Enabling technology for global 3D + 3V hybrid-Vlasov simulations of near-Earth space

Ganse, Urs

2023-04-13

Enabling technology for global 3D + 3V hybrid-Vlasov simulations of near-Earth space

Urs Ganse ✉ ; Tuomas Koskela; Markus Battarbee; ... et. al

Check for updates

View Online   Export Citation   CrossMark

Articles You May Be Interested In

Wave dispersion in the hybrid-Vlasov model: Verification of Vlasiator

Physics of Plasmas (November 2013)

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

Physics of Plasmas (October 2022)

Hybrid-Vlasov modeling of three-dimensional dayside magnetopause reconnection

Physics of Plasmas (September 2020)

Physics of Plasmas

Features in Plasma Physics Webinars

Submit Today!

AIP Publishing

Enabling technology for global 3D + 3V hybrid-Vlasov simulations of near-Earth space

View Online　　Export Citation　　CrossMark

Urs Ganse,[1,a] Tuomas Koskela,[2] Markus Battarbee,[1] Yann Pfau-Kempf,[1] Konstantinos Papadakis,[1] Markku Alho,[1] Maarja Bussov,[1] Giulia Cozzani,[1] Maxime Dubart,[1] Harriet George,[1] Evgeny Gordeev,[1] Maxime Grandin,[1] Konstantinos Horaites,[1] Jonas Suni,[1] Vertti Tarvus,[1] Fasil Tesema Kebede,[1] Lucile Turc,[1] Hongyang Zhou,[1] and Minna Palmroth[1,3]

AFFILIATIONS

[1]Department of Physics, University of Helsinki, P.O. Box 4, Helsinki 00014, Finland
[2]University College London, London WC1E 6BT, United Kingdom
[3]Finnish Meteorological Institute, Space and Earth Observation Centre, PO Box 503, Helsinki 00101, Finland

[a]Author to whom correspondence should be addressed: urs.ganse@helsinki.fi

ABSTRACT

We present methods and algorithms that allow the Vlasiator code to run global, three-dimensional hybrid-Vlasov simulations of Earth's entire magnetosphere. The key ingredients that make Vlasov simulations at magnetospheric scales possible are the sparse velocity space implementation and spatial adaptive mesh refinement. We outline the algorithmic improvement of the semi-Lagrangian solver for six-dimensional phase space quantities, discuss the coupling of Vlasov and Maxwell equations' solvers in a refined mesh, and provide performance figures from simulation test runs that demonstrate the scalability of this simulation system to full magnetospheric runs.

I. INTRODUCTION

Vlasiator is a hybrid-Vlasov space plasma simulation system, specifically designed to perform kinetic simulations of the near-Earth environment.[1] Its goal has been to use it to perform global, three-dimensional simulations of Earth's magnetosphere and its interaction with the solar wind without a fixed prescription of the shape of the particles' velocity distribution function [as would be the case in magnetohydrodynamics (MHD)]. Being an implementation of a hybrid kinetic method, Vlasiator simulates the phase-space evolution of ion species by propagating the phase space density on a Cartesian grid, modeling ions as a distribution function in six (three spatial and three velocity) dimensions. Electrons are treated in an indirect way, their effective physical action reduced to charge neutralization, Hall currents, and contributions to Ohm's law via the electron pressure gradient.[2] In Vlasiator's numerical implementation, the deliberate choice of a Eulerian representation of phase space, instead of the common approach of a particle-in-cell (PIC) approximation,[3] means that the simulations are computationally very heavy, typically exceeding millions of CPU-hours for a few tens of minutes of simulated physical time. On the other hand, this choice enables simulation results that are unencumbered by stochastic particle sampling noise that is inherent to the PIC simulation approach. In the end, high-resolution PIC simulations typically end up using particle counts that are roughly equivalent ($n \approx 10^{12}$) to Vlasiator's phase space grid point numbers.[4]

Vlasiator has been employed to study Earth's foreshock,[5–12] magnetosheath waves,[13,14] magnetosheath jets,[15–17] dayside magnetopause magnetic reconnection,[18–20] transient foreshocks,[21] magnetotail reconnection site locations,[22,23] fast magnetotail flows,[24] magnetotail current sheet flapping,[25] and ionospheric precipitation.[26,27] In addition, Refs. 28 and 29 presented simulations, in which full electrostatic Vlasov simulations including kinetic electrons were run. Common to all previous publications was a limitation to two spatial dimensions: For reasons of computational complexity and cost, the simulation runs were chosen to operate in spatially two-dimensional cuts through the Earth's magnetosphere, along the equatorial plane (for foreshock and magnetosheath studies) or the noon-midnight meridional plane (for studies of dayside- and tail reconnection, among others). The dimension perpendicular to this plane extended only a single simulation cell in thickness, with periodic boundary conditions. Their magnetic field had to be modeled as a line dipole, which prevented

fully realistic mapping of inner magnetosphere locations to ionospheric altitudes.

A notable exception is given by Ref. 30, where the simulation setup was a slab of finite extent, to provide a third spatial dimension on a limited scale (but large enough to comfortably exceed ion scales), so that the local kinetic behavior of dayside magnetopause reconnection could be studied without limitation of strict two-dimensionality. Yet, the overall geometry of the simulation domains, and thus the magnetic topology and physical degrees of freedom, remained essentially two-dimensional, so that a comprehensive validation of physical results was required in each of the aforementioned studies, to ensure that their outcomes were not negatively affected by the limitation to two spatial dimensions.

Thanks to recent algorithmic improvements to the code, these limitations have been overcome, fulfilling the original goal outlined in Ref. 31. In the present paper, we describe the technological and conceptual evolution of the Vlasiator code that reduces the computational overhead to a degree that global simulations in full three spatial dimensions (plus three velocity dimensions) have become possible. Scientific results from the first 3D + 3V global hybrid-Vlasov simulation of Earth's magnetosphere performed with the methods presented here will be presented in an upcoming publication.

The structure of this publication is such that the remainder of Sec. I outlines the length- and timescales that magnetospheric hybrid-kinetic simulations need to resolve and how these resolution requirements lead to difficult computational demands. Section II describes Vlasiator's data structures and solvers, with a special focus on the recent developments that enable the combination of spatial adaptive mesh refinement and sparse velocity space structure. Section III, then, presents verification test cases, which show the capabilities of the new solvers. Discussion and outlook toward future developments are presented in Sec. IV.

A. Spatiotemporal scales and computational requirements

As the goal of Vlasiator is to perform hybrid-kinetic simulations of the whole magnetospheric domain, the involved spatiotemporal scales place stringent requirements on the grid resolution and size. Kinetic electron dynamics are neglected in the hybrid approach, so characteristic length scales of interest are dictated by ion kinetic effects. They include, at the small end, the ion inertial length (whose value in the solar wind, for example, is $d_i = c/\omega_i \approx 230$ km, where c is the speed of light, and ω_i is the ion plasma frequency). Wavelengths of mirror modes and electromagnetic ion cyclotron (EMIC) waves in the magnetosheath are $\lambda_M \approx 300$ km.[13,14] The Larmor radius $r_L = \frac{mv}{qB}$ provides yet another kinetic length scale that varies strongly throughout the magnetospheric domain (from $r_L \approx 1500$ km in the solar wind to tens of kilometers in the cusp regions).

In addition to these smallest length scales that need to be resolved, the minimum outer system size of the simulation domain is dictated by the extents of the magnetospheric flow and current systems. With the simulation domain, shaped as a rectangular cuboid box in the GSM (geocentric-solar-magnetospheric) coordinate system, the bow shock standoff distance at $x \approx 12$ Earth Radii (R_E)[32] provides a minimum bound for the Sunward direction. If the simulation is expected to generate a foreshock extending toward the Sun, the box

must be extended to contain the majority of generated foreshock wave power. In the opposite direction, the magnetotail extends far in $-x$. The whole tail, however, cannot be encompassed in the simulation domain, so the constraint is to include the plasma domain relevant to and driving the features of interest. We have chosen tailward simulation domain extents down to $x \approx -100$ R_E (in contrast to much longer tails in MHD simulations, such as Ref. 33), allowing simulation data to be directly compared to spacecraft data from the ARTEMIS mission.[34]

In the two remaining spatial dimensions, the system size is determined by the bow shock shape: The box needs to be large enough to contain a significant part of the shock surface. Downstream from locations where the shock surface is allowed to intersect the box, the pressure balance has proven to be unreliable, which can negatively influence the model's physical validity. In Vlasiator global runs, the extents are, thus, typically chosen to range from $y, z \approx [-60; 60]$ R_E to provide generous leeway away from the inner magnetosphere and magnetotail.

In near-Earth space, the velocity space extents are dictated by the typical bulk velocity of the solar wind and non-thermal velocity structures (such as non-relativistic particle beams accelerated at the bow shock or by magnetic reconnection). Satellite observations and previous simulation runs[6,35] motivate velocity limits of $v_x, v_y, v_z \in [-4000; 4000]$ km/s under regular solar wind conditions. For the velocity space resolution, i.e., the size of a velocity space cell, a systematic study by Pfau-Kempf *et al.*[36] found that $\Delta v \approx 40$ km/s yields reasonable results for the physical processes relevant to magnetospheric simulations.

Given the resolution and spatial extent requirements discussed above, Table I tabulates the amount of grid points that the phase space needs to comprise in each of its six dimensions. From the table, we conclude that a fully resolved simulation would require a total of $N_r = n_x \times n_y \times n_z = 3822^3 = 4.5 \times 10^8$ 3D simulation cells, each of which contain a velocity space of $N_v = n_{v_x} \times n_{v_y} \times n_{v_z} = 200^3 = 8 \times 10^6$ velocity space cells. In a naïve approach to uniform grid discretization, the 6D phase space would, thus, contain $N_c = N_r \times N_v = 3.5 \times 10^{15}$ phase space cells, which leads to a memory requirement of about 72 exabytes. Obviously, this is far beyond anything a computing facility can be expected to provide in the foreseeable future.

The key strategy hitherto employed in Vlasiator to achieve a significant reduction of computational complexity is the *sparse velocity space* representation.[37] It makes use of the observed property of plasma velocity distribution functions in near-Earth space tend to be compactly localized in the velocity domain. Very often, high phase space densities are present in (near) Maxwellian core populations around the bulk velocity, sometimes with beam distributions clearly separated from the thermal core. Long tails of the distribution, such as power-law or kappa distribution functions, are present and important for high-energy particle transport studies, but their overall kinematic impact on global plasma dynamics is small compared to the bulk and beam structures. Large volumes of the velocity space are filled with phase space density values that are many orders of magnitude lower than the dynamically relevant peaks. Vlasiator deliberately discards those parts and stores and processes only the parts of velocity space in which the phase space density exceeds a set threshold value $f_{\min}$. In magnetospheric setups, we found that a sparsity threshold value of $f_{\min} = 10^{-15}$ s^3/m^6 led to a 98% reduction in modeled phase space volume,[36] while resulting in an overall mass loss of <1%. This directly

TABLE I. Listing of typical simulation extent and resolution requirements for global kinetic magnetospheric simulations in all six spatial and velocity dimensions. A straightforward Cartesian grid without refinement would lead to unfeasible simulation cell counts.

Dim. (GSM)	Physical extent L	Resolution requirement Δl	Required cell count n
x	$[-100; 20]\, \mathrm{R_E}$	1000 km (0.156 $\mathrm{R_E}$)	3822
y	$[-60; 60]\, \mathrm{R_E}$	1000 km (0.156 $\mathrm{R_E}$)	3822
z	$[-60; 60]\, \mathrm{R_E}$	1000 km (0.156 $\mathrm{R_E}$)	3822
v_x, v_y, v_z	$[-4000; 4000]\, \mathrm{km/s}$	40 km/s	200

translates to a 98% reduction in computational requirements, which made the spatially 2D Vlasiator simulations and scientific studies possible. This threshold value was also empirically found to retain foreshock beam distributions, thus leaving ion-kinetic beam instabilities in the foreshock intact.[9] For a more detailed description of the sparse velocity space mechanism, see Refs. 37 and 1.

However, this mechanism alone is not sufficient to make 3D + 3V global magnetospheric simulations feasible, as the memory requirement for the simulation data would still be of the order of (tens of) exabytes. To enable spatially 3D simulations, a further reduction in computational intensity is required. A common approach, especially in magnetohydrodynamic (MHD) modeling,[33,38,39] is to change the spatial grid structure to be non-uniformly discretized. Typically, a distinction is made between adaptive mesh refinement (AMR) approaches and unstructured sparse grids.

In an AMR setup, the simulation domain is initially discretized on a coarse Cartesian base mesh. The mesh resolution in certain areas of the domain is successively doubled by splitting simulation cells into smaller subcells. Thus, parts of the domain have double mesh resolution, parts have quadruple resolution, etc.

The choice for when to refine must be made judiciously based on the problem at hand: In some cases, the refinement will be made based on predefined geometric constraints (such as the choice of higher resolution for certain areas of interest). More generally, however, the *adaptivity* of the refinement process is expressed through a run-time evaluated refinement criterion. This refinement criterion continuously reevaluates whether any given simulation cell should be coarser or finer than it is at the moment.

A simulation code that employs a refined mesh needs to be designed such that it can operate on different grid resolutions and especially handle situations at refinement interfaces, where simulation cells have multiple neighbors of higher or lower resolution in one or multiple directions. Section II describes the details of how Vlasiator's solvers have been adapted and optimized for this task. Using these optimizations, the amount of simulation cells for a global 3D + 3V Vlasiator simulation is reduced to $N_c \approx 1.3 \times 10^{12}$ cells, leading to a total memory requirement of 33.7 terabytes, which is well within the capabilities of modern supercomputers.

Section II details the design choices and algorithmic changes to the Vlasiator data structures and solvers that make computationally efficient 3D + 3V simulations with a refined mesh geometry possible.

II. METHODS
A. The grid structure

Vlasiator's basic simulation grid is based on the distributed Cartesian cell-refinable grid (DCCRG) library,[40] which provides

memory layout, mesh geometry, and refinement and a parallel message passing interface (MPI) communication framework.[41]

The DCCRG framework enables adaptive mesh refinement at arbitrary locations of a Cartesian simulation domain, by splitting a cubic simulation cell into eight cubic subcells with twice the spatial resolution (due to its choice of solvers, Vlasiator currently requires exactly cubical cells). This refinement process can be iterated multiple times. The domain decomposition is neither tree nor patch based,[42,43] as every cell can be individually refined or unrefined without any cooperation of its local neighborhood (with the limitation that only one jump of refinement level is allowed at a time, and refinement interfaces need to be separated by at least two cells). As a result, the refinement domains can be freely located and the refinement interfaces can have arbitrary shapes. The solvers operating on that domain need to, therefore, be able to handle changes of refinement level on a cell-by-cell basis.

The current refinement strategy for Vlasiator global simulations is based on regions of interest. These regions are identified in a low-resolution run (and/or using estimates by well-established empirical relations for the bow shock and magnetopause positions,[44]) and are, thereby, predefined for the high-resolution simulation. Figure 1 shows an example of the mesh refinement geometry employed in such a situation. For future runs, however, dynamic refinement criteria are being investigated, based on local physical parameters and their gradients, as is commonly done in global MHD simulations (e.g., Ref. 33).
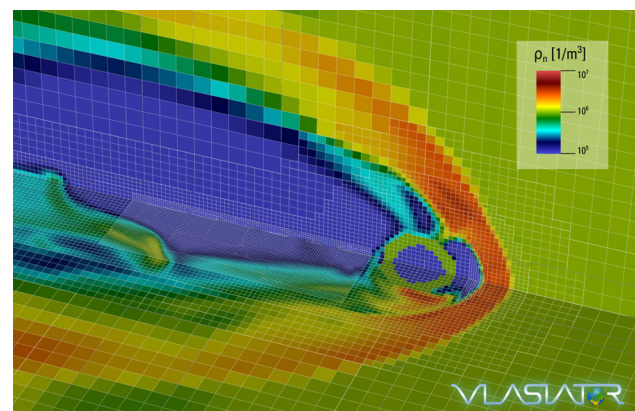

FIG. 1. Snapshot of a Vlasiator global magnetospheric simulation, showing the plasma density color-coded. The adaptive mesh refinement (AMR) is visible in the large contrast of cell sizes between the inflowing solar wind (coarsest cells) and the magnetotail (finest cells). In this simulation, refinement regions are predefined through simple geometric primitives.

While the DCCRG grid of Vlasiator stores the phase space density data as solved by the Vlasov equation, the electromagnetic fields are *not* discretized on the same mesh. They are propagated with a finite difference upwind field solver,[45] which stores its data on a simple uniform Cartesian array grid.[46] This grid does not employ mesh refinement (as the field solver does not need it), but is discretized at a constant resolution matching the finest AMR refinement level of the DCCRG grid.

As the computational demands of these simulations require running on large distributed-memory systems, both the Vlasov and field solver grids use a spatial domain decomposition approach. The strongly varying computational complexity of the sparse velocity space and AMR implementations makes it necessary for the Vlasov solver grid to dynamically rebalance the load of its decomposition domains. A combined action by the DCCRG grid and the Zoltan library[47] takes care of this process. The field solver, on the other hand, has a constant computational load per simulation cell, so that a simple MPI Cartesian communicator is used for initial grid partition; no rebalancing is required during run time.

This multi-mesh structure is referred to as *heterologous domain decomposition*. The intricacies of coupling between these two grids are discussed in a recent publication.[48]

B. The 6D Vlasov solver

At the heart of the numerical implementation of Vlasiator lies the phase space distribution function $f_s(\vec{x}, \vec{v}, t)$ for an ion species s (with charge q_s and mass m_s). Its dynamic evolution in interaction with the electromagnetic fields $\vec{E}$ and $\vec{B}$ is described by the Vlasov equation,[49]

$$\frac{\partial f_s}{\partial t} + \vec{v} \cdot \frac{\partial f_s}{\partial \vec{x}} + \frac{q_s}{m_s}(\vec{E} + \vec{v} \times \vec{B}) \cdot \frac{\partial f_s}{\partial \vec{v}} = 0. \tag{1}$$

This equation corresponds to the Boltzmann equation without collisions and the assumption that the only external force is given by the Lorentz force. Note that Vlasov's equation can be viewed as a multidimensional advection equation without any source terms.

Vlasiator directly solves Vlasov's equation and propagates the ion phase space density using a semi-Lagrangian solver based on the SLICE-3D method,[50] which was originally designed to solve hydrodynamic equations on a spherical surface. The six phase space dimensions (three real-space, and three velocity space dimensions, "3D + 3V") are treated independently, by reordering Eq. (1) as two 3D advection equation in position (velocity), with the velocity (spatial) derivative forming the source term on the right hand side,

$$\frac{\partial f_s}{\partial t} + \vec{v} \cdot \frac{\partial f_s}{\partial \vec{x}} = -\frac{q_s}{m_s}(\vec{E} + \vec{v} \times \vec{B}) \cdot \frac{\partial f_s}{\partial \vec{v}}, \tag{2}$$

$$\frac{\partial f_s}{\partial t} + \frac{q_s}{m_s}(\vec{E} + \vec{v} \times \vec{B}) \cdot \frac{\partial f_s}{\partial \vec{v}} = -\vec{v} \cdot \frac{\partial f_s}{\partial \vec{x}}. \tag{3}$$

This approach treats spatial transport (in real space) and acceleration of the distribution function (in velocity space) in separate solver steps that are performed consecutively. In the following deliberations, they are, respectively, referred to as the *translation* and the *acceleration* steps. Formally, this corresponds to a Strang-splitting approach to solving the Vlasov equation as a 6D hyperbolic conservation law.[51,52]

As the near-Earth plasmas modeled by Vlasiator are in good approximation non-relativistic, relativistic effects are neglected in the implementation. This has several benefits for the semi-Lagrangian structure of the translation step: Transport in all three spatial dimensions can be treated independently, as Newtonian kinematics does not introduce a Lorentz factor of relativistic kinematics. For each dimension, transport in the (x_i, v_{x_i}) subspace, hence, forms a linear shear of the distribution, as illustrated in Fig. 2.

In a similar way, certain properties of the Lorentz force acting on the plasma medium allow the acceleration step to be simplified. In the nonrelativistic Vlasov equation, the action of the Lorentz force $\vec{F}_l = q_s(\vec{v} \times \vec{B} + \vec{E})$ transforms the phase space distribution inside one simulation time step Δt like a solid rotator in velocity space: The magnetic field causes a gyration of the distribution function by the Larmor motion $\Delta\vec{v} = \partial\vec{v}/\partial t \, \Delta t = \frac{q_s}{m_s}(\vec{v} \times \vec{B} + \vec{E})\Delta t$. Any non-zero electric field accelerates electrically charged matter, which is a simple linear shift of velocity space.

Hence, the overall velocity space transformation is a rotation, whose axis direction matches the local magnetic field vector. The gyration center velocity is given by the convective plasma velocity $\vec{E} \times \vec{B}/B^2$. This transformation can be decomposed into three consecutive shear motions[53,54] along the coordinate axes, thus repeating the same fundamental algorithmic structure as the spatial translation update. The acceleration update affects the velocity space locally within each spatial simulation cell, so its properties are unaffected by the introduction of spatial AMR. This paper will, hence, focus on the changes to the translation solver step.

Iterating through the storage data structure of Vlasiator's sparse phase space has a high numerical complexity. In contrast, the access patterns of the solver are predictable and linear: The semi-Lagrangian solver performs the shear transformation in-order, in one dimension at a time. To benefit from this, phase space data are organized into a so-called *pencil* structure at the beginning of each solver step, in which spatially adjacent data, in one Cartesian direction at a time, are stored linearly in memory. These pencils share no data dependencies during the shear and can be processed in parallel.

In a refined mesh, construction of these pencils carries an additional complication, since neighboring cells at the mesh refinement boundaries can have different sizes. The algorithm performs the following steps for every spatial update direction (as illustrated in Fig. 2):

(1) Find pencil seed cells
    The goal is to build pencils of spatially adjacent cells. This process starts with the *seed cells* at one boundary of the domain. These are all cells that do not contain a local neighbor in the backwards direction of the chosen coordinate.
(2) Pencil construction
    The pencil is built by identifying the neighbors of the seed cell in the positive coordinate direction and appending them to the pencil data structure. This process continues until no further neighbor cells are found (either at the opposite simulation boundary, or the edge of the local MPI domain).
(3) Pencil splitting
    When a refinement interface is encountered in this process, the next neighbors are four cells at higher mesh resolution. Here, the existing pencil structure is stopped, and four new separate pencils continue. To minimize the amount of data movement, the split pencils also contain cells from the coarser domain

corresponding to the semi-Lagrangian update's stencil size (two cells in common simulation runs). This process advances recursively across any further refinement interface.

In the case where a refined pencil encounters a re-coarsening of the mesh, the pencil construction stops, again taking padding by the stencil size into account. A new seed point (and hence, a new pencil) is started to continue at the coarser level.

(4) Result merging and remote neighbor contributions

Even a completed pencil still has a mass outflow toward a further neighbor cell. This can, in many cases, mean that a boundary of the spatial decomposition over parallel processes was encountered, and a remote neighbor cell exists, into which the transported phase space density flows. This is called the *remote neighbor contribution* of the pencil.

In situations where the end of the pencil coincides with a coarsening refinement interface, four pencils will feed their remote neighbor contribution into the same target cell, so the target cell provides a target buffer for up to four incoming phase space density values, which are summed after the transport step is completed.

Once the complete set of pencils has been constructed, meaning that the whole 3D space is organized into pencil structures, the next step of the solver acts on one phase space cell at a time, interpolating the discrete phase space values of the pencils by a reconstruction polynomial (depicted as the red curve in Fig. 2). The phase space integral values of each cell are taken as control points for a polynomial using the nonuniform piecewise parabolic reconstruction method of Ref. 55. A slope limiter is used at this point, to ensure the total variation diminishing (TVD) property of the overall solver step.

The reconstructed polynomial gets shifted in phase space to perform the actual phase space transport. The distance of this shift for a translation update is given by

$$\Delta x_i = v_i \, \Delta t, \tag{4}$$

where the velocity v_i is constant for the whole pencil.

In the acceleration update, the shift in velocity,

$$\Delta v_i = a_i \, \Delta t, \tag{5}$$

is determined by the acceleration characteristic a_i from the Lorentz force.

The shifted polynomial is integrated over the target cell extent to obtain updated phase space density values. These are transferred back from the pencil structure into the phase space grid data structure.

This entire process gets repeated for all three spatial dimensions, leading to an effective three-dimensional spatial transport step. As a performance optimization, the pencilization process is only performed once and its metadata are retained until the MPI load balances change the spatial decomposition of the simulation domain.

C. Coupling between Vlasov and field solver

To complete the description of a plasma system in the hybrid-kinetic regime, the Vlasov equation [Eq. (1)] is solved together with Maxwell's equations in the Darwin approximation.[1] The electric field $\vec{E}$ and magnetic field $\vec{B}$ act on the distribution functions, and in return, statistical moments of the particle distribution functions (overall mass density ρ_m, charge density ρ_q, bulk velocity $\vec{V}$, and pressure tensor $\vec{P}$) are fed into the field solver. Hence, data need to be exchanged between the two equations' solver procedures.[48]

As Vlasiator's field solver and Vlasov solver operate on separate grids, they both require a coupling process during each simulation time step, to be used as source terms for each other's update. The association between the two grids is calculated at initialization time and updated after any load balancing step. Reference 48 presents an in-depth discussion of details of the coupling procedure between the two meshes, including the spatial filtering step employed to smooth out discontinuities in areas of resolution mismatch.

D. Boundary conditions

Vlasiator offers a selection of boundary conditions to handle different simulation scenarios, described in detail in Ref. 1 Inflowing boundary conditions enable solar wind to enter the simulation, with arbitrary time-varying plasma conditions.[56] Periodic boundary conditions are used to enable simulation in one and two spatial dimensions, and outflow boundary conditions are used to enable plasma and
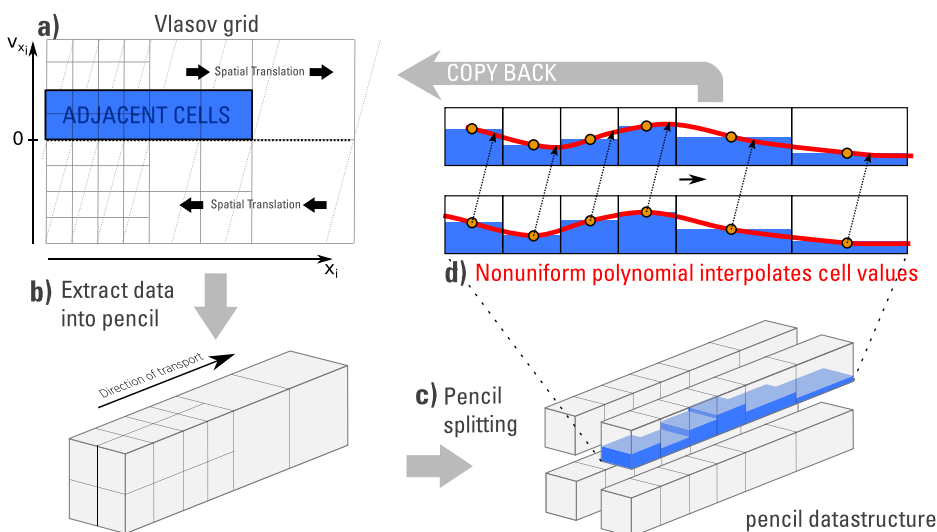
FIG. 2. Illustration of the elementary semi-Lagrangian shear step that underlies the Vlasiator Vlasov solver. Panel (a) shows the example of a 1D + 1D real space/velocity space cut through the simulation domain, as it shears during a translation update. Panel (b) presents how phase space density information from adjacent cells in the update direction is extracted into a linear pencil array. This pencil is split to match its finest resolution cells in panel (c). An interpolating polynomial is reconstructed through the phase space density control points at cell centers, shown in panel (d). This polynomial is advected, and the resulting target phase space values are integrated over the volume of the pencil intersecting the cell and written back into the phase space data structure.
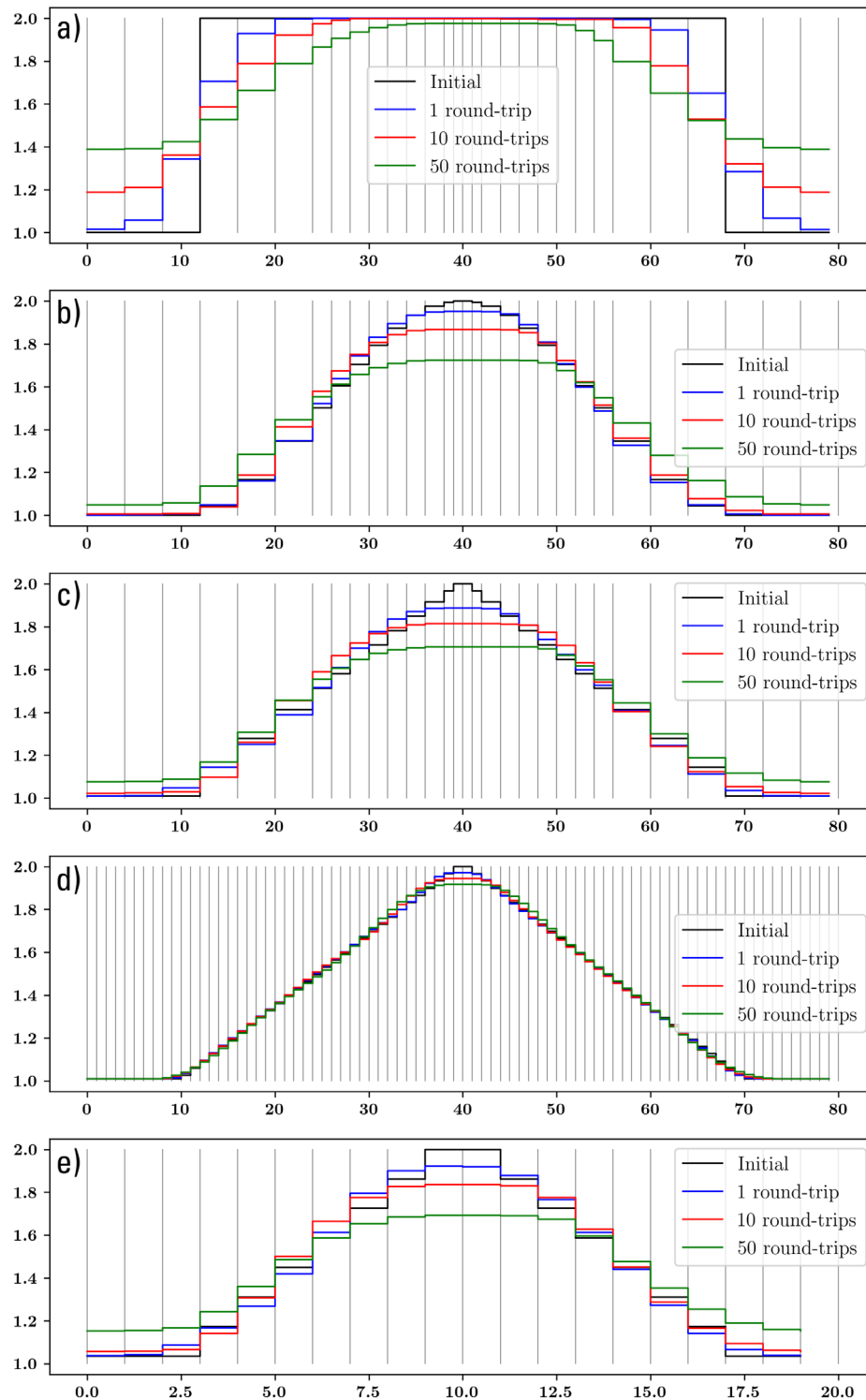
FIG. 3. Accuracy of translation method with square, sinusoidal, and triangular density modulations. In panels (a)–(c), they were propagated over a partially refined periodic simulation domain. Panels (d) and (e) present the behavior of the same triangular modulation in a fully refined and completely unrefined mesh, for comparison. The original signal is shown in black, and the signal after propagation is shown in blue (having traveled once around the periodic boundaries), red (ten crossing times), and green (50 crossing times). Vertical lines indicate the simulation resolution.

electromagnetic fields to exit the simulation domain. Outflowing boundary conditions are implemented as homogeneous Neumann boundaries, essentially copying the values from the neighboring simulation cells. The inner boundary of the magnetospheric setup is terminated by an ionospheric shell, set at about $4.7 R_E$ and modeled as a perfect conductor. The ionospheric boundary cells' phase space density is initialized with a Maxwellian distribution and remains constant throughout the run. The electric field at the ionosphere is set to zero, and the magnetic field component is initialized with values corresponding to the Earth dipole at those locations.

III. RESULTS

Section III presents a number of validation test runs, in which the performance of the Vlasov solver has been tested. Full discussion and in-depth analysis of the physical properties of our large-scale 3D magnetospheric simulations are subject of an upcoming publication.

A. Performance of the semi-Lagrangian solver

To ensure that the propagation algorithm performs well across simulation refinement interfaces, three tailored simulations were performed, where a test distribution was propagated in a periodic box with a size of $20 \times 1 \times 1$ cells at coarsest refinement level. In these tests, the semi-Lagrangian solver was run in isolation from the rest of the simulation system (especially without any interaction of the field solver). A monoenergetic plasma (moving at constant velocity v) with a spatial density modulation of square, sinusoidal, or triangular form was initialized in the center of the domain, with two levels of mesh refinement implemented in the same region. A fixed time step $\Delta t = 0.8 \frac{\Delta x}{v}$ (corresponding to a spatial Courant–Friedrichs–Lewy (CFL) value of 0.8 at the finest refinement level) was used. In 100 simulation time steps, the density modulation propagates once around the boxes' periodic boundaries and arrives at its starting position. Figures 3(a)–3(c) show the spatial profiles for all three signals at the start of the simulation, after 100, 1000, and 5000 time steps (corresponding to 1, 10, and 50 complete transitions of the simulation box). In panels (d) and (e), the triangular setup has been repeated with a uniform grid discretization at the highest and lowest resolution level, respectively.

It is apparent that the use of slope limiters caused some diffusion, but overall the signals are conserved adequately. In particular, there are no distortions or artifacts created at the interfaces between refinement regions. Comparison of the mesh-refined triangle run [panel (c)] with the high- and low-resolution uniform mesh runs [panels (d) and (e)] shows that the AMR mechanism does not add any extra diffusivity, and its effective diffusive behavior lies between the high- and low-resolution case. The small asymmetries visible in the final result stem from the selected propagation direction ($+x$) and were mirrored when the tests where repeated in the opposite flow direction. These asymmetries stem from the applied parabolic polynomial fitting method that uses four data points, which cannot be selected around the source cell in a symmetric fashion.

B. Speedup and computational effort

As a benchmark problem for realistic plasma simulations, a shock-in-a-box test was run, in which an elongated rectangular simulation domain (with a size of $L_x = 3.2 \times 10^4$ km, L_y, and $L_z = 2 \times 10^2$ km) is initially filled with two plasma regions that correspond to the Rankine–Hugoniot conditions of an interplanetary shock in the de Hoffmann–Teller frame.[57,58] These simulations have a similar setup as employed in Ref. 36, with the important distinction that they are fully three-dimensional meshes. Figure 4 shows the
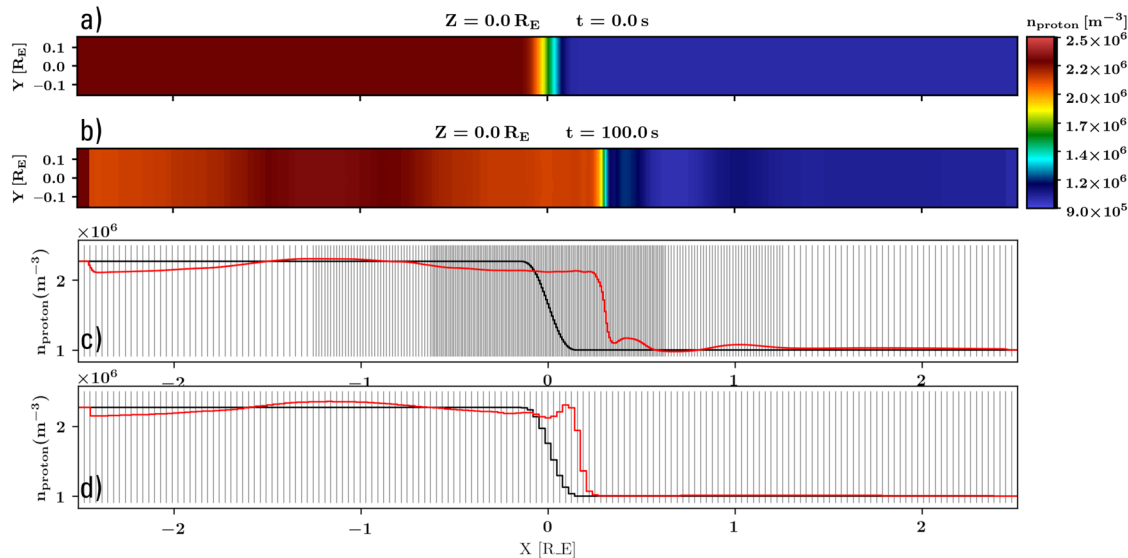

FIG. 4. Setup and results of shock-in-a-box simulations used for computational scaling tests. Panel (a) shows the initial simulation state: The box is an elongated rectangle, inside which a standing shock wave is initialized. Panel (b) presents the situation after $t = 100$ s. A foreshock has formed and kinetic effects breaking the Rankine–Hugoniot conditions have shifted the position of the shock. Panel (c) outlines a 1D cut through the mesh geometry (showing the simulation cells) and density profile at $t = 0$ s (black) and $t = 100$ s (red) of simulation run `AMR-18`. In panel (d), the same information is presented for the unrefined run `low-1`.

simulation box, mesh refinement structure, and resulting density profiles, and Ref. 59 contains full simulation configuration and output files.

The parameters are chosen such that the shock has zero velocity in the simulation frame and remains quasi-stationary in the central region of the box. Note that in the short simulation time frame, complex 3D shock dynamics do not have time to form. The ion inertial length is $c/\omega_{pi} = 223$ km upstream and 148 km downstream of the shock. Table II compares five different runs of this scenario. In the "low-1" run, the simulation was discretized with a unrefined mesh of $160 \times 10 \times 10$ cells in size (yielding a spatial resolution of $\Delta x = 200$ km). For "med-8" and "high-64," this was increased to $320 \times 20 \times 20$ cells ($\Delta x = 100$ km) and $640 \times 40 \times 40$ cells ($\Delta x = 50$ km), respectively, so these three simulations provide a weak-scaling set of the solver implementation without AMR.

The results were obtained on the "Vorna" cluster at the University of Helsinki, consisting of 180 nodes with Intel Xeon E5–2670 CPUs at 2.60 GHz and an InfiniBand QDR interconnect.

Simulation "AMR-18" from Table II was set up with the same low base resolution as "low-1," but the central part of the domain ($x \in [-4 \times 10^6, 4 \times 10^6]$ m) underwent two levels of mesh refinement, giving that region the same effective 50 km resolution as "high-64." Furthermore, simulation "eqv-18" was initialized with $416 \times 26 \times 26$ uniform cells resulting in a spatial cell count (281 216) close to what the AMR run achieved (296 000), which yielded a uniform resolution of $\Delta x = 76.92$ km throughout the box.

The simulation time step Δt was determined from the advection CFL condition. As this value depends on the size of spatial cells, each run had to perform a different amount of time steps in order to reach the simulation target time of 100 s.

As the figures in the table show, the AMR simulations' amount of simulation cells N_c lies far below that of "high-64" and its memory consumption and total computation resource use are reduced accordingly. As the AMR solver pencilization process is more complex than that in the unrefined case, the effective computational effort per phase-space cell, here expressed as $t_{wall} \cdot N_{CPU}/(N_f \cdot N_{tsteps})$ (μs), is noticeably higher. This can also directly be inferred from comparing the resources spent by "AMR-18" and "eqv-18." The increased computational complexity of the AMR solver, however, is offset by the increase in effective physical phase space volume per computational unit, so that AMR Vlasov simulations clearly outperform unrefined setups in terms of overall computational effort per physical volume or conversely allow for a much finer resolution at the physically relevant regions at the same computational cost.
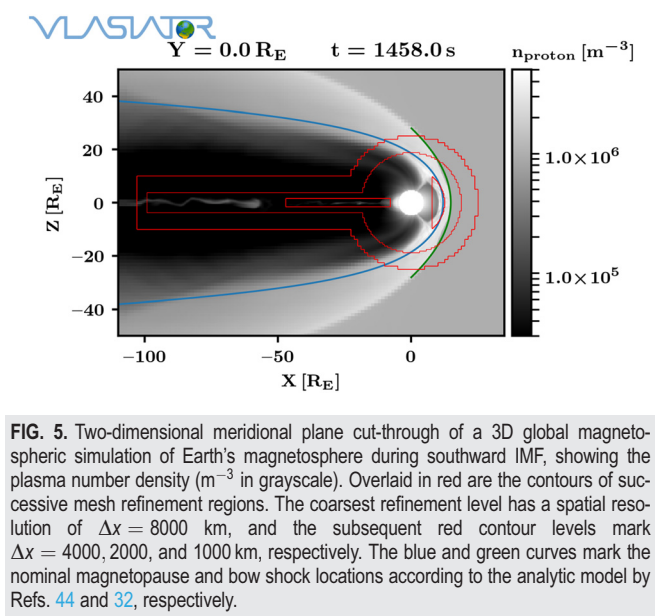
C. Global magnetospheric simulations

Finally, the scientific use case for adaptive mesh refinement in Vlasiator is the study of global, three-dimensional hybrid kinetic simulations of Earth's entire magnetosphere.

Figures 1 and 5 present different visualizations of the mesh refinement structure currently employed in Vlasiator global runs. This run modeled the plasma around Earth, in a box extending in geocentric solar ecliptic (GSE) coordinates $x \in [-110; 50]; R_E, y, z \in [-57; 57] R_E$, with an incoming, steady solar wind density of $n_{SW} = 10^6$ m^{-3}, flowing at a speed of $v_{SW} = 750$ km/s in the $-x$ direction. The solar wind carried an interplanetary magnetic field of $B_z = -5$ nT in the purely southward direction, resulting in spatially well-known positions for the day- and nightside reconnection regions. The simulation's spatial grid had a coarsest resolution level of $\Delta x = 8000$ km with three subsequent levels of refinement, yielding a finest resolution of $\Delta x = 1000$ km. The refinement regions were chosen to give maximum resolution coverage for the dayside magnetopause and tail reconnection regions. Note that the ion inertial length $c/\omega_{pi} \approx 230$ km in this simulation is smaller than even the finest grid resolution. This was likewise the case in previous publications,[14,30,36] which have nonetheless provided valuable insights into physics of the magnetosphere, that would be hard to obtain using other simulation methods. As this manuscript focusses on the implementation details of adaptive mesh refinement, the inclined reader is referred to Ref. 61, which presents a recent study further quantifying the physical fidelity of Vlasiator simulations at low resolution and presenting a subgrid method to mitigate some of its effects.

In Fig. 5, the blue line provides an analytic estimate of the magnetopause location using the model of Ref. 44 calculated from the solar wind parameters of the simulation. The magnetospheric geometry that has self-consistently developed in the simulation matches this prediction quite well, but the magnetopause standoff distance appears to mismatch by about 2 R_E. The authors of Ref. 62, however, have argued that the model by Ref. 44 overestimates the radial distance by about 1 R_E, an effect that is most likely even more exacerbated by the fact that this simulation has an unusually high solar wind temperature of $T = 0.5$ MK, leading to an enhanced thermal pressure in the magnetosheath and somewhat stronger compression of the magnetosphere, resulting in a smaller magnetopause standoff distance. Most importantly, the simulation shows no sudden discontinuity or departure from the prediction in different refinement regions.

The green line in Fig. 5 presents the analytic bow shock location model by Ref. 32. It again shows that the Vlasiator bow shock geometry matches well with observational behavior. Some departure from

TABLE II. Memory usage and wall time requirement of shock-in-a-box simulation test runs. N_c is the number of spatial cells, and N_f is the overall number of phase space cells in the simulation. "Mem" denotes the simulations' total resident memory as measured by the PAPI library.[60] The wall time t_{wall} and the core count N_{CPU} provide the total computational resource use CPUh. All simulations ran for $t_{phys} = 100$ s using different numbers of simulation steps N_{steps}. The efficiency parameter **Eff** $= t_{wall} \cdot N_{CPU}/(N_f \cdot N_{tsteps})$ (μs) denotes the processing time spent per step per phase-space cell.

Run	N_c	N_f	Δx (km)	N_{CPU}	Mem (GB)	t_{wall} (s)	CPUh	N_{tsteps}	Eff (μs)
low-1	16 000	1.1×10^9	200	32	15.74	5500	48.9	495	0.33
med-8	128 000	7.7×10^9	100	256	182.7	12883	916	989	0.43
high-64	1 024 000	6.1×10^{10}	50	2048	1545.9	31695	18030	1978	0.53
AMR-18	296 000	1.8×10^{10}	50–200	608	524.9	28955	4890	1758	0.55
eqv-18	281 216	1.7×10^{10}	76.92	608	429.5	17548	2963	1286	0.49

FIG. 5. Two-dimensional meridional plane cut-through of a 3D global magnetospheric simulation of Earth's magnetosphere during southward IMF, showing the plasma number density (m^{-3} in grayscale). Overlaid in red are the contours of successive mesh refinement regions. The coarsest refinement level has a spatial resolution of $\Delta x = 8000$ km, and the subsequent red contour levels mark $\Delta x = 4000, 2000$, and 1000 km, respectively. The blue and green curves mark the nominal magnetopause and bow shock locations according to the analytic model by Refs. 44 and 32, respectively.

the ideal parabolic shape of the model is visible, but this is likely due to kinetic effects of the cusps and transient magnetosheath phenomena, as previously presented in Refs. 19 and 21, and stays within the confidence range given in Ref. 32.

Direct performance comparison with a high-resolution 2D Vlasiator production run using the Vlasiator solver without AMR was performed on the *Mahti* supercomputer at the CSC—Center for Scientific Computing in Kajaani, Finland, a BullSequana XH2000 system with AMD Rome 7H12 CPUs at 2.6 GHz and an InfiniBand HDR100 interconnect. Table III outlines the cell counts, memory requirement, and effective calculation time per core per cell of both runs. As before, the AMR solver carries a performance penalty in terms of computing time per simulated phase space cell. There is also still somewhat of a resolution gap between the 2D simulation (at $\Delta x = 300$ km) and the 3D run (at $\Delta x = 1000$ km), but the overall reduction in required phase space cells means that a simulated 3D volume, instead of a 2D slice, is now possible and meaningful scientific results can now be attained.

IV. CONCLUSIONS AND OUTLOOK

Studies of the near-Earth plasma environment are strongly dependent on space plasma simulations, whose computational requirements continue to use a major share of supercomputing resources worldwide. The methods and results presented in this paper show that, using a combination of sparse velocity space representation and

spatial adaptive mesh refinement, it is possible to simulate the entire outer extents of Earth's magnetosphere close to hybrid-kinetic scales in three dimensions using a Vlasov simulation code. This outcome opens the door for modeling studies of magnetospheric plasma phenomena in three dimensions, among these complex phenomena benefiting from a global view (without the previously required couplings of different simulation approaches[63] or limitation to stochastic sampling of distribution functions as in the PIC approach[4]).

Section II introduces the dimensional splitting approach used in Vlasiator's solver and described the dimension-by-dimension dynamic pencilization and semi-Lagrangian solver strategy. The combination of adaptive mesh refinement and a massively parallelized, low-diffusive phase space solver is novel, and its fields of applicability are only just starting to be explored.

The scaling results in Subsection III B demonstrate how the AMR implementation provides a very effective reduction of memory usage and overall computation time compared to unrefined simulations. Through the introduction of the AMR solver, individual computations for each computational cell had incurred a computation time overhead of about 50%. Yet, this is counterbalanced by the fact that the number of simulation cells directly reduces by a factor of eight for every introduced refinement level, thus leading to an overall significant performance gain.

Global magnetospheric simulations performed with these new methods have been validated against the Ref. 44 magnetopause and the Ref. 32 bow shock model and show that magnetospheric dynamics remains intact and consistent across all refinement levels. The simulation extents and resolutions that Vlasiator achieves using the methods outlined here are now approaching those of global MHD simulations[64] and are able to encompass kinetic phenomena that before escaped global modeling. Yet, even with the current improvements, resolution of the ion inertial length is still beyond the numerical capabilities of supercomputers in the unrefined parts of the simulation domain. Improvement of simulation techniques and implementations are ongoing.

The current mesh refinement process is only adaptive in its spatial dimensions, while the velocity resolution remains fixed. One potential future optimization would be the implementation of an adaptively refined mesh also in the velocity domain of each cell, thus yielding a fully six-dimensional mesh refinement and hopefully providing another order-of-magnitude improvement of computational performance. Another avenue is to decouple the time stepping in regions of different resolution and different physical properties (as in Ref. 65) enabling longer time steps where possible and focusing the computational power on the regions of dynamics at small spatiotemporal scales.

TABLE III. Memory usage and wall time requirement of global magnetospheric simulation runs. Column labels are identical to Table II.

Run	N_c	N_f	Δx (km)	N_{CPU}	Mem (TiB)	Eff (μs)
2D	6×10^6	1.9×10^{11}	300	128 000	11.89	4.18
3D	1.3×10^6	1.56×10^9	1000	64 000	4.82	7.08

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Urs Ganse: Conceptualization (equal); Formal analysis (equal); Funding acquisition (equal); Investigation (lead); Methodology (supporting); Software (equal); Validation (equal); Visualization (equal); Writing – original draft (lead); Writing – review & editing (lead). **Harriet George:** Writing – review & editing (supporting). **Evgeny Gordeev:** Writing – review & editing (supporting). **Maxime Grandin:** Writing – review & editing (supporting). **Konstantinos Horaites:** Writing – review & editing (supporting). **Jonas Suni:** Writing – review & editing (supporting). **Vertti Tarvus:** Writing – review & editing (supporting). **Fasil Tesema Kebede:** Writing – review & editing (supporting). **Lucile Turc:** Writing – review & editing (supporting). **Hongyang Zhou:** Writing – review & editing (supporting). **Minna Palmroth:** Funding acquisition (equal); Project administration (lead); Resources (lead); Writing – review & editing (supporting). **Tuomas Koskela:** Conceptualization (equal); Methodology (lead); Software (lead); Writing – review & editing (supporting). **Markus Battarbee:** Conceptualization (equal); Formal analysis (equal); Investigation (equal); Methodology (supporting); Software (equal); Visualization (equal); Writing – review & editing (supporting). **Yann Pfau-Kempf:** Conceptualization (equal); Formal analysis (equal); Methodology (supporting); Software (equal); Visualization (supporting); Writing – review & editing (supporting). **Konstantinos Papadakis:** Software (supporting); Validation (supporting); Writing – review & editing (supporting). **Markku Alho:** Writing – review & editing (supporting). **Maarja Bussov:** Writing – review & editing (supporting). **Giulia Cozzani:** Writing – review & editing (supporting). **Maxime Dubart:** Writing – review & editing (supporting).

DATA AVAILABILITY

The Vlasiator simulation code is distributed under the GPL-2 open-source license at https://github.com/fmihpc/vlasiator.[66] Full simulation data for the presented analysis are stored in the University of Helsinki Datacloud. Data presented in this paper can be accessed by following the data policy on the Vlasiator website https://helsinki.fi/vlasiator. Data handling and visualization are supported through the Analysator python package https://github.com/fmihpc/analysator as well as a plugin for the VisIt visualization system.[67] Simulation configuration files and raw performance data are available for download from Ref. 59.

REFERENCES

[1]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 Rev. Comput. Astrophys. **4**(1), 1 (2018).

[2]D. Winske, L. Yin, N. Omidi, H. Karimabadi, and K. Quest, "Hybrid simulation codes: Past, present and future—A tutorial," in *Space Plasma Simulation* (Springer Berlin Heidelberg, 2003), pp. 136–165.

[3]R. W. Hockney and J. W. Eastwood, *Computer Simulation Using Particles* (Taylor & Francis, Bristol, 1988).

[4]Y. Lin, X. Y. Wang, M.-C. Fok, N. Buzulukova, J. D. Perez, L. Cheng, and L.-J. Chen, "Magnetotail-inner magnetosphere transport associated with fast flows based on combined global-hybrid and CIMI simulation," J. Geophys. Res.: Space Phys. **126**, e2020JA028405, https://doi.org/10.1029/2020JA028405 (2021).

[5]Y. Kempf, D. Pokhotelov, O. Gutynska, L. B. Wilson III, B. M. Walsh, S. V. Alfthan, O. Hannuksela, D. G. Sibeck, and M. Palmroth, "Ion distributions in the Earth's foreshock: Hybrid-Vlasov simulation and THEMIS observations," J. Geophys. Res.: Space Phys. **120**, 3684–3701, https://doi.org/10.1002/2014JA020519 (2015).

[6]M. Palmroth, M. Archer, R. Vainio, H. Hietala, Y. Pfau-Kempf, S. Hoilijoki, O. Hannuksela, U. Ganse, A. Sandroos, S. V. Alfthan, and J. P. Eastwood, "ULF foreshock under radial IMF: THEMIS observations and global kinetic simulation Vlasiator results compared," J. Geophys. Res.: Space Phys. **120**, 8782–8798, https://doi.org/10.1002/2015JA021526 (2015).

[7]L. Turc, U. Ganse, Y. Pfau-Kempf, S. Hoilijoki, M. Battarbee, L. Juusola, R. Jarvinen, T. Brito, M. Grandin, and M. Palmroth, "Foreshock properties at typical and enhanced interplanetary magnetic field strengths: Results from hybrid-Vlasov simulations," J. Geophys. Res.: Space Phys. **123**, 5476–5493, https://doi.org/10.1029/2018JA025466 (2018).

[8]L. Turc, O. W. Roberts, M. O. Archer, M. Palmroth, M. Battarbee, T. Brito, U. Ganse, M. Grandin, Y. Pfau-Kempf, C. P. Escoubet, and I. Dandouras, "First observations of the disruption of the earth's foreshock wave field during magnetic clouds," Geophys. Res. Lett. **46**, 12644–12653, https://doi.org/10.1029/2019GL084437 (2019).

[9]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," Ann. Geophys. **38**, 1081–1099 (2020).

[10]L. Turc, V. Tarvus, A. P. Dimmock, M. Battarbee, U. Ganse, A. Johlander, M. Grandin, Y. Pfau-Kempf, M. Dubart, and M. Palmroth, "Asymmetries in the Earth's dayside magnetosheath: Results from global hybrid-Vlasov simulations," Ann. Geophys. **38**, 1045–1062 (2020).

[11]K. Takahashi, L. Turc, E. Kilpua, N. Takahashi, A. Dimmock, P. Kajdic, M. Palmroth, Y. Pfau-Kempf, J. Soucek, T. Motoba, M. D. Hartinger, A. Artemyev, H. Singer, U. Ganse, and M. Battarbee, "Propagation of ultralow-frequency waves from the ion foreshock into the magnetosphere during the passage of a magnetic cloud," J. Geophys. Res.: Space Phys. **126**, e2020JA028474, https://doi.org/10.1029/2020JA028474 (2021).

[12]A. Johlander, M. Battarbee, A. Vaivads, L. Turc, Y. Pfau-Kempf, U. Ganse, M. Grandin, M. Dubart, Y. V. Khotyaintsev, D. Caprioli, C. Haggerty, S. J. Schwartz, B. L. Giles, and M. Palmroth, "Ion acceleration efficiency at the earth's bow shock: Observations and simulation results," Astrophys. J. **914**, 82 (2021).

[13]S. Hoilijoki, M. Palmroth, B. M. Walsh, Y. Pfau-Kempf, S. von Alfthan, U. Ganse, O. Hannuksela, and R. Vainio, "Mirror modes in the earth's magnetosheath: Results from a global hybrid-Vlasov simulation," J. Geophys. Res.: Space Phys. **121**, 4191–4204, https://doi.org/10.1002/2015JA022026 (2016).

[14]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," Ann. Geophys. **38**, 1283–1298 (2020).

[15]M. Palmroth, H. Hietala, F. Plaschke, M. Archer, T. Karlsson, X. Blanco-Cano, D. Sibeck, P. Kajdič, U. Ganse, Y. Pfau-Kempf, M. Battarbee, and L. Turc, "Magnetosheath jet properties and evolution as determined by a global hybrid-Vlasov simulation," Ann. Geophys. **36**, 1171–1182 (2018).

[16]M. Palmroth, S. Raptis, J. Suni, T. Karlsson, L. Turc, A. Johlander, U. Ganse, Y. Pfau-Kempf, X. Blanco-Cano, M. Akhavan-Tafti, M. Battarbee, M. Dubart, M. Grandin, V. Tarvus, and A. Osmane, "Magnetosheath jet evolution as a function of lifetime: Global hybrid-Vlasov simulations compared to MMS observations," Ann. Geophys. **39**, 289 (2021).

[17]J. Suni, M. Palmroth, L. Turc, M. Battarbee, A. Johlander, V. Tarvus, M. Alho, M. Bussov, M. Dubart, U. Ganse, M. Grandin, K. Horaites, T. Manglayev, K. Papadakis, Y. Pfau-Kempf, and H. Zhou, "Connection between foreshock structures and the generation of magnetosheath jets: Vlasiator results," Geophys. Res. Lett. **48**(20), e2021GL095655, https://doi.org/10.1029/2021GL095655 (2021).

[18]S. Hoilijoki, U. Ganse, Y. Pfau-Kempf, P. A. Cassak, B. M. Walsh, H. Hietala, S. von Alfthan, and M. Palmroth, "Reconnection rates and X line motion at the magnetopause: Global 2D-3V hybrid-Vlasov simulation results," J. Geophys. Res.: Space Phys. **122**, 2877–2888, https://doi.org/10.1002/2016JA023709 (2017).

[19]R. Jarvinen, R. Vainio, M. Palmroth, L. Juusola, S. Hoilijoki, Y. Pfau-Kempf, U. Ganse, L. Turc, and S. von Alfthan, "Ion acceleration by flux transfer events in the terrestrial magnetosheath," Geophys. Res. Lett. **45**, 1723–1731, https://doi.org/10.1002/2017GL076192 (2018).

[20]S. Hoilijoki, U. Ganse, D. G. Sibeck, P. A. Cassak, L. Turc, M. Battarbee, R. C. Fear, X. Blanco-Cano, A. P. Dimmock, E. K. J. Kilpua, R. Jarvinen, L. Juusola, Y. Pfau-Kempf, and M. Palmroth, "Properties of magnetic reconnection and FTEs on the dayside magnetopause with and without positive IMF Bx component during southward IMF," J. Geophys. Res.: Space Phys. **124**, 4037–4048, https://doi.org/10.1029/2019JA026821 (2019).

[21]Y. Pfau-Kempf, H. Hietala, S. E. Milan, L. Juusola, S. Hoilijoki, U. Ganse, S. von Alfthan, and M. Palmroth, "Evidence for transient, local ion foreshocks caused by dayside magnetopause reconnection," Ann. Geophys. **34**, 943–959 (2016).

[22]M. Palmroth, S. Hoilijoki, L. Juusola, T. I. Pulkkinen, H. Hietala, Y. Pfau-Kempf, U. Ganse, S. von Alfthan, R. Vainio, and M. Hesse, "Tail reconnection in the global magnetospheric context: Vlasiator first results," Ann. Geophys. **35**, 1269–1274 (2017).

[23]A. Runov, M. Grandin, M. Palmroth, M. Battarbee, U. Ganse, H. Hietala, S. Hoilijoki, E. Kilpua, Y. Pfau-Kempf, S. Toledo-Redondo, L. Turc, and D. Turner, "Ion distribution functions in magnetotail reconnection: Global hybrid-Vlasov simulation results," Ann. Geophys. **39**, 599–612 (2021).

[24]L. Juusola, S. Hoilijoki, Y. Pfau-Kempf, U. Ganse, R. Jarvinen, M. Battarbee, E. Kilpua, L. Turc, and M. Palmroth, "Fast plasma sheet flows and X line motion in the earth's magnetotail: Results from a global hybrid-Vlasov simulation," Ann. Geophys. **36**, 1183–1199 (2018).

[25]L. Juusola, Y. Pfau-Kempf, U. Ganse, M. Battarbee, T. Brito, M. Grandin, L. Turc, and M. Palmroth, "A possible source mechanism for magnetotail current sheet flapping," Ann. Geophys. **36**, 1027–1035 (2018).

[26]M. Grandin, M. Battarbee, A. Osmane, U. Ganse, Y. Pfau-Kempf, L. Turc, T. Brito, T. Koskela, M. Dubart, and M. Palmroth, "Hybrid-Vlasov modelling of nightside auroral proton precipitation during southward interplanetary magnetic field conditions," Ann. Geophys. **37**, 791–806 (2019).

[27]M. Grandin, L. Turc, M. Battarbee, U. Ganse, A. Johlander, Y. Pfau-Kempf, M. Dubart, and M. Palmroth, "Hybrid-Vlasov simulation of auroral proton precipitation in the cusps: Comparison of northward and southward interplanetary magnetic field driving," J. Space Weather Space Clim. **10**, 51 (2020).

[28]M. Battarbee, T. Brito, M. Alho, Y. Pfau-Kempf, M. Grandin, U. Ganse, K. Papadakis, A. Johlander, L. Turc, M. Dubart, and M. Palmroth, "Vlasov simulation of electrons in the context of hybrid global models: An eVlasiator approach," Ann. Geophys. **39**, 85–103 (2021).

[29]M. Alho, M. Battarbee, Y. Pfau-Kempf, Y. V. Khotyaintsev, R. Nakamura, G. Cozzani, U. Ganse, L. Turc, A. Johlander, K. Horaites, V. Tarvus, H. Zhou, M. Grandin, M. Dubart, K. Papadakis, J. Suni, H. George, M. Bussov, and M. Palmroth, "Electron signatures of reconnection in a global eVlasiator simulation," Geophys. Res. Lett. **49**, e2022GL098329, https://doi.org/10.1029/2022GL098329 (2022).

[30]Y. Pfau-Kempf, M. Palmroth, A. Johlander, L. Turc, M. Alho, M. Battarbee, M. Dubart, M. Grandin, and U. Ganse, "Hybrid-Vlasov modeling of three-dimensional dayside magnetopause reconnection," Phys. Plasmas **27**, 092903 (2020).

[31]M. Palmroth, "Daring to think of the impossible: The story of vlasiator," Front. Astron. Space Sci. **9**, 952248 (2022).

[32]J. Merka, A. Szabo, J. A. Slavin, and M. Peredo, "Three-dimensional position and shape of the bow shock and their variation with upstream Mach numbers and interplanetary magnetic field orientation," J. Geophys. Res.: Space Phys. **110**(A4), A04202, https://doi.org/10.1029/2004JA010944 (2005).

[33]P. Janhunen, M. Palmroth, T. Laitinen, I. Honkonen, L. Juusola, G. Facskó, and T. Pulkkinen, "The GUMICS-4 global MHD magnetosphere–ionosphere coupling simulation," J. Atmos. Sol.-Terr. Phys. **80**, 48–59 (2012).

[34]V. Angelopoulos, "The Artemis mission," Space Sci. Rev. **165**, 3–25 (2011).

[35]V. Angelopoulos, "The Themis mission," Space Sci. Rev. **141**, 5–34 (2008).

[36]Y. Pfau-Kempf, M. Battarbee, U. Ganse, S. Hoilijoki, L. Turc, S. von Alfthan, R. Vainio, and M. Palmroth, "On the importance of spatial and velocity resolution in the hybrid-Vlasov modeling of collisionless shocks," Front. Phys. **6**, 44 (2018).

[37]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," J. Atmos. Sol.-Terr. Phys. **120**, 24–35 (2014).

[38]J. Lyon, J. Fedder, and C. Mobarry, "The Lyon–Fedder–Mobarry (LFM) global MHD magnetospheric simulation code," J. Atmos. Sol.-Terr. Phys. **66**, 1333–1350 (2004).

[39]T. I. Gombosi, D. L. Dezeeuw, C. P. T. Groth, K. G. Powell, C. R. Clauer, and P. Song, "From sun to earth: Multiscale MHD simulations of space weather," in Space Weather (American Geophysical Union (AGU), 2001), pp. 169–176.

[40]I. Honkonen, S. von Alfthan, A. Sandroos, P. Janhunen, and M. Palmroth, "Parallel grid library for rapid and flexible simulation development," Comput. Phys. Commun. **184**, 1297–1309 (2013).

[41]MPI Forum. "MPI: A message-passing interface standard—Version 3.1," (HLRS, Stuttgart, 2015).

[42]M. Berger and P. Colella, "Local adaptive mesh refinement for shock hydrodynamics," J. Comput. Phys. **82**, 64–84 (1989).

[43]B. T. Gunney and R. W. Anderson, "Advances in patch-based adaptive mesh refinement scalability," J. Parallel Distrib. Comput. **89**, 65–84 (2016).

[44]J.-H. Shue, J. K. Chao, H. C. Fu, C. T. Russell, P. Song, K. K. Khurana, and H. J. Singer, "A new functional form to study the solar wind control of the magnetopause size and shape," J. Geophys. Res.: Space Phys. **102**, 9497–9511, https://doi.org/10.1029/97JA00196 (1997).

[45]P. Londrillo and L. Del Zanna, "On the divergence-free condition in godunov-type schemes for ideal magnetohydrodynamics: The upwind constrained transport method," J. Comput. Phys. **195**, 17–48 (2004).

[46]M. Palmroth and the Vlasiator Team (2021). "FsGRID—a lightweight, static, Cartesian grid for field solvers," Github. https://github.com/fmihpc/fsgrid

[47]E. G. Boman, U. V. Catalyurek, C. Chevalier, and K. D. Devine, "The Zoltan and Isorropia parallel toolkits for combinatorial scientific computing: Partitioning, ordering, and coloring," Sci. Program. **20**, 713587 (2012).

[48]K. Papadakis, Y. Pfau-Kempf, U. Ganse, M. Battarbee, M. Alho, M. Grandin, M. Dubart, L. Turc, H. Zhou, K. Horaites, I. Zaitsev, G. Cozzani, M. Bussov, E. Gordeev, F. Tesema, H. George, J. Suni, V. Tarvus, and M. Palmroth, "Spatial filtering in a 6D hybrid-Vlasov scheme for alleviating AMR artifacts: A case study with vlasiator, versions 5.0, 5.1, 5.2.1," Geosci. Instrum., Methods Data Syst. **2022**, 7903–7912.

[49]A. A. Vlasov, Many-Particle Theory and Its Application to Plasma (Gordon & Breach Science Publishers Ltd., 1961).

[50]M. Zerroukat and T. Allen, "A three-dimensional monotone and conservative semi-Lagrangian scheme (SLICE-3D) for transport problems," Quart. J. Roy. Meteorol. Soc. **138**, 1640–1651 (2012).

[51]G. Strang, "On the construction and comparison of difference schemes," SIAM J. Numer. Anal. **5**, 506–517 (1968).

[52]L. Einkemmer and A. Ostermann, "Convergence analysis of Strang splitting for Vlasov-type equations," SIAM J. Numer. Anal. **52**, 140–155 (2014).

[53]A. W. Paeth, "A fast algorithm for general raster rotation," in Proceedings of Graphics Interface and Vision Interface '86, GI '86 (Canadian Man-Computer Communications Society, Toronto, ON, Canada, 1986), pp. 77–81.

[54]B. Chen and A. E. Kaufman, "3D volume rotation using shear transformations," Graphical Models **62**, 308–322 (2000).

[55]L. White and A. Adcroft, "A high-order finite volume remapping scheme for nonuniform grids: The piecewise quartic method (PQM)," J. Comput. Phys. **227**, 7394–7422 (2008).

[56]H. Zhou, L. Turc, Y. Pfau-Kempf, M. Battarbee, V. Tarvus, M. Dubart, M. Grandin, U. Ganse, M. Alho, A. Johlander, H. George, J. Suni, M. Bussov, K. Papadakis, K. Horaites, I. Zaitsev, G. Cozzani, F. Tesema, E. Gordeev, and M. Palmroth, "Magnetospheric responses to solar wind Pc5density fluctuations: Results from 2D hybrid Vlasov simulation," Front. Astron. Space Sci. **9**, 984918 (2022).

[57]F. De Hoffmann and E. Teller, "Magneto-hydrodynamic shocks," Phys. Rev. **80**, 692 (1950).

[58]V. Krasnoselskikh, M. Balikhin, S. N. Walker, S. Schwartz, D. Sundkvist, V. Lobzin, M. Gedalin, S. D. Bale, F. Mozer, J. Soucek, Y. Hobara, and H. Comisel, "The dynamic quasiperpendicular shock: Cluster discoveries," in *Microphysics of Cosmic Plasmas* (Springer US, Boston, MA, 2014), pp. 459–522.

[59]U. Ganse (2022). "Shock-simulation settings, input- and performance files," iDA Fairdata Repository. https://doi.org/10.23729/75f137f5-ce43-4d8b-8787-33b27eeb79b8

[60]D. Terpstra, H. Jagode, H. You, and J. Dongarra, "Collecting performance data with Papi-c," in *Tools for High Performance Computing 2009*, edited by M. S. Müller, M. M. Resch, A. Schulz, and W. E. Nagel (Springer Berlin Heidelberg, Berlin, Heidelberg, 2010), pp. 157–173.

[61]M. Dubart, M. Battarbee, U. Ganse, A. Osmane, F. Spanier, J. Suni, A. Johlander, M. Alho, M. Bussov, G. Cozzani, H. George, M. Grandin, K. Horaites, K. Papadakis, Y. Pfau-Kempf, V. Tarvus, L. Turc, I. Zaitsev, H. Zhou, and M. Palmroth, "Sub-grid modeling of pitch-angle diffusion for ion-scale waves in hybrid-Vlasov simulations with Cartesian velocity space," Phys. Plasmas **29**, 103902 (2022).

[62]N. A. Case and J. A. Wild, "The location of the earth's magnetopause: A comparison of modeled position and in situ cluster data," J. Geophys. Res.: Space Phys. **118**, 6127–6135, https://doi.org/10.1002/jgra.50572 (2013).

[63]G. Tóth, X. Jia, S. Markidis, I. B. Peng, Y. Chen, L. K. S. Daldorff, V. M. Tenishev, D. Borovikov, J. D. Haiducek, T. I. Gombosi, A. Glocer, and J. C. Dorelli, "Extended magnetohydrodynamics with embedded particle-in-cell simulation of Ganymede's magnetosphere," J. Geophys. Res.: Space Phys. **121**, 1273–1293, https://doi.org/10.1002/2015JA021997 (2016).

[64]K. A. Sorathia, V. G. Merkin, E. V. Panov, B. Zhang, J. G. Lyon, J. Garretson, A. Y. Ukhorskiy, S. Ohtani, M. Sitnov, and M. Wiltberger, "Ballooning-interchange instability in the near-earth plasma sheet and auroral beads: Global magnetospheric modeling at the limit of the MHD approximation," Geophys. Res. Lett. **47**(14), e2020GL088227, https://doi.org/10.1029/2020gl088227 (2020).

[65]Y. A. Omelchenko and H. Karimabadi, "HYPERS: A unidimensional asynchronous framework for multiscale hybrid simulations," J. Comput. Phys. **231**, 1766–1780 (2012).

[66]M. Palmroth, and the Vlasiator Team (2020). "Vlasiator: Hybrid-Vlasov simulation code," Github. https://github.com/fmihpc/vlasiator

[67]H. Childs, E. Brugger, B. Whitlock, J. Meredith, S. Ahern, "D. Pugmire, K. Biagas, M. Miller, C. Harrison, G. H. Weber, H. Krishnan, T. Fogal, A. Sanderson, C. Garth, E. W. Bethel, D. Camp, O. Rübel, M. Durant, J. M. Favre, and P. Navrátil, Visit: An end-user tool for visualizing and analyzing very large data," in *High Performance Visualization–Enabling Extreme-Scale Scientific Insight* (Chapman and Hall/CRC, 2012), pp. 357–372.