This is probably the effect we noticed for low  $N$  during the visual inspection of Fig. 3. According to Aarseth, Lin, & Papaloizou (1988), this *collective* effect is dominant over  $N$ -body relaxation, and, as confirmed by our detailed numerical tests in Appendix B, is not significantly influenced by softening. This effect is probably also at the origin of instabilities observed in the  $R = 0.1$   $N$ -body simulations. It could as well be at the origin of the deviations between **Vlasolve** and **Gadget** observed when measuring the statistical estimators defined in this paper, but it is again not obvious how much close encounters actually contribute to these deviations.

Note as well that shot noise creates anisotropies in the system, i.e. deviations from spherical symmetry that can get eventually amplified. Aarseth, Lin, & Papaloizou (1988) argue that this effect is subdominant compared to the radial component of the noise induced perturbation. Although their calculation are performed only prior to collapse and in

the cold case, we believe that the general conclusions still stand here.

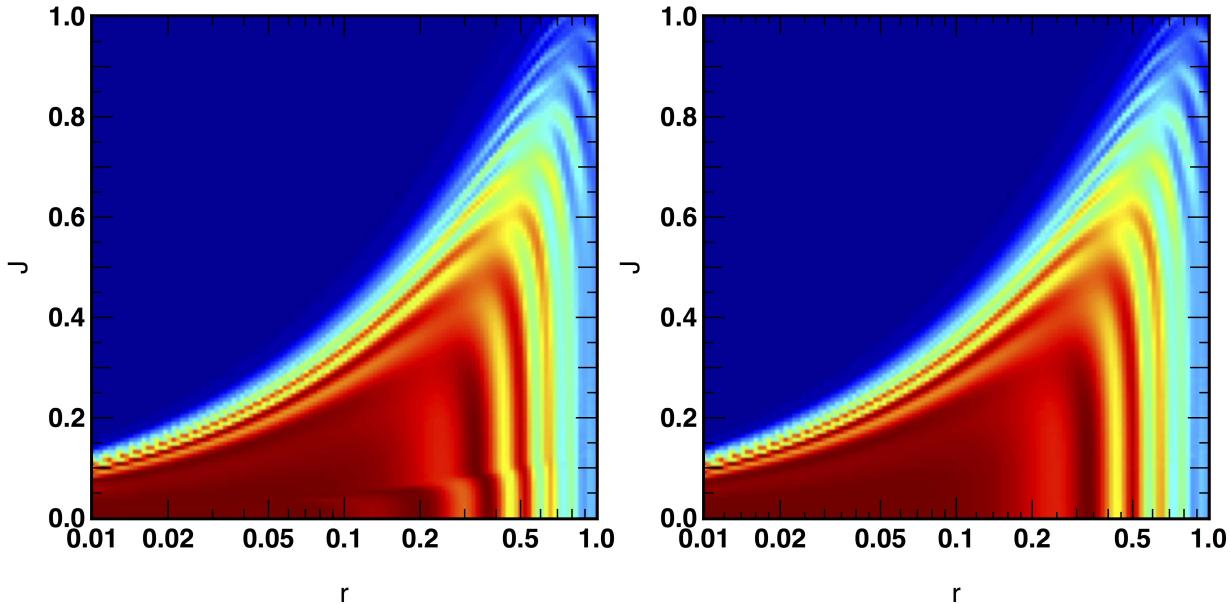
Clearly, this collective effect is a real problem for simulations of close to cold spherical systems when it comes to examine fine structures of the phase-space density. We could not prove convergence of the phase-space density in the  $R = 0.1$  case even for a 100 millions particles simulation. This might, by extension, have non trivial consequences on the fine structure of simulated dark matter halos, where numerical convergence in terms of number of particles might not have been reached yet despite the numerous convergence studies existing. Indeed, convergence toward the continuous limit might be much slower than expected, hence giving the false impression that it is achieved.

## ACKNOWLEDGEMENTS

We thank Jérôme Perez for useful discussions. This work has been funded in part by ANR grant ANR-13-MONU-0003.

## REFERENCES

- Aarseth S. J., Lin D. N. C., Papaloizou J. C. B., 1988, ApJ, 324, 288  
 Alard C., Colombi S., 2005, MNRAS, 359, 123  
 Barnes E. I., Lanzel P. A., Williams L. L. R., 2009, ApJ, 704, 372  
 Bertschinger E., 1998, ARA&A, 36, 599  
 Besse N., Sonnendrücker E., 2003, JCoPh, 191, 341  
 Besse N., Latu G., Ghizzo A., Sonnendrücker E., Bertrand P., 2008, JCoPh, 227, 7889  
 Binney J., 2004, MNRAS, 350, 939  
 Campos Pinto M., 2011, arXiv, arXiv:1112.1859  
 Cheng C. Z., Knorr G., 1976, JCoPh, 22, 330  
 Colombi S., 2001, NewAR, 45, 373  
 Colombi S., Touma J., 2008, CNSNS, 13, 46  
 Colombi S., Touma J., 2014, MNRAS, 441, 2414  
 Crouseilles N., Latu G., Sonnendrücker E., 2009, JCoPh, 228, 1429  
 Crouseilles N., Mehrenberger M., Sonnendrücker E., 2010, JCoPh, 229, 1927  
 Dehnen W., Read J. I., 2011, EPJP, 126, 55  
 Dolag K., Borgani S., Schindler S., Diaferio A., Bykov A. M., 2008, SSRv, 134, 229  
 Filbet F., Sonnendrücker E., Bertrand P., 2001, JCoPh, 172, 166  
 Fujiwara T., 1981, PASJ, 33, 531  
 Fujiwara T., 1983, PASJ, 35, 547  
 Gott, J.R. III, 1973, ApJ, 186, 481  
 Güçlü Y., Christlieb A. J., Hitchon W. N. G., 2014, JCoPh, 270, 711  
 Hahn O., Abel T., Kaehler R., 2013, MNRAS, 434, 1171  
 Hénon M., 1964, AnAp, 27, 83  
 Melott A. L., 2007, arXiv, arXiv:0709.0745  
 Melott A. L., Shandarin S. F., Splinter R. J., Suto Y., 1997, ApJ, 479, L79  
 Moore B., Governato F., Quinn T., Stadel J., Lake G., 1998, ApJ, 499, L5  
 Nishida M. T., Yoshizawa M., Watanabe Y., Inagaki S., Kato S., 1981, PASJ, 33, 567



**Figure 8.** Comparison between the reflecting central sphere method (left panel) and our improved delayed central sphere implementation (right panel). A simulation of a Hénon sphere with  $(N_r, N_v, N_j) = (200, 200, 200)$  and a virial ratio  $R = 0.5$  is shown at  $t = 30$  in the  $(r, u = 0, j)$  plane. The systematic artificial speed increase undergone by orbits that penetrate the central region compared to their higher angular momentum counterparts can clearly be observed at low  $j$  on the left panel where a reflective sphere is used, while the distribution function does not exhibit such spurious features when a delayed kernel is used (right panel).

- Power C., Navarro J. F., Jenkins A., Frenk C. S., White S. D. M., Springel V., Stadel J., Quinn T., 2003, MNRAS, 338, 14  
 Rossmannith J. A., Seal D. C., 2011, JCoPh, 230, 6203  
 Roy F., Perez J., 2004, MNRAS, 348, 62  
 Shandarin S., Habib S., Heitmann K., 2012, PhRvD, 85, 083005  
 Shoucri M. M., Gagne R. R. J., 1978, JCoPh, 27, 315  
 Sonnendrücker E., Roche J., Bertrand P., Ghizzo A., 1999, JCoPh, 149, 201  
 Springel V., 2005, MNRAS, 364, 1105  
 Springel V., et al., 2008, MNRAS, 391, 1685  
 Springel V., Yoshida N., White S. D. M., 2001, NewA, 6, 79  
 Stadel J., Potter D., Moore B., Diemand J., Madau P., Zemp M., Kuhlen M., Quilis V., 2009, MNRAS, 398, L21  
 Umeda T., 2008, EP&S, 60, 773  
 van Albada T. S., 1982, MNRAS, 201, 939  
 Watanabe Y., Inagaki S., Nishida M. T., Tanaka Y. D., Kato S., 1981, PASJ, 33, 541  
 Yoshikawa K., Yoshida N., Umemura M., 2013, ApJ, 762, 116

## APPENDIX A: VLASOV SOLVER: DETAILS ON THE ALGORITHM

### A1 Reflecting boundaries with time delay

In this appendix, we explain how reflecting boundaries conditions with time delay are implemented in **Vlasolve**.

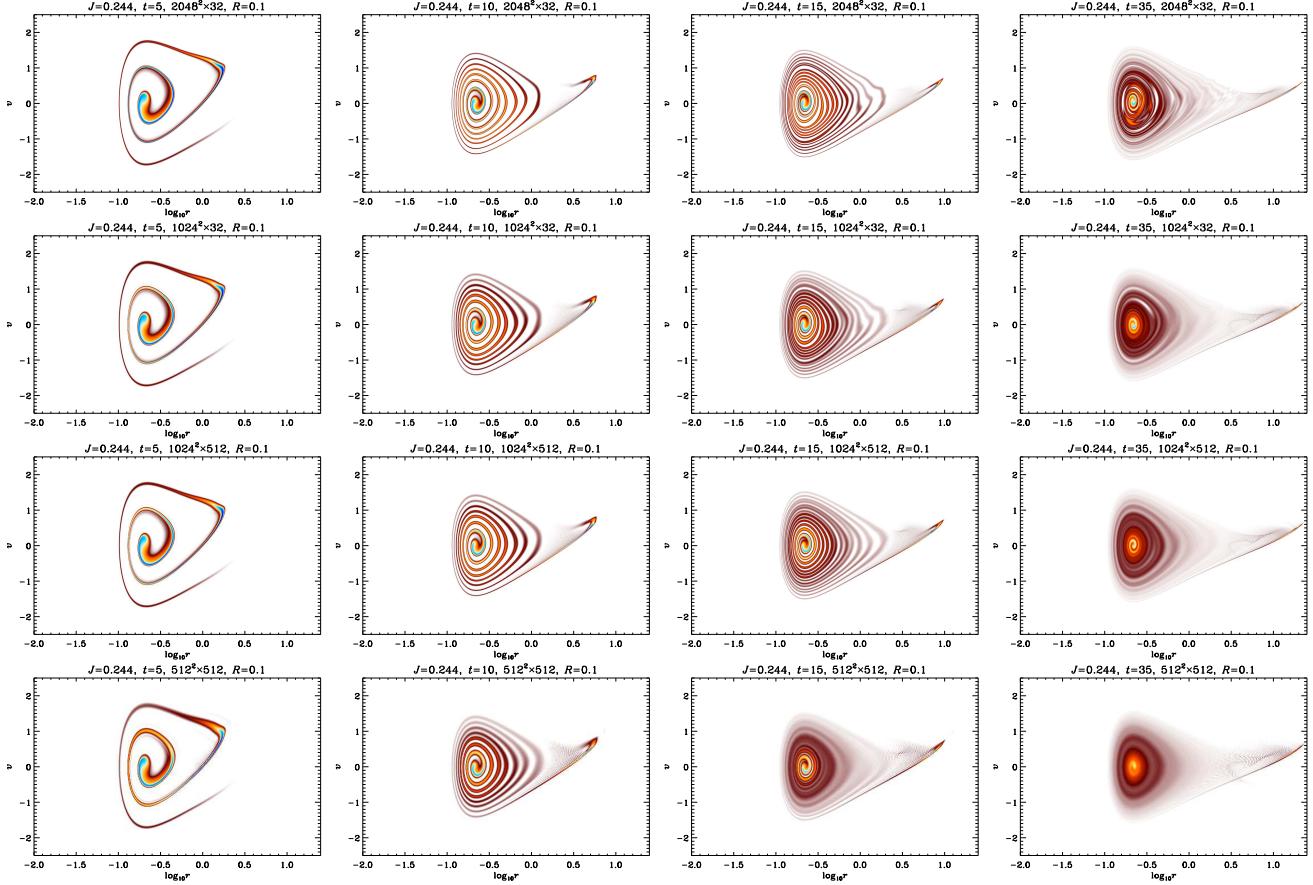
If the mass inside the sphere of radius  $R_{\min}$  is neglected, the trajectories followed by each test particle associated to a grid site that penetrates the sphere are fixed and do not depend on time. This property, combined with the fact that we

use a constant time step, allows us to pre-compute these trajectories once and for all. The delayed central sphere method is then implemented by associating a linked list to each grid site whose associate test particle radial position  $r$  half a time step backward in time are such that  $r \leq R_{\min}$ . Each linked list contains as many elements as the number of time steps needed for the particle to travel a distance of  $2R_{\min}$  and the  $n^{\text{th}}$  element in the list stores the coordinates of the test particle  $n$  time steps backward in time. Before starting the simulation, we initialize each element coordinate and the corresponding value of the initial distribution function. For each time step, the value of each element is then simply updated by assigning to it the value of its successor while the last element value, whose coordinates fall inside the computing domain,  $r \geq R_{\min}$ , is interpolated. A comparison of the results obtained with the reflective central sphere to our improved delayed central sphere is shown on figure 8. The improvements are unquestionable.

### A2 Parallelization issues

We implemented a hybrid shared and distributed memory version of **Vlasolve** via the OpenMP and MPI libraries, respectively.

Shared memory parallelism is relatively straightforward to achieve in the spherically symmetric case, by taking advantage of the fact that the angular momentum  $j$  is a conserved quantity. Spline interpolations, which represent the most expensive part of the code, can thus be computed independently for each slice of constant  $j$ . We therefore easily reach an almost perfect parallelization up to a number of tasks equal to the grid resolution  $N_j$  of angular momentum space, which is typically larger than the number of available cores on a shared memory system.



**Figure A1.** Effect of resolution in the Vlasov code: phase-space density for  $R = 0.1$  and  $j = 0.244$ . Each column of panels corresponds to a given value of time  $t$ , increasing from left to right, while each line correspond to a given resolution,  $(N_r, N_v, N_j) = (2048, 2048, 32)$ ,  $(1024, 1024, 32)$ ,  $(1024, 1024, 512)$  and  $(512, 512, 512)$  from top to bottom, as indicated on each panel. The pictures show only the  $f \geq 0$  part of the phase-space density, while it can actually become negative because of aliasing. However, this choice of representation does not hide aliased regions. The prominent one corresponds to the textured zone above the large  $r$  tail of the system on the right panels.

Distributed memory parallelization via MPI is not as simple. Indeed, spline interpolations are intrinsically non-local, which makes the parallelization along dimensions other than  $j$  non trivial. Sticking with the trivial parallelization described above unfortunately limits the maximum total number of processes running in parallel to  $N_j$ , which is suboptimal. We overcome this limitation by performing MPI domain decomposition in  $(r, v)$  space, following the approach of Crouseilles & al. (2009), who propose to localize the cubic spline interpolation to each domain by using Hermite boundary conditions between the domains with an ad hoc reconstruction of the derivatives.

### A3 Effects of resolution

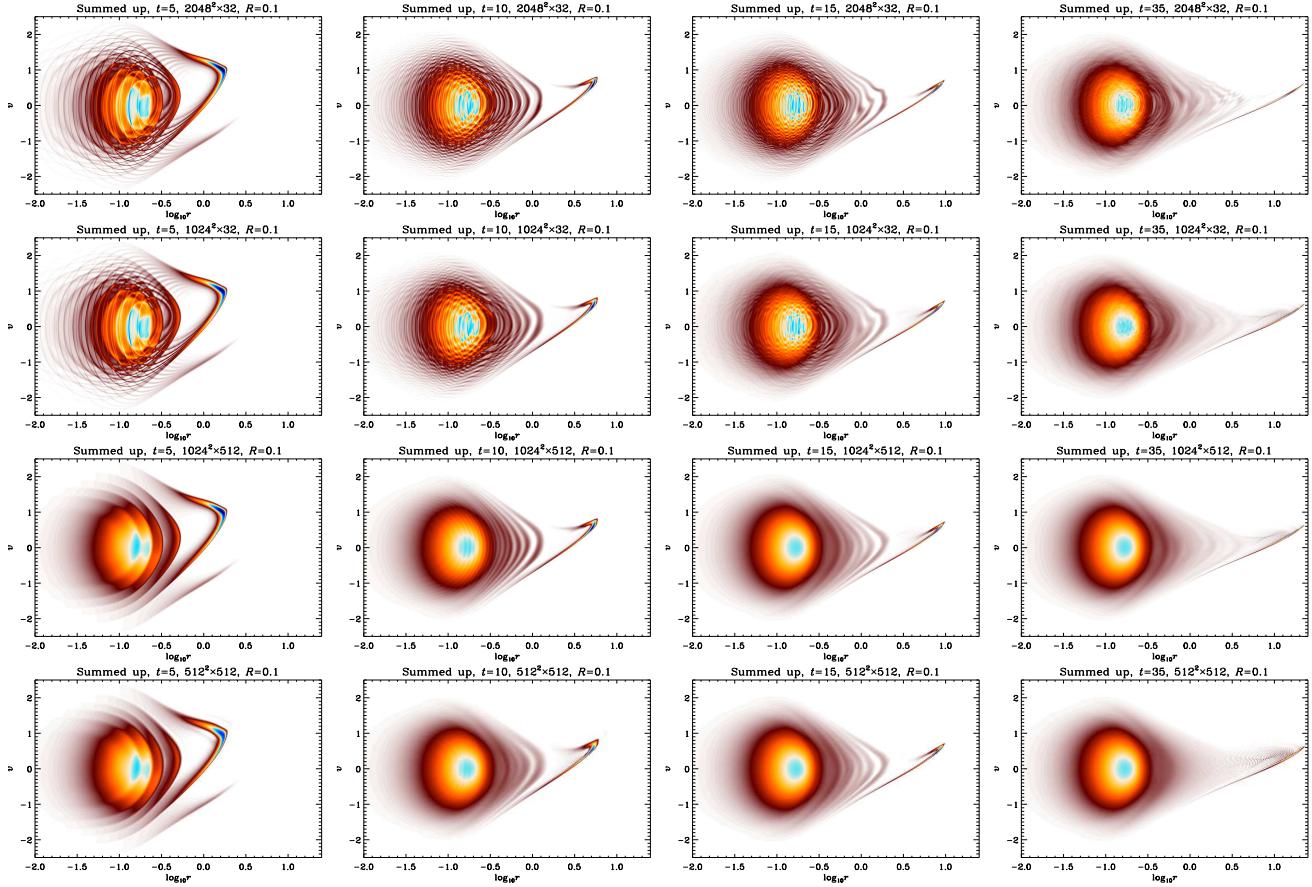
Figures A1 and A2 show, respectively for  $j = 0.244$  and integrated over angular momentum, the phase-space distribution function measured in `Vlasolve` simulations with different resolutions. These simulations have been performed for a Hénon sphere with initial virial ratio  $R = 0.1$ . Beside the very good global agreement between the various runs, these figures bring out three effects, which increase when the resolution of the phase-space grid is reduced:

- Diffusion smearing out fine details that build up in

phase-space during the course of dynamics, for instance clearly visible when one compares top to bottom middle panels of Fig. A1. One concern with diffusion is that it might prevent the appearance of unstable modes. However, we did not perform any simulation in this work that would prove this.

- Aliasing due to artificial oscillations in the spline interpolation: for the problem studied here, aliasing becomes particularly visible after relaxation in the region above the large  $r$  tail, but this does not have significant impact on the dynamics.

- Aliasing due to undersampling angular momentum space: it is visible at all time when one examines the phase-space distribution function integrated over angular momentum (top panels of Fig. A2) and can have dramatic consequences on the dynamics. The two top lines of panels of Fig. A1 and A2, corresponding to a sparse sampling in  $j$  space with only 32 slices, indeed show the appearance of an instability, which presents, on the third column of these figures, the same pattern whether  $(N_r, N_v) = (2048, 2048)$  or  $(1024, 1024)$ . This instability is not present in the simulations with higher resolution in  $j$ , as shown by the two bottom lines of panels. Note that the presence of this instability depends on initial conditions: for  $R = 0.5$ , we did not



**Figure A2.** Same as in Fig. A1, but the phase-space distribution function has now been summed up over the whole available range of values of  $j \in [0, J_{\max} = 1.6]$ , where  $J_{\max}$  is the maximum sampled value of  $j$ .

notice it for the time coverage considered,  $t \leq 100$  (upper line of panels of Figs. 2 and 3).

## APPENDIX B: $N$ -BODY SIMULATIONS: EXPLORATION OF THE CONTROL PARAMETER SPACE

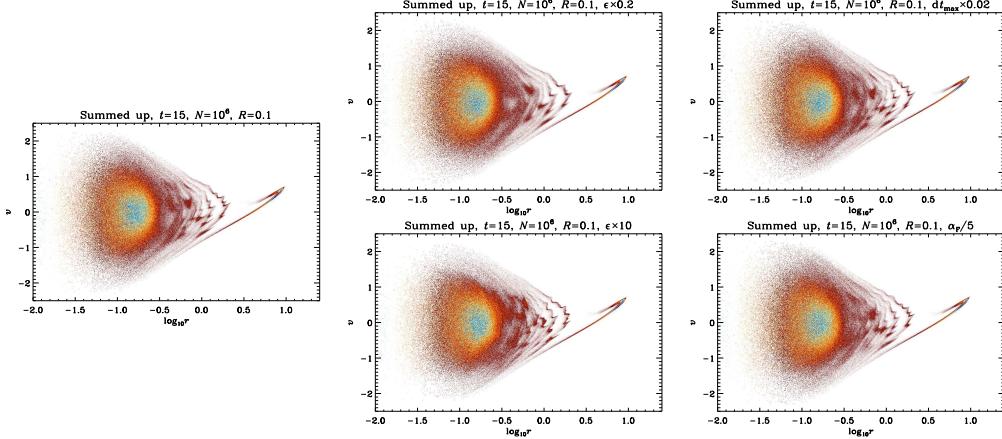
In § 5 we noticed the presence of an instability in the  $R = 0.1$   $N$ -body simulations. One aim of this appendix is to confirm that this instability is related to the number of particles used in the simulations and not to any other control parameter of the `Gadget` code. In the same time, it is of course an opportunity to check that our fiducial choice of the `Gadget` control parameters, given in § 3, is correct.

Figure B1 illustrates the main results of the tests we performed for simulations with  $10^6$  particles. These tests consisted in changing the softening length of the force, the maximum time step value and the tolerance parameter  $\alpha_F$  controlling the errors on the force. Improving the accuracy of the force calculation or dividing the maximum time step  $dt_{\max}$  by a factor 50, which corresponds to imposing  $dt \leq 2 \times 10^{-4}$ , does not change the results. This is confirmed as well by the measurements of the correlators  $C_k$  introduced in § 6, that we do not show here for simplicity. Only the value of the softening parameter of the force  $\epsilon$  have an impact on the dynamics for the tests we did. Re-

ducing  $\epsilon$  by a factor 5 seems to slightly blur the phase-space density, although this effect is difficult to decipher, while increasing  $\epsilon$  by a factor 10 sharpens the fine structures of the phase-space density. Since  $\epsilon$  controls the intensity of close encounters between particles, this is not surprising. Note that increasing  $\epsilon$  by a factor 10 is probably an exaggeration, because it *worsens dramatically* the match during the mixing phase between the  $N$ -body simulation and the Vlasov code when examining the correlators  $C_k$ , a sign that  $\epsilon$  is probably getting too close to a physical characteristic scale of the system.<sup>2</sup> We indeed noticed that increasing  $\epsilon$  only by a factor 5 does not have any impact, on the other hand, on  $C_k$ . However, all these effects do not affect the amplitude of the large scale irregularities on the pattern of  $f(r, v, j)$ , which are present whatever value of  $\epsilon$ . This is also a strong indication that close particle encounters are not at the origin of these irregularities.

We can therefore only conclude that these irregularities and the associated nonlinear instability are the result of non trivial collective effects related to particle shot noise. This argument is also supported by the fact that in addition, the moment of their appearance is particle number dependent, as discussed in § 5.

<sup>2</sup> Increasing  $\epsilon$  by a factor ten gives  $\epsilon = 0.02$ , to be compared for example to the size of the core of the system after relaxation,  $R_c \simeq 0.1$ .



**Figure B1.** Effect of changing the important control parameters in **Gadget**. The phase-space density is shown at  $t = 15$  for **Gadget** simulations with the same initial conditions corresponding to the Hénon sphere with  $R = 0.1$  and involving  $N = 10^6$  particles. In each of the simulations, one control parameter was changed compared to the fiducial simulation shown on left panel and which uses the settings of § 3. On top-left and top-right panels, the softening length of the force was decreased by a factor 5 and increased by a factor 10, respectively. In bottom-left panel, the maximum possible time step was divided by a factor 50, while in the bottom-right panel, the tolerance parameter  $\alpha_F$  defined in § 3 was divided by a factor 5.