Sub-grid modeling of pitch-angle diffusion for ion-scale waves in hybrid-Vlasov simulations with Cartesian velocity space

Dubart, M.

2022-10

Dubart , M , Battarbee , M , Ganse , U , Osmane , A , Spanier , F , Suni , J , Johlander , A , Alho , M , Bussov , M , Cozzani , G , George , H , Grandin , M , Horaites , K , Papadakis , K , Pfau-Kempf , Y , Tarvus , V , Turc , L , Zaitsev , I , Zhou , H & Palmroth , M 2022 , ' Sub-grid modeling of pitch-angle diffusion for ion-scale waves in hybrid-Vlasov simulations with Cartesian velocity space ' , Physics of Plasmas , vol. 29 , no. 10 , 103902 . https://doi.org/10.1063/5.0096361

http://hdl.handle.net/10138/351507
https://doi.org/10.1063/5.0096361

unspecified
acceptedVersion

Sub-grid modeling of pitch-angle diffusion for ion-scale waves in hybrid-Vlasov simulations with Cartesian velocity space

M. Dubart,[1, a)] M. Battarbee,[1] U. Ganse,[1] A. Osmane,[1] F. Spanier,[2] J. Suni,[1] A. Johlander,[1, 3] M. Alho,[1] M. Bussov,[1] G. Cozzani,[1] H. George,[1] M. Grandin,[1] K. Horaites,[1] K. Papadakis,[1] Y. Pfau-Kempf,[1] V. Tarvus,[1] L. Turc,[1] I. Zaitsev,[1] H. Zhou,[1] and M. Palmroth[1, 4]

[1)] *Department of Physics, University of Helsinki, Uusimaa, PO Box 68, 00014 University of Helsinki, Finland*

[2)] *Center for Astronomy, Institute for Theoretical Astrophysics, Heidelberg University, 69120 Heidelberg, Germany*

[3)] *Swedish Institute of Space Physics, Box 537, SE-751 21 Uppsala, Sweden*

[4)] *Space and Earth Observation Centre, Finnish Meteorological Institute, P.O. BOX 503, FI-00101 Helsinki, Finland*

(Dated: 7 December 2022)

Numerical simulations have grown to play a central role in modern sciences over the years. The ever improving technology of supercomputers has made large and precise models available. However, this accuracy is often limited by the cost of computational resources. Lowering the simulation's spatial resolution in order to conserve resources can lead to key processes being unresolved. We have shown in a previous study how insufficient spatial resolution of the proton cyclotron instability leads to a misrepresentation of ion dynamics in hybrid-Vlasov simulations. This leads to larger than expected temperature anisotropy and loss-cone shaped velocity distribution functions. In this study, we present a sub-grid numerical model to introduce pitch-angle diffusion in a 3D Cartesian velocity space, at a spatial resolution where the relevant wave-particle interactions were previously not correctly resolved. We show that the method is successfully able to isotropize loss-cone shaped velocity distribution functions, and that this method could be applied to simulations in order to save computational resources and still correctly model wave-particle interactions.

[a)] Electronic mail: maxime.dubart@helsinki.fi

I. INTRODUCTION

Modern sciences rely heavily on numerical simulations. They can provide detailed descriptions of physical processes, either in synergy with observations to understand complex phenomena, or in cases where long, expensive and often specific in situ experiments or observations are not available. However, a trade-off exists between the accuracy of the simulated model and the cost of computational resources, and ultimately the feasibility of the simulation. This trade-off is often the main limiting factor. In near-Earth space simulations, this often impacts the ability to resolve key plasma instabilities, such as the proton cyclotron instability[1–3] and the mirror instability[4–7]. These two instabilities are important sources of energy transfer and dissipation in the Earth's magnetosheath[8], and arise from temperature anisotropy of the ions. We showed previously[9] that the proton cyclotron instability is not resolved in the Earth's magnetosheath at low spatial resolution in hybrid-Vlasov simulations, i.e. $\Delta r = 900$ km, resulting in the dominance of mirror-like structures and inducing a depletion of phase space density in the "loss-cone" region[10]. This in turn gives rise to a higher than expected temperature anisotropy. While the proton cyclotron instability is correctly resolved at higher resolutions such as $\Delta r = 300$ km, this is too numerically expensive for large 3D global hybrid-Vlasov simulations to be performed given the currently available computational resources. Therefore we have developed a method to model the effect of the proton cyclotron instability on the velocity distribution functions, without increasing the resolution of the simulation, which we present here.

The proton cyclotron instability[1–3] is excited by a temperature anisotropy where the ion perpendicular temperature $T_\perp$ is larger than the parallel temperature $T_\parallel$. It isotropises ions in the magnetosheath through pitch-angle scattering[11]. This diffusion occurs along the direction of the pitch-angle θ, the angle between the vector of the particle velocity and the magnetic field. It is governed by the pitch-angle diffusion coefficient $D_{\mu\mu}$, where $\mu = \cos\theta$ is called the pitch-cosine. This coefficient is a key input to the Fokker-Planck equation, which describes the evolution of plasmas, and is usually estimated using quasi-linear theory[12]. Multiple simulations have also been used to evaluate $D_{\mu\mu}$, such as in the Earth's radiation belt to calculate electron diffusion[13], using test particles in magnetohydrodynamic (MHD) simulations[14], in Particle-in-Cell (PiC) simulations[15], or using distribution functions in hybrid simulations[16]. They have also been extended to model the diffusion itself, either using

MHD[17,18] or PiC[15] simulations. However, the spatial resolution used in these simulations is designed to resolve scales in which pitch-angle scattering occurs. This can be an issue for larger global simulations.

Sub-grid models can be developed to resolve scales inaccessible by the main simulation grid. A sub-grid model evolves with a separate time step with respect to the main grid of the simulation. At each simulation time step, the sub-grid model is performed, solving the small-scale processes the main simulation cannot resolve. Sub-grid models are frequently used in astrophysical plasmas, for example to model fluid scales in interstellar medium[19], unresolved energies[20] or including kinetic physics into MHD models[21]. They are also largely used in meteorological models (e.g. 22 and 23).

In this study we propose a sub-grid model for pitch-angle diffusion in 3D velocity space for hybrid-Vlasov simulations. We do so by mapping the 3D Cartesian velocity space to a 2D (μ, v) space, where v is the plasma velocity. We test the method in a simple test case without spatial propagation of the plasma to ensure it diffuses correctly. We then extend this method in a case with spatial propagation of the plasma. The purpose of the model is to reduce the temperature anisotropy in the simulation. Additionally, we require this model to evolve the initial anisotropic bi-Maxwellian velocity distribution function (VDF) towards a more isotropic distribution, as pitch-angle scattering would do, ensuring the physical processes at play are simulated correctly.

II. GLOBAL HYBRID-VLASOV MODEL

We performed this study using the Vlasiator model. Vlasiator is a hybrid-Vlasov plasma simulation code producing high-fidelity simulations of near-Earth space[24–26]. The global configuration consists of a Cartesian 2D or 3D spatial grid containing the Earth's magnetosphere, magnetosheath, bow shock and foreshock. A uniform Cartesian 3D velocity space grid is contained within each of the ordinary space cells, where the ions (which are simplified to protons in most runs) are described as velocity distribution functions $f(v_x, v_y, v_z)$. In order to save on resources, Vlasiator uses a sparse phase space mechanism[25]. This mechanism ensures that velocity cells with a phase space density beneath the sparsity threshold $f_{\text{threshold}}$ are discarded, or "emptied", preventing the model from having to save the whole velocity space. The model solves the time evolution of the protons in phase space by solving

the Vlasov equation, coupled with the electric and magnetic fields. The electromagnetic fields are propagated using Maxwell's equations in Darwin's approximation[27]. Closure of the system is performed with the generalised Ohm's law including the Hall and electron pressure terms. Electrons are considered an adiabatic, massless, charge-neutralising fluid. The Vlasiator model, and hybrid-Vlasov simulations in general, have the advantage of being noise-free[26], unlike Particle-in-Cell simulations.

In this study however, instead of performing global magnetospheric simulation runs, we present results from our diffusion method in a small simulation box. First, we validate our method in the smallest, simplest test case possible. Then we apply this method to a more advanced case where we initialise the velocity distribution functions with a loss-cone, and with spatial extents large enough to ensure the apparition of the wave modes of interest. This setup is meant to mimic the situation demonstrated in Ref. 9, where a loss-cone has formed due to the simulation not resolving the proton cyclotron instability. We also compare this setup with an identical one where the proton cyclotron instability is resolved. Therefore, we build three different setups:

- Configuration 1 consists of a single 3D cubic cell in ordinary space, with width $\Delta r =$ 900 km. The spatial domain is set to have periodic boundary conditions. The spatial cell contains a 3D Cartesian velocity space grid, consisting of 268 velocity cells in each dimension. Each cubic velocity cell has a width of $\Delta V = 30$ km/s. The sparsity threshold is set to $f_{\text{threshold}} = 10^{-14}$ m^{-6} s^3. The magnetic field is initialised as $\mathbf{B} = B\mathbf{e}_x$, with $B = 17$ nT. The initial proton density is $n = 3.0$ cm^{-3} and the initial anisotropic temperature is set to $T_x = 5.8$ MK, $T_y = 23.2$ MK and $T_z = 23.2$ MK. This setup ensures that no spatial propagation of plasma occurs inside the simulation, i.e. we only allow the velocity space to evolve, unaffected by electric and magnetic fields.

- In Configuration 2, the ordinary space contains 32 cells in both the X-direction and the Y-direction, extending from $X = 0$ to $X = 28800$ km and from $Y = 0$ to $Y =$ 28800 km, and one cell in the Z-direction. Each cell has a width of $\Delta r = 900$ km. The initial magnetic field and temperature are identical to Configuration 1 and initialised uniform throughout the simulation domain. The boundary conditions are set to be periodic in all three directions. The velocity grid cell is identical to Configuration 1,

with the exception that we initialise the velocity distribution functions with an artificial loss-cone by setting the phase-space density to 0 when $\mu \leq -0.5$ or $\mu \geq 0.5$. Due to this initialisation, the initial density in the simulation is $n = 2.26$ cm^{-3}. We initialise each simulation cell with a random normal distributed bulk velocity fluctuation of amplitude 5 km s^{-1}, in order to trigger perturbations. In this setup, spatial propagation and electromagnetic field transport happen as normally in the Vlasiator model. These conditions are favorable to the development of the proton cyclotron instability[3,28], if the spatial resolution were to allow it. However, the resolution of $\Delta r = 900$ km is deliberately chosen so that "the range of **k**-vectors the simulation is able to model is not large enough to allow the proton cyclotron instability to grow"[9]. Therefore, we expect only the presence of mirror modes in this configuration, similar to Panels (i)-(l) in Fig. 8 of Ref. 9. We enable our artificial diffusion method in this configuration.

- Configuration 3 is identical to Configuration 2, but each cell in ordinary space has a width of $\Delta r = 300$ km. The spatial domain extends from $X = 0$ to $X = 9600$ km and from $Y = 0$ to $Y = 9600$ km, and contains one cell in the Z-direction. The initial magnetic field, density, temperature, and boundary conditions are identical to Configuration 2. The velocity distribution functions are also initialised with a loss-cone. These conditions are favorable for the development of both the proton cyclotron and mirror instabilities. This configuration is simulated without artificial diffusion.

III. METHOD

A. Artificial pitch-angle diffusion evaluated in (μ,v) space

Because Vlasiator uses a Cartesian discretization for its 3D velocity space grid, we initialise configuration 1 of our simulation with a bi-Maxwellian distribution function described as follows, in order to trigger the desired instabilities:

$$f_{\mathrm{cart}}^{\mathrm{3D}}(v_\parallel, v_\perp) = \left(\frac{m}{2\pi k_B}\right)^{3/2} \frac{n}{T_\parallel^{1/2} T_\perp} \exp\left[-\frac{m}{2k_B}\left(\frac{v_\parallel^2}{T_\parallel} + \frac{v_\perp^2}{T_\perp}\right)\right], \tag{1}$$

with m the proton mass, k_B the Boltzmann constant, $v_\perp$ and $T_\perp$ are the components of the velocity and temperature of the plasma perpendicular to the magnetic field direction, and $v_\parallel$ and $T_\parallel$ are the components of the velocity and temperature of the plasma parallel

to the magnetic field direction. However, with a spatial resolution of $\Delta r = 900$ km such as in configuration 1, the proton cyclotron instability cannot develop and trigger pitch-angle diffusion as expected. Here, we assume that diffusive pitch-angle scattering of charged particles is the dominant process of diffusion in the unresolved small scales of our simulation domain[11,16]. Therefore, we build a sub-grid model to introduce pitch-angle diffusion in velocity space by explicitly using the pitch-angle diffusion equation derived from the Fokker-Planck equation[12,16]

$$\frac{\partial f^{\mathrm{2D}}_{\mu v}(\mu, v, t)}{\partial t} = \frac{\partial}{\partial \mu}\left[(1-\mu^2) D_{\mu\mu} \frac{\partial f^{2D}_{\mu v}(\mu, v, t)}{\partial \mu}\right], \tag{2}$$

where $D_{\mu\mu}$ is the diffusion coefficient. Eq. 2 uses a 2D distribution function in (μ, v) space defined as[29]

$$f^{\mathrm{2D}}_{\mu v}(\mu, v) = \frac{1}{\sqrt{2\pi}} \left(\frac{m}{k_B}\right)^{3/2} \frac{n v^2}{T_{\parallel}^{1/2} T_{\perp}} \exp\left[-\frac{m}{2k_B}\left(\frac{v^2\mu^2}{T_{\parallel}} + \frac{v^2(1-\mu^2)}{T_{\perp}}\right)\right]. \tag{3}$$

The transformation relations between Eq. 1 and Eq. 3 are given by[29]

$$f^{\mathrm{2D}}_{\mu v}(\mu, v) = 2\pi v^2 f^{\mathrm{3D}}_{\mathrm{cart}}(v\mu, v\sqrt{1-\mu^2}) \tag{4}$$

where the factor $2\pi v^2$ results from the Jacobian determinant for the transformation to the (μ, v) space[29]. Here we use $v_{\parallel} = \mu v$ and $v_{\perp} = v\sqrt{1-\mu^2}$, and

$$n = \int d\mathbf{v} f^{\mathrm{3D}}_{\mathrm{cart}}(v_{\parallel}, v_{\perp}) = \int_{-1}^{1}\int_{0}^{\infty} f^{\mathrm{2D}}_{\mu v}(\mu, v) dv d\mu \tag{5}$$

$$T_{\parallel} = \frac{m}{n k_B} \int d\mathbf{v} v_{\parallel}^2 f^{\mathrm{3D}}_{\mathrm{cart}}(v_{\parallel}, v_{\perp}) = \frac{m}{n k_B} \int_{-1}^{1}\int_{0}^{\infty} v^2\mu^2 f^{\mathrm{2D}}_{\mu v}(\mu, v) dv d\mu \tag{6}$$

$$T_{\perp} = \frac{m}{2n k_B} \int d\mathbf{v} v_{\perp}^2 f^{\mathrm{3D}}_{\mathrm{cart}}(v_{\parallel}, v_{\perp}) = \frac{m}{2n k_B} \int_{-1}^{1}\int_{0}^{\infty} v^2(1-\mu^2) f^{\mathrm{2D}}_{\mu v}(\mu, v) dv d\mu. \tag{7}$$

From a numerical point of view, we now need to map the 3D Cartesian velocity distribution function $f^{\mathrm{3D}}_{\mathrm{cart}}(v_x, v_y, v_z)$ into the 2D (μ, v) space, in order to be able to use Eq. 2. First, we discretize the (μ, v) space with a grid consisting of $\mathrm{N}_{\mu}\times\mathrm{M}_{v}$ bins. When we choose how to discretize the (μ, v) space, two cases are to be considered:

- Undersampling: The discretization of the (μ, v) space is determined such that the width of a bin in μ would cover a large range of this parameter. These would result in 3D velocity space cells with significantly differing values of (μ, v) to be mapped to the same (μ, v) bin. This would then lead to a loss of information about the state of the velocity space, erroneous approximations, and ultimately obscuring the physics at play.

- Oversampling: The discretization of the (μ, v) space is determined such that there would be more bins than required to describe the Cartesian 3D velocity space, and thus some (μ, v) bins would remain empty. This needs to be considered for the calculation of the derivatives in Eq. 2. Numerical derivatives need to be calculated using the closest non-zero neighbors in order to avoid local extrema introduced by empty bins, and thus possible inaccurate derivatives.

We then map the 3D Cartesian velocity space into the 2D (μ, v) space. In this process, we assume the distribution to be mostly gyrotropic, which is a fair estimate in plasmas experiencing significant pitch-angle diffusion. For each cell in velocity space, we calculate its velocity v in the plasma frame by subtracting the bulk velocity. We also calculate its pitch-cosine $\mu = v_\parallel / v$. Each velocity cell is then mapped to its corresponding (μ, v)-bin, including the Jacobian factor of $2\pi v^2$ according to Eq. 4. We finally normalize the total value of $f^{\mathrm{2D}}_{\mu v}(\mu, v)$ in each bin by the count of cells mapped into each bin.

We now calculate the first and second order derivatives in the pitch-angle diffusion equation. For numerical efficiency, we discretize the first and second derivatives separately. As a first simplifying assumption, we consider $D_{\mu\mu}$ to be a constant independent of μ, v or t here:

$$\frac{\partial f^{\mathrm{2D}}_{\mu v}(\mu, v, t)}{\partial t} = D_{\mu\mu}\left[-2\mu\frac{\partial f^{\mathrm{2D}}_{\mu v}(\mu, v, t)}{\partial \mu} + (1-\mu^2)\frac{\partial^2 f^{\mathrm{2D}}_{\mu v}(\mu, v, t)}{\partial \mu^2}\right]. \tag{8}$$

The model thus provides an averaged, effective $D_{\mu\mu}$ value for the whole velocity space. For each bin $f^{\mathrm{2D}}_{\mu v}(\mu_i, v_j)$, both derivatives are calculated accounting for the possibility of oversampling:

$$\frac{\partial f^{\mathrm{2D}}_{\mu v}(\mu_i, v_j, t)}{\partial \mu} = \frac{f^{\mathrm{2D}}_{\mu v}(\mu_{i+R}, v_j, t) - f^{\mathrm{2D}}_{\mu v}(\mu_{i-L}, v_j, t)}{(R+L)\Delta\mu} \tag{9}$$

$$\frac{\partial^2 f^{\mathrm{2D}}_{\mu v}(\mu_i, v_j, t)}{\partial \mu^2} = \frac{1}{\frac{R+L}{2}\Delta\mu}\left[\frac{f^{\mathrm{2D}}_{\mu v}(\mu_{i+R}, v_j, t) - f^{\mathrm{2D}}_{\mu v}(\mu_i, v_j, t)}{R\Delta\mu} - \frac{f^{\mathrm{2D}}_{\mu v}(\mu_i, v_j, t) - f^{\mathrm{2D}}_{\mu v}(\mu_{i-L}, v_j, t)}{L\Delta\mu}\right]. \tag{10}$$

Here, $\Delta\mu$ is the size of a bin in μ. R and L designate the distance in units of $\Delta\mu$ between the bin $f(\mu_i, v_j, t)$ and its first neighbouring bin onto which a nonzero amount of velocity space cells has been mapped, such that $\mu_{i+R} > \mu_i > \mu_{i-L}$. If R $\neq$ L, the derivatives would be slightly offset, and not centered at the grid index in (μ, v) space. However in Vlasiator, we discretize the velocity space as a cell averaged phase-space density. This means that, in

the case of oversampling of (μ, v) space, the grid covers a larger volume in the (μ, v) space, and the derivatives are evaluated at the centre point of this (μ, v) space volume instead. Therefore the derivatives are accurate considering the appropriate volume. We would like to point out that large discrepancies in R and L are very uncommon, as can be seen in Fig. 2a, as we discretize the (μ, v) space to avoid oversampling. According to Eq. 4, the amount of diffusion occurring in the Cartesian velocity space for each $f^{3\mathrm{D}}_{\mathrm{cart}}(\mathbf{v}, t)$ cell is then:

$$\frac{\partial f^{3\mathrm{D}}_{\mathrm{cart}}(\mathbf{v}, t)}{\partial t} = \frac{1}{2\pi v^2} \frac{\partial f^{2\mathrm{D}}_{\mu v}(\mu, v, t)}{\partial t} \tag{11}$$

using its corresponding, previously mapped $\partial_t f^{2\mathrm{D}}_{\mu v}(\mu, v, t)$.

B. Diffusion sub-stepping

The method described above is performed for every spatial cell in the simulation, at every simulation time step Δt_S. $\partial_t f^{3\mathrm{D}}_{\mathrm{cart}}(\mathbf{v}, t)$ varies greatly between velocity space cells. In order to ensure that any changes to phase-space cells are an accurate description of the diffusion equation, we introduce diffusion sub-stepping. The diffusion process is modeled with its own separate time step $\Delta t_D \leq \Delta t_S$. For each velocity space cell, we limit the discretized proportional change $\partial f / f$ per sub-step Δt_D to a constant C. In the case of this simulation, C was chosen ad-hoc to be equal to 10%. Δt_D is chosen so that no velocity-space cell can diffuse more than the set limit with the formulation

$$\Delta t_D = \min\left(C \frac{f^{3\mathrm{D}}_{\mathrm{cart}}(\mathbf{v}, t)}{|\partial_t f^{3\mathrm{D}}_{\mathrm{cart}}(\mathbf{v}, t)|} \right). \tag{12}$$

For values of f below the sparsity threshold, we use the value $f_{\mathrm{threshold}}$ in evaluating the maximum change in f. The diffusion time step Δt_D is not allowed to exceed the simulation time step Δt_S. After $\partial_t f$ is computed, the model updates the value of each cell in velocity space as:

$$f^{3\mathrm{D}}_{\mathrm{cart}}(\mathbf{v}, t + \Delta t_D) = f^{3\mathrm{D}}_{\mathrm{cart}}(\mathbf{v}, t) + \frac{\partial f^{3\mathrm{D}}_{\mathrm{cart}}(\mathbf{v}, t)}{\partial t} \Delta t_D. \tag{13}$$

IV. VALIDATION

Fig. 1 displays the mathematical evaluation of Eq. 3 and the right-hand side of Eq. 2 in the (μ, v) plane in Panels (a) and (c), and their 3D Cartesian equivalents in the (v_x, v_y)

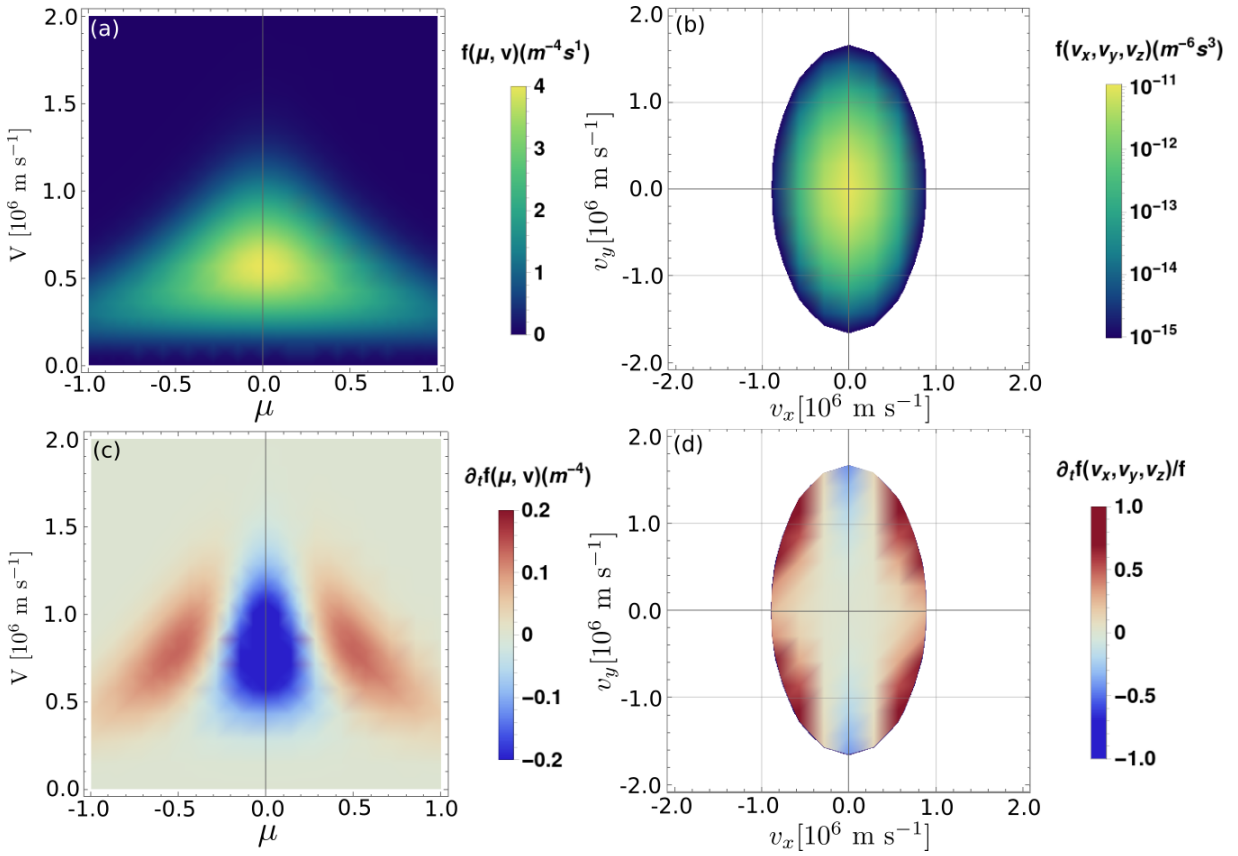

FIG. 1. Representation of the mathematical evaluation of: Eq. 3 (Panel (a)), which displays the distribution function $f(\mu, v)$ in 2D (μ, v) space; Eq. 1 (Panel (b)), which displays the distribution function $f(\mathbf{v})$ in 3D Cartesian velocity space in the (v_x, v_y) plane; the right-hand side of Eq. 2 (Panel (c)), which displays the rate of change $\partial_t f(\mu, v)$ in (μ, v) space; and Eq. 11 divided by f (Panel (d)), which displays the rate of change $\partial_t f(\mathbf{v})$ compared to the original value of f from Panel (b), in 3D Cartesian velocity space in the (v_x, v_y) plane. All equations used $D_{\mu\mu} = 0.01$ s^{-1} and n, $T_\perp$ and $T_\parallel$ as indicated in Configuration 1.

plane, in Panels (b) and (d). We used the arbitrary value of $D_{\mu\mu} = 0.01$ s^{-1}. The values of n, $T_\perp$ and $T_\parallel$ are the same as indicated in Configuration 1. In Panel (c), mass is transported in the 2D (μ, v) space from regions close to $\mu = 0$, highlighted by $\partial_t f(\mu, v) < 0$, towards regions with $\mu \pm 1$, highlighted by $\partial_t f(\mu, v) > 0$ in these regions. Equivalently, in the 3D Cartesian space displayed in Panel (d), phase-space density is transported from regions with v_x close to 0 km s^{-1} towards regions with non-zero v_x, as expected by pitch-angle diffusion.

The initialisation of the numerical application of the method in Configuration 1 can be seen in Fig. 2(a)-(f). Here we choose the same diffusion coefficient $D_{\mu\mu} = 0.01$ s^{-1} as for the

previously mentioned mathematical evaluation. We divide the (μ, v) space with 30 bins in μ and 200 bins in v. Panels (e) and (f) display slices of a bi-Maxwellian velocity distribution function with $T_\parallel < T_\perp$ along two Cartesian directions. Panel (a) displays this distribution after mapping to the (μ, v) plane. Panel (b) displays the evaluation of $\partial_t f(\mu, v)$ at the first step of the simulation. Panels (c) and (d) highlight the amount of the initial distribution which diffuses according to Eq. 11, as slices of the VDF along two Cartesian planes. Panels (a), (b) and (d) are quantitatively consistent with the mathematical evaluation displayed in Fig. 1 and show that the numerical method has been applied successfully at the initial step.

We then launch the simulation, running the Vlasov solver and performing artificial diffusion. After 15 s have elapsed, the previously anisotropic velocity distribution function has diffused into a more isotropic distribution, as depicted in Panels (k) and (l) of Fig. 2. The distribution in the (μ, v) space has diffused from $\mu = 0$ towards $\mu = \pm 1$, as predicted in Fig. 1. The diffusion process has slowed down in the core of the distribution, as highlighted in Panels (i) and (j), as well as in Panel (g) in the (μ, v) space. The derivatives in Panel (h) have decreased in amplitude, indicating that the distribution approaches isotropy and that the method diffuses less as time passes. Our method has been successful in isotropizing the initial velocity distribution function in a simple test case configuration, through pitch-angle diffusion. The Multimedia view for Fig. 2 highlights the diffusion of the distribution from $\mu = 0$ towards $\mu = \pm 1$, controlled by the sub-stepping method. New neighboring velocity cells are initialised as necessary.

We now perform a quantitative check in order to verify that our method models correctly the evolution of the temperature anisotropy $T_\parallel / T_\perp$. We run an idealised numerical test where we initialise a 2D distribution function with Eq. 3, compute the derivatives as in Eq. 8, then evaluate the anisotropy using Eq. 6 and Eq. 7 for 12 s using a time step $\Delta t = 0.001$ s. The results of this test are then compared with the temperature anisotropy given by Vlasiator during the simulation of the diffusion in Configuration 1. We perform this test for several values of $D_{\mu\mu} = 0.1$, 0.05, 0.01, 0.005 and 0.001 s^{-1}. The results are displayed in Fig. 3. Both the idealised test and the simulation provide the same evolution of the temperature anisotropy over time, for each value of the diffusion coefficient.

We now apply the diffusion method in Configuration 2, where $\Delta r = 900$ km, and compare it to Configuration 3, where $\Delta r = 300$ km. Both setups are initialised by setting the phase-space density to 0 when $\mu \leq -0.5$ or $\mu \geq 0.5$ in order to mimic the situation

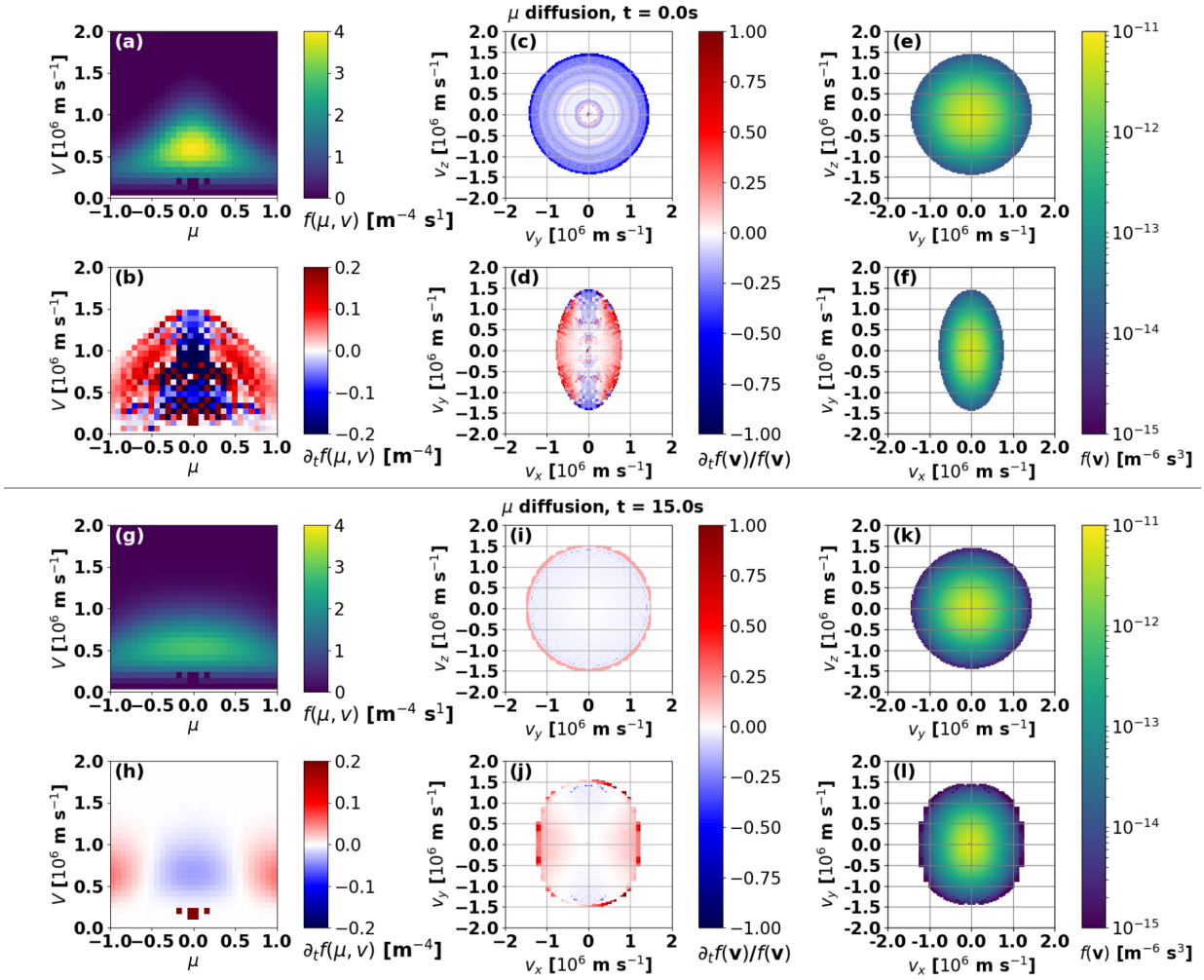

FIG. 2. Panels (a)-(f): initial time step of the diffusion test in Configuration 1. Panels (g)-(l): diffusion test in Configuration 1 after 15 s of simulation have elapsed. Panels (a) and (g) display the distribution function in the (μ, v) space. Panels (b) and (h) display the first time derivative $\partial_t f(\mu, v)$. Panels (c) and (i) display the proportional derivative $\partial_t f(\mathbf{v})/f(\mathbf{v})$ in the (v_y, v_z) plane. Panels (d) and (j) display the proportional derivative $\partial_t f(\mathbf{v})/f(\mathbf{v})$ in the (v_x, v_y) plane. Panels (e) and (k) display the velocity distribution function $f(\mathbf{v})$ in the Cartesian velocity space, in the (v_y, v_z) plane. Panels (f) and (l) display the velocity distribution function $f(\mathbf{v})$ in the Cartesian velocity space, in the (v_x, v_y) plane. The entire diffusion process can be seen on the Multimedia view.

depicted in Fig. 8 of Ref. 9. In Configuration 2, we enable the diffusion method with $D_{\mu\mu} = 0.035$ s^{-1}, whereas in Configuration 3, the simulation follows its course as normally in the Vlasiator model, without artificial diffusion, and allowing the simulation to trigger the

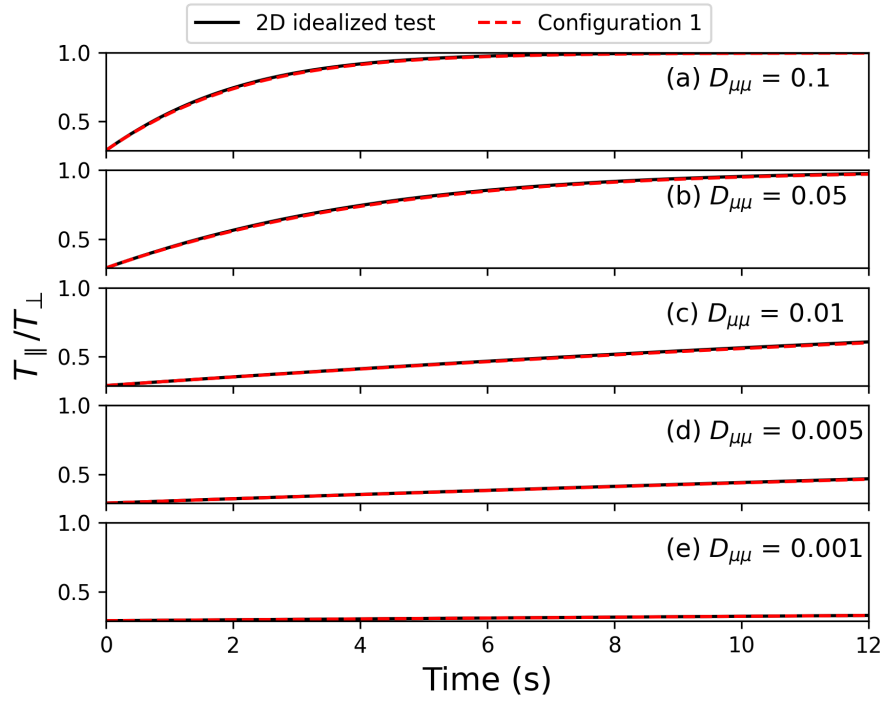

FIG. 3. Comparison of the temperature anisotropy over time between the idealised 2-D diffusion (black) and the simulation (dashed red), for different values of the diffusion coefficient.

proton cyclotron and the mirror instabilities self-consistently. Panel (a) of Fig. 4 displays in orange the maximum growth rate $\gamma/\omega_{\rm ci}$ of the proton cyclotron instability for Configuration 3, where $\omega_{\rm ci} = qB/m \approx 0.163$ s^{-1} is the cyclotron frequency, with q the proton charge and m the proton mass. This growth rate was obtained with the HYbrid Dispersion RelatiOn Solver (HYDROS)[30]. Panel (a) also displays the compressional (dotted green) and non-compressional (solid green) fluctuations of the magnetic field for the same simulation, respectively $\langle \delta \hat{B}^2_{\rm C} \rangle$ and $\langle \delta \hat{B}^2_{\rm NC} \rangle$, with

$$\langle \delta \hat{B}^2_{\rm C} \rangle = \langle (\|\mathbf{B}\| - B_0)^2 \rangle \tag{14}$$

$$\langle \delta \hat{B}^2_{\rm NC} \rangle = \langle \|\mathbf{B} - \mathbf{B}_0\|^2 \rangle \tag{15}$$

where the brackets denote the spatial average over all simulation cells. As shown in Ref. 9, two types of instabilities occur at this spatial resolution: the proton cyclotron instability, which gives rise to Electro-Magnetic Ion Cyclotron waves and can be approximated as non-compressional waves, and the mirror instability, which gives rise to compressional waves. Panel (b) displays the simulation averaged temperature anisotropy $\langle T_\parallel/T_\perp \rangle$ for both Con-

figuration 2 (in red) and Configuration 3 (in blue). The initial growth rate $\gamma/\omega_{\rm ci} \approx 0.675$ indicates that the simulation is indeed favorable to the development of the proton cyclotron instability. The magnetic fluctuations highlight that both wave modes grow as time passes. Wave power grows according to the growth rate, but is not yet strong enough to trigger the diffusion process, as highlighted by the grey region. Pitch-angle diffusion is then triggered at $t \approx 8$ s when the fluctuations are sufficiently strong. Isotropisation begins rougly at this time and lasts until the magnetic fluctuations reach a maximum, around $t \approx 14.4$ s, highlighted by the dotted vertical line. The plasma then becomes stable, and the temperature anisotropy no longer evolves towards 1, having reached a maximum of $T_\parallel/T_\perp \approx 0.74$. In the case of Configuration 2, the artificial diffusion is enabled at $t = 0$ s. Therefore, no waves need to grow in the simulation for the isotropisation process to be triggered, and it starts immediately. In order to better compare both simulations, we shifted the red curve displaying the temperature anisotropy for Configuration 2 to the time where the isotropisation process starts in the case of Configuration 3, at $t \approx 8$ s. We can thus see that, for this particular value of $D_{\mu\mu}$, both diffusion phases isotropise the temperature during the same time span, approximately 7 s. However, in the case of Configuration 2, the temperature anisotropy does not reach a plateau at $T_\parallel/T_\perp \approx 0.74$ but rather keeps approaching a value of 1 when both parallel and perpendicular temperatures reach the same value.

Whilst Fig. 4 displays the result of the process averaged over the whole simulation domain, Fig. 5 shows what happens to a single VDF during both simulations. Panels (a)-(c) display the velocity distribution function in the (v_x, v_y) plane in Configuration 2 at $t = 0$, 2 and 12 s, respectively. Panels (d)-(f) display the velocity distribution function in the (v_x, v_y) plane in Configuration 3 at $t = 0$, 10 and 20 s, respectively. The delay of 8 s has been taken based on Fig. 4 for better comparison between the two setups. Both simulations are initialized with a loss-cone, as shown in Panels (a) and (d). Panels (b) and (e) display a time step during the isotropization process, where phase-space density is transported from $\mu = 0$ towards $\mu \pm 1$, as expected. Panels (c) and (f) show the distributions after the end of the isotropization process, when the temperature anisotropy has reached a value closest to 1 and the plasma has become stable.

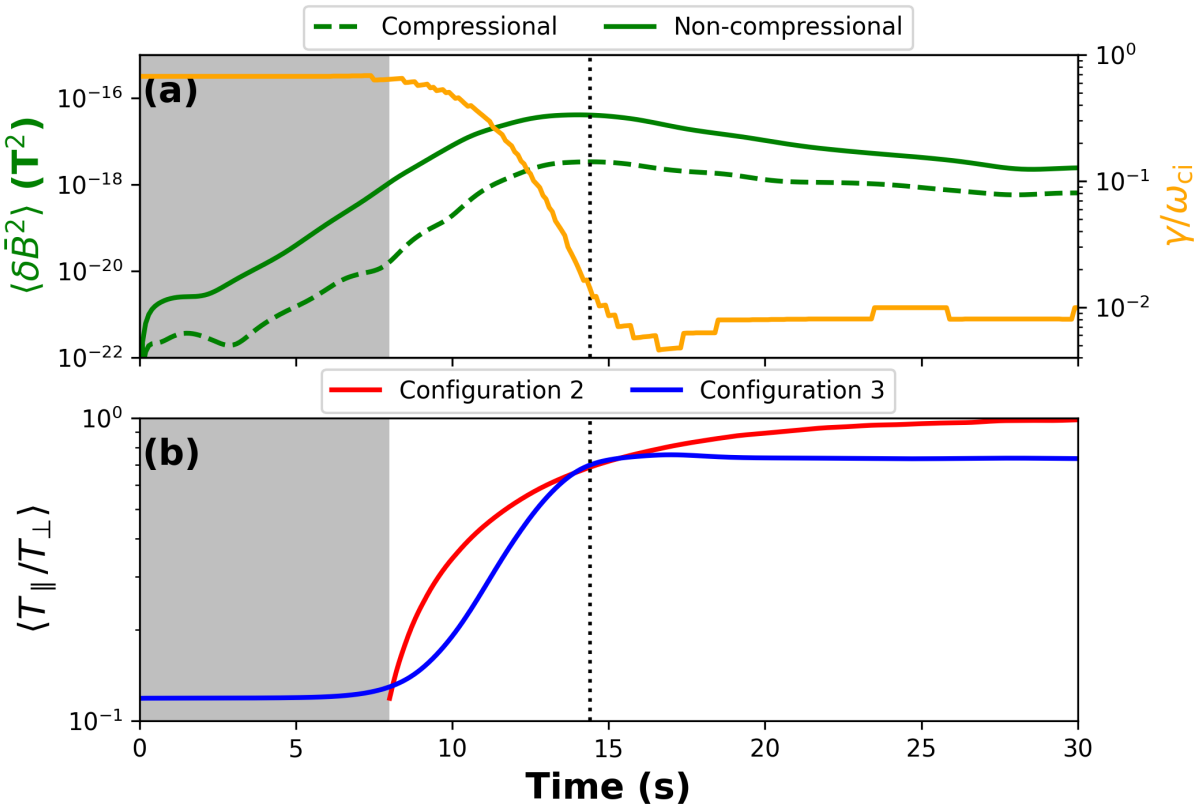

FIG. 4. Panel (a): Evolution with time of the compressional (dotted green) and non-compressional (solid green) magnetic fluctuations $\langle \delta \bar{B}^2 \rangle$, averaged over the whole simulation domain, in Configuration 3 with $\Delta r = 300$ km, and evolution of the maximum growth rate γ/ω_{ci} (orange) of the proton cyclotron instability with time also in Configuration 3, calculated with HYDROS. Panel (b): comparison of the evolution with time of the temperature anisotropy $\langle T_\parallel / T_\perp \rangle$, averaged over the whole simulation domain, in Configuration 2 with $\Delta r = 900$ km (red) and Configuration 3 with $\Delta r = 300$ km (blue). The red curve is shifted forward by 8 s for easier comparison with the blue curve. The grey rectangle highlights the region prior to 8 s during which the growth rate is constant.

V. DISCUSSION

In this paper, we develop a pitch-angle diffusion method suitable for a Cartesian velocity space to be used as a sub-grid model in hybrid-Vlasov simulations. We map the Velocity Distribution Functions from the 3D Cartesian space to the 2D (μ, v) space. We evaluate the derivatives according to Eq. 2. We then map back to the 3D Cartesian velocity space

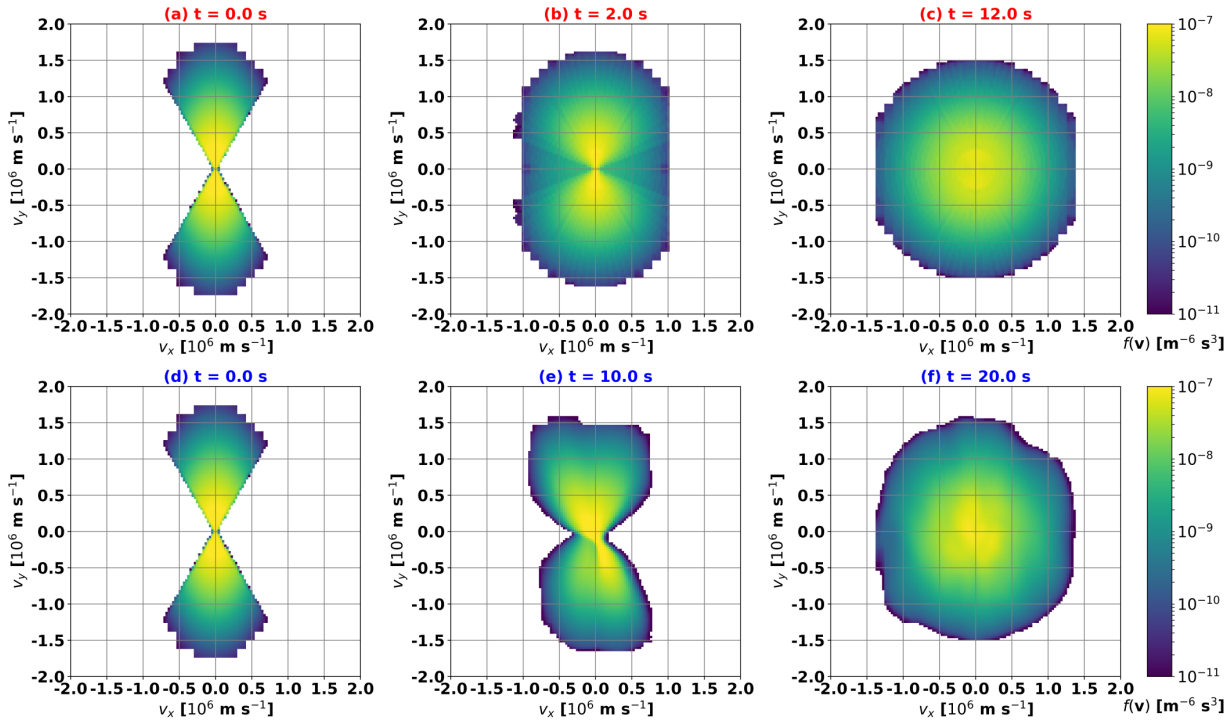

FIG. 5. Panels (a)-(c): Evolution of the velocity distribution function in the (v_x, v_y) plane for $t = 0$ s (a), $t = 2.0$ s (b) and $t = 12.0$ s (c), in Configuration 2 for $\Delta r = 900$ km with artificial diffusion enabled. Panels (d)-(f): Evolution of the velocity distribution function in the (v_x, v_y) plane for $t = 0$ s (a), $t = 10.0$ s (b) and $t = 20.0$ s (c), in Configuration 3 for $\Delta r = 300$ km without artificial diffusion. The time steps for Panels (b) and (e) and Panels (c) and (f) differ by 8 s to account for the shift described in Fig. 4.

and apply the derivatives according to Eq. 11. We test our method first on a single VDF to check that the diffusion occurs as expected, then in a larger test with an initial loss-cone in order to verify that the VDFs are isotropized as expected.

In this study, we make the assumption that pitch-angle diffusion is the dominant process of diffusion at play in our simulation domain. The neglect of the energy diffusion is not valid in all conditions for the case of ion-cyclotron waves. The underlying assumption in our model, is that the ion-cyclotron waves necessary to reproduce the high resolution saturation are parallel propagating ($k_\perp = 0$). In the frame of the wave, the following quantity is conserved

$$(v_\parallel - \omega/k_\parallel)^2 + v_\perp^2 = E \tag{16}$$

and for particles in the tail of the distributions with $v_\parallel \gg \omega/k$, the scattering is predom-

inantly in pitch-angle. Kennel and Engelman[31] provide a heuristic estimate for the time scale associated with energy diffusion T_E and the time scale associated with pitch-angle diffusion T_α and demonstrate that $T_E/T_\alpha \simeq \Omega/\omega \gg 1$ in this limit. The assumption of parallel propagating ion-cyclotron waves is consistent with high resolution Vlasiator simulations of the magnetosheath[9].

Fig. 2 highlights that, in the most simple case without spatial propagation and electromagnetic field transport, our method is consistent with the predicted analytical solution displayed in Fig. 1. The method has diffused accordingly, from pitch-angles $\mu = 0$ towards $\mu = \pm 1$, and the anisotropic VDF in Panel (f) became isotropic in Panel (l). Fig. 3 confirms as well that our method provides the same evolution of the temperature anisotropy than in the case of an idealised 2D diffusion.

Fig. 4 displays a comparison of our method with a test case with parameters similar to those used in the study of the proton cyclotron instability in Ref. 9. This shows that our method is able to reproduce the isotropization process triggered by the proton cyclotron instability after the growth of the corresponding waves, and that the plasma reaches stability.

Fig. 5 highlights that, in a case where spatial propagation and electromagnetic field transport occur, our method is able to diffuse from an initial loss-cone in the VDF for Configuration 2 with $\Delta r = 900$ km (Panels (a)-(c)) to a Maxwellian VDF, where the loss-cone has disappeared, in the same manner as the proton cyclotron instability triggers pitch-angle diffusion for Configuration 3 with $\Delta r = 300$ km (Panels (d)-(f)). This confirms that, in the case where the proton cyclotron instability is not resolved by the simulation due to the spatial resolution, our method is able to isotropise the VDF in order to simulate the effect of the instability on the VDF, similar to Fig. 8 of Ref. 9. We note that the VDF in Panel (e) shows some deviation from a completely gyrotropic distribution, also called agyrotropy. We have calculated the averaged agyrotropy Q_{AG} over the simulation (not shown) according to Ref. 32. $Q_{AG} = 0$ indicates a completely gyrotropic distribution while $Q_{AG} = 1$ indicates a maximal deviation from gyrotropy. The agyrotropy reaches a maximum of $Q_{\mathrm{AG}} \approx 2.2\ 10^{-2}$ during the diffusion process, and drops down to $Q_{\mathrm{AG}} \approx 10^{-3}$ once the plasma reaches a stable state. The intermediate value of order 10^{-2} is transient, and these levels of agyrotropy are low, thus we conclude that they are in agreement with our initial hypothesis. This is also in agreement with values calculated within the magnetosheath, presented by Ref. 33 using a Vlasiator simulation.

However, our method is unable to reproduce the final value of the temperature anisotropy presented in the simulation with Configuration 3 in Fig. 4. In this case, the proton cyclotron instability shows stronger magnetic fluctuations than the mirror instability[34]. Therefore, the proton cyclotron instability isotropizes ions faster than the mirror instability and will consume the available free energy before the mirror instability[35]. In our simulation, it does so before reaching $T_\parallel/T_\perp = 1$. There is thus no remaining energy available to reduce the anisotropy further. In the case of Configuration 2, the proton cyclotron instability cannot develop, and the artificial diffusion reduces the temperature anisotropy according to Eq. 2 until $T_\parallel/T_\perp = 1$. This shows that a constant value of $D_{\mu\mu}$ is insufficient to model perfectly the diffusion process and that our method needs to be further supplemented by a mechanism to determine an appropriate $D_{\mu\mu}$ value based on the physical conditions at any given time step.

Because Vlasiator uses a 3D Cartesian velocity space, one could argue that a method applying the diffusion directly in this 3D Cartesian space could be developed. We discuss here our attempt to build such a model. Such a model would not need to make any assumptions of gyrotropy. Appendix A details the calculations used to transform the diffusion equation Eq. 9 in (μ, v) space into a diffusion equation in 3D Cartesian space (Eq. A16). The implementation of the diffusion sub-stepping for the Cartesian method is identical to the method in (μ, v) space. While the calculation of the derivatives at the initial time step is similar to the method in (μ, v) space and Fig. 2, the Cartesian method presents numerical artefacts later into the simulation. As the initial steps of the Cartesian method behave as expected, we conclude the equations to be correct. However, the numerical implementation, using a simple explicit advance with first-order time-stepping and central difference derivatives is likely the source of the error propagation. Moreover, the Cartesian method requires a derivation specific to the direction of the magnetic field, a constraint the primary presented method does not suffer from. A benefit of evaluating derivatives in the (μ, v) space is that the direction of the magnetic field is managed only when calculating μ and not an inherent part of the equations. It also requires fewer derivatives to be computed, as the Cartesian method requires the first and second derivatives in all three directions.

We showed that the method of diffusion in (μ, v) space is able to accurately reproduce the pitch-angle diffusion process on a VDF in the 3D Cartesian velocity space of a hybrid-Vlasov simulation. With this model acting on a sub-grid scale, one can palliate the absence of the

proton cyclotron instability at lower resolution in the large global runs, such as depicted in Ref. 9 for magnetosheath simulations. This would remove the loss-cone in the VDF and reduce the temperature anisotropy in the simulation, while maintaining a low spatial resolution. While we only demonstrated the feasibility of the method here, we showed the limitations of using an arbitrary diffusion coefficient. In order to use the method in realistic magnetosheath conditions, a further study should concentrate on evaluating an accurate diffusion coefficient, using methods such as in Ref. 16 for hybrid-Vlasov simulations. This coefficient and its application should be dependent on plasma properties both spatially and temporally. Fig. 4 also highlights that it should be damped and not diffuse beyond the availability of wave power.

This method can be used in specific parts of simulations, allowing faster simulation at a lower resolution while still modelling correctly the magnetosheath. A future study should focus on the effect of such a sub-grid model on the global physics of a larger system, such as in the Earth's magnetosheath. This is even more relevant in the case of large, global 3D simulations. This method could also be extended to different diffusion processes such as processes which depend on v or μ, assuming the corresponding description of the diffusion equation is implemented.

VI. CONCLUSIONS

This paper presents a sub-grid model for pitch-angle diffusion in velocity space for hybrid-Vlasov simulations with a Cartesian velocity space, which we demonstrate using Vlasiator. The model solves the pitch-angle diffusion equation derived from the Fokker-Planck equation under the quasi-linear approximation by mapping the 3D Cartesian velocity space into a 2D (μ, v) space. We then evaluate this approach in a small simulation box without spatial propagation of the plasma. We conclude that the (μ, v) approach successfully reduces the temperature anisotropy and isotropises the velocity distribution function. We then tested the (μ, v) approach in a second setup, this time enabling the spatial propagation of the plasma. We again conclude that our model is efficient at correcting the previously formed loss-cone observed in Ref. 9, which forms when there is no pitch-angle diffusion.

Our model could be used to describe pitch-angle diffusion where the proton cyclotron instability is not resolved at low spatial resolution in hybrid-Vlasov simulations. It could

be extended to large global simulations, in 2D or 3D, and utilised in combination with Adaptive Mesh Refinement, where sparse numerical resources are crucial to perform these simulations. This model could also be extended to various types of diffusion. This model therefore enables simulations to evaluate key plasma processes in near-Earth space in a computationally efficient way.

ACKNOWLEDGMENTS

We acknowledge The European Research Council for Starting grant 200141-QuESpace, with which Vlasiator[26] was developed, and Consolidator grant 682068-PRESTISSIMO awarded to further develop Vlasiator and use it for scientific investigations. We gratefully also acknowledge The Finnish Centre of Excellence in Research of Sustainable Space (FORESAIL, Academy of Finland grant number 312351). LT acknowledges the Academy of Finland grant numbers 322544 and 328893, as well as the University of Helsinki (three-year research grand 2020-2022). MG acknowledges the Academy of Finland grant number 338629. YPK acknowledges the Academy of Finland grant number 339756. MP acknowledges the Academy of Finland grant numbers 339327 and 335554. FS thanks Deutsche Forschungsgemeinschaft for support through grant SP1124/9. AO and UG acknowledge funding from the Academy of Finland by the profiling action on Matter and Materials (grant no. 318913). The authors wish to thank the Finnish Grid and Cloud Infrastructure (FGCI) and specifically the University of Helsinki computing services for supporting this project with computational and data storage resources.

CODE AND DATA AVAILABILITY

Vlasiator (`http://www.physics.helsinki.fi/vlasiator/`)[26] is distributed under the GPL-2 open source license at `https://github.com/fmihpc/vlasiator/`[36]. Vlasiator uses a data structure developed in-house (`https://github.com/fmihpc/vlsv/`),[37]. The Analysator software (`https://github.com/fmihpc/analysator/`[38], Hannuksela & the Vlasiator team, 2018) was used to produce the presented figures. The diffusion method can be found at `https://github.com/MaximeII/vlasiator/blob/diffusion/vlasovsolver/velocity_space_diffusion.cpp`. The data described here are kept in storage main-

tained within the Finnish Grid and Cloud Infrastructure (FGCI) and the University of Helsinki. It can be accessed by following the data policy on the Vlasiator website, or downloaded directly from where they are stored `http://urn.fi/urn:nbn:fi:att:85e4e746-55f5-4587-a9de-f6e68f494fec`[39].

AUTHOR DECLARATIONS

The authors declare they have no conflicts of interest.

MD carried out most of the study and the writing of the paper. MB, UG, FS, AO, JS and AJ participated in the development of the numerical and analysis methods presented in this paper. MP is the PI of the Vlasiator model. All co-authors participated in the discussion of the results and contributed in improving the manuscript.

Appendix A: Diffusion equation in Cartesian space

We start from Eq. 8 for the diffusion equation:

$$\frac{\partial f(\mu,t)}{\partial t} = D_{\mu\mu}\left(-2\mu\frac{\partial f}{\partial \mu} + (1-\mu^2)\frac{\partial^2 f}{\partial \mu^2}\right). \tag{A1}$$

We first use the transformation:

$$\mu = \cos\theta \tag{A2}$$

$$v_\parallel = \mu v \tag{A3}$$

$$v_\perp = v\sqrt{1-\mu^2}. \tag{A4}$$

Here we consider $\mathbf{B} = B\mathbf{e}_x$. The first derivative in μ is given by:

$$\begin{aligned}\frac{\partial}{\partial \mu} &= \frac{\partial v_\parallel}{\partial \mu}\frac{\partial}{\partial v_\parallel} + \frac{\partial v_\perp}{\partial \mu}\frac{\partial}{\partial v_\perp} \\ &= v\frac{\partial}{\partial v_\parallel} - v\frac{\mu}{\sqrt{1-\mu^2}}\frac{\partial}{\partial v_\perp}.\end{aligned} \tag{A5}$$

The second derivative in μ can be expressed in terms of the velocity derivatives as:

$$\frac{\partial^2}{\partial \mu^2} = \quad v\frac{\partial}{\partial \mu}\frac{\partial}{\partial v_\parallel} - v\frac{\partial}{\partial v_\perp}\left[\frac{\sqrt{1-\mu^2}+\frac{\mu^2}{\sqrt{1-\mu^2}}}{1-\mu^2}\right] - v\frac{\partial}{\partial \mu}\frac{\partial}{\partial v_\perp}\frac{\mu}{\sqrt{1-\mu^2}}$$

$$
\begin{aligned}
&= v\frac{\partial}{\partial\mu}\frac{\partial}{\partial v_\parallel} - v\frac{\partial}{\partial v_\perp}\left[\frac{1}{(1-\mu^2)^{3/2}} + \frac{\mu}{\sqrt{1-\mu^2}}\frac{\partial}{\partial\mu}\right] \\
&= v^2\frac{\partial^2}{\partial v_\parallel^2} - v^2\frac{\partial}{\partial v_\perp}\frac{\partial}{\partial v_\parallel}\frac{\mu}{\sqrt{1-\mu^2}} - v\frac{\partial}{\partial v_\perp}\frac{1}{(1-\mu^2)^{3/2}} - v^2\frac{\partial}{\partial v_\perp}\frac{\partial}{\partial v_\parallel}\frac{\mu}{\sqrt{1-\mu^2}} \\
&\quad + v^2\frac{\partial^2}{\partial v_\perp^2}\frac{\mu^2}{1-\mu^2} \qquad \text{(A6)}
\end{aligned}
$$

$$
\frac{\partial^2}{\partial\mu^2} = v^2\frac{\partial^2}{\partial v_\parallel^2} - 2v^2\frac{\partial}{\partial v_\perp}\frac{\partial}{\partial v_\parallel}\frac{\mu}{\sqrt{1-\mu^2}} - v\frac{\partial}{\partial v_\perp}\frac{1}{(1-\mu^2)^{3/2}} + v^2\frac{\partial^2}{\partial v_\perp^2}\frac{\mu^2}{1-\mu^2}. \qquad \text{(A7)}
$$

The diffusion equation 8 then becomes:

$$
\begin{aligned}
\frac{\partial f}{\partial t} = \quad & D_{\mu\mu}\left[-2\mu v\frac{\partial f}{\partial v_\parallel} + 2v\frac{\mu^2}{\sqrt{1-\mu^2}}\frac{\partial f}{v_\perp} + (1-\mu^2)\left(v^2\frac{\partial^2 f}{\partial v_\parallel^2}\right.\right. \\
& \left.\left. -2v^2\frac{\mu}{\sqrt{1-\mu^2}}\frac{\partial^2 f}{\partial v_\perp \partial v_\parallel} - v\frac{1}{(1-\mu^2)^{3/2}}\frac{\partial f}{\partial v_\perp} + v^2\frac{\mu^2}{1-\mu^2}\frac{\partial^2 f}{\partial v_\perp^2}\right)\right]. \qquad \text{(A8)}
\end{aligned}
$$

We now transform this equation into the Cartesian coordinates system:

$$
\begin{aligned}
\frac{\partial}{\partial v_\parallel} &= \frac{\partial}{\partial v_x} && \text{(A9)} \\
\frac{\partial^2}{\partial v_\parallel^2} &= \frac{\partial^2}{\partial v_x^2} && \text{(A10)} \\
\frac{\partial}{\partial v_\perp} &= \sin\phi\frac{\partial}{\partial v_y} + \cos\phi\frac{\partial}{\partial v_z} && \text{(A11)} \\
\frac{\partial^2}{\partial v_\perp^2} &= \frac{v_z^2}{v_z^2+v_y^2}\frac{\partial^2}{\partial v_z^2} + \frac{2v_z v_y}{v_z^2+v_y^2}\frac{\partial^2}{\partial v_y \partial v_z} + \frac{v_y^2}{v_z^2+v_y^2}\frac{\partial^2}{\partial v_y^2} && \text{(A12)} \\
\frac{\partial^2}{\partial v_\perp v_\parallel} &= \sin\phi\frac{\partial^2}{\partial v_y \partial v_x} + \cos\phi\frac{\partial^2}{\partial v_z v_x}, && \text{(A13)}
\end{aligned}
$$

in this coordinate system, ϕ is the polar angle defined here as: $\phi = \arctan(v_y/v_z)$ and so does not depend on v_x. Inputting this to Eq. A8:

$$
\begin{aligned}
\frac{\partial f}{\partial t} = \quad & D_{\mu\mu}\left[-2\mu v\frac{\partial f}{\partial v_x} + 2v\frac{\mu^2}{\sqrt{1-\mu^2}}\left(\sin\phi\frac{\partial f}{\partial v_y} + \cos\phi\frac{\partial f}{\partial v_z}\right) + (1-\mu^2)v^2\frac{\partial^2 f}{\partial v_x^2}\right. \\
& -2v^2\mu\sqrt{1-\mu^2}\left(\sin\phi\frac{\partial^2 f}{\partial v_y \partial v_x} + \cos\phi\frac{\partial^2 f}{\partial v_z \partial v_x}\right) - v\frac{1}{\sqrt{1-\mu^2}}\left(\sin\phi\frac{\partial f}{\partial v_y} + \cos\phi\frac{\partial f}{\partial v_z}\right) \\
& \left.+v^2\mu^2\left(\frac{v_z^2}{v_z^2+v_y^2}\frac{\partial^2 f}{\partial v_z^2} + 2\frac{v_z v_y}{v_z^2+v_y^2}\frac{\partial^2 f}{\partial v_y \partial v_z} + \frac{v_y^2}{v_z^2+v_y^2}\frac{\partial^2 f}{\partial v_y^2}\right)\right]. \qquad \text{(A14)}
\end{aligned}
$$

Replacing $\mu = v_x/v$ gives:

$$
\begin{aligned}
\frac{\partial f}{\partial t} = \quad & D_{\mu\mu}\left[-2v_x\frac{\partial f}{\partial v_x} + 2\frac{v_x^2}{\sqrt{v_y^2+v_z^2}}\left(\sin\phi\frac{\partial f}{\partial v_y} + \cos\phi\frac{\partial f}{\partial v_z}\right) + (v_y^2+v_z^2)\frac{\partial^2 f}{\partial v_x^2}\right.
\end{aligned}
$$

$$-2v_x\sqrt{v_y^2+v_z^2}\left(\sin\phi\frac{\partial^2 f}{\partial v_y\partial v_x}+\cos\phi\frac{\partial^2 f}{\partial v_z\partial v_x}\right)-\frac{v^2}{\sqrt{v_y^2+v_z^2}}\left(\sin\phi\frac{\partial f}{\partial v_y}+\cos\phi\frac{\partial f}{\partial v_z}\right)$$
$$+v_x^2\left(\frac{v_z^2}{v_z^2+v_y^2}\frac{\partial^2 f}{\partial v_z^2}+2\frac{v_z v_y}{v_z^2+v_y^2}\frac{\partial^2 f}{\partial v_y\partial v_z}+\frac{v_y^2}{v_z^2+v_y^2}\frac{\partial^2 f}{\partial v_y^2}\right)\Bigg]. \quad \text{(A15)}$$

The pitch-angle diffusion equation in Cartesian coordinates is therefore given by:

$$\begin{aligned}\frac{\partial f}{\partial t}= \quad & D_{\mu\mu}\Bigg[-2v_x\frac{\partial f}{\partial v_x}+\frac{2v_x^2-v^2}{\sqrt{v_y^2+v_z^2}}\left(\sin\phi\frac{\partial f}{\partial v_y}+\cos\phi\frac{\partial f}{\partial v_z}\right)\\ & -2v_x\sqrt{v_y^2+v_z^2}\left(\sin\phi\frac{\partial^2 f}{\partial v_y\partial v_x}+\cos\phi\frac{\partial^2 f}{\partial v_z\partial v_x}\right)\\ & +2\frac{v_x^2 v_z v_y}{v_z^2+v_y^2}\frac{\partial^2 f}{\partial v_y\partial v_z}+(v_y^2+v_z^2)\frac{\partial^2 f}{\partial v_x^2}+\frac{v_x^2 v_z^2}{v_z^2+v_y^2}\frac{\partial^2 f}{\partial v_z^2}+\frac{v_x^2 v_y^2}{v_z^2+v_y^2}\frac{\partial^2 f}{\partial v_y^2}\Bigg]. \quad \text{(A16)}\end{aligned}$$

REFERENCES

[1]C. F. Kennel and H. E. Petschek, "Limit on stably trapped particle fluxes," Journal of Geophysical Research (1896-1977) **71**, 1–28 (1966).

[2]R. C. Davidson and J. M. Ogden, "Electromagnetic ion cyclotron instability driven by ion energy anisotropy in high-beta plasmas," Physics of Fluids **18**, 1045–1050 (1975).

[3]S. P. Gary, "The mirror and ion cyclotron anisotropy instabilities," Journal of Geophysical Research: Space Physics **97**, 8519–8529 (1992).

[4]S. Chandrasekhar, A. N. Kaufman, and K. M. Watson, "The stability of the pinch," Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences **245**, 435–455 (1958).

[5]A. Hasegawa, "Drift mirror instability in the magnetosphere," The Physics of Fluids **12**, 2642–2650 (1969).

[6]D. J. Southwood and M. G. Kivelson, "Mirror instability: 1. physical mechanism of linear instability," Journal of Geophysical Research: Space Physics **98**, 9181–9187 (1993).

[7]M. G. Kivelson and D. J. Southwood, "Mirror instability ii: The mechanism of nonlinear saturation," Journal of Geophysical Research: Space Physics **101**, 17365–17371 (1996).

[8]S. J. Schwartz, D. Burgess, and J. J. Moses, "Low-frequency waves in the Earth's magnetosheath: present status," Annales Geophysicae **14**, 1134–1150 (1997).

[9]M. Dubart, U. Ganse, A. Osmane, A. Johlander, M. Battarbee, M. Grandin, Y. Pfau-Kempf, L. Turc, and M. Palmroth, "Resolution dependence of magnetosheath waves in global hybrid-vlasov simulations," Annales Geophysicae **38**, 1283–1298 (2020).

[10]S. Ichimaru, *Basic Principles of Plasma Physics: a Statistical Approach* (Addison-Wesley, 1980).

[11]M. Tanaka, "Simulations of heavy ion heating by electromagnetic ion cyclotron waves driven by proton temperature anisotropies," Journal of Geophysical Research: Space Physics **90**, 6459–6468 (1985).

[12]J. R. Jokipii, "Cosmic-Ray Propagation. I. Charged Particles in a Random Magnetic Field," The Astrophysical Journal **146**, 480 (1966).

[13]S. A. Glauert and R. B. Horne, "Calculation of pitch angle and energy diffusion coefficients with the padie code," Journal of Geophysical Research: Space Physics **110** (2005), 10.1029/2004JA010851.

[14]G. Qin and A. Shalchi, "Pitch-angle diffusion coefficients of charged particles from computer simulations," Astrophysical Journal **707** (2009).

[15]C. Schreiner and F. Spanier, "Wave-particle-interaction in kinetic plasmas," Computer Physics Communications **185**, 1981–1986 (2014).

[16]A. Ivascenko, S. Lange, F. Spanier, and R. Vainio, "Determining pitch-angle diffusion coefficients from test particle simulations," The Astrophysical Journal **833**, 223 (2016).

[17]F. Spanier and M. Wisniewski, "Simulation of charged particle diffusion in mhd plasmas," Astrophys. Space Sci. Trans **7**, 21–27 (2011).

[18]S. Lange, F. Spanier, M. Battarbee, R. Vainio, and T. Laitinen, "Particle scattering in turbulent plasmas with amplified wave modes," Astronomy & Astrophysics **553**, A129 (2013).

[19]M. J. Colbrook, X. Ma, P. F. Hopkins, and J. Squire, "Scaling laws of passive-scalar diffusion in the interstellar medium," Monthly Notices of the Royal Astronomical Society **467**, 2421–2429 (2017).

[20]P. F. Hopkins, J. Squire, T. K. Chan, E. Quataert, S. Ji, D. Kereš, and C.-A. Faucher-Giguère, "Testing physical models for cosmic ray transport coefficients on galactic scales: self-confinement and extrinsic turbulence at GeV energies," Monthly Notices of the Royal Astronomical Society **501**, 4184–4213 (2020).

[21]A. Marcowith, A. J. van Marle, and I. Plotnikov, "The cosmic ray-driven streaming instability in astrophysical and space plasmas," Physics of Plasmas **28**, 080601 (2021).

[22]M. Müller and D. Scherer, "A grid and subgrid-scale radiation parameterization of topographic effects for mesoscale weather forecast models," Monthly Weather Review - MON

WEATHER REV **133**, 1431–1442 (2005).

[23]L. Rottner-Peyrat, C. Baehr, F. Couvreux, G. Canut, and T. Rieutord, "A new downscaling method for sub-grid turbulence modeling," Atmospheric Chemistry and Physics **17**, 6531–6546 (2017).

[24]M. Palmroth, I. Honkonen, A. Sandroos, Y. Kempf, S. von Alfthan, and D. Pokhotelov, "Preliminary testing of global hybrid-Vlasov simulation: Magnetosheath and cusps under northward interplanetary magnetic field," Journal of Atmospheric and Solar-Terrestrial Physics **99**, 41 – 46 (2013).

[25]S. von Alfthan, D. Pokhotelov, Y. Kempf, S. Hoilijoki, I. Honkonen, A. Sandroos, and M. Palmroth, "Vlasiator: First global hybrid-Vlasov simulations of Earth's foreshock and magnetosheath," Journal of Atmospheric and Solar-Terrestrial Physics **120**, 24 – 35 (2014).

[26]M. Palmroth, U. Ganse, Y. Pfau-Kempf, M. Battarbee, L. Turc, T. Brito, M. Grandin, S. Hoilijoki, A. Sandroos, and S. von Alfthan, "Vlasov methods in space physics and astrophysics," Living Reviews in Computational Astrophysics **4** (2018), 10.1007/s41115-018-0003-2.

[27]P. Londrillo and L. del Zanna, "On the divergence-free condition in Godunov-type schemes for ideal magnetohydrodynamics: the upwind constrained transport method," Journal of Computational Physics **195**, 17–48 (2004), arXiv:astro-ph/0310183 [astro-ph].

[28]B. Remya, R. V. Reddy, B. T. Tsurutani, G. S. Lakhina, and E. Echer, "Ion temperature anisotropy instabilities in planetary magnetosheaths," Journal of Geophysical Research: Space Physics **118**, 785–793 (2013).

[29]D. Moseev and M. Salewski, "Bi-maxwellian, slowing-down, and ring velocity distributions of fast ions in magnetized plasmas," Physics of Plasmas **26**, 020901 (2019).

[30]D. Told, J. Cookmeyer, P. Astfalk, and F. Jenko, "A linear dispersion relation for the hybrid kinetic-ion/fluid-electron model of plasma physics," New Journal of Physics **18**, 075001 (2016).

[31]C. F. Kennel and F. Engelmann, "Velocity space diffusion from weak plasma turbulence in a magnetic field," The Physics of Fluids **9**, 2377–2388 (1966).

[32]M. Swisdak, "Quantifying gyrotropy in magnetic reconnection," Geophysical Research Letters **43**, 43–49 (2016), https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1002/2015GL066980.

[33]M. Battarbee, X. Blanco-Cano, L. Turc, P. Kajdič, A. Johlander, V. Tarvus, S. Fuselier,

K. Trattner, M. Alho, T. Brito, U. Ganse, Y. Pfau-Kempf, M. Akhavan-Tafti, T. Karlsson, S. Raptis, M. Dubart, M. Grandin, J. Suni, and M. Palmroth, "Helium in the earth's foreshock: a global vlasiator survey," Annales Geophysicae **38**, 1081–1099 (2020).

[34]M. E. McKean, D. Winske, and S. P. Gary, "Kinetic properties of mirror waves in magnetosheath plasmas," Geophysical Research Letters **19**, 1331–1334 (1992).

[35]M. E. McKean, D. Winske, and S. P. Gary, "Mirror and ion cyclotron anisotropy instabilities in the magnetosheath," Journal of Geophysical Research: Space Physics **97**, 19421–19432 (1992).

[36]M. Palmroth and the Vlasiator Team, "Vlasiator: hybrid-Vlasov simulation code, Github repository," (2019).

[37]A. Sandroos, "VLSV: file format and tools, Github repository," (2019).

[38]M. Battarbee, O. A. Hannuksela, Y. Pfau-Kempf, S. von Alfthan, U. Ganse, R. Jarvinen, Leo, J. Suni, M. Alho, lturc, Ilja, tvbrito, and M. Grandin, "fmihpc/analysator: v0.9," (2021).

[39]M. Dubart, M. Battarbee, and U. Ganse, "Vlasiator artificial diffusion data (configurations 1, 2 & 3)," `http://urn.fi/urn:nbn:fi:att:85e4e746-55f5-4587-a9de-f6e68f494fec` (2022).