T. Pollinger, J. Rentrop, D. Pflüger, K. Kormann

A mass-conserving sparse grid combination technique with biorthogonal hierarchical basis functions for kinetic simulations

A mass-conserving sparse grid combination technique with biorthogonal hierarchical basis functions for kinetic simulations

Theresa Pollinger, Johannes Rentrop, Dirk Pflüger, Katharina Kormann

September 23, 2022

Abstract

The exact numerical simulation of plasma turbulence is one of the assets and challenges in fusion research. For grid-based solvers, sufficiently fine resolutions are often unattainable due to the curse of dimensionality. The sparse grid combination technique provides the means to alleviate the curse of dimensionality for kinetic simulations. However, the hierarchical representation for the combination step with the state-of-the-art hat functions suffers from poor conservation properties and numerical instability.

The present work introduces two new variants of hierarchical multiscale basis functions for use with the combination technique: the biorthogonal and full weighting bases. The new basis functions conserve the total mass and are shown to significantly increase accuracy for a finite-volume solution of constant advection. Further numerical experiments based on the combination technique applied to a semi-Lagrangian Vlasov–Poisson solver show a stabilizing effect of the new bases on the simulations.

1 The curse of dimensionality in kinetic simulations

The simulation of the time evolution of plasma in magnetic confinement fusion devices is an important part of plasma fusion research, and as such, serves both as an asset and a challenge. While a magnetohydrodynamic (MHD) description of the plasma can capture many properties, important phenomena—as for instance the Landau damping phenomenon—are not present in the MHD model. In this case, the more comprehensive kinetic description in phase-space is necessary. Such a description is computationally much more demanding due to the phase-space being six dimensional. Under certain conditions, the gyrokinetic and the drift-kinetic models—that reduce the dimensionality to five or four—provide an accurate description with a reduced dimensionality in velocity space. In addition to the high dimensionality, a further challenge stems from the fact that important plasma features can occur on small scales, in a way that they only start to appear at fine resolutions in the simulation. The so-called curse of dimensionality—the exponential increase in the number of degrees of freedom (DOF) when increasing the dimensionality—haunts any approach that fits a regular grid into the phase space for direct simulation. Taking these two properties (high dimensionalities and small scales) together, it becomes obvious that simulations that are resolved finely enough to have predictive quality can be too large in terms of compute time and/or memory footprint even for today's largest compute systems.

One approach to alleviate the curse of dimensionality is to use the sparse grid combination technique. It combines differently resolved numerical solutions multiple times throughout the course of the simulation. The combination technique allows to re-use existing solvers in a black-box fashion and adds an extra level of parallelism to the parallelism that may already be present in the solver [27]. For instance, previous work applying the gyrokinetic solver GENE [21] has shown that the eigenvalues in the linear part of the simulation can be obtained by a combination technique solution [25]. However, current approaches usually did not include the conservation of invariants—such as mass—and even had to deal with a more general problem: When simulating up to the

1

nonlinear quasi-stationary turbulent phase of an ITG instability scenario, the GENE combination technique might become numerically unstable [27, p. 319]. The effect was more prevalent when using shorter recombination intervals, a greater number of grids, or higher resolutions [29, section 3.6.4]. This presented a severe problem for running large-scale combination technique plasma instability simulations at physically meaningful resolutions.

This work now employs two solvers that are easier to analyze numerically than GENE because they preserve known invariants: a finite volume solver and a semi-Lagrangian Vlasov solver implemented in the SeLaLib library [23] that both conserve mass up to machine precision. With the finite volume solver we consider a simple advection equation where the solution is known analytically, which provides a verifiable baseline. For the semi-Lagrangian solver we then study two benchmark problems from plasma physics: the Landau damping phenomenon and the two-stream plasma instability.

We introduce two variants of mass-conserving hierarchical multiscale bases that turn the combination step into a mass-conserving operation and establish the connection to sparse grid and combination technique theory. Both types of basis function are rooted in the theory of biorthogonal [6] and lifting [38] wavelets. The exact hierarchization and dehierarchization steps are illustrated, and we draw parallels to related work that solves the Vlasov equation directly in the hierarchical sparse grid space [17; 2; 8; 9; 22; 16; 39] or using the combination technique [26; 27]. The mass-conserving basis functions are shown to reduce the error between combination technique and full grid solution in the advection scenario. When using an equal number of degrees of freedom, the mass-conserving combination technique delivers results that are comparable in accuracy to a full grid solution in the cases of Landau damping and two-stream instability. Furthermore, the most important finding is that the mass-conserving bases provide stability in combination technique scenarios where the current state of the art, based on interpolets, becomes numerically unstable. This fact may permit future exa-scale plasma simulations to draw from all benefits of the sparse grid combination technique.

1.1   Related work

There is a wide field of closely and loosely related work touching on the main topic of this paper: using sparse-grid techniques to efficiently compute solutions for kinetic problems. A necessary distinction is whether the numerical scheme (differential operators, integral operators, time-stepping) is formulated in terms of the multiscale basis functions, or if the combination technique—also known as Smolyak sparse grids—is used, which allows to have the solver operate with an arbitrary basis representation.

The works by Haefele et al. [17] and Besse et al. [2] fall into the first category. These works are particularly related to our work, since they use a semi-Lagrangian scheme for the solution of the Vlasov–Poisson system, employing interpolet—a.k.a. hat—basis functions. A similar approach is taken by Deriaz and Peirani [8], using interpolets with finite difference schemes. Earlier work by Deriaz and Perrier [9] focuses on divergence- and curl-free—i.e., conservative—schemes with wavelet constructions for flow simulations. While neither of the mentioned lines of work explicitly uses a sparse grid construction, they employ adaptive hierarchical schemes, which will result in basis constructions that are similar to spatially adaptive hierarchical sparse grid bases [32]. Kormann and Sonnendrücker [22] implemented hierarchical basis operators for the Vlasov–Poisson equations into the SeLaLib [23] code, using the classical hat functions and higher order polynomials. The work introduces a multiplicative δf ansatz to cope with the problem of approximating Gaussians with the sparse grid technique. More recently, a line of work by Guo and Cheng [16; 39] successfully used an adaptive multiwavelet basis for Discontinuous Galerkin simulations of the Vlasov–Poisson and Vlasov–Maxwell systems. Notably, the multiwavelet basis also conserves momenta up to the chosen order of the DG basis. Griebel and Koster [15; 24] employed spatially adaptive sparse grids with biorthogonal basis constructions—like ψ^{bo} introduced in Section 2.2.2—for the simulation of turbulent flows.

The second category, combination technique based solvers, employs the hierarchical basis only between the solver steps to exchange the information between the component grids. For flow

simulations, Huber [20] formulated custom hierarchization and dehierarchization operators to make the simulation stable. These operators were problem and domain specific, taking into account the physical properties to be conserved and the geometry of the domain in the simulation.

Previous work for plasma simulations could extrapolate eigenvalues in the linear part of the simulation by way of the combination technique [25]. The combination technique—with some adaptation—also performed well for linearized gyrokinetic simulations [26]. For turbulent plasma simulations however, as discussed before, the combination technique applied to GENE would often become numerically unstable [27, p. 319].

Our new results are based on the semi-Lagrangian approach to the Vlasov–Poisson system; in contrast to e. g., Haefele et al. [17] and Deriaz and Peirani [8], this work solves the systems of equations not directly in the hierarchical basis, but employs the combination technique to couple solvers for structured grids. It introduces biorthogonal linear wavelets as intermediate representations in the combination technique, since the interpolet basis is not mass-conserving and can lead to the simulation becoming unstable. The results presented here may allow to revisit some of the aforementioned experiments with relatively little overhead at improved stability and accuracy.

To the best of our knowledge, so far no attempts have been made at higher-dimensional combination technique time-dependent simulations with bases other than hierarchical hat functions.

2 Wavelets, hierarchical basis and basis transformations

To introduce the necessary concepts used in this paper, we give a short overview on the theory of multiresolution analysis and the wavelet transform [5]. In this context, we will introduce biorthogonal wavelet bases [6], and, as a special case, the hierarchical hat function basis. We consider an unbounded domain in this section and discuss boundary conditions for compact domains in section 2.4.

2.1 Multiresolution analysis and biorthogonal wavelets

A multiresolution analysis on $L^2(\mathbb{R})$ is a hierarchy of function spaces

$$\ldots V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \cdots \subset L^2(\mathbb{R}) \tag{1}$$

with the following properties:

- The union of all function spaces is dense in $L^2(\mathbb{R})$ and there is no redundancy:

$$\overline{\cup_{l \in \mathbb{Z}} V_l} = L^2(\mathbb{R}) , \quad \cap_{l \in \mathbb{Z}} V_l = \{0\} . \tag{2}$$

- Each space is spanned by dilates/translates of a *scaling function* ϕ (it is possible to have a set of scaling functions, but we focus on the case with only one scaling function):

$$\phi_{ls}(x) := \phi(2^l x - s) , \quad V_l = \mathrm{span}\{\phi_{ls}\}_{s \in \mathbb{Z}} . \tag{3}$$

One can think of the series of spaces as successive approximation spaces for a function $f \in L^2(\mathbb{R})$. Each projection $\mathcal{P}_l f \in V_l$ is an approximation with resolution 2^{-l}. We call l the *level* of the approximation. The level $l = 0$ corresponds to the coarsest possible level on the unit interval, which will be discussed later.

Since $\phi \in V_0 \subset V_1 = \mathrm{span}\{\phi_{1s}\}_{s \in \mathbb{Z}}$ the scaling function has to satisfy the multiscale equation

$$\phi(x) = \sum_{s \in \mathbb{Z}} h_s \phi(2x - s) \tag{4}$$

with coefficients $h_s \in \mathbb{R}$, which explains its name.

3

Let us consider the complement space of V_{l-1} in V_l, also known as the *hierarchical increment space* W_l

$$V_l = V_{l-1} \oplus W_l , \qquad V_{l-1} \cap W_l = \{0\} . \tag{5}$$

Applying this relation iteratively leads to the decomposition (setting $W_0 := V_0$)

$$V_l = \oplus_{l'=0}^{l} W_{l'} , \qquad L^2(\mathbb{R}) = \overline{\oplus_{l'=0}^{\infty} W_{l'}} . \tag{6}$$

We aim to define basis functions ψ_{ls} for W_l, so that $W_l = \mathrm{span}\{\psi_{ls}\}_{s \in \mathbb{Z}}$. These functions will be called *wavelets*. Then, every function $f \in L^2(\mathbb{R})$ can be represented as

$$f = \sum_{l=0}^{\infty} \sum_{s \in \mathbb{Z}} \alpha_{ls} \psi_{ls} , \tag{7}$$

where we set $\psi_{0s} := \phi_{0s}$. We call α_{ls} the *hierarchical* or *wavelet coefficients*.

However, the space W_l is not uniquely defined, it needs further specification. Requiring it to be the orthogonal complement, for example, leads to *orthonormal wavelets* [7]. Here, instead, we only demand that the ψ_{ls} are dilates/translates of a *mother wavelet* ψ, i.e. $\psi_{ls}(x) := \psi(2^{l-1}x - s)$, in particular $\psi_{10} = \psi$. Since $W_1 \subset V_1$, there are—similar to Equation (4)—coefficients $g_s \in \mathbb{R}$, so that

$$\psi(x) = \sum_{s \in \mathbb{Z}} g_s \phi(2x - s) . \tag{8}$$

In order to uniquely define the mother wavelet ψ, we employ *biorthogonal wavelets*. For this purpose, we require a *dual* multiresolution analysis with a dual scaling function $\tilde{\phi}$, a dual mother wavelet $\tilde{\psi}$ and coefficients $\tilde{h}_s, \tilde{g}_s$, so that

$$\tilde{\phi}(x) = 2 \sum_{s \in \mathbb{Z}} \tilde{h}_s \tilde{\phi}(2x - s) , \qquad \tilde{\psi}(x) = \sum_{s \in \mathbb{Z}} \tilde{g}_s \tilde{\phi}(2x - s) . \tag{9}$$

Theorem 3.8 in [6] states that, if (h_s) and $(\tilde{h}_s)$ are finite sequences with $\sum_s h_s \tilde{h}_{s-2k} = \delta_{k0}$, if the Fourier transforms of the corresponding ϕ and $\tilde{\phi}$ are bounded and in $L^2(\mathbb{R})$, and if we define

$$g_s := (-1)^{1-s} \tilde{h}_{1-s} , \qquad \tilde{g}_s := (-1)^{1-s} h_{1-s} , \tag{10}$$

then the $\psi_{ls}, \tilde{\psi}_{ls}$ constitute two dual frames, so that for any $f \in L^2(\mathbb{R})$,

$$f = \sum_{l=0}^{\infty} \sum_{s \in \mathbb{Z}} 2^l \langle \tilde{\psi}_{ls}, f \rangle \psi_{ls} = \sum_{l=0}^{\infty} \sum_{s \in \mathbb{Z}} 2^l \langle \psi_{ls}, f \rangle \tilde{\psi}_{ls} . \tag{11}$$

Furthermore, if and only if $\int \phi(x) \tilde{\phi}(x - s) \mathrm{d}x = \delta_{s0}$, the $\psi_{ls}, \tilde{\psi}_{ls}$ are two dual Riesz bases and we have the following biorthogonality relations

$$\begin{aligned} \langle \phi_{ls}, \tilde{\phi}_{ls'} \rangle &= 2^{-l} \delta_{ss'} , & \langle \psi_{ls}, \tilde{\psi}_{l's'} \rangle &= 2^{-l} \delta_{ll'} \delta_{ss'} , \\ \langle \phi_{(l-1)s}, \tilde{\psi}_{ls'} \rangle &= 0 , & \langle \psi_{ls}, \tilde{\phi}_{(l-1)s'} \rangle &= 0 , \end{aligned} \tag{12}$$

where $\langle \cdot, \cdot \rangle$ is the L^2 scalar product. Note that we use different normalization factors compared to the original paper.

In practice, we are interested in calculating the *wavelet transform* or *hierarchization* of a function whose approximation in V_l is given by $f_l := \mathcal{P}_l f = \sum_{s \in \mathbb{Z}} c_{ls} \phi_{ls}$. This amounts to computing $\alpha_{l's} = 2^{l'} \langle \tilde{\psi}_{l's}, f_l \rangle$ for $0 \le l' \le l$, $s \in \mathbb{Z}$. One starts by choosing the two sequences (h_s) and $(\tilde{h}_s)$ in such a way that they meet the above mentioned conditions. The scaling functions and wavelets do not have to be known explicitly. From an algorithmic point of view, it suffices to know (h_s), $(\tilde{h}_s)$ and, by consequence, (g_s), $(\tilde{g}_s)$ in order to perform the well-known wavelet transform:

Due to Equation (5) we can expand f_l as

$$f_l = \sum_{s \in \mathbb{Z}} c_{ls} \phi_{ls} = \sum_{s \in \mathbb{Z}} c_{(l-1)s} \phi_{(l-1)s} + \sum_{s \in \mathbb{Z}} \alpha_{ls} \psi_{ls} \; . \tag{13}$$

Taking the scalar product with $2^l \tilde{\psi}_{lk}$ and using Equation (9) and Equation (12) we obtain

$$\alpha_{lk} = 2^l \langle \tilde{\psi}_{lk}, f_l \rangle = 2^l \sum_{s \in \mathbb{Z}} c_{ls} \langle \tilde{\psi}_{lk}, \phi_{ls} \rangle = \sum_{s \in \mathbb{Z}} \sum_{s' \in \mathbb{Z}} c_{ls} \tilde{g}_{s'} 2^l \langle \tilde{\phi}_{l(s'+2k)}, \phi_{ls} \rangle = \sum_{s \in \mathbb{Z}} \sum_{s' \in \mathbb{Z}} c_{ls} \tilde{g}_{s'} \delta_{(s'+2k)s}$$

$$= \sum_{s \in \mathbb{Z}} c_{ls} \tilde{g}_{(s-2k)} \; . \tag{14}$$

Analogously, taking the scalar product with $2^{l-1} \tilde{\phi}_{(l-1)k}$, we get

$$c_{(l-1)k} = 2^{l-1} \langle \tilde{\phi}_{(l-1)k}, f_l \rangle = 2^{l-1} \sum_{s \in \mathbb{Z}} c_{ls} \langle \tilde{\phi}_{(l-1)k}, \phi_{ls} \rangle = \sum_{s \in \mathbb{Z}} \sum_{s' \in \mathbb{Z}} c_{ls} \tilde{h}_{s'} 2^l \langle \tilde{\phi}_{l(s'+2k)}, \phi_{ls} \rangle$$

$$= \sum_{s \in \mathbb{Z}} c_{ls} \tilde{h}_{(s-2k)} \; . \tag{15}$$

Replacing l by $l - 1$, these two steps can be applied iteratively down to $l' = 0$. Alternatively, the transform could just be carried through to some other level $l^{\min} \geq 0$ in order to retain a hybrid representation of f_l. This is a beneficial property when discussing the sparse grid combination technique.

Equations (14) and (15) have the structure of a convolution with the *filter mask* $(\tilde{g}_s)$ resp. $(\tilde{h}_s)$, which is displaced with a stride of two. $(\tilde{h}_s)$ covers the even indices, while $(\tilde{g}_s)$ covers the odd ones (due to the shift of one in Definition (10)).

The inverse wavelet transform or *dehierarchization* is obtained by taking the scalar product of (13) with $2^l \tilde{\phi}_{lk}$,

$$c_{lk} = 2^l \langle \tilde{\phi}_{lk}, f_l \rangle = \sum_{s \in \mathbb{Z}} c_{(l-1)s} 2^l \langle \tilde{\phi}_{lk}, \phi_{(l-1)s} \rangle + \sum_{s \in \mathbb{Z}} \alpha_{ls} 2^l \langle \tilde{\phi}_{lk}, \psi_{ls} \rangle$$

$$= \sum_{s \in \mathbb{Z}} c_{(l-1)s} h_{k-2s} + \sum_{s \in \mathbb{Z}} \alpha_{ls} g_{k-2s} \; , \tag{16}$$

which, again, can be computed iteratively from level zero or some $l^{\min}$ up to l. Due to Equations (14) to (16), $(\tilde{h}_s)$ and $(\tilde{g}_s)$ are called *decomposition filters* while (h_s) and (g_s) are called *reconstruction filters*.

2.2 Relevant bases for this paper

The following three types of hierarchical basis functions will be used throughout this paper. They all employ the classical hat function, either as scaling function or dual scaling function,

$$\phi(x) = \max(1 - |x|, 0) \; . \tag{17}$$

This function has compact support, which leads to finite filter lengths, and it is interpolating, thus $\phi(s) = \delta_{s0}$ for $s \in \mathbb{Z}$. The latter implies that the coefficients for the representation of $f_l \in V_l$ in terms of the ϕ_{ls} are just the function values at the grid points, i.e. $c_{ls} = f(2^{-l}s)$. Since the hat function is piecewise linear, f_l will be a linear spline approximating f in V_l. It is easy to verify from the multiscale equation (4) that the corresponding filter coefficients are given by

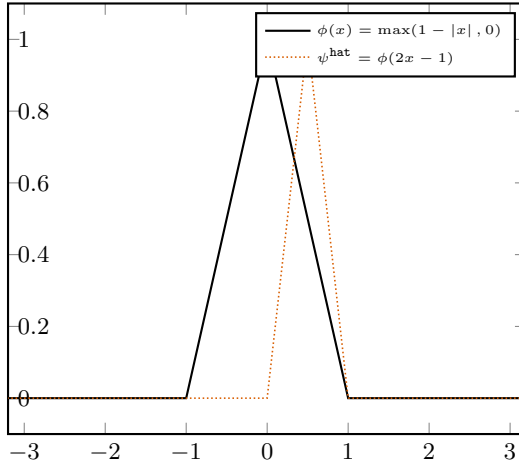$$(h_s)_{-1 \leq s \leq 1} = (\tfrac{1}{2}, 1, \tfrac{1}{2}) \; . \tag{18}$$

5

Figure 1: "Hat" function: scaling function for piecewise linear multi-resolution analysis, and standard choice for hierarchical basis $\psi^{\mathtt{hat}}$.

2.2.1 Hierarchical hat basis

The hierarchical hat basis is the standard basis used in most of the sparse grid literature. This is why we use it in this paper as a point of reference. Outside of the sparse grid literature, it is often framed as the interpolet basis of first order [10]. Here, the dual scaling function is the Dirac distribution $\tilde{\phi}(x) = \delta(x)$ [38]. In this case $\tilde{\phi} \notin L^2(\mathbb{R})$ and all scalar products have to be understood as a dual pairing. Also, this violates the requirements of Theorem 3.8 in [6] and one can show that the hierarchical hat basis is indeed not a Riesz basis.

The Dirac distribution satisfies the functional equation $\delta(x) = 2\delta(2x)$, which immediately serves as its multiscale equation. Hence, the corresponding filters have only one non-vanishing entry, $\tilde{h}_s = \delta_{s0}$. It follows from Equation (10) that $g_s = \delta_{s1}$. Therefore, the wavelets themselves are also hat functions that live at odd grid points, and we will refer to them as $\psi^{\mathtt{hat}}$ (cf. Figure 1). In addition, the second step of the decomposition, Equation (15), becomes redundant. The c_{ls} stay the same throughout the levels. This makes the hierarchical basis transform especially efficient ("lazy wavelets", cf. [38]). In summary, the full set of filters is given by

$$
\begin{aligned}
(h_s)^{\mathtt{hat}}_{-1 \leq s \leq 1} &= \left(\tfrac{1}{2}, 1, \tfrac{1}{2}\right) , \quad (g_s)^{\mathtt{hat}}_{0 \leq s \leq 2} = (0, 1, 0) , \\
(\tilde{h}_s)^{\mathtt{hat}}_{-1 \leq s \leq 1} &= (0, 1, 0) , \quad (\tilde{g}_s)^{\mathtt{hat}}_{0 \leq s \leq 2} = \left(-\tfrac{1}{2}, 1, -\tfrac{1}{2}\right) .
\end{aligned}
\tag{19}
$$

2.2.2 2,2-biorthogonal bases

The simplest biorthogonal wavelet basis that uses the hat function as scaling function, is the example with $(N, \tilde{N}) = (2, 2)$ in chapter 6.A of [6]. We will denote it by $\psi^{\mathtt{bo}}$. The full set of filters is given by

$$
\begin{aligned}
(h_s)^{\mathtt{bo}}_{-1 \leq s \leq 1} &= \left(\tfrac{1}{2}, 1, \tfrac{1}{2}\right) , \quad (g_s)^{\mathtt{bo}}_{-1 \leq s \leq 3} = \left(-\tfrac{1}{8}, -\tfrac{1}{4}, \tfrac{3}{4}, -\tfrac{1}{4}, -\tfrac{1}{8}\right) , \\
(\tilde{h}_s)^{\mathtt{bo}}_{-2 \leq s \leq 2} &= \left(-\tfrac{1}{8}, \tfrac{1}{4}, \tfrac{3}{4}, \tfrac{1}{4}, -\tfrac{1}{8}\right) , \quad (\tilde{g}_s)^{\mathtt{bo}}_{0 \leq s \leq 2} = \left(-\tfrac{1}{2}, 1, -\tfrac{1}{2}\right) .
\end{aligned}
\tag{20}
$$

The third type of basis we use in this paper is almost identical to the last example. Only the roles of ϕ and $\tilde{\phi}$ are interchanged. This means that the decomposition filters become the analysis filters and vice versa:

$$
\begin{aligned}
(h_s)^{\mathtt{fw}}_{-2 \leq s \leq 2} &= \left(-\tfrac{1}{4}, \tfrac{1}{2}, \tfrac{3}{2}, \tfrac{1}{2}, -\tfrac{1}{4}\right) , \quad (g_s)^{\mathtt{fw}}_{0 \leq s \leq 2} = \left(-\tfrac{1}{2}, 1, -\tfrac{1}{2}\right) , \\
(\tilde{h}_s)^{\mathtt{fw}}_{-1 \leq s \leq 1} &= \left(\tfrac{1}{4}, \tfrac{1}{2}, \tfrac{1}{4}\right) , \quad (\tilde{g}_s)^{\mathtt{fw}}_{-1 \leq s \leq 3} = \left(-\tfrac{1}{8}, -\tfrac{1}{4}, \tfrac{3}{4}, -\tfrac{1}{4}, -\tfrac{1}{8}\right) .
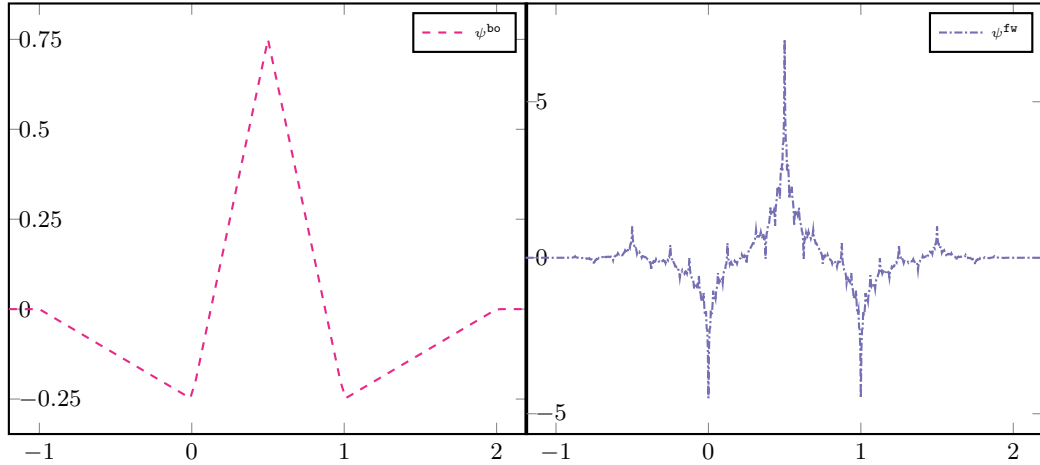\end{aligned}
\tag{21}
$$

Figure 2: The two choices of mass-conserving wavelet functions: $\psi^{\mathtt{bo}}$ to the left and $\psi^{\mathtt{fw}}$ to the right.

We call this version the *full weighting* basis, $\psi^{\mathtt{fw}}$, because the decomposition filter $(\tilde{h}_s)$ in this case is known from the full weighting restriction in the multigrid literature.

The shapes of $\psi^{\mathtt{bo}}$ and $\psi^{\mathtt{fw}}$ can be seen in Figure 2. $\psi^{\mathtt{bo}}$ is a linear B-spline wavelet just like $\psi^{\mathtt{hat}}$, while $\psi^{\mathtt{fw}}$ is the wavelet that serves as the dual for $\psi^{\mathtt{bo}}$. Although the filters in the biorthogonal case are wider than for $\psi^{\mathtt{hat}}$, they can be computed in-place with two data sweeps per level instead of one, where the intermediate filters are always three elements wide. This is due to their possible decomposition as lifting wavelets, as discussed in [38]. Figure 3 displays the hierarchization and dehierarchization operations resulting from the bases introduced in this section, and highlights the decomposition and reconstruction filters where applicable.

2.3 Conservation of mass

For our application, the motivating advantage of the biorthogonal bases over the hierarchical basis is the fact that they have a vanishing integral or zeroth moment. For the two cases presented in this work, this can easily be shown by Equation (8), together with the fact that in both cases ϕ is integrable (as proved in [6]) and $\sum_{s\in\mathbb{Z}} g_s = 0$. This implies for the hybrid representation

$$f_l = \sum_{s\in\mathbb{Z}} c_{l^{\min}s}\phi_{l^{\min}s} + \sum_{l^{\min}<l'\leq l}\sum_{s\in\mathbb{Z}} \alpha_{l's}\psi_{l's} \tag{22}$$

that the zeroth moment is fully concentrated in the minimal level and, consequently, that it remains the same throughout all approximations, independent of the level.

In a physical sense, this signifies the conservation of mass throughout the levels of approximation. Since the conservation of mass is a key property of the type of PDEs considered in this paper, this can potentially play an important role when applying the sparse grid combination technique, cf. Section 3.2. The reason is that the latter is based on an extrapolation using approximations on different levels. Conservation of mass throughout the levels results in a conserved mass for the extrapolant.

Conversely, this is not the case when using the hierarchical hat basis. In this scenario, the wavelets themselves are hat functions with a non-vanishing integral, so the contributions to the integral are spread among the levels. As an extreme example, consider a function whose mass is distributed very granularly, so that all $c_{l^{\min}s}$ and α_{ls} for $l < L$ are zero. All approximations with $l < L$ will carry no mass and the extrapolant will effectively consist of only one partial approximation at level L.

7
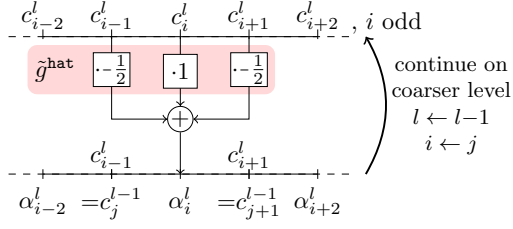
(a) Hierarchical hat basis: For the hierarchization, new values on the coarser level are identical to the previous ones.
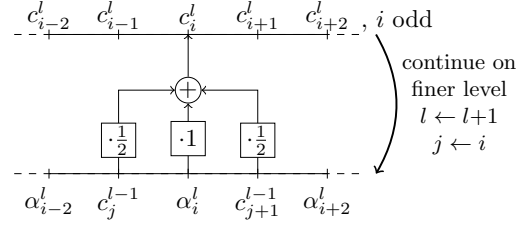
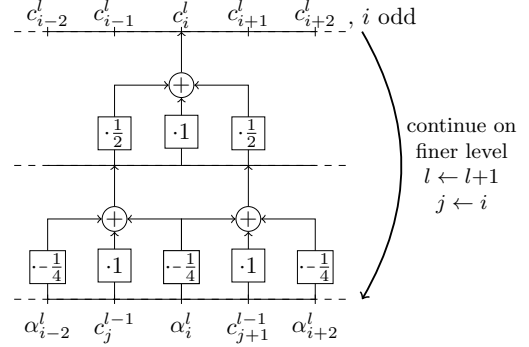(b) Hierarchical hat basis: Coarser values are kept, finer values are obtained by the sum of interpolation and hierarchical surplus.
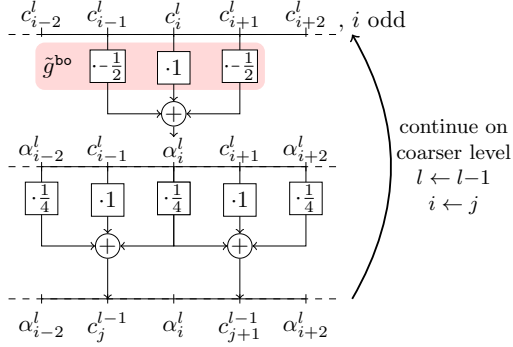
(c) Biorthogonal basis: Hierarchical coefficients are calculated as before, but then the scaling function coefficients are updated, too.

(d) Biorthogonal basis: The pattern of inverse signs for odd-numbered stencil positions compared to the hierarchization is visible.
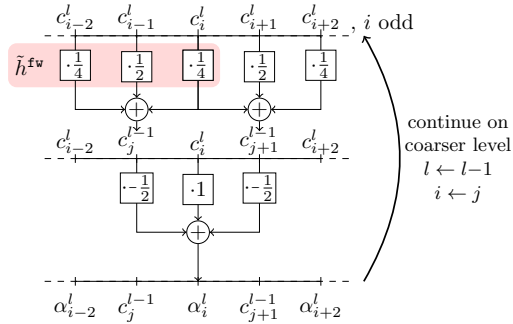
(e) Full weighting basis: The filters are virtually the same as in the biorthogonal case, but applied in reverse order.
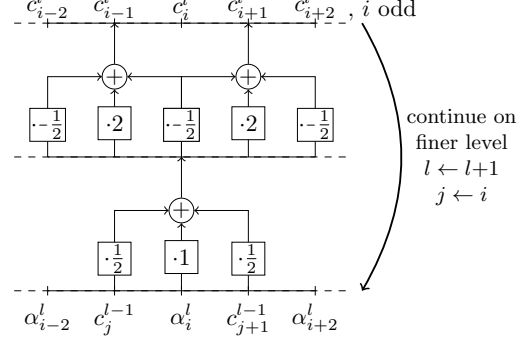
(f) Full weighting basis: One can observe the "lifting" property, allowing to greedily compute the transform in-place.

Figure 3: Filters for the hierarchization and dehierarchization operations for the different multiscale bases. The hierarchization (left column) is described by Equations (14) and (15); the dehierarchization (right column) is described by Equation (16). In Figures (c) to (f), filters from Equations (20) and (21) that are not explicitly depicted arise from the subsequent execution of the two lifting steps.

This leads to the hypothesis that the biorthogonal wavelet bases are better suited for the context of mass conservation than the hierarchical basis. Section 5 is going to validate this proposition by three numerical studies.

8

2.4 Wavelets on compact domains

In this work, only functions on compact domains will be considered, which can be scaled to the unit interval $[0, 1]$ or the hypercube $[0, 1]^d$ in d dimensions. Usually, one would need to make boundary adjustments for the basis functions close to the boundary in order to account for the desired regularity properties, when faced with reduced stencil sizes due to missing neighboring points outside the domain. For instance, a suitable boundary treatment for ψ^{bo} is derived in [24]. In all cases considered here, we will use periodic boundary conditions, which means that the boundaries can be treated like the boundaries at internal partitions of the solver. This removes the need for boundary adjusted basis functions.

3 Sparse grids and the combination technique

In this chapter we introduce sparse grids and, based on that, the sparse grid combination technique [3]. The concept is based on the hierarchical subspace splitting presented in the last chapter.

3.1 Sparse grids

Sparse grids can be seen as a compression technique for the representation of multidimensional functions. Consider a function $f : [0, 1]^d \to \mathbb{R}$. In order to discretize it, we choose a hybrid wavelet representation similar to Equation (22). We use tensor products of the one-dimensional approximation spaces to generalize them to multiple dimensions. For example, the two-dimensional version of the subspace decomposition in (5) reads as

$$
\begin{aligned}
V_{l_1, l_2} = V_{l_1} \otimes V_{l_2} &= (V_{l_1-1} \oplus W_{l_1}) \otimes (V_{l_2-1} \oplus W_{l_2}) \\
&= (V_{l_1-1} \otimes V_{l_2-1}) \oplus (V_{l_1-1} \otimes W_{l_2}) \oplus (V_{l_2-1} \otimes W_{l_1}) \oplus (W_{l_1} \otimes W_{l_2}) .
\end{aligned}
\tag{23}
$$

Again, this can iteratively be expanded into hierarchical increment spaces down to a minimal level. In order to account for the remaining cross terms between V and W, we define the multidimensional scaling resp. wavelet functions as

$$
\phi_{\boldsymbol{ls}}(\boldsymbol{x}) = \prod_{i=1}^{d} \phi_{l_i s_i}(x_i) , \qquad \psi_{\boldsymbol{ls}}(\boldsymbol{x}) = \prod_{i=1}^{d} \psi'_{l_i s_i}(x_i) , \qquad \psi'_{l_i s_i} = \begin{cases} \phi_{l_i s_i} & l_i = l_i^{\min} \\ \psi_{l_i s_i} & l_i > l_i^{\min} \end{cases} , \tag{24}
$$

where bold indices are multi-indices, e.g. $\boldsymbol{l} = (l_1, \dots, l_d)$. The multidimensional (de-)hierarchization operation reduces to a sequential execution of one-dimensional (de-)hierarchization operations.

The classical discretization on a *full* grid, i.e. a regular rectangular grid of level $\boldsymbol{l}$, is then given by

$$
f_{\boldsymbol{l}} = \sum_{\boldsymbol{s}} c_{\boldsymbol{l}^{\min} \boldsymbol{s}} \phi_{\boldsymbol{l}^{\min} \boldsymbol{s}} + \sum_{\boldsymbol{l}^{\min} \leq \boldsymbol{l}' \leq \boldsymbol{l}} \sum_{\boldsymbol{s}} \alpha_{\boldsymbol{l}' \boldsymbol{s}} \psi_{\boldsymbol{l}' \boldsymbol{s}} . \tag{25}
$$

Binary operators between multi-indices are understood component-wise. Also, whenever the first term is written out for clarity, it should implicitly be excluded from the sum in the second term. The number of basis functions and thus the number of grid points in each direction is given by $N_i := 2^{l_i} + 1$ (including both boundaries). For an isotropic discretization ($N_i = N$) the total number of grid points is then given by N^d. The exponential dependency on the number of dimensions d is often referred to as the curse of dimensionality.

In order to lift this curse, we introduce the so called *generalized sparse grid* approximation

$$
f_{\boldsymbol{l}}^{(\mathrm{s})} = \sum_{\boldsymbol{s}} c_{\boldsymbol{l}^{\min} \boldsymbol{s}} \phi_{\boldsymbol{l}^{\min} \boldsymbol{s}} + \sum_{\boldsymbol{l}' \in \mathcal{I}_{\boldsymbol{l}}} \sum_{\boldsymbol{s}} \alpha_{\boldsymbol{l}' \boldsymbol{s}} \psi_{\boldsymbol{l}' \boldsymbol{s}} . \tag{26}
$$

Here, the second sum does not run over all levels up to $\boldsymbol{l}$ but, instead, the levels are chosen from an index set $\mathcal{I}_{\boldsymbol{l}}$. This index set is usually designed to exclude many of the levels with high isotropic resolution compared to the full grid, while most of the anisotropic levels are kept. This reduces the

number of grid points drastically. The resulting *target level* $\boldsymbol{l}$ of the set is defined by the maximal occurring l'_i in each direction.

The specific design of the index set depends on the decay behavior of the wavelet coefficients, which is linked to the regularity properties of the considered function class. It is also possible to let the index set be chosen by an adaptive algorithm [11; 32]. The only requirement for the index set is to be downward closed, i.e. $\forall \boldsymbol{l'} \in \mathcal{I}_{\boldsymbol{l}} : \boldsymbol{l}^{\min} \leq \boldsymbol{j} \leq \boldsymbol{l'} \Rightarrow \boldsymbol{j} \in \mathcal{I}_{\boldsymbol{l}}$, which ensures that the (de-)hierarchization operations exist.

Let us, as an example, consider the classical sparse grid for an isotropic target resolution $\boldsymbol{l} = (n, \dots, n)$ for some $n \in \mathbb{N}$, and denote it by $f_n^{(\mathrm{s})}$. The corresponding level set is given by

$$\mathcal{I}_n = \left\{ \boldsymbol{l'} \in \mathbb{N}^d : |\boldsymbol{l'}|_1 \leq n \right\} \tag{27}$$

with $\boldsymbol{l}^{\min} = (0, \dots, 0)$ and the ℓ^1 norm $|\cdot|_1$. Geometrically, this level set is merely a simplex in $\mathbb{N}^d$ compared to the full hypercube that corresponds to Equation (25). This choice is optimal for the approximation in the L^2 norm of functions from Sobolev spaces with dominating mixed smoothness, $f \in H^2_{\mathrm{mix}}$. This Sobolev space consists of all functions whose mixed weak derivatives up to second order are bounded in L^2,

$$H^2_{\mathrm{mix}} := \left\{ f \in L^2 : \|f\|_{H^2_{\mathrm{mix}}} < \infty \right\} , \qquad \|f\|_{H^2_{\mathrm{mix}}} := \left(\sum_{|\boldsymbol{m}|_\infty \leq 2} \|D^{\boldsymbol{m}} f\|_{L^2}^2 \right)^{1/2} , \tag{28}$$

where $D^{\boldsymbol{m}} f := \frac{\partial^{|\boldsymbol{m}|_1}}{\partial_{x_1}^{m_1} \dots \partial_{x_d}^{m_d}}$ is the $\boldsymbol{m}$-th weak derivative.

For the case of the hierarchical basis, it can be shown [3] that the bound for the representation error is given by ($N := 2^n$)

$$\left\| f - f_n^{(\mathrm{s})} \right\|_{L^2} \lesssim N^{-2} (\log N)^{d-1} \|f\|_{H^2_{\mathrm{mix}}} , \tag{29}$$

which is only worse by a logarithmic factor compared to the usual $\mathcal{O}(N^{-2})$ for a piece-wise linear approximation on a full grid. In contrast, the number of points on the sparse grid behaves asymptotically like $\mathcal{O}(N(\log N)^{d-1})$. This is orders of magnitude less than in the full grid case, since the exponential dependency on the dimension is only present in the log-Term.

3.2 The combination technique

The combination technique (CT) is an alternative representation of a sparse grid function (26). The goal is to represent the sum over hierarchical increment spaces by a linear combination of full grid functions (25), which can be decomposed into the necessary hierarchical subspaces, i.e., all spaces in the index set $\mathcal{I}$. Since many subspaces on lower levels will then be counted multiple times, one has to account for it by subtracting suitable full grid functions containing those subspaces. This is similar to the inclusion exclusion principle from combinatorics. For example, for the classical sparse grid (27) in 2D this leads to the combination formula

$$f_n^{(\mathrm{s})} = \sum_{l_1 + l_2 = n} f_{l_1, l_2} - \sum_{l_1 + l_2 = n-1} f_{l_1, l_2} . \tag{30}$$

That is, all full grid functions with levels that lie on the bounding diagonal of the level set have a combination coefficient of $\lambda = +1$ while functions with levels on the diagonal below have a coefficient of $\lambda = -1$. For a general index set, this leads to the formula

$$f_{\boldsymbol{l}}^{(\mathrm{s})} = \sum_{\boldsymbol{l'} \in \mathcal{I}_{\boldsymbol{l}}} \lambda_{\boldsymbol{l'}} f_{\boldsymbol{l'}} , \qquad \text{with} \quad \lambda_{\boldsymbol{l'}} = \sum_{\boldsymbol{z} = \boldsymbol{0}}^{\boldsymbol{1}} (-1)^{|\boldsymbol{z}|_1} \chi^{\mathcal{I}_{\boldsymbol{l}}} (\boldsymbol{l'} + \boldsymbol{z}) , \tag{31}$$

where χ denotes the indicator function [18], defined by the index set $\mathcal{I}$.

In this paper we use a combination scheme that allows for anisotropic minimal and target levels. For a more distinctive notation we will denote minimal and target level by $\boldsymbol{l}^{\min}$ and $\boldsymbol{l}^{\max}$, respectively. The level set has the form

$$\mathcal{I}_{\boldsymbol{l}^{\min},\boldsymbol{l}^{\max}} = \left\{ \boldsymbol{l}' \in \mathbb{N}^d : \boldsymbol{l}' \in \text{conv}(\boldsymbol{l}^{\min}, \boldsymbol{l}^{\min} + \boldsymbol{d}^{(1)}, \ldots, \boldsymbol{l}^{\min} + \boldsymbol{d}^{(d)}) \right\} , \qquad (32)$$

where $\boldsymbol{d}^{(i)} = (0, \ldots, l_i^{\max} - l_i^{\min}, \ldots, 0)$, and conv denoting the convex hull. This again describes a simplex, but one that is situated at $\boldsymbol{l}^{\min}$ and stretches out to $l_i^{\max}$ in the i-th direction. The potential for anisotropic minimal and target levels can be used to increase the resolution for more important directions versus directions that may be sampled at lower resolutions, compare [12]. Additionally, some solvers inherently require a minimal number of grid points, e.g. for the stencils they use, or to make sure that some relevant physical scale can be resolved.

In the case of interpolating a known function, the representations (26) and (31) are identical. Thus, the representation error of the combination technique is the same as in Equation (29). But if the full grid functions constituting the sum in (31) are derived in a separate way, e.g. by solving a PDE, they carry an additional discretization error and it is not guaranteed that Equation (29) holds for the total error. However, there are cases where it can be shown that it does. Firstly, if the full grid solutions satisfy an error splitting assumption of the form

$$(f - f_{\boldsymbol{l}})(x_{\boldsymbol{l}}) = \sum_{i=1}^{d} \sum_{\{j_1,\ldots,j_i\} \subset \{1,\ldots,d\}} \kappa_{j_1,\ldots,j_i}(x_{\boldsymbol{l}}; h_{j_1},\ldots,h_{j_i}) h_{j_1}^2 \cdots h_{j_i}^2 \qquad (33)$$

at all full grid points $x_{\boldsymbol{l}}$, with bounded $|\kappa_{j_1,\ldots,j_i}(x_{\boldsymbol{l}}; h_{j_1},\ldots,h_{j_i})| \leq K$ and $h_j := 2^{-j}$, then the point-wise total error of the combined function has the same asymptotic rate as in Equation (29) [14]. The same can be shown for the L^2-error of the combined function, if the full grid functions are solutions of operator equations with elliptic operators acting on arbitrary Gelfant triples [13]. Further analysis for the advection-diffusion equation is found in [36], where the same asymptotic convergence rates are shown. Error analysis for the combination technique with multiple time intervals in the case of the advection equation can be found in [28].

Still, a general theory for arbitrary, particularly nonlinear, PDEs is not available. For the case of kinetic equations, such as the Vlasov–Poisson system or the gyrokinetic equations, the notion of convergence itself has to be defined first. While in certain scenarios or temporal regimes, where the nonlinearity of the equations is negligible, a convergence of $f_{\boldsymbol{l}}(t)$ to the exact solution is expected, this is not the case anymore for the long term simulation of turbulent/chaotic behavior. In this situation, the combination technique can only be expected to work well for the solution of the PDE for very small time frames ΔT [28]. In practice, we therefore iterate the application of the combination technique m times in order to arrive at a final time $m\Delta T$.

For the combination step that is necessary after every interval, all subproblems on the different full grids have to be combined in the resulting sparse grid space, according to Equation (31). For this purpose, the full grid functions are hierarchized and the resulting hierarchical coefficients are added to or subtracted from the sparse grid solution, where the coefficients from missing subspaces are padded with zeros. After that, the sparse grid solution has to be redistributed to the original full grids. There are several ways to do this. Carrying out an L^2 projection would yield, on each full grid, the best approximation to the sparse grid function in the L^2 norm. However, in practice, such a projection requires more computational effort because a linear system of equations involving the mass matrix $M_{\boldsymbol{l}'\boldsymbol{s}',\boldsymbol{l}\boldsymbol{s}} = \langle \psi_{\boldsymbol{l}'\boldsymbol{s}'}, \psi_{\boldsymbol{l}\boldsymbol{s}} \rangle$ has to be solved. Thus, we use the method of keeping the coefficients of all hierarchical difference spaces contained in the full grid and omitting the rest:

$$\mathcal{P}_{\boldsymbol{l}} f^{(\text{s})} = \mathcal{P}_{\boldsymbol{l}} \left(\sum_{\boldsymbol{l}' \in \mathcal{I}_{\boldsymbol{l}^{\min},\boldsymbol{l}^{\max}}} \sum_{\boldsymbol{s}} \alpha_{\boldsymbol{l}'\boldsymbol{s}} \psi_{\boldsymbol{l}'\boldsymbol{s}} \right) := \sum_{\boldsymbol{l}^{\min} \leq \boldsymbol{l}' \leq \boldsymbol{l}} \sum_{\boldsymbol{s}} \alpha_{\boldsymbol{l}'\boldsymbol{s}} \psi_{\boldsymbol{l}'\boldsymbol{s}} , \qquad \forall \boldsymbol{l} \in \mathcal{I}_{\boldsymbol{l}^{\min},\boldsymbol{l}^{\max}} . \qquad (34)$$

Afterwards, the dehierarchization operation is applied on each full grid.

For an orthogonal multiscale basis, e.g. orthonormal wavelets, this would be identical to the L^2 projection, since in this case $M_{l's',ls} = \delta_{l's',ls}$. For biorthogonal wavelet bases it is not, so the best approximation property is lost. However, as noted earlier, the mass is conserved by this projection when using biorthogonal wavelet bases, since it is wholly contained in the subspace $l^{\min}$, which is present in all full grids. We argue that, in this instance, mass conservation is the more important property compared to L^2 best approximation.

Finally, quantitative analysis of the solution is commonly done by studying certain quantities of interest that are typically averaged over the phase-space. This is why, in our simulations of the Vlasov-Poisson system, we assess the quality of the combined solution by analyzing its performance in two benchmark problems via the time traces of the potential energy. A similar approach was taken in [27] for simulations of the gyrokinetic equations with the code GENE [21]. A full convergence study of the combination technique failed, however, since for many index sets the simulations became unstable. Therefore, the main goal of this paper is to show that the use of biorthogonal wavelet bases helps to alleviate this problem.

4 Mass-conserving solvers for hyperbolic problems

Naturally, the total mass can only be conserved in a combination technique with the novel hierarchical bases if the underlying solvers are also mass-conserving. We therefore choose a finite volume advection solver and a semi-Lagrangian solver for the kinetic equations that both conserve mass and, moreover, operate on a structured six-dimensional grid.

By contrast, plasma turbulence codes such as GENE that are closely inspired by predictions for experimental nuclear fusion devices typically incorporate optimizations and simplifications that make a numerical analysis on the system level difficult. Moreover, they do not conserve mass or other momenta, since they only simulate a perturbation from equilibrium instead of the full distribution function.

4.1 Finite volume advection solver

As a first benchmark problem, we consider the incompressible advection equation

$$\frac{\partial u(\boldsymbol{x}, t)}{\partial t} + \boldsymbol{a} \cdot \nabla_{\boldsymbol{x}} u(\boldsymbol{x}, t) = 0 \tag{35}$$

on the unit hypercube $\Omega = [0, 1]^d$ in d dimensions. The advection velocity $\boldsymbol{a}$ is $\boldsymbol{1}$, constant throughout Ω. Note that a purely diagonal advection is the most challenging direction for combination technique simulations.

The spatial boundary conditions are periodic, which makes this problem well-suited for our purposes, since the analytical solution can be obtained in a straightforward manner by periodically translating the initial distribution. This makes a direct Monte Carlo integration of the error possible, cf. Section 5.1.

A straightforward discretization is the first-order finite volume upwinding scheme for positive velocities

$$\begin{aligned} \nabla_i u = \frac{\partial u}{\partial x_i} &\approx \frac{u(\boldsymbol{x}, t - \Delta t) - u(\boldsymbol{x} - \Delta x_i, t - \Delta t)}{\Delta x_i}, \\ \frac{\partial u}{\partial t} &\approx \frac{u(\boldsymbol{x}, t) - u(\boldsymbol{x}, t - \Delta t)}{\Delta t}, \end{aligned} \tag{36}$$

which conserves the mass $\|u\|_1$—assuming positivity of u—up to machine precision. Note that the numerical value of the mass (at initial time) can vary slightly depending on the resolution due to the numerical interpolation of the initial solution.

Plugging Equation (36) into Equation (35) and solving for $u(\boldsymbol{x}, t)$ results in the explicit Euler time-stepping scheme

$$u(\boldsymbol{x}, t) = u(\boldsymbol{x}, t - \Delta t) - \nabla u \cdot \boldsymbol{a} \cdot \Delta t, \tag{37}$$

which is first-order accurate in both space and time (provided that the loose CFL conditions of the combination technique are fulfilled, cf. [1]). The solver is implemented directly into the distributed combination technique framework DisCoTec.

4.2 Semi-Lagrangian Vlasov plasma solver

In a second example, we study a simulation of the Vlasov–Poisson system in 3d3v (three spatial and three velocity dimensions). We focus on the case of electrons in a neutralizing background. Let $f(\boldsymbol{x}, \boldsymbol{v}, t)$ be the distribution function of the electron species of a plasma in phase space $(\boldsymbol{x}, \boldsymbol{v}) \in \Omega \times \mathbb{R}^3$, $\Omega \subset \mathbb{R}^3$. This distribution function evolves according to the Vlasov equation

$$\partial_t f(\boldsymbol{x}, \boldsymbol{v}, t) + \boldsymbol{v} \cdot \nabla_{\boldsymbol{x}} f(\boldsymbol{x}, \boldsymbol{v}, t) + \frac{q}{m} \left(\boldsymbol{E}(\boldsymbol{x}, t) + \boldsymbol{v} \times \boldsymbol{B}(\boldsymbol{x}, t) \right) \cdot \nabla_{\boldsymbol{v}} f(\boldsymbol{x}, \boldsymbol{v}, t) = 0, \qquad (38)$$

where q and m denote electron charge and mass, respectively, $\boldsymbol{E}$ the electric and $\boldsymbol{B}$ the magnetic field. The self-consistent fields are given by Maxwell's equations or—for low-frequency phenomena—by Poisson's equation. In this paper, we focus on the Poisson equation given by

$$-\Delta_{\boldsymbol{x}} \phi(\boldsymbol{x}, t) = 1 - \int f(\boldsymbol{x}, \boldsymbol{v}, t) \, \mathrm{d}\boldsymbol{v}, \quad \boldsymbol{E}(\boldsymbol{x}, t) = -\nabla_{\boldsymbol{x}} \phi(\boldsymbol{x}, t). \qquad (39)$$

and $\boldsymbol{B}(\boldsymbol{x}, t) = 0$.

Several discretization schemes are possible in order to approximate the Vlasov equation: methods of Eulerian, semi-Lagrangian, and Lagrangian type. While Lagrangian particle methods overcome the curse of dimensionality to some extent, they suffer from inherent numerical noise. In this paper, instead, we propose to use the sparse grid combination technique applied to a grid-based method in order to overcome the curse of dimensionality with a noise-free method. We choose the semi-Lagrangian method as the basis for our full grid simulations, since semi-Lagrangian schemes allow for larger time steps in explicit time propagation schemes compared to Eulerian methods.

The semi-Lagrangian method discretizes the distribution function on a grid and the propagation is based on the characteristic equations

$$\frac{\mathrm{d}\boldsymbol{X}}{\mathrm{d}t} = \boldsymbol{V}, \quad \frac{\mathrm{d}\boldsymbol{V}}{\mathrm{d}t} = \frac{q}{m} \left(\boldsymbol{E} + \boldsymbol{V} \times \boldsymbol{B} \right) \qquad (40)$$

along which the distribution function is constant, i.e.

$$f(\boldsymbol{x}, \boldsymbol{v}, t) = f_0(\boldsymbol{X}(0; \boldsymbol{x}, \boldsymbol{v}, t), \boldsymbol{V}(0; \boldsymbol{x}, \boldsymbol{v}, t)),$$

where $(\boldsymbol{X}(\tau; \boldsymbol{x}, \boldsymbol{v}, t), \boldsymbol{V}(\tau; \boldsymbol{x}, \boldsymbol{v}, t))$ is the solution of the characteristic curve at time τ starting at $(\boldsymbol{x}, \boldsymbol{v})$ at time t and f_0 is the initial solution at time $\tau = 0$.

Since the fields depend on the distribution function, we cannot follow the characteristics from final to initial time. However, over small time intervals, approximating the fields by a constant value in time is a good approximation. Based on this assumption, the Vlasov equation can be solved in two steps: Solving the ODE system of the characteristics is followed by evaluating the interpolated distribution function at the foot of the characteristic. Using an operator splitting method and solving the $\boldsymbol{x}$ and the $\boldsymbol{v}$ advection separately, the advection coefficients are constant in each time step so that the solution of the system of ODEs can be found analytically. This algorithm has been proposed in [4]. Several schemes are conceivable for the one-dimensional interpolations. In our implementation, we use Lagrangian interpolation, since the interpolation is local, a fact which is favorable in the context of distributed domains. It can be shown that this scheme conserves both mass and momentum up to round-off errors.

For our simulations, we use the semi-Lagrangian method implemented for high performance computing simulations in SeLaLib [23]. In order to couple SeLaLib to the DisCoTec framework, some adaptations were implemented regarding computation and communication of the distribution function. Fortunately, the coordinate placement used in SeLaLib naturally results in nested coordinates when doubling the resolution, which matches the requirements of the combination technique perfectly.

5   Combination technique experiments

In our combination technique simulation experiments, the solvers described in Section 4 are used to perform a time step update on each of the component grids. These solver updates are independent between the component grids and can be performed in parallel. Information between the component grids is only exchanged in the combination step: The recombination transforms the data on each component grid into their hierarchical representations—coefficients corresponding to either of $\psi^{\mathtt{hat}}$, $\psi^{\mathtt{bo}}$, $\psi^{\mathtt{fw}}$—and sums them up to a common sparse grid according to the combination formula (Equation (31)). The decombination then "scatters", i. e., projects, the sparse grid data back to the component grids and, afterwards, performs the inverse transformation on each grid to recover the interpolating coefficients. Then, the next time step can begin.

The three numerical experiments were designed to investigate different behaviors: First, to verify that mass is now conserved up to machine precision; second, to showcase the error properties of the combination technique using the different basis functions w.r.t. an analytical solution, cf. Section 5.1; and, third, to apply the different combination technique approaches to plasma physics scenarios while comparing them to the monolithic full grid solution at a fixed amount of available memory, cf. Sections 5.2 and 5.3.

In order to achieve a fair comparison, the number of degrees of freedom for the full grid simulations is calculated as the number of points on that single grid

$$\#\mathrm{DOF}_{\mathrm{fg}} = \sum_{i=1}^{d} 2^{l_i^{\mathrm{fg}}}, \tag{41}$$

whereas for the combination technique, we need to consider the sum of points on all component grids

$$\#\mathrm{DOF}_{\mathrm{ct}} = \sum_{\boldsymbol{l} \in \mathcal{I} | \lambda_{\boldsymbol{l}} \neq 0} \sum_{i=1}^{d} 2^{l_i}. \tag{42}$$

This implies that the combination technique solution is not necessarily "cheaper" than a full grid of level $\boldsymbol{l}^{\max}$, and that the index set $\mathcal{I}$ needs to be chosen carefully. We use only double-precision floats for the representation of the (interpolating and hierarchical) basis coefficients, such that the memory consumption can be obtained as $\#\mathrm{DOF} \cdot 8/2^{30}$ GiB.

Another aspect that is common to all experiments is the choice of the recombination interval ΔT. Lastdrager et al. [28] analyzed the different error terms introduced by the combination technique with hat functions for advection-type problems. They could show that by recombination at constant simulation time intervals throughout the simulation, the leading time-error term would be of first order [28, Eq. (32)]. However, they did not take the solver's own time-dependent error into account, but only the solver's spatial error. Hence, we restrict ourselves to recombination after each solver time step Δt for our experiments, i. e., $\Delta T = \Delta t$, in order to avoid unforeseen numerical artifacts originating from different time ranges and error expansions between the solver and the combination technique.

In the simulations, the mass (and other properties) of the combination technique solution are obtained by evaluating the same property on each component grid and summing with the combination formula

$$Q_{\mathrm{ct}}(f) := \sum_{\boldsymbol{l} \in \mathcal{I}} \lambda_{\boldsymbol{l}} \cdot Q(f_{\boldsymbol{l}}). \tag{43}$$

Note that the formula is only equivalent to the interpolated sparse grid properties

$$Q_{\mathrm{sg}}(f) := Q\left(\sum_{\boldsymbol{l} \in \mathcal{I}} \lambda_{\boldsymbol{l}} \cdot f_{\boldsymbol{l}} \right) \tag{44}$$

for linear operators, e.g. the mass. For instance in the maximum norm, one can easily imagine that the combined maximum is not the same as the sparse grid maximum, since the maximal

function value will usually be attained at different points in the component grids—and the sparse grid point with maximal value will likely not be present in each of the component grids.

Sections 5.2 and 5.3 are going to deal with both the (linear) mass and the (nonlinear) potential energy $\phi(f)$ as combined quantities $Q_{\text{ct}}(f)$. We validated the approach of using the combined potential energy with coarsely resolved CT plasma simulations, where both $\phi_{\text{ct}}(f)$ and the interpolated $\phi_{\text{sg}}(f)$ could be feasibly computed. The result was that, for those time intervals where the simulation was stable, the properties would be in close agreement. We can therefore assume that the combination technique linearization assumption

$$Q_{\text{sg}}(f) \approx Q_{\text{ct}}(f) \tag{45}$$

for a general quantity of interest $Q(\cdot)$ holds sufficiently in the cases that are physically relevant. And conversely, we observed that physical bounds—such as the positivity of the potential energy— are only violated at the onset of numerical instabilities.

The following sections use the combined quantities of interest unless otherwise indicated.

5.1 Advecting a Gaussian

The finite volume advection solver introduced in Section 4.1 is used to compare the different hierarchical basis functions on an analytically solvable scenario in arbitrary dimensions.

The unknown concentration u is initialized at $t = 0$ by a Gaussian distribution

$$u(\boldsymbol{x}, t = 0) = \exp\left(-\sum_{i=1}^{d}\left(x_i - \frac{1}{2}\right)^2 \cdot \frac{1}{\sigma^2}\right), \tag{46}$$

that is, a multivariate normal distribution with standard deviation $\sigma = \frac{1}{3}$. It is normalized, such that the maximum value is always 1, independently of d. Gaussians are the typical initial conditions in velocity space for plasma physics simulations, and they can be particularly challenging to be represented on sparse grids [32].

The solver time step size Δt is set to 1×10^{-4}, to fulfill the CFL condition for all considered resolutions, and recombination takes place after each time step. While the full grid reference simulations are each run on a single grid at different isotropic resolutions, all combination technique simulations have a fixed minimum level of $\boldsymbol{l}^{\min} = (2)^d$ but different target levels $\boldsymbol{l}^{\max}$, which correspond to the resolutions of the reference simulations. This results in an increasing number of component grids as $\boldsymbol{l}^{\max}$ increases.

We first investigate the combination technique's behavior with respect to the one known invariant of the solver: u_{exact} is chosen such that the mass $\|u_{\text{exact}}\|_1$—assuming positivity of u—is approximately 0.570792^d, cf. Equation (46). The full grid simulations all conserve the mass up to machine precision. Figure 4 shows the mass fluctuations throughout the simulation time in the combination technique solutions. It clearly states that the ψ^{bo} and ψ^{fw} bases lead to the desired conservation of mass, in contrast to ψ^{hat}.
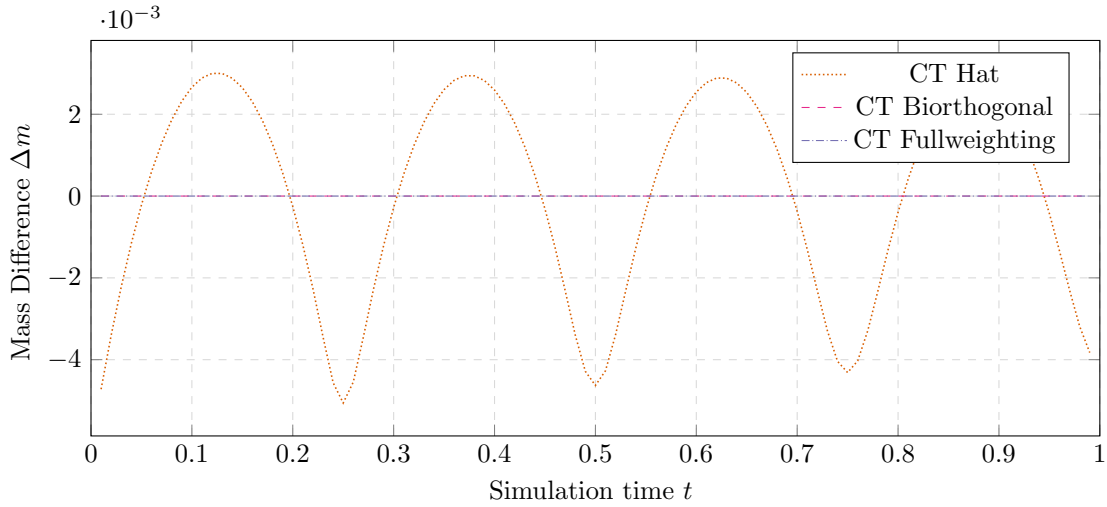
Figure 4: Difference Δm between the combined masses $\sum_{l \in \mathcal{I}} \lambda_l \cdot \int u_l(\boldsymbol{x}, t) \, d\boldsymbol{x}$ and analytical mass $\int u_{\text{exact}}(\boldsymbol{x}, \cdot) \, d\boldsymbol{x} \approx 0.186$ for a combination technique 3D advection scenario with $\boldsymbol{l}^{\min} = (2, 2, 2)$, $\boldsymbol{l}^{\max} = (11, 11, 11)$ (136 component grids). Using hierarchical hat functions leads to fluctuations of up to 2.7% of the analytical mass. The maxima in the mass are reached whenever the Gaussian's maximum passes through those regions of the sparse grid where there are fewer degrees of freedom. By comparison, the biorthogonal and full weighting bases conserve the mass in the simulation up to an accuracy of 1×10^{-10}.

Next, we consider different errors at $t = 1.0$, i.e., after one full cycle of diagonally advecting the initial solution through the domain. Since the true solution is known analytically, the relative error norms w.r.t. the analytical solution are obtained by Monte Carlo integration

$$
\begin{aligned}
\frac{\|u_{\text{ct}} - u_{\text{exact}}\|_1}{\|u_{\text{exact}}\|_1}(t) &\approx \frac{\sum_{j=1}^N |u_{\text{ct}}(\boldsymbol{x}_j, t) - u_{\text{exact}}(\boldsymbol{x}_j, t)|}{\sum_{j=1}^N |u_{\text{exact}}(\boldsymbol{x}_j, t)|}, \\
\frac{\|u_{\text{ct}} - u_{\text{exact}}\|_2}{\|u_{\text{exact}}\|_2}(t) &\approx \frac{\sum_{j=1}^N |u_{\text{ct}}(\boldsymbol{x}_j, t) - u_{\text{exact}}(\boldsymbol{x}_j, t)|^2}{\sum_{j=1}^N |u_{\text{exact}}(\boldsymbol{x}_j, t)|^2}, \\
\frac{\|u_{\text{ct}} - u_{\text{exact}}\|_\infty}{\|u_{\text{exact}}\|_\infty}(t) &\approx \frac{\max_{\boldsymbol{x}_j} |u_{\text{ct}}(\boldsymbol{x}_j, t) - u_{\text{exact}}(\boldsymbol{x}_j, t)|}{\max_{\boldsymbol{x}_j} |u_{\text{exact}}(\boldsymbol{x}_j, t)|},
\end{aligned} \tag{47}
$$

on $N = 1 \times 10^5$ randomly sampled coordinates $\boldsymbol{x}_j$ (the pseudorandom numbers are generated using the Mersenne Twister algorithm). N is chosen numerically such that the variance of the Monte Carlo integrals over multiple runs with different random coordinates is sufficiently small. Using random coordinates—not sparse grid collocation points—is particularly important since the analytical solution is not contained in the linear ansatz space; it is necessary to capture the error between collocation points especially for low-resolution simulations.

Figure 5 displays several characteristics of the different solutions: For the same $\boldsymbol{l}^{\max}$, the combination technique solutions require drastically smaller numbers of degrees of freedom than the full grid solutions, while providing similar numerical accuracy in all considered error norms. Furthermore, using the mass-conserving basis functions reduces the errors further, an effect that gets more significant as the dimensionality increases.

These errors can be split up in a meaningful way: For a combination scheme with minimum level $\boldsymbol{l}^{\min}$ and maximum level $\boldsymbol{l}^{\max}$, it seems unlikely to attain errors below that of the reference full grid solution $u_{\text{fg}, \boldsymbol{l}^{\max}}$, since all coordinates contained in the sparse grid are a subset of the full grid's DOF. We therefore take the full grid errors as a bound to the combination technique errors. Thus, we may decompose the total error into the solver's error itself and the error introduced by
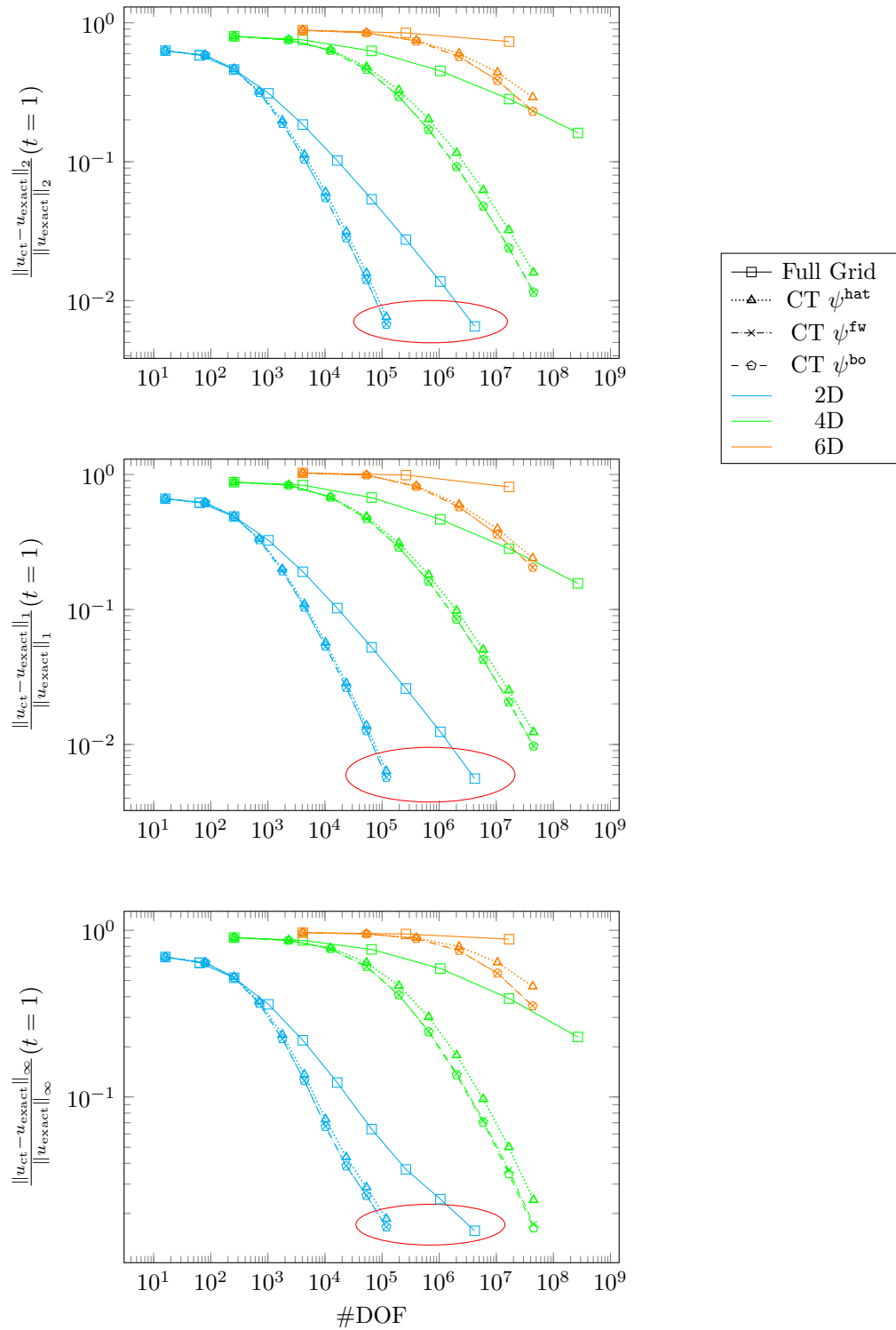
16

Figure 5: Monte Carlo errors for the advection solver over the number of degrees of freedom used. Curves are shown for dimensionalities of 2, 4, and 6, comparing the full grid reference solutions to combination technique solutions with different hierarchical basis functions at $t = 1$. For the comparison in the red ellipses (2D, $\boldsymbol{l}^{\min} = (2, 2)$, $\boldsymbol{l}^{\max} = (11, 11)$), the u fields are plotted in Figure 7 at $t = 0.5$
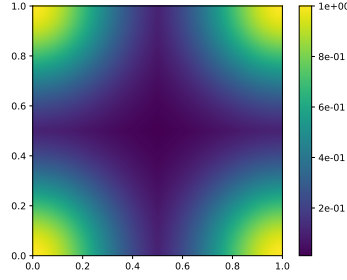
.

17

Figure 6: Analytical solution u_{exact} at time $t = 0.5$. The Gaussian has moved along half of the diagonal, such that the maximum is attained exactly at the corners of the domain.

hierarchical hat basis
ψ^{hat}

hierarchical hat basis
ψ^{hat}

biorthogonal basis
ψ^{bo}

$=$

biorthogonal basis
ψ^{bo}

$+$

full weighting basis
ψ^{fw}

full weighting basis
ψ^{fw}

(a) Total combination technique errors $u_{\text{ct}} - u_{\text{exact}}$

(b) Full grid solver error $u_{\text{fg},\boldsymbol{l}^{\max}} - u_{\text{exact}}$

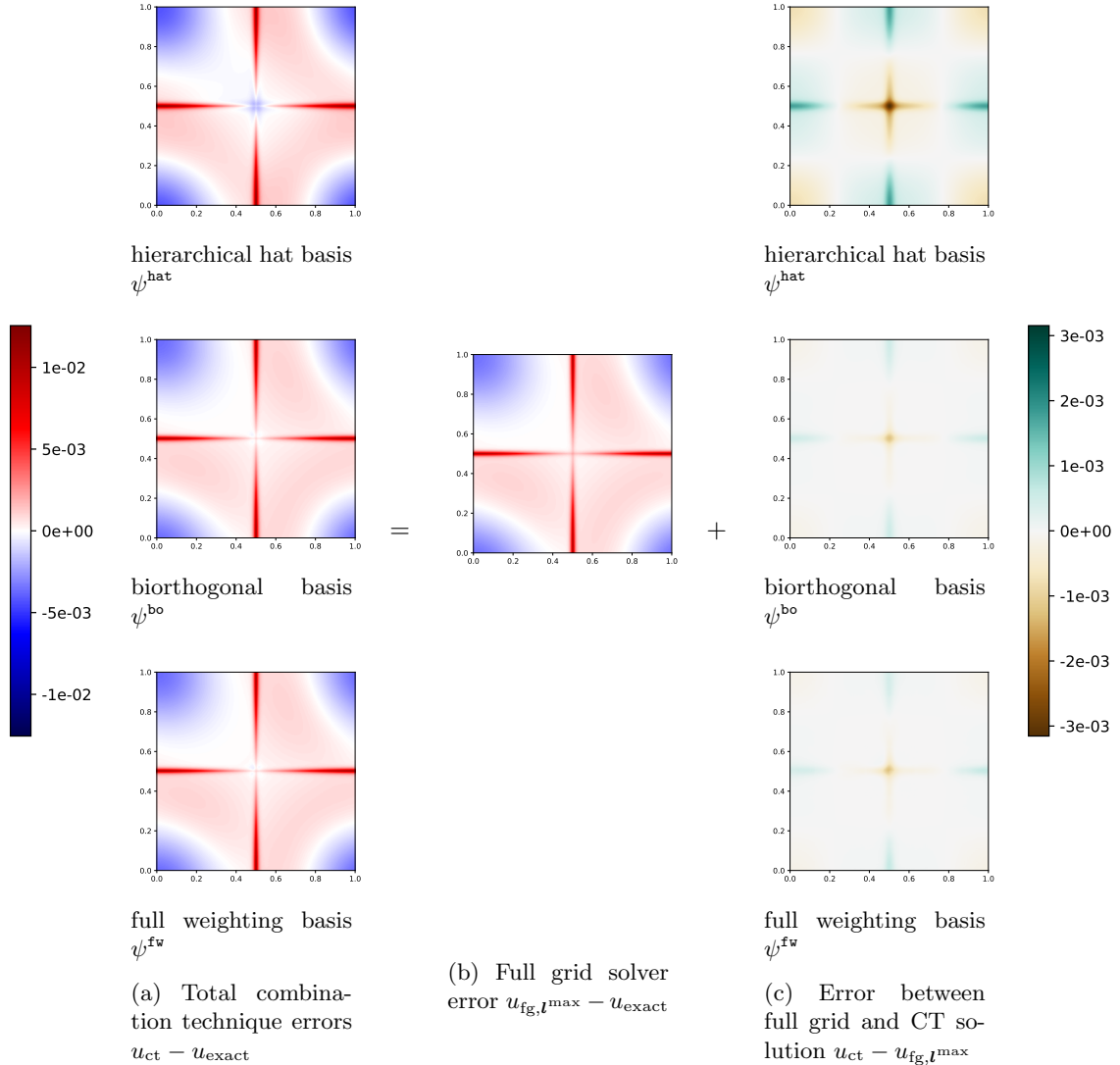(c) Error between full grid and CT solution $u_{\text{ct}} - u_{\text{fg},\boldsymbol{l}^{\max}}$

Figure 7: Error decomposition for the 2D advection solution at $t = 0.5$: The combination technique error is split into the full grid error and the difference between full grid and combination technique solution. All solvers used $\boldsymbol{l}^{\max} = (11, 11)$, and the combination technique used $\boldsymbol{l}^{\min} = (2, 2)$, leading to approximately 4.19×10^6 DOF for the full grid and 1.19×10^5 DOF for the combination technique simulations, cf. the ellipses in Figure 5. We use $t = 0.5$ instead of $t = 1.0$, because the slight differences in Figure 7a are easier to compare when the maxima of the errors are centered. The difference between the full grid and CT solutions is visible on a smaller scale only, hence the separate color bar on the right.

18

the combination technique scheme:

$$u_{\text{ct}} - u_{\text{exact}} = (u_{\text{fg},\boldsymbol{l}^{\text{max}}} - u_{\text{exact}}) + (u_{\text{ct}} - u_{\text{fg},\boldsymbol{l}^{\text{max}}}) \,,$$
$$\left\| u_{\text{ct}} - u_{\text{exact}} \right\| \leq \left\| u_{\text{fg},\boldsymbol{l}^{\text{max}}} - u_{\text{exact}} \right\| + \left\| u_{\text{ct}} - u_{\text{fg},\boldsymbol{l}^{\text{max}}} \right\| \,. \tag{48}$$

Visually, it is instructive to investigate one fixed set of data points, namely the two-dimensional problem with maximum level $\boldsymbol{l}^{\text{max}} = (11, 11)$ (cf. the ellipse in Figure 5). The analytical solution at $t = 0.5$, interpolated onto $\boldsymbol{l} = \boldsymbol{l}^{\text{max}}$ (i.e., $N = (2048, 2048)$), is displayed in Figure 6.

In Figures 6 and 7, the solutions are displayed at time $t = 0.5$, i.e., the Gaussian in the analytical solution has moved along the diagonal, such that the maximum of 1 is now exactly on the corner points. The error field w.r.t. the analytical solution is shown in Figure 7—making it a visualization of the error decomposition, Equation (48). To plot the two-dimensional fields, we choose $t = 0.5$ instead of $t = 1.0$ (like in Figure 5) because the slight differences in Figure 7a are easier to compare when the maxima of the errors are centered.

For all solver schemes, most of the error is introduced along the middle axes. This is to be expected, given that the analytical solution is C^∞ in most of the domain, but only C^0 along the middle axes, since the middle axes at $t = 0.5$ are the boundary of the initial solution $u(t = 0)$ that is being advected through the domain. High values of u_{exact} are underestimated while low values tend to be overestimated. We observe that the errors are dominated by the full grid error $\left\| u_{\text{fg},\boldsymbol{l}^{\text{max}}} - u_{\text{exact}} \right\|$, and the sparse grid projection error $\left\| u_{\text{ct}} - u_{\text{fg},\boldsymbol{l}^{\text{max}}} \right\|$ is more than one order of magnitude lower for all choices of hierarchical basis functions—note the different color scale in Figure 7. Also, the sign of the sparse grid projection error appears to form a checkerboard pattern: The values are underestimated at the center of the domain, partially compensating the full grid solver's error. In the other regions of Ω, the full grid solver's error is slightly amplified by the sparse grid projection error.

Yet, strikingly, the difference between the mass-conserving CT solutions and the full grid solutions $\left\| u_{\text{ct}} - u_{\text{fg},\boldsymbol{l}^{\text{max}}} \right\|$—the combination technique projection error—is significantly lower than for the hat basis combination technique: It generally has the same shape but is lower by a factor of ≈ 3 throughout the domain.

This also explains the differences in the error norms highlighted in Figure 5: For the same $\boldsymbol{l}^{\text{max}}$, the projection error measure $\left\| u_{\text{ct}} - u_{\text{fg},\boldsymbol{l}^{\text{max}}} \right\|_p$ is decreased by a factor of ≈ 3 when using ψ^{bo} or ψ^{fw} instead of ψ^{hat} for the combination. This error reduction is observed in all advection simulations considered here—across all dimensionalities d and norm parameters p.

In addition to maintaining the conservation of mass, the increased accuracy is a substantial benefit for combination technique simulations.

5.2 Landau damping with the semi-Langrangian solver

For the Vlasov–Poisson system, we study two typical benchmark problems, Landau damping and the two-stream instability. Landau damping refers to the effect of exponential damping of longitudinal waves in plasmas. It can be studied when simulating the Vlasov–Poisson system with the initial condition

$$f_0(\boldsymbol{x}, \boldsymbol{v}) = \left(1 + \varepsilon \sum_{i=1}^{3} \cos(kx_i) \right) \frac{1}{(2\pi)^{3/2}} \exp\left(-\frac{\|\boldsymbol{v}\|_2^2}{2} \right) \tag{49}$$

on the domain $[0, \frac{2\pi}{k}]^3 \times \mathbb{R}^3$ for certain ranges of k, typically $k = 0.5$.

A linear dispersion analysis reveals the damping rate and the oscillation frequency in time to first order in ε. For the mode $k = 0.5$ the damping rate is $\omega = -0.1533$. For a small value of $\varepsilon = 0.01$, the electric energy practically coincides with the linear prediction from the second oscillation onwards. In grid-based simulations, however, the so-called numerical recurrence phenomenon occurs: After a certain time depending on the resolution of the velocity grid, more precisely when $T = \frac{2\pi}{k\Delta v}$ with Δv being the (minimal) resolution in velocity space, a sudden growth in potential energy occurs. This phenomenon is a purely numerical artefact.

19

For low-resolution test runs of the combination technique, we could verify that the combined potential energy $\|\phi\|_{2,\mathrm{ct}}^2$ is a very good approximation to the sparse grid potential energy $\|\phi\|_{2,\mathrm{sg}}^2$, cf. Equation (45): The sparse grid's energy $\|\phi\|_{2,\mathrm{sg}}^2$ is obtained by interpolating f_{ct} onto $\boldsymbol{l}^{\mathrm{max}}$ and then computing $\|\phi\|_2^2$ on the finely-resolved grid. This holds at least for the regions where the simulation is stable—in those cases where $\|\phi\|_{2,\mathrm{ct}}^2$ becomes smaller than 0, it is plausibly quite different from $\|\phi\|_{2,\mathrm{sg}}^2$, since the latter is guaranteed to be positive.
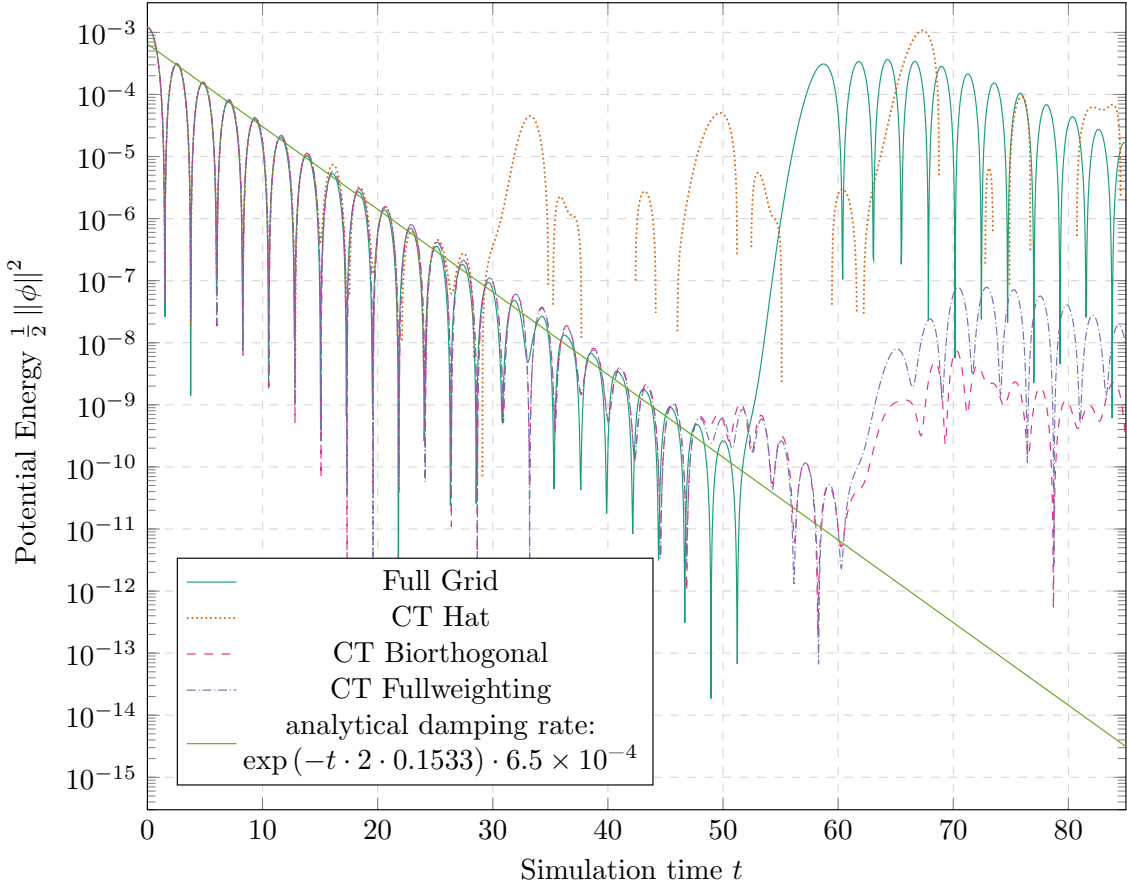

Figure 8: Landau Damping: The 5.5 GiB full grid with a resolution of $\boldsymbol{N} = (16, 16, 16, 57, 57, 57)$ is compared to combination technique solutions with $\boldsymbol{l}^{\mathrm{min}} = (4, 4, 4, 2, 2, 2)$, $\boldsymbol{l}^{\mathrm{max}} = (4, 4, 4, 8, 8, 8)$ at 5.4 GiB for the f data structures. This means that the $\boldsymbol{x}$ space resolutions are exactly the same between the full and combination simulations, and combination is only employed in the $\boldsymbol{v}$ space, resulting in 64 component grids for the combination technique simulations. For the combination technique simulations, combination is performed after every time step of $\Delta t = 0.01$.

Initial Landau damping test simulations showed that the hierarchical coefficients rapidly approach zero for finer $\boldsymbol{x}$ space resolutions, such that the $\boldsymbol{x}$ levels are uniformly chosen at a relatively coarse resolution of $l_{\boldsymbol{x}} = 4$ for this experiment. This also becomes clear from the structure of the problem, since the perturbation is chosen aligned with the grid. This alignement persists in the dynamics of the linearized equation that decouples the three one-dimensional perturbations. Thus, we are effectively reducing the combination dimensions to the three velocity components v_1, v_2, v_3.

The exact combination technique scenario is $\boldsymbol{l}^{\mathrm{min}} = (4, 4, 4, 2, 2, 2)$, $\boldsymbol{l}^{\mathrm{max}} = (4, 4, 4, 8, 8, 8)$, resulting in 64 component grids, which take up a total of 5.4 GiB for the distribution functions' data structures. We compare this to the same simulation run on a single full grid with resolution

$\boldsymbol{N} = (16, 16, 16, 57, 57, 57)$ (i.e, $\boldsymbol{l} \approx (4, 4, 4, 5.83, 5.83, 5.83)$)), totalling ≈ 5.5 GiB for the f field and matching the $\boldsymbol{x}$ resolutions of the combination technique simulations exactly.

The simulation is performed with a three-point Lagrange interpolation used for the semi-Lagrangian method. Using this relatively low-order interpolation, we have a minimal overhead due to ghost-cell transfer but on the other hand, we clearly see some deviations of the simulation from the analytical damping rate. For the full grid solution, we expect the recurrence at time $T = \frac{2\pi \cdot 57}{0.5 \cdot 12} \approx 59.7$ which can also be observed (cf. Fig. 8). The combination technique solutions show no clear recurrence but we observe smaller jumps in the damping profile of the energy curve. With the pure hat function sparse grids, this occurs already very early around time 30. For the full weighting and biorthogonal solutions, on the other hand, the effect occurs at a time similar to the recurrence time.

5.3 Plasma instability with the semi-Langrangian solver

Next, we consider the two-stream instability with initial condtion

$$
\begin{aligned}
f_0(\boldsymbol{x}, \boldsymbol{v}) = {} & \left(1 + \varepsilon \sum_{i=1}^{3} \cos(0.2 x_i)\right) \frac{1}{2(2\pi)^{3/2}} \cdot \\
& \left(\exp\left(-\frac{(v_1 - 2.4)^2}{2}\right) + \exp\left(-\frac{(v_1 + 2.4)^2}{2}\right)\right) \exp\left(-\frac{v_2^2 + v_3^2}{2}\right)
\end{aligned}
\tag{50}
$$

on the domain $\left[0, \frac{2\pi}{0.2}\right]^3 \times \mathbb{R}^3$. A linear dispersion analysis shows that there is a growing mode with growth rate given by $\omega = 0.2258$. The two-stream instability is bound to appear along the first dimension while Landau damping dominates along the others. For a perturbation with $\varepsilon = 0.001$, the potential energy curve follows the linear dispersion after some initial oscillations with a growth rate of ω. However, at some point the perturbation from equilibrium becomes so strong that nonlinear effects start dominating: Particles get trapped and the electric energy oscillates around a certain energy level.

Typical pitfalls of numerical methods during the nonlinear phase are too high oscillations or numerical damping of the electric energy, as observed in previous experiments [22].

Here, we compare a sparse grid combination solution with 84 component grids of $\boldsymbol{l}^{\min} = (3, 3, 3, 3, 3, 3)$, $\boldsymbol{l}^{\max} = (6, 6, 6, 6, 6, 6)$ and a single full grid solution with $\boldsymbol{N} = (22, 22, 22, 24, 24, 24)$ (i.e, $\boldsymbol{l} \approx (4.46, 4.46, 4.46, 4.58, 4.58, 4.58)$), since both configurations result in a similar memory footprint. For the linear regime we recover the growth rate with very good accuracy in all simulations. Therefore, in Figure 9, we focus on the nonlinear part of the simulation. It can be seen that the pure combination technique solution based on hat functions becomes unstable just before time 106 while the other three simulations oscillate around the reference solution from a simulation on a full grid with $\boldsymbol{l} = \boldsymbol{l}^{\max}$. In particular, no numerical damping is observed. The medium-resolution full grid solution contains a rather uniform oscillation with a larger amplitude and can therefore be considered slightly worse than the ones obtained with a full weighting or a biorthogonal combination technique.

Figure 10 shows the deviation of the mass over time. It can be seen that the full grid solution conserves mass to machine precision and the full weighting and biorthogonal combination technique solutions show a slow and small variation in mass. We attribute this to the accumulated errors from the individual hierarchization and dehierarchization operations on each grid: By definition, the hierarchical increments α_{ls} should be very small in smooth regions of the solution. In calculating them, we are subtracting very similar values from each other, which results in numerical cancellation for α to some degree. The visible mass changes for the ψ^{bo} and ψ^{fw} at $\approx 10^{-11}$ are due to the output accuracy of 11 digits in the SeLaLib diagnostic files; the mass change slowly accumulates on each component grid but only becomes visible in Figure 10 once it has reached the output threshold of 1×10^{-11} for the common sparse grid sum. This is also how the tiny spikes are introduced at the changes of mass, for instance in the case of ψ^{bo} at $t \approx 185$: Some of the component grids display the change in mass one or two time steps earlier than the others,
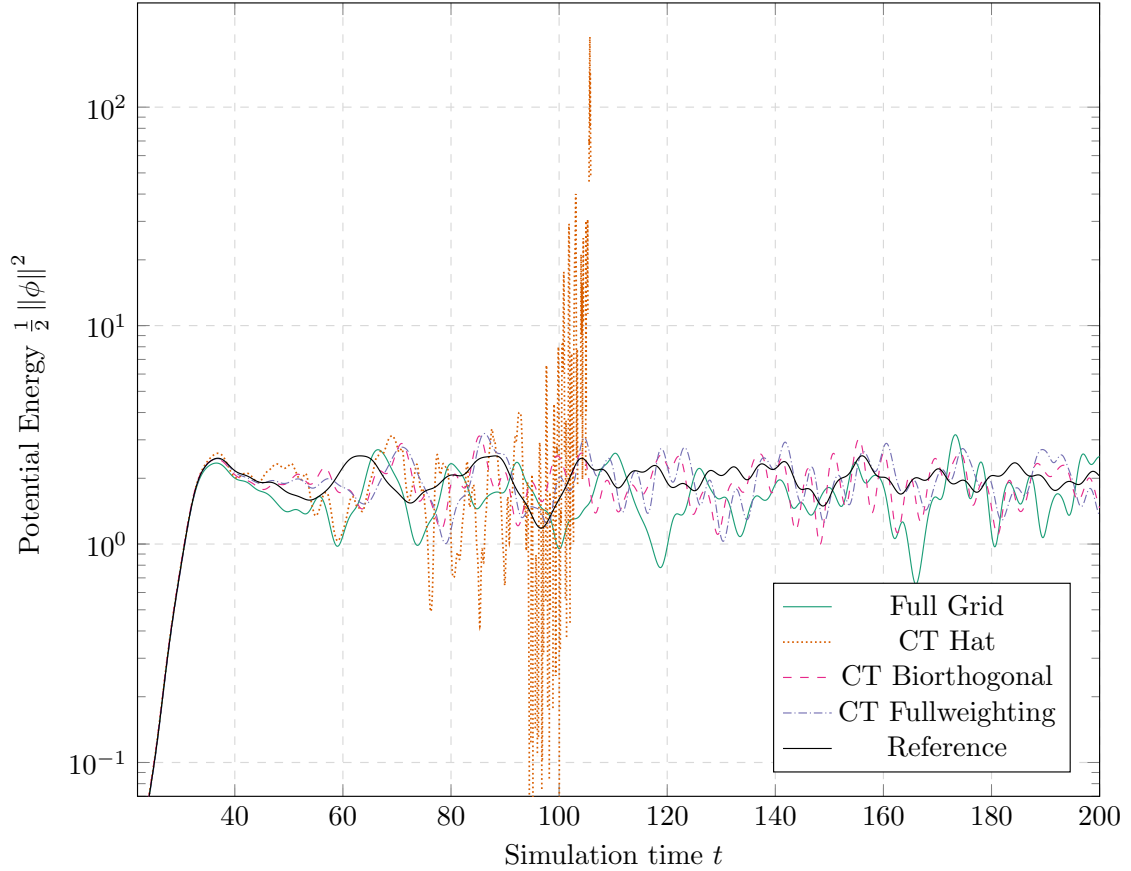
Figure 9: The potential energy over time in the two-stream instability scenario. The combination technique simulations use $\boldsymbol{l}^{\min} = (3, 3, 3, 3, 3, 3)$, $\boldsymbol{l}^{\max} = (6, 6, 6, 6, 6, 6)$ (84 component grids) and the full grid solution uses $\boldsymbol{N} = (22, 22, 22, 24, 24, 24)$—both resulting in a memory footprint of $\approx 1.1\,\mathrm{GiB}$ for the distribution function. The reference solution has a resolution level of $\boldsymbol{l}^{\max}$, taking up $512\,\mathrm{GiB}$ in memory for the plain f data. For the combination technique simulations, combination is performed after every time step of $\Delta t = 0.01$.
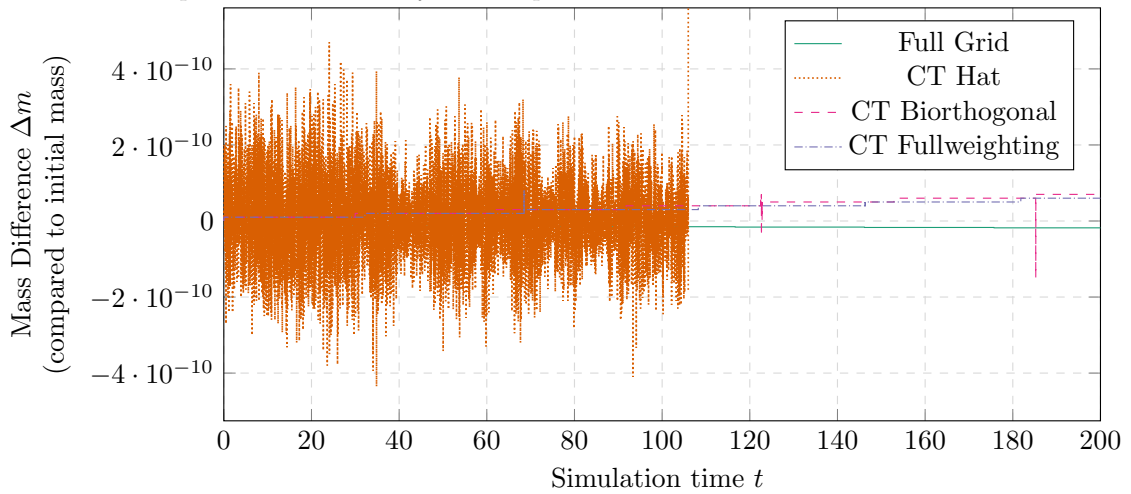

Figure 10: Difference between the mass $m_{\mathrm{ct}}(t) = \sum_{\boldsymbol{l} \in \mathcal{I}} \lambda_{\boldsymbol{l}} \cdot \int f_{\boldsymbol{l}}(x, t)\, dx$ and the initial mass $m_{\mathrm{ct}}(0)$ for the two-stream instability scenario.

and it may happen that those grids' combination coefficients have the same sign, which leads to the spikes. The combination coefficients also help to understand the high-frequent behavior when using $\psi^{\mathtt{hat}}$ for the combination technique: Even though the combination technique using the hierarchical hat basis is not mass-conserving, the error in mass shows an oscillatory behavior with rather small deviations up to the point where the simulation becomes unstable. In particular, mass conservation remains on the order of 10^{-10} way beyond time ≈ 94 where the values of the potential energy start to strongly oscillate, including negative values (which may occur due to the extrapolation of the potential energy from the values of the component grids, cf. Equation (43)).

Thus, it appears that the conservation of mass itself is not the determining factor for the numerical stability of the problem. Rather, the causality may be in the reverse direction: The choice of stable basis functions also introduces conservation of mass in the distribution function.

6 Discussion of kinetic CT simulations with mass-conserving basis functions

In summary, this work introduces the mass-conserving basis functions $\psi^{\mathtt{bo}}$ and $\psi^{\mathtt{fw}}$ for combination technique simulations. While the biorthogonal basis has been previously used for sparse grid simulations [24], the full weighting basis is a new approach that shows a connection to geometric multigrid methods.

A starting point for our research was the conservation of mass in the individual component grids, which is generally not to be expected if the combination technique is performed with hat functions. The present work shows the conservation of mass by $\psi^{\mathtt{bo}}$ and $\psi^{\mathtt{fw}}$ theoretically as well as experimentally for a finite volume advection solver and the semi-Lagrangian SeLaLib solver, which both use periodic boundary conditions. For solvers with non-periodic boundary conditions, theory tells that also the first moment—the location of the center of mass—should be conserved, which corresponds to conservation of momentum when considering velocity coordinates like in SeLaLib and other grid-based Vlasov solvers.

Furthermore, the direct comparison to the full grid solution in Section 5.1 showed that the mass-conserving basis functions decrease the projection error of the combination technique solution by a factor of ≈ 3 as opposed to hat functions.

Perhaps the most important feature of the introduced mass-conserving bases is that they numerically stabilize the simulation. In the Landau damping test case, cf. Section 5.2, the standard combination technique with the hat basis loses numerical stability even before the first numerical recurrence occurs. This problem is not encountered for the mass-conserving bases: When allocating the same number of DOF, the biorthogonal and full weighting combination technique solutions are very similar to the full grid solution for the time before recurrence.

Fre the plasma instability scenario discussed in Section 5.3 the hat basis leads to the simulation becoming numerically unstable after some simulation time, which even causes it to abort. By contrast, the biorthogonal and full weighting functions let the simulation run stable well into the quasi-stationary turbulent phase. This fact is particularly interesting in the context of previous work in the EXAHD project [33; 27; 29] which was severely impacted by recombination simulations with hat functions becoming numerically unstable. Those cases that were most likely to become unstable were the ones that had many grids, fine resolutions, and short recombination intervals [29]. The results of the plasma scenarios in Sections 5.2 and 5.3 show that the mass-conserving basis functions solve this problem, at least for the range of resolutions that was tested. It is worth noting that, contrary to previous numerical experiments directly in the hierarchical hat basis [22], no additional (unphysical) damping occurs in the two-stream instability scenario. Further investigations are required to apply the approach to other grid-based solvers, in particular solvers that do not conserve any moments of the distribution function.

The time-based comparison of mass and potential energy in Section 5.3 suggests that the conservation of mass and the increased stability of the basis are closely linked: both result from the increased regularity of the biorthogonal wavelets $\tilde{\psi}$ [6]. Analytical results obtained by Koster [24]

had predicted insufficient numerical stability for some PDE computations directly in the hierarchical hat basis sparse grid representation. The same effect appears to hold in the SeLaLib simulations presented here, although the time-stepping does not take place in the hierarchical basis, but in the nodal representation on each component grid individually. Only the numerical "synchronization" is carried out in the hierarchical basis, yet this is enough to skew the component solutions in a way that makes the simulation unstable over time.

Thus far, we observed no reason to prefer one of the two mass-conserving multiscale bases over the other. The 2D errors plotted in Section 5.1 illustrate that there is a slight difference in the shape of the error introduced, although it is very small and not distinguishable in the measured Monte Carlo error plots. Nevertheless, the long-term behavior in Figure 9 shows that these ever-so-slight discrepancies lead to differences in the time trajectory of the system. At this point, it is not obvious whether one of $\psi^{\mathtt{bo}}$ and $\psi^{\mathtt{fw}}$ should be preferred over the other for combination technique simulations, but they both clearly outperform $\psi^{\mathtt{hat}}$ in terms of conservation of mass, accuracy, and stability of the solution.

7 Benefits and potential of mass-conserving basis functions for combination technique simulations

From our numerical experiments, it can be concluded that the newly introduced full weighting basis and the biorthogonal basis positively influence the numerical properties of the combination technique applied to high-dimensional advection problems compared to the state-of-the-art hat functions.

It is worth noting that the kinetic simulations performed in Section 5 use relatively simple, "standard" combination schemes. By employing dimensionally [11; 37] or spatially [30] adaptive schemes, the accuracy could be improved further for the same number of degrees of freedom. This applies to both selecting the initial set of component grid resolutions as well as automatically updating the scheme during the course of a simulation, in analogy to the adaptivity in the hierarchical bases as described in [16]. Furthermore, the benefit of mass conservation is not limited to hyperrectangles. As long as the domain transform's Jacobian can be formulated in terms of a tensor product (such as for cylindrical coordinates), a straightforward transformation of the basis functions is possible.

And even for lower-dimensional simulations, it can be sensible to employ the combination technique with $\psi^{\mathtt{bo}}$ and $\psi^{\mathtt{fw}}$: If the combination technique does not introduce any additional numerical instabilities, as is possible with the basis functions introduced in this work, it can be a viable option for finely resolved simulations on 2D or 3D structured grids.

The computational overhead introduced by the mass-conserving hierarchization and dehierarchization operations affects the additional parallelism provided by the combination technique only to a small extent [35]. This fact allows to revisit previous attempts at exa-scale combination technique plasma scenarios [31; 19] with the newly introduced basis functions. Even though codes such as GENE do not generally conserve mass, the increased accuracy and stability properties should be beneficial for the simulation of experimental nuclear fusion devices. This includes results for algorithm-based fault tolerance in the presence of hard and soft faults [31] as well as the distribution across compute systems [34].

Furthermore, the concept of biorthogonal basis functions shows a way of extending the conservation to higher-order moments of the combined function, such as velocity moments of the Vlasov distribution function. To this end, it is necessary to use higher-order ansatz functions in the simulations, such as a B-spline basis [40]. Then, longer filters $h, \tilde{h}, g, \tilde{g}$ can be applied to retain more moments [6] of the combined function.

All of these properties—stability, conservation, adaptivity, fault tolerance, and parallelizability—are highly desirable for predictive plasma turbulence simulations at scale. For grid-based solvers, the combination technique offers a black-box approach to break the curse of dimensionality, without the complexity of involved multiscale sparse grid solver schemes. In this way, the mass-

conserving hierarchical basis functions will hopefully foster the usage of the combination technique in plasma (and other) simulations.

Reproducibility references

The simulations were generated using the codes DisCoTec (`https://github.com/SGpp/DisCoTec`) and SeLaLib (`https://github.com/selalib/selalib`). The exact build instructions, parameters and outputs used for this work are made available at `https://doi.org/10.18419/darus-2790`.

Acknowledgments

The authors would like to thank Michael Obersteiner for his help in coupling the codes DisCoTec and SeLaLib and for fruitful discussions. Discussions with Tilman Dannert in an early phase of the project are gratefully acknowledged. Furthermore, the authors thank Michael Griebel for proofreading and helpful comments on the numerics of the combination technique.

References

[1] Thomas Bellebaum. Evaluation of different time-synchronization schemes for the combination technique. Master's thesis, Technische Universität München, München, 2018. URL `https://mediatum.ub.tum.de/1456382`.

[2] Nicolas Besse, Guillaume Latu, Alain Ghizzo, Eric Sonnendrücker, and Pierre Bertrand. A wavelet-MRA-based adaptive semi-Lagrangian method for the relativistic Vlasov–Maxwell system. *Journal of Computational Physics*, 227(16):7889–7916, 2008. ISSN 0021-9991. doi: 10.1016/j.jcp.2008.04.031. URL `https://www.sciencedirect.com/science/article/pii/S0021999108002672`.

[3] Hans-Joachim Bungartz and Michael Griebel. Sparse Grids. *Acta Numerica*, 13:147–269, 2004.

[4] Chiozong Cheng and Georg Knorr. The integration of the Vlasov equation in configuration space. *Journal of Computational Physics*, 22(3):330–351, 1976. ISSN 0021-9991. doi: 10.1016/0021-9991(76)90053-X. URL `http://www.sciencedirect.com/science/article/pii/002199917690053X`.

[5] A. Cohen. *Numerical Analysis of Wavelet Methods*. ISSN. Elsevier Science, 2003. ISBN 9780080537856. URL `https://books.google.at/books?id=Dz9RnDItrAYC`.

[6] Albert Cohen, Ingrid Daubechies, and Jean-Christophe Feauveau. Biorthogonal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, 45(5):485–560, 1992. ISSN 1097-0312. doi: 10.1002/cpa.3160450502. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/cpa.3160450502`.

[7] Ingrid Daubechies. Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, 41(7):909–996, 1988. doi: https://doi.org/10.1002/cpa.3160410705. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/cpa.3160410705`.

[8] Erwan Deriaz and Sébastien Peirani. Six-dimensional adaptive simulation of the Vlasov equations using a hierarchical basis. *Multiscale Modeling & Simulation*, 16(2):583–614, 2018. ISSN 1540-3459. doi: 10.1137/16M1108649. URL `https://epubs.siam.org/doi/abs/10.1137/16M1108649`. Publisher: Society for Industrial and Applied Mathematics.

[9] Erwan Deriaz and Valérie Perrier. Divergence-free and curl-free wavelets in two dimensions and three dimensions: application to turbulent flows. *Journal of Turbulence*, 7:N3, 2006. doi: 10.1080/14685240500260547. URL https://doi.org/10.1080/14685240500260547.

[10] Gilles Deslauriers and Serge Dubuc. Symmetric iterative interpolation processes. In Ronald A. DeVore and Edward B. Saff, editors, *Constructive Approximation: Special Issue: Fractal Approximation*, Constructive Approximation, pages 49–68. Springer US, 1989. ISBN 978-1-4899-6886-9. doi: 10.1007/978-1-4899-6886-9_3. URL https://doi.org/10.1007/978-1-4899-6886-9_3.

[11] Thomas Gerstner and Michael Griebel. Dimension–adaptive tensor–product quadrature. *Computing*, 71(1):65–87, 2003. doi: 10.1007/s00607-003-0015-5.

[12] M. Griebel and H. Harbrecht. On the construction of sparse tensor product spaces. *Mathematics of Computations*, 82(282):975–994, apr 2013.

[13] M. Griebel and H. Harbrecht. On the convergence of the combination technique. In *Sparse grids and Applications*, volume 97 of *Lecture Notes in Computational Science and Engineering*, pages 55–74. Springer, 2014.

[14] M. Griebel, M. Schneider, and C. Zenger. A combination technique for the solution of sparse grid problems. In P. de Groen and R. Beauwens, editors, *Iterative Methods in Linear Algebra*, pages 263–281. IMACS, Elsevier, North Holland, 1992.

[15] Michael Griebel and Frank Koster. Multiscale methods for the simulation of turbulent flows. In Ernst Heinrich Hirschel, editor, *Numerical Flow Simulation III*, Notes on Numerical Fluid Mechanics and Multidisciplinary Design (NNFM), pages 203–214. Springer, 2003. ISBN 978-3-540-45693-3. doi: 10.1007/978-3-540-45693-3_13.

[16] Wei Guo and Yingda Cheng. An adaptive multiresolution discontinuous Galerkin method for time-dependent transport equations in multidimensions. *SIAM Journal on Scientific Computing*, 39(6):A2962–A2992, 2017. ISSN 1064-8275. doi: 10.1137/16M1083190.

[17] Matthieu Haefele, Guillaume Latu, and Michaël Gutnic. A parallel Vlasov solver using a wavelet based adaptive mesh refinement. In *2005 International Conference on Parallel Processing Workshops (ICPPW'05)*, pages 181–188, 2005. doi: 10.1109/ICPPW.2005.13. ISSN: 2332-5690.

[18] Brendan Harding. Adaptive sparse grids and extrapolation techniques. In Jochen Garcke and Dirk Pflüger, editors, *Sparse Grids and Applications - Stuttgart 2014*, Lecture Notes in Computational Science and Engineering, pages 79–102. Springer International Publishing, 2016. ISBN 978-3-319-28262-6. doi: 10.1007/978-3-319-28262-6_4.

[19] Mario Heene and Dirk Pflüger. Efficient and scalable distributed-memory hierarchization algorithms for the sparse grid combination technique. *Parallel Computing: On the Road to Exascale*, pages 339–348, 2016. doi: 10.3233/978-1-61499-621-7-339. URL https://ebooks.iospress.nl/doi/10.3233/978-1-61499-621-7-339. Publisher: IOS Press.

[20] Walter Huber. *Turbulenzsimulation mit der Kombinationsmethode auf Workstation-Netzen und Parallelrechnern*. Herbert Utz Verlag, 1996. ISBN 978-3-89675-139-3. URL https://www5.in.tum.de/publikat/diss/huberw.ps.gz.

[21] Frank Jenko, Daniel Told, Tobias Görler, Jonathan Citrin, Alejandro Bañón Navarro, Clarisse Bourdelle, Stephan Brunner, Garrard D Conway, Tilman Dannert, Hauke Doerk, David R Hatch, J. W. (Willem) Haverkort, Joerg Hobirk, G. M. D. Hogeweij, Paola Mantica, M. J. Pueschel, Olivier Sauter, Laurent Villard, and Elisabeth Wolfrum. Global and local gyrokinetic simulations of high-performance discharges in view of ITER. *Nuclear Fusion*, 53(7):073003, 2013. ISSN 0029-5515. doi: 10.1088/0029-5515/53/7/073003. URL https://doi.org/10.1088%2F0029-5515%2F53%2F7%2F073003.

[22] Katharina Kormann and Eric Sonnendrücker. Sparse grids for the Vlasov–Poisson equation. In Jochen Garcke and Dirk Pflüger, editors, *Sparse Grids and Applications - Stuttgart 2014*, Lecture Notes in Computational Science and Engineering, pages 163–190. Springer International Publishing, 2016. ISBN 978-3-319-28262-6.

[23] Katharina Kormann, Klaus Reuter, and Markus Rampp. A massively parallel semi-Lagrangian solver for the six-dimensional Vlasov-Poisson equation. *The International Journal of High Performance Computing Applications*, 2019. ISSN 1094-3420. doi: 10.1177/1094342019834644. URL https://doi.org/10.1177/1094342019834644.

[24] Frank Koster. *Multiskalen-basierte Finite-Differenzen-Verfahren auf adaptiven dünnen Gittern*. Thesis, Universität Bonn, 2002. URL https://bonndoc.ulb.uni-bonn.de/xmlui/handle/20.500.11811/1696.

[25] Christoph Kowitz and Markus Hegland. The sparse grid combination technique for computing eigenvalues in linear gyrokinetics. *Procedia Computer Science*, 18(0):449–458, 2013. ISSN 1877-0509. doi: 10.1016/j.procs.2013.05.208. URL http://www.sciencedirect.com/science/article/pii/S1877050913003517. 2013 International Conference on Computational Science.

[26] Christoph Kowitz, Dirk Pflüger, Frank Jenko, and Markus Hegland. The combination technique for the initial value problem in linear gyrokinetics. In Jochen Garcke and Michael Griebel, editors, *Sparse Grids and Applications*, Lecture Notes in Computational Science and Engineering, pages 205–222. Springer, 2013. ISBN 978-3-642-31703-3. doi: 10.1007/978-3-642-31703-3_10.

[27] Rafael Lago, Michael Obersteiner, Theresa Pollinger, Johannes Rentrop, Hans-Joachim Bungartz, Tilman Dannert, Michael Griebel, Frank Jenko, and Dirk Pflüger. EXAHD: A massively parallel fault tolerant sparse grid approach for high-dimensional turbulent plasma simulations. In *Software for Exascale Computing - SPPEXA 2016-2019*, pages 301–329. Springer, 2020. ISBN 978-3-030-47955-8. doi: 10.1007/978-3-030-47956-5_11.

[28] Boris Lastdrager, Barry Koren, and Jan Verwer. The sparse-grid combination technique applied to time-dependent advection problems. *Applied Numerical Mathematics*, 38(4):377–401, September 2001. ISSN 0168-9274. doi: 10.1016/s0168-9274(01)00030-7.

[29] Michael Obersteiner. *A spatially adaptive and massively parallel implementation of the fault-tolerant combination technique*. Dissertation, Technische Universität München, 2021. URL https://mediatum.ub.tum.de/doc/1613369/1613369.pdf.

[30] Michael Obersteiner and Hans-Joachim Bungartz. A generalized spatially adaptive sparse grid combination technique with dimension-wise refinement. *SIAM Journal on Scientific Computing*, 43(4):A2381–A2403, 2021. ISSN 1064-8275. doi: 10.1137/20M1325885. URL https://epubs.siam.org/doi/abs/10.1137/20M1325885. Publisher: Society for Industrial and Applied Mathematics.

[31] Michael Obersteiner, Alfredo Parra Hinojosa, Mario Heene, Hans-Joachim Bungartz, and Dirk Pflüger. A highly scalable, algorithm-based fault-tolerant solver for gyrokinetic plasma simulations. In *Proceedings of the 8th Workshop on Latest Advances in Scalable Algorithms for Large-Scale Systems*, pages 1–8, 2017.

[32] Dirk Pflüger. *Spatially Adaptive Sparse Grids for High-Dimensional Problems*. Verlag Dr. Hut, 2010.

[33] Dirk Pflüger, Hans-Joachim Bungartz, Michael Griebel, Frank Jenko, Tilman Dannert, Mario Heene, Christoph Kowitz, Alfredo Parra Hinojosa, and Peter Zaspel. EXAHD: An exascalable two-level sparse grid approach for higher-dimensional problems in plasma physics and beyond. In *Euro-Par 2014: Parallel Processing Workshops*, volume 8806 of *Lecture Notes in Computer Science*, pages 565–576. Springer International Publishing, 2014.

[34] Theresa Pollinger, Marcel Hurler, Michael Obersteiner, and Dirk Pflüger. Distributing higher-dimensional simulations across compute systems: A widely distributed combination technique. In *2021 IEEE/ACM International Workshop on Hierarchical Parallelism for Exascale Computing (HiPar)*, pages 1–9, 2021. doi: 10.1109/HiPar54615.2021.00006.

[35] Theresa Pollinger, Katharina Kormann, and Dirk Pflüger. Scaling the plasma simulation while conserving the mass: A massively-parallel semi-Lagrangian solver with the sparse grid combination technique. Poster, 2022. URL `https://pasc22.pasc-conference.org/program/schedule/presentation/?id=pos109&sess=sess181`.

[36] Christoph Reisinger. Analysis of linear difference schemes in the sparse grid combination technique. *Ima Journal of Numerical Analysis*, 33:544–581, 2007.

[37] Alexander Rüttgers and Michael Griebel. Multiscale simulation of polymeric fluids using the sparse grid combination technique. *Applied Mathematics and Computation*, 319:425–443, 2018. ISSN 0096-3003. doi: 10.1016/j.amc.2017.04.025. URL `https://www.sciencedirect.com/science/article/pii/S0096300317302771`.

[38] Wim Sweldens. The lifting scheme: A construction of second generation wavelets. *SIAM Journal on Mathematical Analysis*, 29(2):511–546, 1998. ISSN 0036-1410. doi: 10.1137/S0036141095289051. URL `https://epubs.siam.org/doi/abs/10.1137/S0036141095289051`.

[39] Zhanjing Tao, Wei Guo, and Yingda Cheng. Sparse grid discontinuous Galerkin methods for the Vlasov-Maxwell system. *Journal of Computational Physics: X*, 3:100022, 2019. doi: 10.1016/j.jcpx.2019.100022.

[40] Julian Valentin. *B-Splines for Sparse Grids: Algorithms and Application to Higher-Dimensional Optimization*. Dissertation, Universität Stuttgart, 2019.