Sparse grid time-discontinuous Galerkin method with streamline diffusion for transport equations

Andreas Zeiser

Received: 19 September 2022 / Accepted: 20 June 2023 / Published online: 1 August 2023
© The Author(s) 2023

Abstract
High-dimensional transport equations frequently occur in science and engineering. Computing their numerical solution, however, is challenging due to its high dimensionality. In this work we develop an algorithm to efficiently solve the transport equation in moderately complex geometrical domains using a Galerkin method stabilized by streamline diffusion. The ansatz spaces are a tensor product of a sparse grid in space and discontinuous piecewise polynomials in time. Here, the sparse grid is constructed upon nested multilevel finite element spaces to provide geometric flexibility. This results in an implicit time-stepping scheme which we prove to be stable and convergent. If the solution has additional mixed regularity, the convergence of a $2d$-dimensional problem equals that of a $d$-dimensional one up to logarithmic factors. For the implementation, we rely on the representation of sparse grids as a sum of anisotropic full grid spaces. This enables us to store the functions and to carry out the computations on a sequence regular full grids exploiting the tensor product structure of the ansatz spaces. In this way existing finite element libraries and GPU acceleration can be used. The combination technique is used as a preconditioner for an iterative scheme to solve the transport equation on the sequence of time strips. Numerical tests show that the method works well for problems in up to six dimensions. Finally, the method is also used as a building block to solve nonlinear Vlasov-Poisson equations.

Keywords Sparse grid · High-dimensional transport equations · Streamline diffusion · Combination technique · Vlasov-Poisson equations

Mathematics Subject Classification 65M60 · 65M12

This article is part of the section “Computational Approaches” edited by Siddhartha Mishra.

Andreas Zeiser
andreas.zeiser@htw-berlin.de

1 Fachbereich 1 Ingenieurwissenschaften – Energie und Information, HTW Berlin, Wilhelminenhofstr.
75A, Berlin 12459, Germany
1 Introduction

High-dimensional transport equations frequently arise in science and engineering. In this paper we will consider the linear equation

$$\partial_t u + \beta(t, r) \cdot \nabla r u + \sigma(t, r) u = f(t, r), \quad \text{in } (0, T) \times \Omega$$

subject to the inflow boundary condition

$$u = g \quad \text{on } \Gamma^- = \{(t, r) \in (0, T) \times \partial \Omega \mid \beta(t, r) \cdot n(r) < 0\}$$

where $n(r)$ is the outward unit normal of the boundary, and $\beta, f, \sigma, g, u_0$ are smooth enough functions. We will focus on the case where the variable $r$ can be partitioned into $r = (x, v)$ and $x, v$ are $d$-dimensional variables with $d = 1, 2, 3$ (i.e., space and velocity). We assume that $\Omega = \Omega^{(1)} \times \Omega^{(2)}$ is a product domain with $\Omega^{(1)}, \Omega^{(2)} \subset \mathbb{R}^d$, i.e.

$$u : (0, T) \times \Omega^{(1)} \times \Omega^{(2)} \rightarrow \mathbb{R}$$

is a function in time and two variables with dimension $d$.

Simulations of such systems are computationally demanding since the time evolution of an up to six-dimensional function has to be calculated. Applying standard discretization schemes leads to an evolution equation in $O(n^{2d})$ degrees of freedom, where $n$ is the number of grid points in one dimension and a corresponding high computational effort. To tackle the problem, methods such as particle methods [1], adaptive multiscale methods [2, 3] and tensor product methods [4] have been used.

Sparse grids [5] are a means of overcoming the curse of dimensionality and have been applied to transport equations [6] as well as to kinetic equations: in [7, 8] interpolation on sparse grids is used in a semi-Lagrangian method while [9] uses sparse grids with discontinuous ansatz functions. In these approaches tensor products of one-dimensional bases are used, restricting the domain to rectangular regions. However, finite element spaces can also be used in the construction of sparse grid spaces [10, 11] allowing for greater geometric flexibility.

The related problem of radiative transfer has been studied widely. In [12] standard finite element discretization has been used to solve the stabilized equation. To cope with the high dimensionality, sparse grids of finite element spaces [13], sparse tensor product of multilevel finite element spaces and spherical harmonics [14] as well as a corresponding sparse grid combination technique [15] have been used. More recently adaptive schemes are applied to solve the parameter dependent transport equation [16, 17].

Our aim is to use sparse grid spaces based upon finite element spaces to devise a stable, convergent and efficient method for the solution of kinetic equations in moderately complex geometrical domains. We also focus on their efficient implementation using well-established finite element libraries and fast algorithms.

For the discretization of space continuous piecewise polynomial finite element spaces for each domain $\Omega^{(i)}$ are used. Compared to discontinuous elements less degrees of freedom are needed to represent these functions. This is especially important as this factor enters quadratically in the degrees of freedom of the sparse grid space.

The time domain is partitioned and on each time strip space-time elements are constructed by tensorizing polynomials in time with sparse grids in space. Using this space in a Galerkin method results in transport equations on each time strip, where the result of the previous

\[ \text{Springer} \]
step enters the current step as an initial condition. Enforcing this condition weakly leads to discontinuous functions in time.

However, it is known that the finite element method is unstable if the solution has discontinuities and hence has to be stabilized. One option is to use the streamline diffusion method [18]. There, an artificial diffusion is added in the direction of the streamline leading to a stable scheme, while still keeping the order of the method.

This method can be interpreted as an implicit time-stepping scheme, where stabilized transport equations have to be solved on each time strip. For that purpose matrix-free iterative methods are applied. For an efficient calculation the sparse grid functions are represented as a linear combination of functions in anisotropic full grid spaces, which is also used in the combination technique (see [19] and references therein). This leads to regular data structures and fast applications of the discretization matrices by exploiting the tensor product structure. Finally, the combination technique will serve as a preconditioner for the iterative solution of the corresponding equations [20].

We will proceed along the following lines. In Sect. 2 sparse grid spaces are introduced, and the weak form, including streamline diffusion, is employed to derive discrete equations for each time step. This algorithm is analyzed in Sect. 3 in the case of constant coefficients with respect to stability and convergence. Section 4 describes the algorithm for efficient computations based on the tensor product structure and multilevel spaces. The algorithm is applied to test cases in Sect. 5 as well as to Vlasov-Poisson equations in one and two spatial dimensions.

In the following we denote by $C \lesssim D$ that $C$ is bounded by a multiple of $D$ independently of the level of the sparse grid. $C \sim D$ is defined as $C \lesssim D$ and $D \lesssim C$.

2 Discretization and weak formulation

In this section we introduce the weak formulation of the transport problem including streamline diffusion [18] for stabilization. For the discretization we will use a sparse grid based on finite elements and discontinuous polynomials in time. Finite element spaces have been used in the construction of sparse grids, realized either by wavelet-type [13, 21] or multilevel frames [11]. We will represent sparse grids as in [22]. This enables us to use classical nested finite element spaces. We partition the time domain, use discontinuous piecewise polynomials and a full tensor product with the sparse grid, see [23] in the case of parabolic equations. Eventually this will lead us to an implicit time-stepping scheme.

Let $\Omega^{(1)}$ and $\Omega^{(2)}$ be two open polyhedral Lipschitz domains in $d$ dimensions and $V^{(i)}$ conforming finite element spaces on $\Omega^{(i)}$ of piecewise polynomial ansatz functions. Define nested multilevel finite element spaces

$$V_{\ell}^{(i)} \subset V^{(i)}_2 \subset \ldots \subset H^1(\Omega^{(i)}), \quad i = 1, 2$$

by uniform refinement such that $\dim V_{\ell}^{(i)} \sim 2^{d\ell}$. Based on these spaces the sparse tensor product space of level $L = 1, 2, \ldots$

$$V_L = \sum_{|\ell_1| = L+1} V_{\ell_1}^{(1)} \otimes V_{\ell_2}^{(2)}, \quad \ell = [\ell_1, \ell_2], \quad |\ell|_1 = \ell_1 + \ell_2$$

is constructed with $\dim V_L \lesssim L^{2dL}$ [22]. Up to a logarithmic factor the number of degrees of freedom scales like a standard globally refined finite element discretization of a $d$-dimensional domain.
In the time domain $J = (0, T)$ define $0 = t_0 < t_1 < \cdots < t_N = T$ and the partition
\[ \mathcal{I} = \{ I_j \}_{j=1}^{N}, \quad I_j = (t_{j-1}, t_j), \quad |\mathcal{I}| = \max_{j=1}^{N} (t_j - t_{j-1}). \]

The discrete function space is defined as the set of all piecewise polynomials of order $r$ with coefficients in $V_L$, i.e.
\[ V_{L, \mathcal{I}} = \{ v \in L^2(J; V_L) \mid v|_{I_j} \in \mathbb{P}^r(I; V_L) \text{ for all } I_j \in \mathcal{I} \}, \]
where $\mathbb{P}^r$ denotes the polynomials degree $r$.

We will use the space $V_{L, \mathcal{I}}$ in a Galerkin scheme to compute an approximate solution of (1) and (2).

For that purpose, we use the streamline diffusion method on each time strip $I_j$ [18]. The initial condition of (1) and the inflow condition (2) are enforced in a weak sense:

For $j = 1, 2, \ldots, N$ find a function $U_j \in \mathbb{P}^r(I_j, V_L)$ on the time strip $I_j \in \mathcal{I}$, $j = 1, \ldots, N$, such that
\[ a_j^{(\delta)}(U_j, w) = b_j^{(\delta)}(w), \quad \text{for all } w \in \mathbb{P}^r(I_j, V_L) \quad (4) \]
where $\delta > 0$ is the parameter for the streamline diffusion and
\[ a_j^{(\delta)}(v, w) = \int_{I_j} (\partial_t v + \partial_\beta v + \sigma v, w + \delta (\partial_t w + \partial_\beta w)) - \langle v, w \rangle_{\Gamma^-} \, dt + (v^+_{j-1}, w^+_{j-1}) \]
\[ b_j^{(\delta)}(w) = \int_{I_j} (f, w + \delta (\partial_t w + \partial_\beta w)) - \langle g, w \rangle_{\Gamma^-} \, dt + (U^-_{j-1}, w^+_{j-1}). \]

Here $\partial_\beta u = \beta \cdot \nabla u$, $u^+_{j} = \lim_{\tau \to I_j^+} u(t, \cdot)$, $U^-_0 = u_0$. By $(\cdot, \cdot)$ and $\| \cdot \|$ denote the inner product and norm on $L^2(\Omega)$, respectively. For the boundary define
\[ \hat{\Gamma}^\pm(t) = \{ \mathbf{r} \in \partial \Omega \mid \beta(t, \mathbf{r}) \cdot \mathbf{n}(\mathbf{r}) \geq 0 \} \]
and for any $\hat{\Gamma} \subset \partial \Omega$
\[ \langle u, v \rangle_{\hat{\Gamma}} = \int_{\hat{\Gamma}} (\beta \cdot n) \, uv \, ds, \quad |u|^2_{\hat{\Gamma}} = \int_{\hat{\Gamma}} |\beta \cdot n| uv^2 \, ds. \]

Equation (4) can be solved successively, resulting in an implicit time-stepping procedure. Finally, the approximate solution $U \in V_{L, \mathcal{I}}$ is composed of the solutions $U_j$ with potential jumps at $t_j$.

### 3 Analysis—constant coefficients

In this section the stability and convergence of the discretization is analyzed for the case of constant coefficients $\beta$ and $\sigma$. Hence, for the rest of this section we assume $\beta \in \mathbb{R}^{2d}$, $\sigma \geq 0$, $f \in L^2(J \times \Omega)$ and $g \in L^2(J \times \hat{\Gamma}^-)$. A similar analysis for stationary transport dominated problems has been carried out in [6] and for parabolic equations in [23]. For an introduction to discontinuous time-stepping schemes and finite element spaces see [24].

However, these results have to be adapted to the current setting. With respect to stability, the weak formulation of the boundary condition has to be taken into account. To show convergence, classical approximation results for piecewise polynomial ansatz functions have to be combined with sparse grid estimates to handle the tensor product ansatz functions.
For the analysis we formulate the problem on the space $V_{L,I}$ by summing up (4) for $j = 1, \ldots, N$. This gives

$$A_I^{(δ)}(U, w) = B_I^{(δ)}(u_0, f, g; w) \text{ for all } w \in V_{L,I},$$

(5)

where

$$A_I^{(δ)}(v, w) = \sum_{j=1}^{N} \int_{I_j} \left( \partial_t v + \partial_\beta v + \sigma v, w + \delta(\partial_t w + \partial_\beta w) \right) - (v, w)_{\hat{T}^-} \, dt$$

$$+ \sum_{j=1}^{N-1} \left( [v]_j, w_j^+ \right) + \left( v_0^+, w_0^+ \right)$$

(6)

$$B_I^{(δ)}(u_0, f, g; w) = \sum_{j=1}^{N} \int_{I_j} \left( f, w + \delta(\partial_t w + \partial_\beta w) \right) - (g, w)_{\hat{T}^-} \, dt + \left( u_0, w_0^+ \right)$$

(7)

with the jump term $[v]_j = v_j^+ - v_j^-$. Note that the boundaries $\hat{T}^{\pm}$ are now time independent.

For the analysis the streamline diffusion norm

$$\|v\|_δ^2 = \sum_{j=1}^{N} \int_{I_j} \sigma \|v\|^2 + \delta \|\partial_t v + \partial_\beta v\|^2 + |v|^2_{\partial\Omega} \, dt$$

$$+ \sum_{j=1}^{N-1} \|\left[ v \right]_j \|^2 + \|v_0^+\|^2 + \|v_N^-\|^2$$

(8)

will play a central role. This norm gives extra control of the variations in the direction of the streamline.

3.1 Stability

In the first step we show that the method is stable with respect to the streamline diffusion norm. The following equivalence plays a central role.

Lemma 1 Let $v \in H^1(J \times \Omega), \sigma \geq 0, 0 < \delta$ and $\delta < 1/\sigma$ in the case $\sigma > 0$. Then

$$\frac{1}{2} \|v\|^2_δ \leq A_I^{(δ)}(v, v) \leq \frac{3}{2} \|v\|^2_δ.$$

(9)

Proof Applying Green’s identity

$$\int_{I_j} (\partial_t v, w) \, dt = (v_j^-, w_j^-) - (v_{j-1}^+, w_{j-1}^+) - \int_{I_j} (v, \partial_t w) \, dt$$

$$\int_{I_j} (\partial_\beta v, w) \, dt = \int_{I_j} (v, \partial_\beta w) \, dt - (v, \partial_\Omega w) \, dt$$
on (6) gives
\[
A^{(d)} \left( v, w \right) = \sum_{j=1}^{N} \int_{I_j} \left( -v, \partial_t w + \partial_\beta w \right) + (\sigma v, w + \delta (\partial_t w + \partial_\beta w)) \, dt
\]
\[
+ \sum_{j=1}^{N} \int_{I_j} \delta (\partial_t v + \partial_\beta v, \partial_t w + \partial_\beta w) + \langle v, w \rangle_{\Gamma_+} \, dt
\]
\[
+ \sum_{j=1}^{N-1} (v_j^-, -w_j^+) + (v_N^-, w_N^-). \tag{10}
\]
Averaging the two forms for \( A^{(d)} \left( v, v \right) \) gives
\[
A^{(d)} \left( v, v \right) = \sum_{j=1}^{N} \int_{I_j} \left( 1 + \frac{1}{2} \| v \|_{\Omega}^2 + \frac{1}{2} \| \partial_t v + \partial_\beta v \|_{\Omega}^2 \right) \, dt
\]
\[
+ \sum_{j=1}^{N} \int_{I_j} \delta (\partial_t v + \partial_\beta v, \partial_t v + \partial_\beta v) \, dt + \frac{1}{2} \sum_{j=1}^{N} \| [v]_j \|_{\Gamma_1}^2 + \frac{1}{2} \| v_0^+ \|_{\Gamma_1}^2 + \frac{1}{2} \| v_N^- \|_{\Gamma_1}^2. \tag{11}
\]
For \( \sigma = 0 \)
\[
A^{(d)} \left( v, v \right) = \sum_{j=1}^{N} \int_{I_j} \frac{1}{2} \| v \|_{\Omega}^2 + \delta \| \partial_t v + \partial_\beta v \|_{\Omega}^2 \, dt + \frac{1}{2} \sum_{j=1}^{N} \| [v]_j \|_{\Gamma_1}^2 + \frac{1}{2} \| v_0^+ \|_{\Gamma_1}^2 + \frac{1}{2} \| v_N^- \|_{\Gamma_1}^2,
\]
hence (9) holds. In the case \( \sigma > 0 \) the second term can be estimated by
\[
\left| \sum_{j=1}^{N} \int_{I_j} \delta (\partial_t v + \partial_\beta v) \, dt \right| \leq \sum_{j=1}^{N} \int_{I_j} \frac{\delta \sigma^2}{2} \| v \|_{\Omega}^2 + \frac{\delta}{2} \| \partial_t v + \partial_\beta v \|_{\Omega}^2 \, dt.
\]
Using \( \delta < 1/\sigma \), the equivalence follows directly using the definition of the streamline diffusion norm (8).

Now we use the Lemma to prove stability.

**Theorem 2** If \( \delta > 0 \) and \( \delta < 1/\sigma \) in the case \( \sigma > 0 \), the weak form of the streamline diffusion equation with constant coefficients (5) has a unique solution \( U \). If \( \sigma > 0 \) the system is stable in the sense that the solution satisfies
\[
\| U \|^2_{\delta} \lesssim \sum_{j=1}^{N} \int_{I_j} \| f \|^2 + |g|^2_{\Gamma_+} \, dt + \| u_0 \|^2.
\]
For the case \( \sigma = 0 \) a similar bound can be derived where, however, the constant depends exponentially on the final time \( T \).

**Proof** For \( \sigma > 0 \) applying Cauchy and Young’s inequality on (7) gives
\[
B^{(d)} (u_0, f, g; v)
\leq \sum_{j=1}^{N} \int_{I_j} \left( \frac{1}{\sigma} + \delta \right) \| f \|^2 + \frac{\sigma}{4} \| v \|^2 + \frac{\delta}{4} \| \partial_t v + \partial_\beta v \|^2 + |g|^2_{\Gamma_+} + \frac{1}{4} v_0^+ \|_{\Gamma_+}^2 \, dt
\]
\[\square\] Springer
\[ + \|u_0\|^2 + \frac{1}{4}\|v_0^+\|^2 \]
\[ \leq \frac{1}{2} A^{(\delta)}_L (u, v) + \sum_{j=1}^{N} \int_{I_j} \left( \frac{1}{\sigma} + \delta \right) \|f\|^2 + |g|^2_{\Gamma_+} \, dt + \|u_0\|^2, \]

where Lemma 1 was used in the last step. The solution \( U \) satisfies
\[ A^{(\delta)}(U, U) = B^{(\delta)}(u_0, f, g; U) \]
\[ \leq \frac{1}{2} A^{(\delta)}(U, U) + \sum_{j=1}^{N} \int_{I_j} \left( \frac{1}{\sigma} + \delta \right) \|f\|^2 + |g|^2_{\Gamma_+} \, dt + \|u_0\|^2. \]

Bringing \( A^{(\delta)}(U, U) \) to the left side and using (9) gives the inequality which also shows the existence and uniqueness of the solution.

For the case \( \sigma = 0 \) one can introduce a positive constant term by change of the unknown to \( w(t, \cdot) = e^{-\alpha t} u(t, \cdot) \) which leads to an exponential factor \( e^{\alpha T} \) in the bound [18].

### 3.2 Approximation

In the following we will derive error estimates for the orthogonal projection onto \( V_{L,\mathcal{I}} \), which will be used in the convergence proof in Sect. 3.3. For the result we will combine estimates for sparse grids and well known approximation results from one-dimensional finite element discretization for the time domain.

Following [22] we assume that the approximation property
\[ \inf_{v_\ell \in V_\ell} \|u - v_\ell\|_{H^s(\Omega^{(i)})} \lesssim h^{s-q}_\ell \|u\|_{H^{s-q}(\Omega^{(i)})}, \quad u \in H^s(\Omega^{(i)}), \quad i = 1, 2 \]
holds for \( q < \gamma, q \leq s \leq r + 1 \) uniformly in the level \( \ell \) (note that we replaced \( r \) by \( r + 1 \)). Here \( h_\ell = 2^{-\ell} \) and
\[ \gamma = \sup \{ s \in \mathbb{R} \mid V_\ell^{(1)} \subset H^s(\Omega^{(1)}), \ V_\ell^{(2)} \subset H^s(\Omega^{(2)}) \}. \]

Define for \( s > 0 \) the Sobolev spaces of mixed order [25]
\[ H^s_{\text{mix}}(\Omega) = H^{s,s}_{\text{mix}}(\Omega), \quad H^{s_1,s_2}_{\text{mix}}(\Omega) = H^{s_1}(\Omega^{(1)}) \otimes H^{s_2}(\Omega^{(2)}) \]
and denote the corresponding norm by \( \| \cdot \|_{H^s_{\text{mix}}} \). Then the following approximation results hold.

**Theorem 3** ([22]) Let \( 0 < s \leq r \) and denote by \( P_L \) the \( L^2 \)-orthogonal projection onto \( V_L \). Then
\[ \|u - P_L u\|_{L^2(\Omega)} \lesssim 2^{-(s+1)L} \sqrt{L} \|u\|_{H^{s+1}_{\text{mix}}(\Omega)} \]
\[ \|u - P_L u\|_{H^1(\Omega)} \lesssim 2^{-sL} \sqrt{L} \|u\|_{H^{s+1}_{\text{mix}}(\Omega)} \]
if \( u \in H^{s+1}_{\text{mix}}(\Omega) \).

**Proof** The inequalities directly follow from [22, Thm. 1] for the case \( n_1 = n_2 = d, \sigma = 1 \), and Examples 1 and 2. 

\[ \text{Springer} \]
Finally, define the space
\[ \mathcal{H}^s(J \times \Omega) = H^s(J) \otimes L^2(\Omega) \cap L^2(J) \otimes H^s_{\text{mix}}(\Omega) \]
with classical regularity with respect to time and a mixed regularity with respect to \( x \) and \( v \).

The corresponding norm reads
\[ \| u \|_{\mathcal{H}^s(J \times \Omega)}^2 = \int_0^T \| \partial_t^s u \|_{L^2(\Omega)}^2 + \| u \|_{H^s_{\text{mix}}(\Omega)}^2 \, dt. \]

With this type of regularity we can show the following approximation result.

**Theorem 4** Let \( L \in \mathbb{N} \) and let \( I \) be a time partition such that \( |I| \sim h = 2^{-L} \). For \( u \in L^2(J \times \Omega) \) define
\[ u_h = (P_I \otimes P_L) u, \]
where \( P_I \) is the \( L^2 \)-orthogonal projection onto the continuous piecewise polynomials of degree \( r \) with respect to the partition \( I \). For \( 0 < s \leq r \) and \( u \in \mathcal{H}^{s+1} \)
\[ \| u - u_h \|_{L^2(J \times \Omega)} \lesssim 2^{-(s+1)L} \sqrt{L} \| u \|_{\mathcal{H}^{s+1}(J)}, \quad \| v - P_I v \|_{H^s(J)} \lesssim |I|^s \| \partial_t^{s+1} v \|_{L^2(\Omega)} \]
for \( v \in H^{s+1}(I) \). Hence, the norm corresponding operators can be bounded by
\[ \| I - P_I \|_{H^{s+1}(J) \rightarrow L^2(\Omega)} \lesssim 2^{-(s+1)L}, \quad \| I - P_I \|_{H^s(J) \rightarrow H^1(\Omega)} \lesssim 2^{-sL}. \]

For the projectors onto \( V_L \) it follows from Theorem 3 that
\[ \| I - P_L \|_{H^{s+1}_{\text{mix}}(\Omega) \rightarrow L^2(\Omega)} \lesssim 2^{-(s+1)L} \sqrt{L}, \quad \| I - P_L \|_{H^{s+1}_{\text{mix}}(\Omega) \rightarrow H^1(\Omega)} \lesssim 2^{-sL} \sqrt{L}. \]

Now, the projection error with respect to \( L^2(J \times \Omega) \) can be estimated by
\[ \| u - u_h \|_{L^2(J \times \Omega)} \]
\[ = \| (I - P_I) \otimes P_L u + P_I \otimes (I - P_L) u + (I - P_I) \otimes (I - P_L) u \|_{L^2(J \times \Omega)} \]
\[ \leq \| I - P_I \|_{H^{s+1}(J) \rightarrow L^2(\Omega)} \| P_L \|_{L^2(\Omega) \rightarrow L^2(\Omega)} \| u \|_{H^{s+1}(J) \otimes L^2(\Omega)} \]
\[ + \| P_I \|_{L^2(\Omega) \rightarrow L^2(\Omega)} \| I - P_L \|_{H^{s+1}_{\text{mix}}(\Omega) \rightarrow L^2(\Omega)} \| u \|_{L^2(\Omega) \otimes H^{s+1}_{\text{mix}}(\Omega)} \]
\[ + \| I - P_I \|_{H^{s+1}(J) \rightarrow L^2(\Omega)} \| I - P_L \|_{L^2(\Omega) \rightarrow L^2(\Omega)} \| u \|_{H^{s+1}(J) \otimes L^2(\Omega)}. \]

Using the bounds of the projection operators and the fact that
\[ \| P_I \|_{L^2(\Omega) \rightarrow L^2(\Omega)}, \quad \| P_L \|_{L^2(\Omega) \rightarrow L^2(\Omega)}, \quad \| I - P_L \|_{L^2(\Omega) \rightarrow L^2(\Omega)} \lesssim 1 \]
due to the orthogonality of the projectors it follows that
\[ \| u - u_h \|_{L^2(J \times \Omega)} \lesssim 2^{-(s+1)L} \| u \|_{H^{s+1}(J) \otimes L^2(\Omega)} + 2^{-(s+1)L} \sqrt{L} \| u \|_{L^2(J) \otimes H^{s+1}_{\text{mix}}(\Omega)} \]
\[ + 2^{-(s+1)L} \| u \|_{H^{s+1}(J) \otimes L^2(\Omega)} \]
\[
\lesssim 2^{-(s+1)L} \sqrt{L} \|u\|_{\mathcal{H}^{L+1}}.
\]

For the second inequality of the theorem note that
\[
H^1(J \times \Omega) = H^1(J) \otimes L^2(\Omega) \cap L^2(J) \otimes H^1_{\text{mix}}(\Omega) \cap L^2(J) \otimes H^0_{\text{mix}}(\Omega)
\]
We estimate each contribution like in the previous case:
\[
\|u - u_h\|_{H^1(J) \otimes L^2(\Omega)} \lesssim \|I - P_I\|_{H^{s+1}(J) \to H^1(J)} \|P_L\|_{L^2(\Omega) \to L^2(\Omega)} \|u\|_{H^{s+1}(J) \otimes L^2(\Omega)}
\]
\[
+ \|P_I\|_{L^2(\Omega) \to H^1(J)} \|I - P_L\|_{H^1(\Omega) \otimes H^s_{\text{mix}}(\Omega)} \|u\|_{L^2(J) \otimes H^s_{\text{mix}}(\Omega)}
\]
\[
+ \|I - P_I\|_{H^{s+1}(J) \to H^1(J)} \|I - P_L\|_{L^2(\Omega) \to L^2(\Omega)} \|u\|_{H^{s+1}(J) \otimes L^2(\Omega)}
\]
and use the inverse inequality of \(P_I\) to bound \(\|P_I\|_{L^2(\Omega) \to H^1(J)} \lesssim |I|^{-1} \sim 2^L\) which leads to
\[
\|u - u_h\|_{H^1(J) \otimes L^2(\Omega)} \lesssim 2^{-sL} \sqrt{L} \|u\|_{\mathcal{H}^{L+1}}.
\]

In the same manner
\[
\|u - u_h\|_{L^2(\Omega) \otimes H^0_{\text{mix}}(\Omega)} \lesssim \|I - P_I\|_{H^{s+1}(J) \to L^2(\Omega)} \|P_L\|_{L^2(\Omega) \to H^0_{\text{mix}}(\Omega)} \|u\|_{H^{s+1}(J) \otimes H^0_{\text{mix}}(\Omega)}
\]
\[
+ \|P_I\|_{L^2(\Omega) \to L^2(\Omega)} \|I - P_L\|_{H^1(\Omega) \otimes H^0_{\text{mix}}(\Omega)} \|u\|_{L^2(J) \otimes H^0_{\text{mix}}(\Omega)}
\]
\[
+ \|I - P_I\|_{L^2(\Omega) \to L^2(\Omega)} \|I - P_L\|_{H^1(\Omega) \otimes H^0_{\text{mix}}(\Omega)} \|u\|_{L^2(J) \otimes H^1_{\text{mix}}(\Omega)}
\]
\[
\lesssim 2^{-sL} \sqrt{L} \|u\|_{\mathcal{H}^{L+1}}.
\]

Here we use the inverse inequality \(\|P_L\|_{L^2(\Omega) \to H^1(\Omega)} \lesssim 2^L\). The last norm with respect to \(L^2(J) \otimes H^0_{\text{mix}}(\Omega)\) is treated analogously. Summing up all contributions proves the second inequality. \(\square\)

### 3.3 Convergence

Using the approximation result from the previous section we can prove the convergence of the discrete solution.

**Theorem 5** Let \(u\) be a solution of the transport problem (1) and (2) in the case of constant coefficients and \(\sigma \geq 0\) and assume that \(u \in \mathcal{H}^{r+1}\) for \(0 < s \leq r\). Let \(U\) be the solution of the discrete equations (5) for level \(L\), where \(|I| \sim h = 2^{-L}\) and \(\delta \sim h\). Then
\[
\|U - u\|_{\delta} \lesssim h^{s+1/2} (\ln h)^{1/2} \|u\|_{\mathcal{H}^{L+1}}.
\]

**Proof** The proof follows along the lines of [24, Thm. 13.7] and [6]. Decompose the error
\[
e = U - u = (U - u_h) + (u_h - u) = \theta + \rho, \quad u_h = (P_I \otimes P_L)u.
\]
The streamline diffusion norm can be estimated by
\[
\|e\|_{\delta} \lesssim \|\theta\|_{\delta} + \|\rho\|_{\delta}.
\]
For the first term note that
\[
\|\theta\|_{\delta}^2 \lesssim A_I^{(\delta)}(\theta, \theta) = -A_I^{(\delta)}(\rho, \theta),
\]
where the Galerkin orthogonality has been used. Applying (10) for the right hand side, using Young’s inequality and Lemma 1 gives the estimate

\[
|A^{(\delta)}_I (\rho, \theta)| 
\leq \sum_{j=1}^{N} \int_{I_j} 2\delta^{-1} \|\rho\|^2 + \frac{\delta}{8} \|\partial_t \rho + \partial_\beta \rho\|^2 + \sigma \|\rho\|^2 + \frac{\sigma}{4} \|\theta\|^2 + \|\rho\|_{H^1}^2 + \frac{1}{4} |\theta|^2_{H^1} + 2\delta \|\partial_t \rho + \partial_\beta \rho\|^2 + \frac{\delta}{8} \|\partial_t \rho + \partial_\beta \rho\|^2 \, dt 
\]

\[
+ \sum_{j=1}^{N} \|\rho_j\|^2 + \frac{1}{4} \sum_{j=1}^{N-1} \|\theta_j\|^2 + \frac{1}{4} |\theta_N|^2 
\]

\[
\leq \frac{1}{2} A^{(\delta)}_I (\theta, \theta) 
+ C \left[ \sum_{j=1}^{N} \int_{I_j} (1 + \delta^{-1}) \|\rho\|^2 + \delta \|\partial_t \rho + \partial_\beta \rho\|^2 + |\rho|^2_{H^1} + \int_{I_j} \|\rho_j\|^2 \right].
\]

Here we used the fact that \( \rho \) is continuous in time. Hence it follows that

\[
\|\theta\|^3_3 \lesssim \sum_{j=1}^{N} \int_{I_j} (1 + \delta^{-1}) \|\rho\|^2 + \delta \|\partial_t \rho + \partial_\beta \rho\|^2 + |\rho|^2_{H^1} + \int_{I_j} \|\rho_j\|^2.
\]

For the second term in (12)

\[
\|\rho\|^2_3 \lesssim \sum_{j=1}^{N} \int_{I_j} \|\rho\|^2 + \delta \|\partial_t \rho + \partial_\beta \rho\|^2 + \|\rho\|^2_{H^1(I_j \times \Omega)} + \|\rho_0\|^2 + \|\rho_N\|^2,
\]

where again the continuity of \( \rho \) was used. Applying the trace inequality on each strip \( I_j \times \Omega \) gives

\[
\|\rho_{j-1}\|^2 + \|\rho_j\|^2 + \int_{I_j} |\rho|_{H\text{lip}(\partial \Omega)}^2 \, dt \lesssim \|\rho\|_{L^2(I_j \times \Omega)} \cdot \|\rho\|_{H^1(I_j \times \Omega)} 
\lesssim \delta^{-1} \|\rho\|_{L^2(I_j \times \Omega)}^2 + \delta \|\rho\|_{H^1(I_j \times \Omega)}^2.
\]

Using

\[
\int_{I_j} \|\partial_t \rho + \partial_\beta \rho\|^2 \, dt \lesssim \|\rho\|_{H^1(I_j \times \Omega)}^2,
\]

summing up and applying Theorem 4 gives

\[
\|\varepsilon\|^2_3 \lesssim (1 + \delta^{-1}) \|\rho\|_{L^2(J \times \Omega)}^2 + \delta \|\rho\|_{H^1(J \times \Omega)}^2 \lesssim h_L^{2\epsilon+1} + \ln h_L \|u\|_{H^1}^2
\]

which completes the proof.

\[\square\]

4 Numerical algorithm

In this section we describe how the discrete equation (4) on each time strip \( I_j, j = 1, \ldots, N \) is solved. Now we allow the coefficients \( \beta, \sigma \) to be functions with sufficient regularity. The aim
is to use data structures which enable the efficient representation of the sparse grid functions as well as efficient calculations. For that purpose, we use standard finite element libraries and exploit the tensor product structure of the problem. Therefore we will concentrate on coefficient functions which exhibit a tensor product structure.

### 4.1 Sparse grid representation

Traditionally sparse grid spaces are built on multilevel decomposition and require special basis functions resulting in quite sophisticated operations in Galerkin schemes. This problem can be overcome by representing sparse grid functions as a combination of anisotropic full grid spaces and performing all the computations on these regular spaces. For sparse grids based on finite element spaces this approach has been used in [11, 22], for example. This idea is closely connected with the sparse grid combination technique introduced in [26], see also [19].

Here in this section, the focus is on the representation of sparse grid functions. In contrast to the combination technique, we will continue to solve the problem on the whole space $V_L$ and not only combining the solutions of the anisotropic spaces. However, later in Sect. 4.3 we will use the combination technique as a preconditioner for the sparse grid system.

The key is the representation of the space $V_L$ in (3) as a sum of anisotropic full grid spaces $V_{\ell_1}^{(1)} \otimes V_{\ell_2}^{(2)}$ with $\ell_1 + \ell_2 = L + 1$. Note that this is not a direct sum. For representing a function we use the finite element basis

$$V_{\ell}^{(i)} = \text{span} \{ \varphi_{\ell, k}^{(i)} | k = 1, \ldots, n_{\ell}^{(i)} \}, \quad n_{\ell}^{(i)} = \dim V_{\ell}^{(i)}$$

for $i = 1, 2$ and a basis for the polynomials on $I_j$:

$$\mathbb{P}^r(I_j, \mathbb{R}) = \text{span} \{ \eta_{s}^{(j)} | s = 0, \ldots, r \}.$$ 

For notational simplicity we will omit the index $j$ in the following.

Hence a function $U \in \mathbb{P}^r(I_j, V_L)$ can be written as

$$U(t, x, v) = \sum_{|\ell|_1=L+1} \sum_{0 \leq s \leq r} \sum_{k \leq n_\ell} u_{s,k}^{(\ell)} \eta_{s}(t) \varphi_{\ell_1,k_1}^{(1)}(x) \varphi_{\ell_2,k_2}^{(2)}(v), \quad \mathbf{n}_\ell = [n_{\ell_1}^{(1)}, n_{\ell_2}^{(2)}].$$

The representation however is not unique since the functions

$$\eta_{s} \otimes \varphi_{\ell_1,k_1}^{(1)} \otimes \varphi_{\ell_2,k_2}^{(2)}, \quad 0 \leq s \leq r, \quad k \leq n_\ell, \quad |\ell|_1 = L + 1,$$

are just a spanning set and not a basis. This leads to additional degrees of freedom, which are negligible for the interesting case of $d = 2, 3$, however.

### 4.2 Discrete equations and matrices

We will now use the spanning set (13) to derive a system of linear equations for the discrete equation (4) in weak form for a time strip $I_j$. Omitting again the index $j$ gives

$$A^{(\delta)} \mathbf{u} = \mathbf{b}^{(\delta)}, \quad \mathbf{u} = [u_{\ell}]_{|\ell|_1=L+1}, \quad u_{\ell} = [u_{s,k}^{(\ell)}]_{0 \leq s \leq r, \quad k \leq n_\ell}.$$  

(14)

where the blocks are given by

$$A_{\ell, \ell'}^{(\delta)} = \left[ a_{j}^{(\delta)} \eta_{s} \otimes \varphi_{\ell_1,k_1}^{(1)} \otimes \varphi_{\ell_2,k_2}^{(2)} \otimes \eta_{s'} \otimes \varphi_{\ell_1,k_1}^{(1)} \otimes \varphi_{\ell_2,k_2}^{(2)} \right]_{0 \leq s, s' \leq r, \quad k \leq n_\ell, k' \leq n_{\ell'}}$$

\(\heartsuit\) Springer
with indices \( k \) the interval solvable since the right hand side is in the range of the matrix \( [27] \). We will later use iterative methods to solve the system which are based on the application of the matrix to a vector, see Sect. 4.3.

For its efficient calculation we exploit the tensor product structure for each individual block \( A^{(\delta)}_{\ell, \ell'} \). In the first step we use a quadrature rule of sufficient order for the time integration on the interval \( I_j = [t_{j-1}, t_j] \):

\[
\int_{I_j} f(t) \, dt \approx \sum_{\mu=1}^m w_\mu f(\tau_\mu), \quad \tau_\mu \in [t_{j-1}, t_j].
\]

Define for \( 1 \leq \mu \leq m \)

\[
M^{(t)}_\mu = \left[ \eta_{s'}(\tau_\mu) \cdot \eta_{s}(\tau_\mu) \right]_{s, s'}, \quad T^{(t)}_\mu = \left[ \eta_{s'}(\tau_\mu) \cdot \eta_{s}(\tau_\mu) \right]_{s, s'},
\]

\[
C^{(t)}_\mu = \left[ \eta_{s'}(\tau_\mu) \cdot \eta_{s}(\tau_\mu) \right]_{s, s'}, \quad G^{(t)}_\mu = \left[ \eta_{s'}(t_{j-1}) \cdot \eta_{s}(t_{j-1}) \right]_{s, s'}
\]

for \( 0 \leq s, s' \leq r \) and

\[
M^{(r)}_{\mu, \ell, \ell'} = \left[ (\varphi_{\ell'}, k') \cdot (\varphi_{\ell, k}) \right], \quad T^{(r)}_{\mu, \ell, \ell'} = \left[ (\partial_\beta(\tau_\mu, \cdot) \varphi_{\ell'}, k') \cdot (\varphi_{\ell, k}) \right],
\]

\[
C^{(r)}_{\mu, \ell, \ell'} = \left[ (\partial_\beta(\tau_\mu, \cdot) \varphi_{\ell'}, k') \cdot (\partial_\beta(\tau_\mu, \cdot) \varphi_{\ell, k}) \right], \quad K^{(r)}_{\mu, \ell, \ell'} = \left[ (\sigma(\tau_\mu, \cdot) \varphi_{\ell', k'}) \cdot (\varphi_{\ell, k}) \right],
\]

\[
G^{(r)}_{\mu, \ell, \ell'} = \left[ (\varphi_{\ell'}, k') \cdot (\varphi_{\ell, k}) \right]
\]

with indices \( \ell \leq n_\ell, \ell' \leq n_{\ell'} \). Hence one block can be approximated by

\[
A^{(\delta)}_{\ell, \ell'} \approx \tilde{A}^{(\delta)}_{\ell, \ell'}
\]

\[
= \sum_{\mu=1}^m w_\mu \left[ (T^{(t)}_\mu + \delta C^{(t)}_\mu) \otimes M^{(r)}_{\mu, \ell, \ell'} + \delta T^{(r)}_\mu \otimes (T^{(r)}_{\mu, \ell, \ell'})^T + \delta (T^{(t)}_\mu)^T \otimes (T^{(r)}_{\mu, \ell, \ell'} + K^{(r)}_{\mu, \ell, \ell'}) + M^{(r)}_\mu \otimes (T^{(r)}_{\mu, \ell, \ell'} + C^{(r)}_{\mu, \ell, \ell'} + K^{(r)}_{\mu, \ell, \ell'}) + \tilde{K}^{(r)}_{\mu, \ell, \ell'} - G^{(r)}_{\mu, \ell, \ell'} \right] + G^{(r)}_\mu \otimes M^{(r)}_{\mu, \ell, \ell'}.
\]

The matrix \( M^{(r)}_{\mu, \ell, \ell'} \) acting on the variables \( x \) and \( y \) has itself a tensor product structure

\[
M^{(r)}_{\mu, \ell, \ell'} = M^{(1)}_{\ell_1, \ell_1} \otimes M^{(2)}_{\ell_2, \ell_2}, \quad M^{(r)}_{\ell, \ell'} = \left[ (\varphi_{\ell', k'}) \cdot (\varphi_{\ell, k}) L_2(\Omega^{(1)}) \right]_{k, k'},
\]

which can be exploited. In general this is not the case for the other matrices, making a \( 2d \)-dimensional integration necessary. In the following we will therefore assume that the coefficients \( \beta \) and \( \sigma \) have a tensor product structure in the sense of

\[
\beta(t, x, v) = \sum_{i=1}^d a_i(t, x) b_i(t, v) e_i + c_i(t, x) d_i(t, v) e_{i+d},
\]

\[
\sigma(t, x, v) = p(t, x) q(t, v),
\]

\( \square \) Springer
where \( e_i \) is the \( i \)-th unit vector, or can be represented (approximately) in a short sum of such tensor products. In the first case for example

\[
T_{\mu, \ell, \ell'}^{(r)} = \sum_{i=1}^{d} \left[ \left( d_i (\tau_{\mu} \cdot \cdot \cdot) \partial_{x_i} \varphi_{\ell_1, k_1}^{(1)} \cdot \varphi_{\ell_1, k_1}^{(1)} \right)_{L^2(\Omega^{(1)})} \cdot \left( b_i (\tau_{\mu} \cdot \cdot \cdot) \varphi_{\ell_2, k_2}^{(2)} \cdot \varphi_{\ell_2, k_2}^{(2)} \right)_{L^2(\Omega^{(2)})} \right]_{\ell, \ell'}
\]

All other matrices can be treated analogously except the matrices \( \Omega_1^{(1)} \) given by

\[
\ell A
\]

for general matrices \( S \) for the calculation of integrals over \( \Omega^{(1)} \). Therefore to calculate \( \ell \) for \( \ell' < \ell \) denote by \( E_{\ell, \ell'} \) the matrix representation of the embedding from \( \Omega_1^{(1)} \) in \( \Omega_1^{(2)} \). Then for example

\[
A_{\mu, \ell, \ell'}^{(\mu)} = (E_{\ell, \ell'})^T \cdot A_{\mu, \ell, \ell'}^{(\mu)} \cdot E_{\ell, \ell'}
\]

For the calculation of \( A_{\mu, \ell, L} \) classical finite element libraries can be used directly.

The application of the matrix \( A^{(\delta)} \) to a vector \( u \) can be done blockwise, i.e. where block \( \ell \) is given by

\[
\left[ A^{(\delta)} u \right]_\ell = \sum_{|\ell'| = L+1} A^{(\delta)}_{\ell, \ell'} \cdot u_{\ell'}
\]

while the tensor product structure can be exploited for each block. However, the order of application is crucial [21, 28] to prevent the need to store intermediate results on full grids. Therefore to calculate

\[
S^{(r)} \otimes S^{(1)}_{\ell_1, \ell'_1} \otimes S^{(2)}_{\ell_2, \ell'_2} \cdot u_{\ell'}
\]

for general matrices \( S^{(r)} \), \( S^{(1)} \), \( S^{(2)} \) we will use

\[
(S^{(r)} \otimes \text{Id}^{(1)}_{\ell_1} \otimes \text{Id}^{(2)}_{\ell_2}) \cdot (\text{Id}^{(r)} \otimes S^{(1)}_{\ell_1, \ell'_1} \otimes \text{Id}^{(2)}_{\ell_2}) \cdot (\text{Id}^{(r)} \otimes \text{Id}^{(1)}_{\ell_1} \otimes S^{(2)}_{\ell_2, \ell'_2}) \cdot u_{\ell'}
\]

in the case \( \ell' + \ell_2 \leq \ell_1 + \ell'_1 \) and

\[
(S^{(r)} \otimes \text{Id}^{(1)}_{\ell_1} \otimes \text{Id}^{(2)}_{\ell_2}) \cdot (\text{Id}^{(r)} \otimes \text{Id}^{(1)}_{\ell_1} \otimes S^{(1)}_{\ell_2, \ell'_2}) \cdot (\text{Id}^{(r)} \otimes \text{Id}^{(1)}_{\ell_1} \otimes \text{Id}^{(2)}_{\ell_2}) \cdot u_{\ell'}
\]

otherwise, where \( \text{Id} \) the identity matrix in the respective spaces. The application of such a tensor product matrix can be applied in log-linear complexity with respect to the degrees of freedom.
4.3 Solution of linear system of equations

For the solution of (14) we will use an iterative method to exploit the tensor product structure of the block matrices and use the combination technique as a preconditioner.

The sparse grid combination technique was introduced in [26] (see also [19]). Instead of solving the discretized equation on the sparse grid, solutions are computed on a sequence of anisotropic full grid spaces. They are then combined to form a sparse grid function. For a quite general class of symmetric bilinear forms it can be shown that this method has the same order of convergence as the sparse grid solution [22]. Alternatively, it can also be used as a preconditioner for the sparse grid discretization, see [20, 29] in the case of radiative transfer.

In our non-symmetric case we follow the latter approach and use the combination technique as a preconditioner for an outer Richardson iteration, see Algorithm 1. In order to compute the preconditioner we have to solve the transport equation

\[ A^{(δ)}_{ℓ, ℓ} \Delta \widehat{u}_ℓ = r_ℓ \]  

(16)
on small anisotropic full grid spaces with levels \( L \) (line 6) and \( L - 1 \) (line 11).

**Algorithm 1** Richardson iteration with combination technique preconditioner for the solution of (14)

1: \( u ← 0 \)
2: repeat
3: \( r ← A^{(δ)}u - b^{(δ)} \)
4: for \( ℓ_1 = 1, \ldots, L \) do \( \triangleright \) anisotropic spaces of level \( L \)
5: \( ℓ ← (ℓ_1, L + 1 - ℓ_1) \)
6: Solve \( A^{(δ)}_{ℓ, ℓ} \Delta \widehat{u}_ℓ = r_ℓ \)
7: end for
8: for \( ℓ_1 = 1, \ldots, L - 1 \) do \( \triangleright \) anisotropic spaces of level \( L - 1 \)
9: \( ℓ ← (ℓ_1, L - ℓ_1) \)
10: \( r_ℓ ← (\text{Id}^{(1)} ⊗ (E^{(1)}_{ℓ_1+1, ℓ_1})^T ⊗ \text{Id}^{(2)}_{ℓ_2}) r_{ℓ_1+1, ℓ_2} \)
11: Solve \( A^{(δ)}_{ℓ, ℓ} \Delta \widehat{u}_ℓ = r_ℓ \)
12: end for
13: for \( ℓ_1 = 1, \ldots, L \) do \( \triangleright \) combination technique
14: \( ℓ ← (ℓ_1, L + 1 - ℓ_1) \)
15: \( \Delta u_ℓ ← \begin{cases} \Delta \widehat{u}_ℓ & ℓ_1 = 1 \\ \Delta \widehat{u}_ℓ - (E^{(1)}_{ℓ_1, ℓ_1-1} ⊗ \text{Id}^{(2)}_{ℓ_2}) \Delta \widehat{u}_{ℓ_1-1, ℓ_2} & ℓ_1 > 1 \end{cases} \)
16: end for
17: \( ∆u ← [\Delta u_ℓ]_{ℓ_1=L+1} \)
18: \( u ← u − ∆u \) \( \triangleright \) preconditioned Richardson step
19: until \( ∥ ∆u ∥_{H^1} < ϵ \)

For that purpose we use a preconditioned GMRES. As a first preliminary preconditioner for these systems, we use the inverse of

\[ (T^{(i)} + δC^{(i)} + G^{(i)}) ⊗ M^{(1)}_{ℓ_1, ℓ_1} ⊗ M^{(2)}_{ℓ_2, ℓ_2}, \]

which can be computed efficiently using the LU decomposition in each product space. As a consequence the tensor product structure (see Sect. 4.2) can be used for all calculations and neither the full matrix \( A^{(δ)} \) (14) nor the smaller matrices \( A^{(δ)}_{ℓ, ℓ} \) (16) have to be assembled.

Alternatively, we can also use other preconditioners like ILU or direct sparse solvers. For that purpose, however, the matrices \( A^{(δ)}_{ℓ, ℓ} \) for the small anisotropic full grid spaces have to be
assembled. For the future it would be beneficial to devise schemes similar to [27] for the case of symmetric bilinear forms or use more advanced preconditioned iterative schemes for the anisotropic full grids [30].

Using the combination technique we compute the preconditioned residuum in line 15. Note that we used the embedding to represent the solution on spaces with level $|\ell|_1 = L + 1$. For interpolation such a scheme can be shown to give the exact result [19].

The iteration is stopped, if the norm of preconditioned residuum is small enough, i.e. $\|\Delta u\|_{H^1(J \times \Omega)} < \epsilon$, for some given tolerance $\epsilon$.

5 Numerical experiments

In this section we present numerical examples to study the method described in Sect. 4. The implementation is based on the finite element library MFEM [31] and uses its Python wrapper PyMFEM.1 Due to the regular data structure of the combination technique standard numerical routines from numpy2 and scipy3 were used. The cupy4 library was used as a replacement for the numerical libraries to perform the calculations on a GPU with only minor code changes. The computations were carried out on a desktop computer (i9 7900, 128 GB RAM, GTX 1080).

5.1 Linear advection with constant coefficients

As the first example we consider the linear advection equation with constant coefficients in $d + d$ dimensions

$$\partial_t u + \mathbf{1} \cdot \nabla_x u + \mathbf{1} \cdot \nabla_v u = 0, \quad x, v \in [0, 1]^d$$

$$u(0, x, v) = \sin \left(2\pi \sum_{i=1}^{d}(x_i + v_i)\right)$$

with periodic boundary conditions. The solution is given by

$$u(t, x, v) = \sin \left(2\pi \left(\sum_{i=1}^{d}(x_i + v_i) - 2dt\right)\right),$$

which is periodic in $t$ for $T = 1/(2d)$.

The mesh on the coarsest scale on each $\Omega^{(i)}$ consists of $4^d$ $d$-cubes of equal size and is uniformly refined for finer scales. The finite element functions are continuous piecewise polynomials of order $r$ for $r = 1, 2$. The streamline diffusion parameter is chosen as the edge length on the finest scale. The equation is solved on one period $T$ using discontinuous polynomials of the same order $r$ as in the spatial discretization and $2^{L+1}$ steps.

The numerical solution is compared to the interpolation of the analytical solution in $V_{L+1, I}$ and polynomial order $r + 1$. The streamline diffusion norm of the difference is computed and reported in Table 1. There the order is estimated from the errors of two successive runs. The convergence is also depicted in Fig. 1. The numerical results confirm the convergence order of $h^{r+1/2}$ from Theorem 5.

1 https://github.com/mfem/PyMFEM.
2 https://numpy.org/.
3 https://scipy.org/.
4 https://cupy.dev/.
Table 1  Error with respect to the streamline diffusion norm $\parallel \cdot \parallel_5$ and estimated order of the numerical solution of (17) on $d + d$ dimensions after one period for sparse grid spaces with levels $L$ and order $r$ of the ansatz functions; $h$ denotes the grid size of the finest level.

| $L$ | $d = 1$ |  |  | $d = 2$ |  |  | $d = 3$ |  |  |
|-----|---------|---|---|---------|---|---|---------|---|---|
|     | $h$     | dof | Error | Order | $h$      | dof | Error | Order | $h$      | dof | Error | Order |
| 1   | 2.50e−01 | 16  | 8.68e−01 | 64  | 3.09e−01 | 2.50e−01 | 256 | 8.14e−01 | 4096 | 3.92e−01 |
| 2   | 1.25e−01 | 48  | 5.25e−01 | 0.72 | 192      | 1.25e−01 | 1792 | 5.98e−01 | 28,672 | 7.47e−02 |
| 3   | 6.25e−02 | 128 | 2.03e−01 | 1.37 | 512      | 6.25e−02 | 10,240 | 2.64e−01 | 163,840 | 1.34e−02 |
| 4   | 3.12e−02 | 320 | 7.20e−02 | 1.49 | 1280     | 3.12e−02 | 53,248 | 3.39e−02 | 4,194,304 | 1.49e−04 |
| 5   | 1.56e−02 | 768 | 2.55e−02 | 1.50 | 3072     | 1.56e−02 | 262,144 | 3.39e−02 | 19,922,944 | 7.41e−05 |
| 6   | 7.81e−03 | 1792| 8.99e−03 | 1.50 | 7168     | r = 2     | r = 2     | r = 1     | 262,144 | 4.59e−01 |

Fig. 1  Error with respect to the streamline diffusion norm $\parallel \cdot \parallel_5$ of the numerical solution of (17) on $d + d$ dimensions after one period for sparse grid spaces with order $r$ of the ansatz functions; $h$ denotes the grid size of the finest level.
Fig. 2 Left: spatial mesh on the coarsest scale for domain $\Omega^{(1)}$. The initial distribution is supported in the light gray area and is transported mainly along the arrow. It is completely contained in the dark area at $t = 1$; right: initial spatial distribution $\rho(0, x)$

### 5.2 L-shaped spatial domain

In the next example we solve the kinetic equation
\[
\partial_t u + v \cdot \nabla_x u = 0, \quad [0, 1] \times \Omega
\]
on a L-shaped spatial domain
\[
\Omega = \Omega^{(1)} \times \Omega^{(2)}, \quad \Omega^{(1)} = (-1, 1)^2 \setminus [0, 1]^2, \quad \Omega^{(2)} = [-2, 0] \times [0, 2],
\]
see Fig. 2. This demonstrates the use of the method for non-rectangular domains, boundary conditions, and discontinuous solutions. We choose zero inflow boundary condition and
\[
u_0(x, v) = \psi\left(\frac{x_1 - 0.5}{0.25}\right) \cdot \psi\left(\frac{x_2 + 0.5}{0.25}\right) \cdot \psi\left(\frac{v_1 + 1}{0.25}\right) \cdot \psi\left(\frac{v_2 - 1}{0.25}\right)
\]
as the initial condition. Here $\psi$ is a $C^1$ function centered at 0 and supported in $[-1, 1]$:
\[
\psi(\xi) = \begin{cases} 
(\|\xi\| - 1)^2 \cdot (2|\xi| + 1) & |\xi| \leq 1 \\
0 & |\xi| > 1 
\end{cases}
\]
For the assessment of the numerical results we will concentrate on the spatial distribution
\[
\rho(t, x) = \int_{\Omega^{(2)}} u(t, x, v) \dd v.
\]
The initial distribution is supported in $[0.25, 0.75] \times [-0.75, -0.25]$. It is transported mainly in the direction $[-1, 1]^T$ such that it is completely located in $[-1, 0] \times [0, 1]$ at the final time $t = 1$, see Fig. 2. By the method of characteristics the solution is given by
\[
u(1, x, v) = \begin{cases} 
\nu_0(x - v, v) & \text{if } v_1x_2 > v_2x_1 \\
0 & \text{otherwise}
\end{cases}
\]
for $x \in [-1, 0] \times [0, 1], \ v \in [-1.25, -0.75] \times [0.75, 1.25]$. Outside this domain it is zero. Here, the L-shaped domain boundary leads to a discontinuous distribution.

The matrices $G^{(r)}_{\mu, \ell, \ell'}$ from (15) corresponding to the boundary condition can be represented as a short sum of tensor products by partitioning the inflow boundary. For example the inflow
Fig. 3 Left: Computed spatial distribution $\rho(x)$ of (18) for $L = 5$ and $r = 1$; right: error of the computed spatial distribution.

Table 2  Relative error of the spatial distribution (19) with respect to the $L_2$-norm and estimated order of the numerical solution of (18) at $t = 1$ for sparse grid spaces with levels $L$ and linear elements; $h$ denotes the grid size of the finest level.

| $L$ | 2   | 3   | 4   | 5   | 6   |
|-----|-----|-----|-----|-----|-----|
| $h$ | 6.25e−02 | 3.12e−02 | 1.56e−02 | 7.81e−03 | 3.91e−03 |
| rel. err | 5.89e−01 | 2.68e−01 | 6.56e−02 | 2.58e−02 | 1.28e−02 |
| Order | 1.14 | 2.03 | 1.34 | 1.01 |

at $x_1 = -1$ with normal $n = [-1, 0, 0]^T$ is given by

$$\{ -1 \} \times [ -1, 1 ] \times \{ \mathbf{v} \in \Omega (2) \mid -v_1 < 0 \}.$$

Hence its contribution to the element $k, k'$ of $G_{\mu, \ell, \ell'}^{(r)}$ can be calculated by

$$\int_{-1}^{1} \int_{\Omega (2)} \phi_{x_1, k_1}^{(1)} (-1, x_2) \cdot \phi_{x_1, k_1}^{(1)} (-1, x_2) \cdot \chi_{v_1 > 0} (\mathbf{v}) \cdot (-v_1) \cdot \phi_{x_2, k_2}^{(2)} (\mathbf{v}) \cdot \phi_{x_2, k_2}^{(2)} (\mathbf{v}) \, d\mathbf{v} \, dx_2$$

with the characteristic function $\chi$. This integral as well as the integrals over other sections of the inflow boundary can be factorized leading to a sum of tensor product operators for $G_{\mu, \ell, \ell'}^{(r)}$.

The coarsest mesh for the spatial domain $\Omega (1)$ is depicted in Fig. 2 and the domain $\Omega (2)$ is discretized by an $8 \times 8$ regular grid. The equation is simulated for $t \in [0, 1]$ with linear elements ($r = 1$).

Figure 3 (left) shows the resulting spatial distribution for $t = 1$ and $L = 5$. In order to assess the numerical result it is compared to the distribution computed from the analytical solution (20) (right). The integration with respect to $\mathbf{v}$ was carried out numerically. The results for levels up to $L = 6$ are given in Table 2.

Although the solution $u$ is discontinuous the numerical distribution $\rho$ does not show oscillations thanks to the streamline diffusion. The theoretical results do not cover this case and it is too early to assess the order of convergence numerically. However the method seems to perform reasonable in this case.
5.3 Vlasov-Poisson equation

In plasma physics the particle density function $f(t, x, v)$ of electrons in a constant background ion density interacting with electrostatic fields ignoring collisions may be described by the Vlasov-Poisson equations

$$\partial_t f + v \cdot \nabla_x f - E(f) \cdot \nabla_v f = 0$$

$$\nabla_x \cdot E = -\int f(t, x, v) \, dv + 1, \quad \nabla \times E = 0$$

supplemented by appropriate boundary conditions [4]. The first equation fits in the framework of the transport equation (1) except for the non-linearity due to the coupling with the electrical field. Nevertheless we may still use the method in a fixed-point iteration where the transport and field equation are solved alternately.

Given some approximation of the electrical field $E^{(k)}$ we may compute the density function $f^{(k+1)}$ by solving

$$\partial_t f^{(k+1)} + v \cdot \nabla_x f^{(k+1)} - E^{(k)}(t, x) \cdot \nabla_v f^{(k+1)} = 0. \quad (21)$$

using the method for linear transport equation presented here. Now given $f^{(k+1)}$ we compute the resulting electrical field by solving the Poisson equation

$$-\Delta_x \phi = -\int f^{(k+1)}(t, x, v) \, dv + 1 \quad \text{and letting } E^{(k+1)} = -\nabla_x \phi. \quad (22)$$

Hence to solve the coupled equation in a time strip $I_j = [t_{j-1}, t_j]$ we alternate the two steps. Starting with $f^{(0)}$ as the final density $f(t_{j-1})$ of the last step and the resulting electrical field $E^{(0)}$ for all $t \in I_j$ we iterate until

$$\|E^{(k+1)} - E^{(k)}\|_{L^\infty(I_j \times \Omega^{(1)})} + \|f^{(k+1)} - f^{(k)}\|_{H^1(I_j \times \Omega^{(1)} \times \Omega^{(2)})} < \epsilon$$

for some tolerance $\epsilon$.

Note that (22) has to be solved only at certain quadrature points $\tau_{\mu} \in I_j$, see Sect. 4.2, and that the integral of $f$ on the right hand side can be represented by a finite element function on the finest grid of the space domain $\Omega^{(1)}$. Its solution is computed by the finite element method on the same grid. The preconditioner needed to solve (21) is updated if the rate of convergence decreases significantly.

As the first test we consider the classic Landau damping. In the 1+1-dimensional case we use the initial condition

$$f(0, x, v) = f_0(x, v) = \frac{1}{\sqrt{2\pi}} e^{-v^2/2} \left(1 + \alpha \cos(kx)\right)$$

and parameters $\alpha = 10^{-2}$, $k = 1/2$ on a periodic domain $[0, 4\pi) \times (-v_{\text{max}}, v_{\text{max}})$ with $v_{\text{max}}$ big enough [4]. We choose $v_{\text{max}} = 6$.

We simulate the system in the time interval $[0, 50]$. On level $L = 1$ each domain, i. e. space and velocity, is divided into four equally sized subintervals which are refined globally for higher levels. For $L = 5$ one hundred times steps are performed, which are doubled for each increase in level. For the computation of the electrical field the midpoint rule is applied. A direct solver computes the resulting systems of linear equations for the transport as well as the potential equation.
Linear analysis shows that the electric field decays with a rate of $\gamma \approx 0.153$ [32]. In Fig. 4 the electrical energy

$$\frac{1}{2} \int_{\Omega} |E(t, x)|^2 \, dx$$

exhibits the analytical decay rate up to $t = 25$ or $t = 30$ depending on the level $L$. Furthermore the invariants particle number, total energy and entropy, i.e.

$$\int_{\Omega} f(t, x, v) \, dx \, dv, \quad \frac{1}{2} \int_{\Omega} |v|^2 f(t, x, v) \, dx \, dv + \frac{1}{2} \int_{\Omega^\times} |E(t, x)|^2 \, dx,$$

are shown. We see that the mass is almost conserved whereas the energy and entropy only up to a small error.

In the 2+2-dimensional setting we use $\Omega = (0, 4\pi)^2 \times (-6, 6)^2$ and

$$f(0, x, v) = \frac{1}{2\pi} e^{-|v|^2/2} \left( 1 + \alpha \cos(kx_1) + \alpha \cos(kx_2) \right), \quad \alpha = 10^{-2}, \ k = \frac{1}{2}.$$ Again we simulate the system with the same settings as in the 1+1-dimensional case, where the coarsest meshes now consist of four by four squares. The results are shown in Fig. 5. Qualitatively the results of the 1+1-dimensional setting are reproduced with slightly higher errors.

The second example is the two-stream instability, where two beams propagate in opposite directions. Here, small perturbations in the initial condition lead to an exponential increase in electrical energy and a subsequent saturation.
In the 1+1-dimensional case we use the periodic domain $\Omega = (0, 10\pi) \times (-9, 9)$ and the initial condition

$$f(0, x, v) = \frac{1}{2\sqrt{2\pi}} \left( e^{-(v-v_0)^2/2} + e^{-(v+v_0)^2/2} \right) \left( 1 + \alpha \cos(kx) \right),$$

where $\alpha = 10^{-3}$, $k = 1/5$ and $v_0 = 2.4$. The results are shown in Fig. 6.

For all levels used, one observes an almost identical exponential increase of the electrical energy and a saturation where the energy remains almost constant. Regarding the physical invariants, the mass is almost conserved, while the total energy and especially the entropy show a strong increase. Comparing these results with the Landau damping, we see that this is a more challenging problem. We note that the increase in the error occurs during the time when the electrical energy increases exponentially and then remains almost constant.

Now turning to the 2+2-dimensional case we let $\Omega = (0, 10\pi)^2 \times (-9, 9)^2$ with periodic boundary condition and

$$f(0, x, v) = \frac{1}{8\pi} \left( 1 + \alpha \cos(kx_1) + \alpha \cos(kx_2) \right) \prod_{k=1}^2 \left( e^{-(v_i-v_0)^2/2} + e^{-(v_i+v_0)^2/2} \right),$$

where again $\alpha = 10^{-3}$, $k = 1/5$ and $v_0 = 2.4$. The results are shown in Fig. 7.

The energy as well as the invariants show almost identical behavior as in the 1+1-dimensional case. Only the error of mass increases by several orders of magnitude, but is still small. The error of the invariants improves with increasing level $L$.

In conclusion both tests show that the method can be used also in the case of nonlinear problems. However, the execution times are quite high: for the 2+2-dimensional two-stream instability the simulation takes a little less than 3 h for level $L = 5$, two days for $L = 6$, and
Fig. 6  Simulation results of the 1+1-dimensional two-stream instability for linear elements and different levels $L$.

Fig. 7  Simulation results of the 2+2-dimensional two-stream instability for linear elements and different levels $L$. 
two weeks for $L = 7$. Note that the code is still prototypical and not optimized for speed. For example, the problems (16) arising in the computation of the combination technique preconditioner were solved sequentially, although parallelization is possible. Furthermore, the solution of these systems is the most time-consuming part because of the lack of an efficient preconditioner. In contrast, the Richardson iteration as well as the fixed-point iteration of the density and the electric field require only a few iterations.

6 Outlook

In this paper we present a method which is capable of computing the solution of transport equations in moderately complex geometrical domains in up to 3+3-dimensional phase space. By representing sparse grid functions as a linear combination of grid functions in anisotropic full grid spaces, traditional finite element libraries can be used. The resulting regular data structures facilitate parallelization and the use of GPU acceleration. The resulting linear equation can be solved iteratively using a combination technique preconditioner. By using a fixed-point iteration, this can also be used to compute the solution of nonlinear Vlasov-Poisson equations.

We aim to use the method for more complex geometries—for example, to simulate a plasma in an accelerator [33]. For such realistic problems it will be necessary to further parallelize the method. Again, the combination technique naturally leads to independent problems where each problem is a transport equation on anisotropic full grids which itself can be solved on multiple machines. It remains to further investigate more efficient preconditioners for these systems.

Funding Open Access funding enabled and organized by Projekt DEAL.

Data Availability The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Verboncoeur, J.P.: Particle simulation of plasmas: review and advances. Plasma Phys. Control. Fusion 47, 231–260 (2005). https://doi.org/10.1088/0741-3335/47/5A/017
2. Besse, N., Latu, G., Ghizzo, A., Sonnendrücker, E., Bertrand, P.: A wavelet-MRA-based adaptive semi-Lagrangian method for the relativistic Vlasov-Maxwell system. J. Comput. Phys. 227(16), 7889–7916 (2008). https://doi.org/10.1016/j.jcp.2008.04.031
3. Deriaz, E., Peirani, S.: Six-dimensional adaptive simulation of the Vlasov equations using a hierarchical basis. Multiscale Model. Simul. 16(2), 583–614 (2018). https://doi.org/10.1137/16M1108649
4. Einkemmer, L., Lubich, C.: A low-rank projector-splitting integrator for the Vlasov-Poisson equation. SIAM J. Sci. Comput. 40(5), 1330–1360 (2018). https://doi.org/10.1137/18M116383X
5. Bungartz, H.-J., Griebel, M.: Sparse grids. Acta Numer 13, 147–269 (2004). https://doi.org/10.1017/S0962492904000182
6. Schwab, C., Suli, E., Todor, R.A.: Sparse finite element approximation of high-dimensional transport-dominated diffusion problems. ESAIM: Math. Modell. Numer. Anal. 42(5), 777–819 (2008). https://doi.org/10.1051/m2an:2008027

7. Kormann, K., Sonnendrücker, E.: Sparse grids for the Vlasov–Poisson equation. In: Sparse Grids and Applications—Stuttgart 2014. Lecture Notes in Computational Science and Engineering, pp. 163–190. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-28262-6_7

8. Bokanowski, O., Garcke, J., Griebel, M., Klompmaker, I.: An adaptive sparse grid semi-Lagrangian method for first order Hamilton-Jacobi Bellman equations. J. Sci. Comput. 55(3), 575–605 (2013). https://doi.org/10.1007/s10915-012-9648-x

9. Guo, W., Cheng, Y.: A sparse grid discontinuous Galerkin method for high-dimensional transport equations and its application to kinetic simulations. SIAM J. Sci. Comput. 38(6), 3381–3409 (2016). https://doi.org/10.1137/16M1060017

10. Griebel, M., Harbrecht, H.: On the construction of sparse tensor product spaces. Math. Comput. 82, 975–994 (2013). https://doi.org/10.1090/S0025-5718-2012-02638-X

11. Harbrecht, H., Schneider, R., Schwab, C.: Multilevel frames for sparse tensor product spaces. Numer. Math. 110(2), 199–220 (2008). https://doi.org/10.1007/s00211-008-0162-x

12. Kanschat, G.: A robust finite element discretization for radiative transfer problems with scattering. East-West J. Numer. Math. 6 (1998)

13. Widmer, G., Hiptmair, R., Schwab, Ch.: Sparse adaptive finite elements for radiative transfer. J. Comput. Phys. 227(12), 6071–6105 (2008). https://doi.org/10.1016/j.jcp.2008.02.025

14. Grella, K., Schwab, C.: Sparse tensor spherical harmonics approximation in radiative transfer. J. Comput. Phys. 230, 8452–8473 (2011). https://doi.org/10.1016/j.jcp.2011.07.028

15. Grella, K., Schwab, C.: Sparse discrete ordinates method in radiative transfer. Comput. Methods Appl. Math. (2011). https://doi.org/10.2478/cmam-2011-0017

16. Dahmen, W., Kutyniok, G., Lim, W.-Q., Schwab, C., Welper, G.: Adaptive anisotropic Petrov-Galerkin methods for first order transport equations. J. Comput. Appl. Math. 340, 191–220 (2018). https://doi.org/10.1016/j.cam.2018.02.023

17. Dahmen, W., Gruber, F., Mula, O.: An adaptive nested source term iteration for radiative transfer equations. Math. Comput. 89, 1 (2019). https://doi.org/10.1090/mcom/3505

18. Johnson, C., Nävert, U., Pitkäranta, J.: Finite element methods for linear hyperbolic problems. Comput. Methods Appl. Mech. Eng. 45, 285–312 (1984). https://doi.org/10.1016/0045-7825(84)90158-0

19. Garcke, J.: Sparse grids in a nutshell. In: Garcke, J., Griebel, M. (eds.) Sparse Grids and Applications, vol. 88, pp. 57–80. Springer, Berlin (2012). https://doi.org/10.1007/978-3-642-31703-3_3

20. Griebel, M.: A domain decomposition method using sparse grids. In: Quarteroni, A. (ed.) Contemporary Mathematics, Vol. 157, DDM6, pp. 263–296. American Mathematical Society, Providence, Rhode Island (1994)

21. Schwab, Ch., Todor, R.A.: Sparse finite elements for stochastic elliptic problems—higher order moments. Computing 71(1), 43–63 (2003). https://doi.org/10.1007/s00607-003-0024-4

22. Griebel, M., Harbrecht, H.: On the Convergence of the Combination Technique. In: Garcke, J., Pflüger, D. (eds.) Sparse Grids and Applications - Munich 2012. Lecture Notes in Computational Science and Engineering, pp. 55–74. Springer International Publishing, Cham (2014). https://doi.org/10.1007/978-3-319-04537-5_3

23. Hilber, N.W.: Stabilized Wavelet Methods for Option Pricing in High Dimensional Stochastic Volatility Models. PhD thesis, ETH Zürich (2009)

24. Larsson, S., Thomée, V.: Partial Differential Equations with Numerical Methods, Corrected 2nd Printing edn. Texts in Applied Mathematics, vol. 43. Springer, Berlin (2005)

25. Griebel, M., Knapek, S.: Optimized tensor-product approximation spaces. Constr. Approx. 16(4), 525–540 (2000). https://doi.org/10.1007/s003650010010

26. Griebel, M., Schneider, M., Zenger, C.: A combination technique for the solution of sparse grid problems. In: de Groen, P., Golub, G., Ruuth, S. (eds.) Iterative Methods in Linear Algebra, pp. 263–281. IMACS, Elsevier, North Holland (1992)

27. Schwab, Ch., Todor, R.A.: On a multilevel preconditioner and its condition numbers for the discretized laplacian on full and sparse grids in higher dimensions. In: Singular Phenomena and Scaling in Mathematical Models, pp. 263–296. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-00786-1_12

28. Zeiser, A.: Fast matrix-vector multiplication in the sparse-grid Galerkin method. J. Sci. Comput. 47(3), 328–346 (2011). https://doi.org/10.1007/s10915-010-9438-2

29. Widmer, G.: An efficient sparse finite element solver for the radiative transfer equation. J. Heat Transfer. (2009). https://doi.org/10.1115/1.4000190
30. Reisinger, C., Wittum, G.: On multigrid for anisotropic equations and variational inequalities "Pricing multi-dimensional European and American options." Comput. Visual Sci. 7(3), 189–197 (2004). https://doi.org/10.1007/s00791-004-0149-9

31. Anderson, R., Andrej, J., Barker, A., Bramwell, J., Camier, J.-S., Dobrev, J.C.V., Dudouit, Y., Fisher, A., Kolev, T.Z., Pazner, W., Stowell, M., Tomov, V., Akkerman, I., Dahm, J., Medina, D., Zampini, S.: MFEM: A modular finite element methods library. Comput. Math. Appl. 81, 42–74 (2021). https://doi.org/10.1016/j.camwa.2020.06.009

32. Pham, N., Helluy, P., Crestetto, A.: Space-only hyperbolic approximation of the Vlasov equation. ESAIM Proc. 43, 17–36 (2013). https://doi.org/10.1051/proc/201343002

33. Keßler, T., Rjasanow, S., Weißer, S.: Vlasov-Poisson system tackled by particle simulation utilising boundary element methods. SIAM J. Sci. Comput. 42(1), 299–326 (2020). https://doi.org/10.1137/18M1225823