First-order continuous- and discontinuous-Galerkin moment models for a linear kinetic equation: realizability-preserving splitting scheme and numerical analysis

Florian Schneider\textsuperscript{a}, Tobias Leibner\textsuperscript{b,1}

\textsuperscript{a}Fachbereich Mathematik, TU Kaiserslautern, Erwin-Schrödinger-Str., 67663 Kaiserslautern, Germany, schneider@mathematik.uni-kl.de

\textsuperscript{b}Fachbereich Mathematik und Informatik, WWU Münster, Einsteinstrasse 62, 48149 Münster, tobias.leibner@uni-muenster.de

Abstract

We derive a second-order realizability-preserving scheme for moment models for linear kinetic equations. We apply this scheme to the first-order continuous (HFM\textsubscript{n}) and discontinuous (PMM\textsubscript{n}) models in slab and three-dimensional geometry derived in [56] as well as the classical full-moment M\textsubscript{N} models. We provide extensive numerical analysis as well as our code to show that the new class of models can compete or even outperform the full-moment models in reasonable test cases.

Keywords: moment models, minimum entropy, kinetic transport equation, continuous Galerkin, discontinuous Galerkin, realizability

1. Introduction

We consider moment closures, which are a type of (non-linear) Galerkin projection, in the context of kinetic transport equations. Here, moments are defined by taking velocity- or phase-space averages with respect to some (truncated) basis of the velocity space. Unfortunately, the truncation inevitably comes at the cost that information is required from the basis elements which were removed.

The specification of this information, the so-called moment closure problem, distinguishes different moment methods. In the context of linear radiative transport, the standard spectral method is commonly referred to as the P\textsubscript{N} closure [36], where \(N\) is the degree of the highest-order moments in the model. The P\textsubscript{N} method is powerful and simple to implement, but does not take into account the fact that the original function to be approximated, the kinetic density, must be non-negative. Thus, P\textsubscript{N} solutions can contain negative values for the local densities of particles, rendering the solution physically meaningless. Entropy-based moment closures, typically denoted by M\textsubscript{N} models in the context of radiative transport [18, 41], have (for physically relevant entropies) all the properties one would desire in a moment method, namely positivity of the underlying kinetic density, hyperbolicity of the closed system of equations, and entropy dissipation [35]. These models are usually comparatively expensive as they require the numerical solution of an optimization problem at every point on the space-time grid. Practical interest in such models increased recently due to their inherent parallelizability [25]. While the cost of solving the local nonlinear problems in the M\textsubscript{N} model scales strongly with the number of moments \(n\) (since one has to solve square problems of size \(n\)), the desired

\textsuperscript{1}Funding: The author acknowledges funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany’s Excellence Strategy EXC 2044–390685587, Mathematics Münster: Dynamics–Geometry–Structure.

Preprint submitted to arXiv.org
spectral convergence with respect to the moment order \( N \) is only achieved for smooth test cases, which rarely occur in reality. This means that the gain in efficiency by increasing the order of approximation will become rather insignificant.

To increase the accuracy of the \( M_N \) models while maintaining the lower cost for small moment order \( N \), a partition of the velocity space while keeping the moment order fixed is useful, similar to some h-refinement for, e.g., finite element approximations [6]. We focus on the continuous and discontinuous piece-wise linear bases derived in [56], which aim to be a generalization of the special cases provided in [19, 20, 44, 53, 57] in slab geometry and the fully three-dimensional case.

Besides their inherent parallelizability, in order to make these methods truly competitive with more basic discretizations, the gains in efficiency that come from higher-order methods (in space and time) are necessary. Here the issue of realizability becomes a stumbling block. The property of positivity implies that the system of moment equations only evolves on the set of so-called realizable moments. Realizable moments are simply those moments associated with positive densities, and the set of these moments forms a convex cone which is a strict subset of all moment vectors. This property, even though desirable due to its consistency with the original kinetic distribution, can cause problems in numerical simulations. Standard high-order numerical solutions (in space and time) to the Euler equations, which indeed are an entropy-based moment closure, have been observed to have negative local densities and pressures [62]. Similar effects have been reported in the context of elastic flow [46]. This is exactly loss of realizability.

We propose a second-order realizability-preserving scheme, that is based on a splitting technique and analytic solutions of the stiff part, combined with a realizability-preserving reconstruction scheme. It turns out that this scheme is very effective for (medium) smooth and non-smooth test cases, which can also occur in practice. The realizability-preserving property is achieved using the realizability limiter proposed in [2, 15, 51, 54]. This limiter requires information about the set of realizable moments, which turns out to be very simple in the context of our first-order models [56]. Again, this additionally makes the implementation of such models faster (and easier) compared to standard \( M_N \) models.

This paper is organized as follows. First, we shortly recall the transport equation, its moment approximations and the relevant results from [56] (Sections 2 and 3). Then, we propose our second-order realizability-preserving scheme and investigate all the required properties that it should fulfill (Section 4). In Section 5, we discuss some implementation details of our scheme. Finally, in Section 6, we give a comprehensive numerical investigation of our models and the \( M_N \) models in slab geometry and three dimension, to show that our models can indeed compete with or even outperform the full-moment models.

### 2. Modeling

This section closely follows the corresponding part in [56]. We consider the linear transport equation

$$
\partial_t \psi + \Omega \cdot \nabla_x \psi + \sigma_s \psi = \sigma_s \mathcal{C}(\psi) + Q,
$$

(2.1a)

which describes the density of particles with speed \( \Omega \in S^2 \) at position \( x = (x, y, z)^T \in X \subseteq \mathbb{R}^3 \) and time \( t \) under the events of scattering (proportional to \( \sigma_s(t, x) \)), absorption (proportional to \( \sigma_a(x) \)) and emission (proportional to \( Q(x, \Omega) \)). Collisions are modeled using the BGK-type collision operator

$$
\mathcal{C}(\psi) = \int_{S^2} K(\Omega, \Omega') \psi(t, x, \Omega') \, d\Omega' - \int_{S^2} K(\Omega', \Omega) \psi(t, x, \Omega) \, d\Omega'.
$$

(2.1b)

The collision kernel \( K \) is assumed to be strictly positive, symmetric (i.e. \( K(\Omega, \Omega') = K(\Omega', \Omega) \)) and normalized to \( \int_{S^2} K(\Omega', \Omega) \, d\Omega' = 1 \). In this paper, we restrict ourselves to isotropic scattering, where

$$
K(\Omega, \Omega') \equiv \frac{1}{|S^2|} = \frac{1}{4\pi}.
$$

2
The equation is supplemented with initial condition and Dirichlet boundary conditions:
\[
\psi(0, x, \Omega) = \psi_{t=0}(x, \Omega) \quad \text{for } x \in X, \Omega \in S^2 \quad (2.1c)
\]
\[
\psi(t, x, \Omega) = \psi_t(x, \Omega) \quad \text{for } t \in T, x \in \partial X, n \cdot \Omega < 0 \quad (2.1d)
\]
where \( n \) is the outward unit normal vector in \( x \in \partial X \). Parameterizing \( \Omega \) in spherical coordinates we obtain
\[
\Omega = \left( \sqrt{1-\mu^2} \cos(\varphi), \sqrt{1-\mu^2} \sin(\varphi), \mu \right)^T := (\Omega_x, \Omega_y, \Omega_z)^T \quad (2.2)
\]
where \( \varphi \in [0, 2\pi] \) is the azimuthal and \( \mu \in [-1, 1] \) the cosine of the polar angle.

**Definition 2.1.** The vector of functions \( b : S^2 \to \mathbb{R}^n \) consisting of \( n \) basis functions \( b_l, l = 0, \ldots, n-1 \) of maximal order \( N \) (in \( \Omega \)) is called an angular basis.

The so-called moments \( u = (u_0, \ldots, u_{n-1})^T \) of a given distribution function \( \psi \) are then defined by
\[
u = \int_{S^2} b \psi \, d\Omega = \langle b \psi \rangle \quad (2.3)
\]
where the integration is performed component-wise.

Furthermore, the quantity \( \rho = \rho(u) := \langle \psi \rangle \) is called the local particle density. Additionally, \( u_{iso} = \langle b \rangle \) is called the isotropic moment.

Equations for \( u \) can then be obtained by multiplying (2.1) with \( b \) and integration over \( S^2 \), resulting in
\[
\partial_t u + \nabla_x \cdot (\Omega b \psi) + \sigma_a u = \sigma_s (b C (\psi)) + \langle b Q \rangle . \quad (2.4)
\]
Depending on the choice of \( b \) the terms \( \langle \Omega_x b \psi \rangle, \langle \Omega_y b \psi \rangle, \langle \Omega_z b \psi \rangle \), and in some cases even \( \langle b C (\psi) \rangle \), cannot be given explicitly in terms of \( u \). Therefore an ansatz \( \hat{\psi} \) has to be made for \( \psi \) closing the unknown terms. This is called the moment-closure problem.

In this paper the ansatz density \( \hat{\psi} \) is reconstructed from the moments \( u \) by minimizing the entropy-functional
\[
\mathcal{H}(\psi) = \langle \eta(\psi) \rangle \quad \text{under the moment constraints } \langle b \psi \rangle = u. \quad (2.5)
\]
The kinetic entropy density \( \eta : \mathbb{R} \to \mathbb{R} \) is strictly convex and twice continuously differentiable and the minimum is simply taken over all functions \( \psi = \psi(\Omega) \) such that \( \mathcal{H}(\psi) \) is well defined. This problem, which must be solved over the space-time mesh, is typically solved through its strictly convex finite-dimensional dual,
\[
\alpha(u) := \argmin_{\alpha \in \mathbb{R}^n} \langle \eta_s (b \cdot \tilde{\alpha}) - u \cdot \tilde{\alpha} \rangle \quad (2.6)
\]
where \( \eta_s \) is the Legendre dual of \( \eta \). The first-order necessary conditions for the multipliers \( \alpha(u) \) show that the solution to (2.5), if it exists, has the form
\[
\hat{\psi}_u = \eta_s' (b \cdot \alpha(u)) \quad (2.7)
\]
This approach is called the minimum-entropy closure [35]. The resulting model has many desirable properties: symmetric hyperbolicity, bounded eigenvalues of the directional flux Jacobian and the direct existence of an entropy-entropy flux pair (compare [35, 52]).

The kinetic entropy density \( \eta \) can be chosen according to the physics being modelled. As in [25, 35], Maxwell-Boltzmann entropy
\[
\eta(\psi) = \psi \log(\psi) - \psi \quad (2.8)
\]
is used, thus \( \eta_s(p) = \eta'_s(p) = \exp(p) \). This entropy is used for non-interacting particles as in an ideal gas.

Substituting \( \psi \) in (2.4) with \( \hat{\psi}_u \) yields a closed system of equations for \( u \):
\[
\partial_t u + \partial_x \left( \Omega_x b \hat{\psi}_u \right) + \partial_y \left( \Omega_y b \hat{\psi}_u \right) + \partial_z \left( \Omega_z b \hat{\psi}_u \right) + \sigma_a u = \sigma_s \left( b C (\hat{\psi}_u) \right) + \langle b Q \rangle . \quad (2.9)
\]
Remark 2.2. Note that using the entropy \( \eta(\psi) = \frac{1}{2} \psi^2 \) the linear ansatz
\[
\hat{\psi}_u = b \cdot \alpha(u)
\]  
(2.10)
is obtained, leading to standard continuous/discontinuous-Galerkin approaches. If the angular basis is chosen as spherical harmonics of order \( N \), (2.9) turns into the classical \( P_N \) model [9, 11, 58].

For convenience, we write (2.9) in the standard form of a non-linear hyperbolic system of partial differential equations:
\[
\partial_t u + \partial_x f_1(u) + \partial_y f_2(u) + \partial_z f_3(u) = s(u),
\]  
(2.11)
where
\[
f_1(u) = \langle \Omega_x b \hat{\psi}_u \rangle, \quad f_2(u) = \langle \Omega_y b \hat{\psi}_u \rangle, \quad f_3(u) = \langle \Omega_z b \hat{\psi}_u \rangle \in \mathbb{R}^n,
\]  
(2.12a)

\[
s(x,u) = \sigma_s(x) \left( \langle bC(\hat{\psi}_u) \rangle + (bQ(x,\cdot)) - \sigma_a(x) u \right).
\]  
(2.12b)

For ease of visibility, we also consider our models in slab geometry, which is a projection of the sphere onto the \( z \)-axis [58]. The transport equation under consideration then has the form
\[
\partial_t \psi + \mu \partial_x \psi + \sigma_a \psi = \sigma_s C(\psi) + Q, \quad t \in T, z \in X, \mu \in [-1,1].
\]  
(2.13)
The shorthand notation \( \langle \cdot \rangle = \int_{-1}^{1} \cdot d\mu \) then denotes integration over \([-1,1]\) instead of \( S^2 \). Finally, the moment system is given by
\[
\partial_t u + \partial_x \left( \mu \langle b \hat{\psi}_u \rangle \right) + \sigma_a u = \sigma_s \left( \langle bC(\hat{\psi}_u) \rangle + (bQ) \right).
\]  
(2.14)

3. Angular bases

We shortly recall the angular bases under consideration. For a detailed derivation and further information, we refer the reader to [56].

3.1. Slab geometry

- Full-moment basis
  \[
f_N = (1,\mu,\ldots,\mu^N)^T
  \]  
(3.1a)
  or
  \[
f_N = (P_0^0, P_1^0, P_2^0, \ldots, P_N^0)^T
  \]  
(3.1b)
  with the monomials or the Legendre polynomials \( P_l^0, l = 0,\ldots,N \).

- Piecewise-linear angular basis (hat functions, continuous-Galerkin ansatz) \( h_n = (h_0,\ldots,h_{n-1})^T \)
  \[
h_l(\mu) = \mathbb{I}_{l-1} \frac{\mu - \mu_{l-1}}{\mu_l - \mu_{l-1}} + \mathbb{I}_l \frac{\mu - \mu_{l+1}}{\mu_l - \mu_{l+1}},
  \]  
(3.2)
  where \(-1 = \mu_0 < \mu_1 < \ldots < \mu_{n-2} < \mu_{n-1} = 1\) are some angular “grid” points and \( \mathbb{I}_l(\mu) \) is the indicator function on the interval \( I_l = [\mu_l,\mu_{l+1}] \) (with \( \mathbb{I}_{l-1} \equiv \mathbb{I}_{n-1} \equiv 0 \)).

- Partial moments (discontinuous-Galerkin ansatz) \( p = (p_{h_0},\ldots,p_{h_{k-1}}) \)
  \[
p_{I_l} = \mathbb{I}_{I_l} (1,\mu)^T,
  \]  

where \( k \) is the number of intervals.

Definition 3.1. The resulting linear (compare (2.10)) and nonlinear models (compare (2.8)) will be called \( P_N/M_N \) (full moment basis), \( HFP_n/HFM_n \) (hat functions basis) and \( PMP_n/PMM_n \) (partial moment basis), respectively.
3.2. Angular bases in three dimensions

Albeit both approaches are not limited to this, we consider moments on spherical triangles. To that end, let $T_h$ be a spherical triangulation of $S^2$ and $\bar{K} \in T_h$ be a spherical triangle. In this paper, the triangulation $\mathcal{P}$ will be obtained by dyadic refinement of the octants of the sphere $V = S^2$, i.e. the coarsest triangulation contains the eight spherical triangles obtained by projecting the octahedron with vertices $\{(\pm1, 0, 0)^T, (0, \pm1, 0)^T, (0, 0, \pm1)^T\}$ to the sphere and finer partitions are obtained by iteratively subdividing each spherical triangle into four new ones, adding vertices at the midpoints of the triangle edges. After $r$ refinements, we thus obtain $n_v(r) = 4^{r+1} + 2$ vertices and $n_t(r) = 2 \cdot 4^{r+1}$ spherical triangles.

The bases that we use are the following.

- **Full-moment basis**
  \[ f_N = \left( S^m_l(\mu, \varphi); \quad l = 0, \ldots, N, \quad m = -l, \ldots, l \right)^T, \]
  where $S^m_l$ are the real-valued spherical harmonics on the unit sphere [9, 58].

- **Barycentric-coordinate basis functions**
  \[ h_{n_v} = \left( h_0, \ldots, h_{n_v-1} \right), \]
  where $n_v$ is the number of vertices of the triangulation and $h_l$ is the basis function defined using spherical barycentric coordinates on the $l$-th vertex as in [14, 30, 45].

- **Partial moments on the unit sphere**
  \[ p_n = \left( p^{\bar{K}}_{\mu}; \quad \bar{K} \in T_h \right) = \left( \frac{1}{\bar{K}}, \frac{1}{\bar{K}} \Omega; \quad \bar{K} \in T_h \right), \]
  where $n = 4 \cdot |T_h|$ is the number of moments.

Naming of the models will be analogous to the slab-geometry case, compare Definition 3.1.

3.3. Realizability

The minimum-entropy moment problem (2.5) has a solution if and only if the moment vector is realizable.

**Definition 3.2.** The realizable set $\mathcal{R}_b$ is

\[ \mathcal{R}_b = \{ u \in \mathbb{R}^n : \exists \psi(\Omega) \geq 0, \quad \rho = \langle \psi \rangle > 0, \text{ such that } u = \langle b \psi \rangle \}. \]

If $u \in \mathcal{R}_b$, then $u$ is called realizable. Any $\psi$ such that $u = \langle b \psi \rangle$ is called a representing density.

Unfortunately, checking whether a moment vector is realizable is not trivial for general bases. However, for the piecewise linear moment models, the realizability conditions are particularly simple (see [56]).

**Lemma 3.3.** For the hat function basis in one or three dimensions, $u \in \text{cl}(\mathcal{R}_{h_n})$ if and only if $u_l \geq 0$ for all $l = 0, \ldots, n-1$.

**Lemma 3.4.** For the partial moment basis in one dimension (slab geometry), $u \in \text{cl}(\mathcal{R}_{p_n})$ if and only if

\[ u_{2i} \geq 0 \quad \text{and, for } u_{2i} > 0, \quad \frac{u_{2i+1}}{u_{2i}} \in I_i = [\mu_i, \mu_{i+1}] \quad (3.3) \]

for all $i = 0, \ldots, \frac{n}{2} - 1$.

For more details on the realizability of the regarded models, see [56].
4. Second-order realizability-preserving splitting scheme

As already mentioned before, the minimum-entropy moment problem (2.5) has a solution if and only if the moment vector is realizable. This implies that it is mandatory to maintain realizability during the numerical simulation (since otherwise the flux function cannot be evaluated). Explicit high-order schemes have been developed in [2, 54]. Unfortunately, the physical parameters $\sigma_s$ and $\sigma_a$ directly influence the CFL condition, resulting in very small time steps for large scattering/absorption.

This can be overcome by using a first-order implicit-explicit time stepping scheme [48, 49, 51], treating the transport part explicit while implicitly solving the (time-)critical source term. Unfortunately, using higher-order IMEX schemes again results in a CFL condition of the same magnitude as for the fully explicit schemes.

We are interested in a second-order scheme for (2.11). This can be achieved by doing a Strang splitting for

\[ \partial_t u + \partial_x f_1(u) + \partial_y f_2(u) + \partial_z f_3(u) = 0, \]

\[ \partial_t u = s(x, u). \]

A second-order realizability preserving scheme will be obtained if both subsystems are solved with a (at least) second-order accurate and realizability-preserving scheme. For notational simplicity, we show the full scheme for one spatial dimension only. A generalization to structured meshes in higher dimensions is straightforward.

4.1. Source system

Let us start with the stiff part (4.1b) whose finite-volume form is given by

\[ \partial_t \mathbf{u}_i = \frac{1}{\Delta z} \int_{z_{i-\frac{1}{2}}}^{z_{i+\frac{1}{2}}} s(z, \mathbf{u}) \, dz. \]  

(4.2)

Fortunately, using the midpoint rule, it holds that

\[ s(z, \mathbf{u})_i = \frac{1}{\Delta z} \int_{z_{i-\frac{1}{2}}}^{z_{i+\frac{1}{2}}} s(z, \mathbf{u}) \, dz = s(z_i, \mathbf{u}_i) + O(\Delta z^2). \]  

(4.3)

To obtain a second-order accurate solution of (4.2), it is thus sufficient to solve the system

\[ \partial_t \mathbf{u}_i = s(z_i, \mathbf{u}_i), \]  

(4.4)

which is purely an ODE (in every cell). As mentioned above, we restrict ourselves to isotropic scattering, where we have $K(\mu, \mu') = \frac{1}{(1)} = \frac{1}{2}$, i.e.

\[ C(\psi) = \frac{\langle \psi \rangle}{\langle 1 \rangle} - \psi. \]  

(4.5)

The source term now becomes

\[ s(z, \mathbf{u}) = \sigma_s \left( \mathbf{b}C \left( \hat{\psi}_u \right) \right) + \langle \mathbf{b}Q \rangle - \sigma_a \mathbf{u} = \sigma_s \frac{\rho(u)}{\langle 1 \rangle} (\mathbf{b}) + \langle \mathbf{b}Q \rangle - \sigma_t \mathbf{u} \]

\[ = \sigma_s \mathbf{u}_{iso}(\mathbf{u}) + \langle \mathbf{b}Q \rangle - \sigma_t \mathbf{u} = (\sigma_s \mathbf{G} - \sigma_t \mathbf{I}) \mathbf{u} + \langle \mathbf{b}Q \rangle, \]  

(4.6)
where $\mathbf{G} = \frac{\mathbf{u}_{\text{iso}}(\alpha \mathbf{t})^T}{(t)}$ is the matrix mapping the moment vector $\mathbf{u}$ to the isotropic moment vector with the same density $\mathbf{u}_{\text{iso}}(\mathbf{u}) = \mathbf{u}_{\text{iso}} \cdot \frac{\rho(\mathbf{u})}{(t)}$. Here we assumed that there exists a vector $\alpha_0^T$ such that $\alpha_0^T \cdot \mathbf{b} = 1$ (true for all regarded bases: $\alpha_0^T = (1, 0, \ldots, 0)^T$ for Legendre Polynomials, $\alpha_0^T = (\sqrt{\frac{4\pi}{\sigma}}, 0, \ldots, 0)^T$ for real spherical harmonics, $\alpha_0^T = (1, 1, \ldots)^T$ for the hat functions basis, $\alpha_0^T = (1, 0, 1, 0, 0, \ldots)^T$ for the partial moments in slab geometry and $(1, 0, 0, 1, 0, 0, \ldots)^T$ for the partial moment basis in three dimensions).

Since in this case, (4.4) is linear and the parameters $\sigma_x, \sigma_y$, $Q$ are time-independent, we solve it explicitly using matrix exponentials, trivially obtaining a realizable second-order accurate solution of (4.2).

Remark 4.1. Note that in this specific situation, the solution of this sub-step does not depend on the moment closure used in the flux system.

Using the matrix exponential and the variation of constants formula, the solution to (4.4) is

$$
\mathbf{u}(t, z) = \mathbf{e}^{\langle \sigma_x \mathbf{G} - \sigma_y \mathbf{I} \rangle t} \mathbf{u}(0, z) + \left( \int_0^t \mathbf{e}^{\langle \sigma_x \mathbf{G} - \sigma_y \mathbf{I} \rangle (t - s)} ds \right) \langle \mathbf{b} \mathbf{Q} \rangle \quad (4.7)
$$

As $\mathbf{G}$ and $\mathbf{I}$ commute, we have

$$
\mathbf{e}^{\langle \sigma_x \mathbf{G} - \sigma_y \mathbf{I} \rangle t} = \mathbf{e}^{\langle \sigma_x \mathbf{G} \rangle t} \mathbf{e}^{\langle -\sigma_y \mathbf{I} \rangle t} = \mathbf{e}^{\langle \sigma_x \mathbf{G} \rangle t} (\mathbf{e}^{\langle -\sigma_y \mathbf{I} \rangle}) \quad (4.8)
$$

It remains to compute the matrix exponential of $\sigma_x \mathbf{G}$. As $\mathbf{G} \mathbf{u}_{\text{iso}}(\mathbf{u}) = \mathbf{u}_{\text{iso}}(\mathbf{u})$, we have that $\mathbf{G}^k = \mathbf{G}$ for all $k \geq 1$. It follows

$$
\mathbf{e}^{\sigma_x \mathbf{G} t} = \sum_{k=0}^{\infty} \frac{(\sigma_x \mathbf{G} t)^k}{k!} = \mathbf{I} + \sum_{k=1}^{\infty} \frac{(\sigma_x t)^k}{k!} \mathbf{G} = \mathbf{I} + (\mathbf{e}^{\sigma_x t} - 1) \mathbf{G} \quad (4.9)
$$

Inserting (4.9) in (4.8), we get

$$
\mathbf{e}^{\langle \sigma_x \mathbf{G} - \sigma_y \mathbf{I} \rangle t} = \mathbf{e}^{\langle -\sigma_y \mathbf{I} \rangle} (\mathbf{I} + (\mathbf{e}^{\sigma_x t} - 1) \mathbf{G}) \quad (4.10)
$$

Plugging (4.10) into (4.7), we finally get

$$
\mathbf{u}(t) = \mathbf{e}^{\langle -\sigma_y \mathbf{I} \rangle} (\mathbf{I} + (\mathbf{e}^{\sigma_x t} - 1) \mathbf{G}) \mathbf{u}(0, z)
$$

$$
+ \left( \int_0^t \mathbf{e}^{\langle -\sigma_y (t - s) \rangle} (\mathbf{I} + (\mathbf{e}^{\sigma_x (t - s)} - 1) \mathbf{G}) ds \right) \langle \mathbf{b} \mathbf{Q} \rangle \quad (4.11)
$$

If the source is also isotropic then $\mathbf{G} \langle \mathbf{b} \mathbf{Q} \rangle = \langle \mathbf{b} \mathbf{Q} \rangle = \langle \mathbf{b} \rangle Q$ and (4.11) simplifies to

$$
\mathbf{u}(t, z) = \mathbf{e}^{\langle -\sigma_y \mathbf{I} \rangle} (\mathbf{e}^{\langle -\sigma_y \mathbf{I} \rangle} \mathbf{u}(0, z) + (1 - \mathbf{e}^{\langle -\sigma_y \mathbf{I} \rangle}) \mathbf{u}_{\text{iso}}(\mathbf{u}(0, z)))
$$

$$
+ \left( \frac{1 - \exp(-\sigma_y t)}{\sigma_y} (\mathbf{I} - \mathbf{G}) + \frac{1 - \exp(-\sigma_y t)}{\sigma_y} \mathbf{G} \right) \langle \mathbf{b} \mathbf{Q} \rangle \quad (4.12)
$$

which can easily be calculated without explicit calculation of $\mathbf{G}$ or any matrix operations.
4.2. Flux system

Let us now consider the non-stiff part (4.1a). This can be solved using standard realizability-preserving methods [2, 15, 52, 54], which will be summarized in the following.

The standard finite-volume scheme in semi-discrete form for (4.1a) looks like

$$\partial_t u_i = g(u_{i+\frac{1}{2}}, u_{i+\frac{1}{2}}) - g(u_{i-\frac{1}{2}}, u_{i-\frac{1}{2}}),$$

(4.13)

where \( g \) is a numerical flux function. The simplest example is the global Lax-Friedrichs flux

$$g(u_1, u_2) = \frac{1}{2} (f_i(u_1) + f_i(u_2) - C(u_2 - u_1)).$$

(4.14)

The numerical viscosity constant \( C \) is taken as the global estimate of the absolute value of the largest eigenvalue of the Jacobian \( f' \). In our case, the viscosity constant can be set to \( C = 1 \), because for the moment systems used here the largest eigenvalue is bounded in absolute value by one [2, 42, 52].

Another possible choice is the kinetic flux [20, 22, 25, 54]

$$g(u_1, u_2) = \left\langle \mu b \right\rangle_+ + \left\langle \mu b \right\rangle_-,$n

(4.15)

where \( \langle \cdot \rangle_+ \) and \( \langle \cdot \rangle_- \) denote integration over the positive and negative half intervals \([-1, 0]\) and \([0, 1]\), respectively. The kinetic flux is less diffusive than the (global) Lax-Friedrichs flux and admits a more consistent implementation of kinetic boundary conditions (see [52] and Section 5.5). For this reason, we will use (4.15) in all our computations.

4.2.1. Polynomial reconstruction

The value \( u_{i+\frac{1}{2}} \) is the evaluation of a suitable linear reconstruction of \( u \) at the cell interface \( z_{i+\frac{1}{2}} \). In one dimension, it can be obtained from a minmod reconstruction\(^2\)

$$u_i(z) = \bar{u}_i + u'_i(z - z_i)$$

$$u'_i = \frac{1}{\Delta z} \minmod\left(\bar{u}_{i+1} - \bar{u}_i, \bar{u}_i - \bar{u}_{i-1}, \frac{1}{2}(\bar{u}_{i+1} - \bar{u}_{i-1})\right),$$

where \( \minmod(\cdot) \) is the minmod function

$$\minmod(a_1, a_2, a_3) = \begin{cases} 
\text{sign}(a_1) \min(|a_1|, |a_2|, |a_3|) & \text{if sign}(a_1) = \text{sign}(a_2) = \text{sign}(a_3), \\
0 & \text{else}.
\end{cases}$$

applied componentwise. We then set \( u_{i+\frac{1}{2}} = u_i(z_{i+\frac{1}{2}}) \) and \( u^+_{i+\frac{1}{2}} = u_{i+1}(z_{i+\frac{1}{2}}) \).

To avoid spurious oscillations, the reconstruction has to be performed in characteristic variables. They are found by transforming the moment vector \( u \) using the matrix \( V_i \), whose columns hold the eigenvectors of the Jacobian \( f'(\bar{u}_i) \) evaluated at the cell mean \( \bar{u}_i \). This leads to

$$u'_i = \frac{1}{\Delta z} V_i \minmod\left(V_i^{-1}(\bar{u}_{i+1} - \bar{u}_i), V_i^{-1}(\bar{u}_i - \bar{u}_{i-1}), \frac{1}{2} V_i^{-1}(\bar{u}_{i+1} - \bar{u}_{i-1})\right).$$

(4.16)

For details on the eigenvalue computation see Section 5.2. In several dimension, we perform a dimension-by-dimension reconstruction as in [60] using the minmod reconstruction in characteristic variables in each one-dimensional reconstruction step.

\(^2\)Other second-order accurate reconstructions like WENO [16, 28] are also possible.
While this already gives us a second-order scheme, we do not have the realizability-preserving property yet. To achieve this, we need to apply a realizability limiter, ensuring that $u_i(z)$ is point-wise realizable at the interface nodes $z \in \{z_{i-\frac{1}{2}}, z_{i+\frac{1}{2}}\}$. We follow the construction from [2].

We replace $u_i$ with the limited version $u_\theta^i = \theta u_i + (1 - \theta) u_i'(z - z_i)$.

The limiter variable $\theta \in [0, 1]$ dampens the reconstruction from unlimited ($\theta = 0$) to first-order ($\theta = 1$). Assuming that $u_i \in \mathbb{R}^b$ is realizable, there exists at least one $\theta$ (namely $\theta = 1$) such that $u_\theta^i(z) \in \text{cl}(\mathbb{R}_b)$ for every $z$ in the set of quadrature nodes (where $\text{cl}(\mathbb{R}_b)$ is the closure of $\mathbb{R}_b$). Since the realizable set is a convex cone, and by continuity, it is guaranteed that there exists a minimal $\theta$ satisfying this assumption. We are thus searching for the solution of the minimization problem

$$\max_{z \in \{z_{i-\frac{1}{2}}, z_{i+\frac{1}{2}}\}} \min_{\theta \in [0,1]} \quad \text{s. t.} \quad u_\theta^i(z) \in \text{cl}(\mathbb{R}_b)$$

In practice, given some interface node $z$, we search for the intersection of the line $u_\theta^i(z)$ (wrt. $\theta$) with the boundary of the realizable set, check if the value is in $[0, 1]$ and store it in the case that it is.

For the presented first-order moment models, the solution of the above limiter problem can often be computed explicitly (see Section 5.3 for more details).

If we discretize (4.13) with a second-order SSP Runge-Kutta (RK) scheme, e.g. Heun’s method or the general $s$ stage SSP ERK$_2$ [23, 29], a realizability-preserving scheme is obtained under a CFL-like condition if reconstruction and limiting is performed in every stage of the RK method, see Lemma 4.5.

### 4.2.3. Solving the optimization problem

For the minimum-entropy models, in each stage of the time stepping scheme for (4.1a), we have to solve the optimization problem (2.5) once in each cell (to compute the Jacobians) and twice at each interface of the computational mesh (one optimization problem for the left and right reconstructed value at the interface, respectively). This usually accounts for the majority of computation time which makes it mandatory to pay special attention to the implementation of the optimization algorithm. In this section, we will focus on the stopping criteria for the optimization algorithm. For details on the implementation, see Section 5.1.

Recall that the objective function in the dual problem (2.6) is

$$p_u(\alpha) = \langle \eta_u(b \cdot \alpha) \rangle - u \cdot \alpha.$$  

(4.19)

The gradient and Hessian of $p$ are given by

$$g_u(\alpha) = \nabla_\alpha p(\alpha) = \langle b\eta_u'(b \cdot \alpha) \rangle - u$$

and

$$H(\alpha) = D_\alpha g(\alpha) = \langle bb^T \eta_u''(b \cdot \alpha) \rangle,$$

(4.20)

(4.21)

respectively. Note that $\eta_u'' > 0$ since we assumed that $\eta$ (and thus also $\eta_u$) is strictly convex, and remember that the basis functions contained in $b$ are linearly independent. As a consequence, the Hessian $H$ is symmetric positive definite.

To find a minimizer of $p$, we are searching for a root of the gradient $g$ using Newton’s method. For simplicity, we will restrict ourselves to Maxwell-Boltzmann entropy (2.8) such that $\eta_u'(\alpha \cdot b) = \exp(\alpha \cdot b)$. We use
the Newton algorithm from [2, 52, 55] with some adaptions. Before entering the algorithm for the moment vector \(u\) we rescale it to

\[
\phi := \frac{u}{\rho(u)}
\]  

(4.22)
such that \(\rho(\phi) = 1\). Let \(\varrho(\alpha) = \langle \exp(b \cdot \alpha) \rangle\) be the mapping \(\alpha \mapsto \rho(u(\alpha))\) which maps a set of multipliers \(\alpha\) to the density of its associated moment. If the optimization algorithm for \(\phi\) stops at an iterate \(\beta\), we return

\[
\tilde{\alpha} = \beta + \alpha^1_b \log \left( \frac{\rho(u)}{\varrho(\beta)} \right)
\]  

(4.23)
where \(\alpha^1_b\) is the multiplier with the property that \(\alpha^1_b \cdot b \equiv 1\) (see Section 4.1). This ensures that the local particle density is preserved exactly:

\[
\varrho(\tilde{\alpha}) = \langle \exp(b \cdot \tilde{\alpha}) \rangle = \langle \exp(b \cdot \beta) \rangle \frac{\rho(u)}{\varrho(\beta)} = \rho(u).
\]  

(4.24)
Given \(\tau \in \mathbb{R}^+\), \(\varepsilon, \gamma \in (0, 1)\), we will stop the Newton iteration at iterate \(\beta\) if

\[
(1) \quad \|g_\phi(\beta)\|_2 < \tau' := \begin{cases} 
\frac{\tau}{1 + \|\phi\|_2 + \tau} & \text{if } b = f_N \\
\frac{\tau}{1 + \sqrt{n} \|\phi\|_2 + \sqrt{n} \tau} & \text{if } b \in \{h_n, p_n\}, \end{cases}
\]  

(4.25a)
and

\[
(2) \quad u - (1 - \varepsilon, \gamma)u(\tilde{\alpha}) \in \mathcal{R}^+_n,
\]  

(4.25b)
where \(\tilde{\alpha}\) is obtained from \(\beta\) by (4.23) and, as always, \(n\) is the number of moments.

In the following, we will explain the rationale behind these stopping criteria.

**Lemma 4.2.** Let \(\tau \in \mathbb{R}^+\). If (4.25a) is fulfilled, we have that \(\|g_u(\tilde{\alpha})\|_2 \leq \tau\).

**Proof.** First note that, by its definition (4.20), the gradient can be written as

\[
g_u(\tilde{\alpha}) = \langle b \exp(b \cdot \tilde{\alpha}) \rangle - u = u(\tilde{\alpha}) - u
\]  

(4.26)
where \(u(\tilde{\alpha})\) is the moment vector corresponding to the multipliers \(\tilde{\alpha}\).

Let \(\beta' := \beta - \alpha^1_b \log(\varrho(\beta))\). Then it follows that

\[
\rho(u) \|g_u(\beta')\|_2 = \rho(u) \| \left( b \exp(b \cdot \beta') - \phi \right) \|_2 \\
= \| \left( b \exp \left( b \cdot \left( \beta' + \alpha^1_b \log(\rho(u)) \right) \right) - \rho(u) \phi \|_2 \\
= \|b \exp(b \cdot \tilde{\alpha}) - \rho(u) \phi\|_2 \\
= \|u(\tilde{\alpha}) - u\|_2 = \|g_u(\tilde{\alpha})\|_2.
\]

Consequently, we have

\[
\|g_u(\tilde{\alpha})\|_2 = \rho(u) \|g_u(\beta')\|_2 = \rho(u) \|u(\beta') - \phi\|_2 \\
= \rho(u) \left\| \frac{1}{\varrho(\beta)} u(\beta) - \phi \right\|_2 \\
= \rho(u) \left\| \frac{1}{\varrho(\beta)} u(\beta) - \frac{1}{\varrho(\beta)} \phi + \frac{1}{\varrho(\beta)} \phi - \phi \right\|_2 \\
\leq \rho(u) \left( \frac{1}{\varrho(\beta)} \|u(\beta) - \phi\|_2 + \left\| \frac{1}{\varrho(\beta)} \phi - \phi \right\|_2 \right) \\
= \rho(u) \left( \frac{1}{\varrho(\beta)} \|g_u(\beta)\|_2 + \left\| \frac{1 - \varrho(\beta)}{\varrho(\beta)} \|\phi\|_2 \right\|_2 \right).
\]  

(4.27)
Moreover,
\[
|\varrho(\beta) - 1| = |\varrho(\beta) - \rho(\phi)| \\
= |\rho(u(\beta)) - \rho(\phi)| \\
= |\alpha_b^1 \cdot (u(\beta) - \phi)| \\
= \left\{ \begin{array}{ll}
\sum_{l=0}^{n-1} (u_l(\beta) - \phi_l) & \text{for hat functions,} \\
\sum_{m=0}^{k-1} (u_{m,0}(\beta) - \phi_{m,0}) & \text{for partial moments,} \\
u_{0}(\beta) - \phi_0 & \text{for full moments,}
\end{array} \right.
\]
\quad (4.28)
\]
where in the last step we used the explicit forms of \(\alpha_b^1\) for the different bases (see Section 4.1). Since
\[
u(\beta) - \phi = g_\phi(\beta),
\]
it follows from (4.28) and (4.25a) that
\[
|\varrho(\beta) - 1| \leq \|g_\phi(\beta)\|_1 \leq \sqrt{n} \|g_\phi(\beta)\|_2 \leq \sqrt{n} \tau',
\]
for partial moments and hat functions, and
\[
|\varrho(\beta) - 1| \leq \|g_\phi(\beta)\|_\infty \leq \|g_\phi(\beta)\|_2 \leq \tau'
\]
for full moments, which directly gives
\[
\frac{1}{\varrho(\beta)} \leq \frac{1}{1 - \sqrt{n} \tau'} \quad \text{and} \quad \frac{1}{\varrho(\beta)} \leq \frac{1}{1 - \tau'},
\]
respectively. Inserting these bounds in (4.27), we finally obtain
\[
\|g_u(\alpha)\|_2 \leq \rho(u) \left( \frac{\tau'}{1 - \sqrt{n} \tau'} + \frac{\sqrt{n} \tau'}{1 - \sqrt{n} \tau'} \|\phi\|_2 \right) = \rho(u) \left( \frac{\tau' (1 + \sqrt{n} \|\phi\|_2)}{1 - \sqrt{n} \tau'} \right) = \tau
\]
for partial moments and hat functions, and similarly for full moments, removing \(\sqrt{n}\) accordingly.

The second criterion (4.25b) ensures that the ansatz density (2.7) corresponding to the multiplier \(\tilde{\alpha}\) obtained from the Newton iteration is close enough to a representing density for the moments \(u\).

**Lemma 4.3.** Let \(u \in \mathbb{R}_b^+\) and let \(\varepsilon, \gamma \in (0, 1)\), \(\tilde{\alpha} \in \mathbb{R}^n\) such that the second stopping criterion (4.25b) holds. Then there exists a representing distribution \(\psi\) for \(u\), i.e. \(u = \langle b \psi \rangle\), such that
\[
\frac{\psi}{\psi_u(\tilde{\alpha})} \geq \frac{\psi}{\eta^*_\gamma(\tilde{\alpha} \cdot b)} \geq 1 - \varepsilon_\gamma.
\]
\quad (4.30)

**Proof.** If (4.25b) is satisfied, there exists a positive distribution \(\psi^{\varepsilon_\gamma}\) such that
\[
\langle \psi^{\varepsilon_\gamma} b \rangle = u - (1 - \varepsilon_\gamma)u(\tilde{\alpha}).
\]
\quad (4.31)
Then
\[
\psi := \psi^{\varepsilon_\gamma} + (1 - \varepsilon_\gamma)\tilde{\psi}_u(\tilde{\alpha})
\]
is a positive distribution representing \(u\) and satisfying (4.30).

In Section 4.2.4 we will use Lemma 4.3 to show that the scheme is realizability-preserving although the optimization problems are only solved approximately.
Remark 4.4. Note that
\[ u - (1 - \varepsilon_{\gamma})u = \varepsilon_{\gamma}u \]
is realizable for all \( \varepsilon_{\gamma} > 0 \). Due to the openness of \( \mathcal{R}_u^+ \), there exists a \( \delta > 0 \) s.t. \( \tilde{u} \in \mathcal{R}_u^+ \) for all \( \tilde{u} \) with \( \| \tilde{u} - \varepsilon_{\gamma}u \|_2 < \delta \). Note further that
\[
\| u - (1 - \varepsilon_{\gamma})u(\tilde{\alpha}) - \varepsilon_{\gamma}u \|_2 = (1 - \varepsilon_{\gamma}) \| u - u(\tilde{\alpha}) \|_2
\]
so (4.25b) is fulfilled if \( \| g_u(\tilde{\alpha}) \|_2 \leq \frac{\delta}{1 - \varepsilon_{\gamma}} \), i.e. if our numerical solution to the approximation problem is close enough to the exact solution. For moments \( u \) that are very close to the realizable boundary (so \( \delta \) is very small and in addition \( H \) may be very badly conditioned), we might not be able to achieve such an accuracy. In that case, we either use a regularized version of \( u \) (see (5.6)) or disable linear reconstruction (see Item 4 in Section 5). Choosing \( \varepsilon_{\gamma} \) closer to 1 makes it easier to fulfill (4.25b) at the expense of smaller time steps (see Lemma 4.5). In our computations, we used the value \( \varepsilon_{\gamma} = 0.01 \) which worked well in practice.

4.2.4. Time-step restriction

Now we are able to put all the things together to show that one forward-Euler step of our scheme (4.13) is indeed realizability-preserving.

Lemma 4.5. The finite volume scheme (4.13), using the kinetic flux (4.15) and the stopping criteria from Section 4.2.3, on a rectangular grid in \( d \) dimensions preserves realizability under the CFL-like condition
\[
\Delta t < \frac{1 - \varepsilon_{\gamma}}{2\sqrt{d}} \Delta x. \tag{4.33}
\]

Proof. Adapted from [52, Theorem 3.19]. As we are using time stepping schemes that consist of a convex combination of Euler forward steps, it is enough to show realizability preservation in a single Euler forward step. Consider the one-dimensional (\( d = 1 \)) case first. The update formula in one step is
\[
\begin{align*}
\psi_j^{(k+1)} &= \psi_j^{(k)} - \frac{\Delta t}{\Delta x} \left( \langle \mu b \psi_j^{\tau, -\tau} \rangle + \langle \mu b \psi_{j+\frac{1}{2}}^{\tau, +\tau} \rangle - \langle \mu b \psi_{j+\frac{1}{2}}^{\tau, -\tau} \rangle - \langle \mu b \psi_{j-\frac{1}{2}}^{\tau, +\tau} \rangle \right) \\
&= \langle b \psi_j^{(k)} \rangle - \frac{\Delta t}{\Delta x} \left( \langle \max(\mu, 0) b \left( \psi_j^{\tau, -\tau} - \psi_{j-\frac{1}{2}}^{\tau, -\tau} \right) \rangle + \langle \min(\mu, 0) b \left( \psi_{j+\frac{1}{2}}^{\tau, +\tau} - \psi_{j+\frac{1}{2}}^{\tau, -\tau} \right) \rangle \right) \\
&= \langle b \psi_j^{(k)} \rangle - \frac{\Delta t}{\Delta x} \left( \langle \max(\mu, 0) \left( \psi_j^{\tau, -\tau} - \psi_{j-\frac{1}{2}}^{\tau, -\tau} \right) \rangle + \langle \min(\mu, 0) \left( \psi_{j+\frac{1}{2}}^{\tau, +\tau} - \psi_{j-\frac{1}{2}}^{\tau, -\tau} \right) \rangle \right) \\
&= \langle b \psi_j^{(k+1)} \rangle
\end{align*}
\]
where \( \psi_j^{(k)} \) is an arbitrary representing density for \( \psi_j^{(k)} \) and \( \psi_{j+\frac{1}{2}}^{\tau, +\tau} \) is the ansatz distribution obtained from the approximate solution of the optimization problem. To preserve realizability, we have to ensure that \( \psi_j^{(k+1)} \geq 0 \) for all \( \mu \in [-1, 1] \) and all cells \( j \).

For \( \mu > 0 \), after stripping away positive terms and using \( \mu \leq 1 \), we have
\[
\psi_j^{(k+1)} \geq \psi_j^{(k)} - \frac{\Delta t}{\Delta x} \psi_j^{\tau, -\tau} \geq \psi_j^{(k)} - \frac{\Delta t}{\Delta x} \psi_j^{\tau, +\tau}, \tag{4.34}
\]

where $\psi_{j+\frac{1}{2}}$ is the distribution from (4.30).

We have that

$$u_{j\pm\frac{1}{2}} = u_j^{(\kappa)} \pm \frac{1}{2} u_i'$$

(4.35)

where $u_i'$ is the (limited) slope on cell $i$. Thus we have

$$u_j^{(\kappa)} = \frac{u_j^- + u_j^+}{2}$$

(4.36)

and therefore a representing density for $u_j^{(\kappa)}$ is $\psi_{j\pm\frac{1}{2}}$. Inserting this in (4.34) gives

$$\psi_j^{(\kappa+1)} \geq \frac{\psi_j^- + \psi_j^+}{2} - \Delta t \frac{\psi_j^-}{\Delta x} \frac{\psi_j^- \psi_j^+}{2}$$

This is positive under the time step restriction

$$\Delta t < \frac{1 - \varepsilon_\gamma}{2} \Delta x.$$  

(4.37)

The case $\mu \leq 0$ follows in a similar way.

In $d$ dimensions, the update formula changes to

$$u_j^{(\kappa+1)} = u_j^{(\kappa)} - \sum_{l=1}^d \Delta t_{\Delta x_l} \left( g_{l}^{\text{kin}}(u_j^- + u_j^+) - g_{l}^{\text{kin}}(u_j^- - u_j^-) \right)$$

(4.39)

where $j = (1, \ldots, d)^T$ is an index tuple. As in one dimension, we define the representing density $\psi_j^{(\kappa+1)}$ and only regard the case $\Omega_l > 0 \forall l$, the other cases follow similarly. After stripping away positive terms we are left with

$$\psi_j^{(\kappa+1)} - \frac{\psi_j^- + \psi_j^+}{2} - \Delta t \frac{\psi_j^-}{\Delta x} \frac{\psi_j^- \psi_j^+}{2}$$

(4.40)

where $\Delta x = \min_l \Delta x_l$. We proceed as in one dimension and note that

$$\psi_j^{(\kappa+1)} \geq \frac{\psi_j^- + \psi_j^+}{2} - \Delta t \frac{\psi_j^-}{\Delta x} \frac{\psi_j^- \psi_j^+}{2}$$

(4.41)

is a representing density for $u_j^{(\kappa)}$ for any partition of unity $\sum_{l=1}^d w_l = 1$. Inserting this ansatz in (4.40) gives

$$\psi_j^{(\kappa+1)} \geq \sum_{l=1}^d \left( \frac{w_l}{2} - \Omega_l \frac{\Delta t}{\Delta x} \frac{\psi_j^-}{\Delta x(1 - \varepsilon_\gamma)} \right) \frac{\psi_j^- + \psi_j^+}{2} + \sum_{l=1}^d w_l \frac{\psi_j^- - \psi_j^-}{2}$$

(4.42)

This is positive if

$$\frac{\Delta t}{\Delta x} < \min \frac{1 - \varepsilon_\gamma}{2} \frac{w_l}{\Omega_l} \forall \Omega_l > 0 \forall l = 1, \ldots, d.$$  

(4.43)

So for given $\Omega$ we have to find a partition of unity $w$ such that the right-hand side of (4.43) is maximal, i.e., we want to find

$$\min \|\Omega\|_{\infty} \max \|w\|_{\infty} \min \min_{l \in \{1, \ldots, d\}} \frac{w_l}{\Omega_l}$$

(4.44)
Obviously, the maximum is attained if \( \frac{w_{l_1}}{\| \Omega_{l_1} \|} = \frac{w_{l_2}}{\| \Omega_{l_2} \|} \) for all \( l_1, l_2 \) (otherwise we could increase the \( w_l \) which belongs to the minimum and decrease the other ones a little). Taking the partition of unity property into account, we thus have to choose \( w_l = \frac{\Omega_l}{\| \Omega_l \|} \). Inserting this in (4.44) gives

\[
\min_{\| \Omega_l \| \leq 1} \max_{\| w_l \|_1 = 1} \min_{i \in \{1, \ldots, d\}} w_l \frac{\Omega_l}{\| \Omega_l \|} = \min_{\| \Omega_l \| \leq 1} \frac{1}{\| \Omega_l \|} \leq \min_n \frac{\| \Omega \|_2}{\| \Omega \|_1} = \frac{1}{\sqrt{d}}.
\]

Using this in (4.43), we end up with the time-step restriction

\[
\Delta t < \frac{1 - \varepsilon_\gamma}{2\sqrt{d}} \Delta x.
\]

5. Implementation details

We implemented the whole scheme in the generic C++ framework DUNE [7, 8], more specifically in the DUNE generic discretization toolbox dune-gdt [47] and the dune-xt-modules [39, 40].

As mentioned above, we advance the flux system in time using Heun’s method, which is a second-order strong-stability preserving Runge-Kutta scheme [23]. In each stage of the Runge-Kutta scheme, we perform the following steps:

1. Solve the optimization problem for the cell means \( \bar{u}_i \) in each grid cell (see Section 5.1). If regularization is needed, replace \( \bar{u}_i \) by its regularized version\(^3\) (see Section 5.1.2).
2. Reconstruct the values at the cell interfaces using linear reconstruction in characteristic variables (see Section 4.2.1), using the solution of the optimization problems from step 1 to calculate the Jacobians.
3. Perform the realizability limiting (see Section 5.3).
4. Solve the optimization problem for all reconstructed values \( u_{i \pm \frac{1}{2}} \). If the solver fails for a reconstructed value, disable the linear reconstruction in that cell.
5. Evaluate the kinetic flux (4.15) and update the stage values according to (4.13).

In the following, we will give some details on the implementation of these steps.

5.1. Implementation of the minimum-entropy solver

Our solver for the optimization problem is based on the algorithm from [3]. It uses a Newton-type algorithm with Armijo line search, i.e. to find a minimizer of the objective function \( p \) (see (4.19)), we are searching for a root of the gradient \( g \) (see (4.20)) in the Newton direction \( d(\alpha) \) which solves

\[
H(\alpha)d(\alpha) = -g(\alpha).
\]

and then update the multipliers as

\[
\alpha_{k+1} = \alpha_k + \zeta_k d(\alpha_k)
\]

\(^3\)This formally destroys the consistency of the scheme. However, since regularization rarely occurs (and only near the realizability boundary), this effect can usually be neglected in practice.
where $\zeta_k$ is determined by a backtracking line search such that
\[ p(\alpha_{k+1}) < p(\alpha_k) + \xi \zeta_k g(\alpha_k) \cdot d(\alpha_k) \] (5.3)
with $\xi \in (0, 1)$.

We stop the optimization if the new iterate $\alpha_{k+1}$ satisfies the stopping criteria (4.25), except that we use
\[ \|g_\beta(\beta)\|_2 < \min(\tau', \tau) \] (5.4)
as the first stopping criterion instead of simply using (4.25a). This avoids numerical difficulties for moments with small density, where $\tau'$ is in the order of 1 and thus some iterates $\beta$ with very large (in absolute values) entries might fulfill the stopping criterion by chance. Moreover, checking the second stopping criterion (4.25b) might be quite expensive (depending on the basis $b$). We therefore check this criterion only if additionally
\[ 1 - \varepsilon_\gamma < \exp(\|d(\beta)\|_1 + |\log g(\beta)|) \] (5.5)
holds. This criterion approximately ensures (4.30) (see [3, 52]) but, in general, is much easier to evaluate than (4.25b). For the HFM$_n$ models, however, checking realizability is just checking positivity, so in that case we do not need to check (5.5) first.

To improve the performance and stability of the algorithm, we use several additional techniques which we will detail in the following. The values of the algorithms’ parameters that we use in all computations are given in Table 1.

| Newton algorithm | Realizability limiter | Minima |
|------------------|-----------------------|--------|
| $k_0$ | $k_{max}$ | $\varepsilon_\gamma$ | $\varepsilon$ | $\chi$ | $\xi$ | $\tau$ | $\{r_l\}$ |
| 500 | 1000 | $10^{-2}$ | $2^{-52}$ | $1/2$ | $10^{-3}$ | $10^{-9}$ | $\{0, 10^{-8}, 10^{-6}, 10^{-4}, 10^{-3} \}$ |
| $\varepsilon_R$ | $\bar{\varepsilon}$ | $\rho_{vac}$ | $\psi_{vac}$ | $10^{-8}$ | $\rho_{vac}/(1)$ |

Table 1: Parameter choice for the different aspects of the simulation. Notation for the Newton algorithm as in [3].

### 5.1.1. Adaptive change of basis

Though the Hessian $H(\alpha)$ is positive definite and thus invertible, it may be very badly conditioned, especially for multipliers $\alpha$ corresponding to moments $u(\alpha)$ close to the boundary of the realizable set. Moreover, in general, the integral in the definition (4.21) of $H$ can only be calculated approximately using a numerical quadrature (see Section 5.4). If the quadrature is not sufficiently accurate, the approximate Hessian may have a significantly worse condition or may even be numerically singular.

To improve this situation, a change of basis can be performed after each Newton iteration such that the Hessian at the current iterate becomes the unit matrix in the new basis [3]. We use this procedure in our implementation for all bases except for the hat function bases $h_n$.

For the hat function basis, all matrices and vectors required in the optimization algorithm are sparse and exploiting this fact in the implementation greatly speeds up the computations. Including the change of basis destroys the sparsity and thus harms performance. In theory, this could be compensated by faster
convergence and thus less iterations of the algorithm due to the condition improvements. Further, the algorithm with change of basis might use regularization less frequently and thus introduce less errors in the solution, as shown for the full moments in [3]. We thus compared the algorithm with and without change of basis in several test problems. The differences in the results were negligible in all tests cases and the version without change of basis was significantly faster. We thus do not use the adaptive change of basis for the hat functions.

The first-order partial moments have a similarly simple structure as the hat functions, so the adaptive change of basis might also not be needed for these models. However, the change of basis does not have a significant performance impact in this case as the support of each basis function is restricted to a single interval or spherical triangle, and thus all matrix operations can be performed on the $2 \times 2$ or $4 \times 4$ submatrices corresponding to an interval in 1d and a spherical triangle in 3d, respectively. Similar, quadrature evaluations can be performed for each interval or spherical triangle separately. For this reason, we include the adaptive change of basis in our optimization algorithm for the partial moments.

For details on the change of basis algorithm see [3]. Note that the stopping criteria (4.25) are computed in the original basis such that knowledge of the change of basis algorithm is not required to understand the presentation in this paper.

5.1.2. Regularization

For tests with strong absorption, the local particle density may become very small in parts of the domain. As a consequence, also the entries of the Hessian $H$ become very small which may cause numerical problems. We thus choose a “vacuum” density $\psi_{\text{vac}}$ with corresponding local particle density $\rho_{\text{vac}} = \langle \psi_{\text{vac}} \rangle$. We then enforce a minimum local particle density of $\rho_{\text{vac}}$ by replacing moments $u$ with local particle density $\rho(u) < \rho_{\text{vac}}$ by the isotropic moment with vacuum density $\rho_{\text{vac}}$. Obviously, this approach leads to a violation of the conservation properties of the scheme. However, since we only replace moments with very small local particle densities by moments with slightly larger but still very small densities, the effect should be negligible in practice.

Additionally, if the optimizer fails for a moment vector $u$ (for example, by reaching a maximum number of iterations or being unable to solve for the Newton direction) we use the isotropic-regularization technique from [3], i.e. we replace $u$ by the regularized moment vector

$$u^r := (1 - r)u + rGu.$$  \hfill (5.6)

and retry the optimization. If the optimizer still fails, we increase $r$ until the optimizer succeeds, which is guaranteed at least for $r = 1$ where $u^r$ is isotropic. As the regularized moment vector $u^r$ always has the same local particle density as the original moment vector $u$, this technique does not violate the mass conservation of the scheme but it may potentially completely alter the solution. In practice, regularization is only used rarely and if it is used, a small regularization parameters is usually sufficient.

5.1.3. Caching

We use two types of caching. First, for each grid cell we store the moment vector from the last time step and the corresponding multiplier obtained by entropy minimization. In this way we do not have to solve the optimization problem again if the moment vector in that grid cell did not change during the last time step. In addition, we store the last few solutions of the minimization problem with corresponding input moment vectors per thread of execution, so if several grid cells contain the same values, we only have to perform the optimization once and then use the cached values. If we encounter a moment vector that can not be found in the caches, we take the moment vector that is closest to the input vector (in one-norm) and use the corresponding multiplier as an initial guess.
5.1.4. Linear solvers

In each iteration of the Newton scheme, we have to apply the inverse of a positive definite Hessian matrix. We assemble the matrices using the quadratures described in Section 5.4. Inversion is then done by computing a Cholesky factorization of the assembled matrix. For the full moment models, the Hessian matrices are dense, so we use the LAPACK routine \texttt{dpotrf} to compute the factorization and then use \texttt{dtrsv} to actually invert the linear systems. For the PMM\(_n\) models, the Hessian is block-diagonal (each block corresponds to one interval/triangle of the partition) such that we can perform the Cholesky decomposition independently for each block. For the HFM\(_n\) models in one dimension, the Hessian matrices are tridiagonal, so we can use the specialized LAPACK algorithms \texttt{dpotrf} and \texttt{dpttrs}. In three dimensions, the HFM\(_n\) Hessians are not tridiagonal anymore but still sparse, so we use the sparse SimplicialLDLT solver from the \texttt{Eigen} library [24].

5.2. Solving the eigenvalue problems

To avoid spurious oscillations, the reconstruction has to be performed in characteristic coordinates (see Section 4.2.1). For that reason, we have to compute the eigenvectors of the flux Jacobians

\[
\mathbf{f}'(\mathbf{u}) = \mathbf{J}(\mathbf{u}) \mathbf{H}^{-1}(\mathbf{u})
\]

where

\[
\mathbf{J}(\mathbf{u}) := \left\langle \mu \mathbf{b} (\mathbf{b} \cdot \mathbf{u}) \right\rangle
\]

and

\[
\mathbf{H}(\mathbf{u}) := \mathbf{H}(\mathbf{u}) = \left\langle \mathbf{b} (\mathbf{b} \cdot \mathbf{u}) \right\rangle
\]

(compare Section 4.2.3). Note that, in general, the Jacobian (5.7) is not symmetric. However, since \(\mathbf{J}\) is symmetric and \(\mathbf{H}\) symmetric positive definite (see Section 4.2.3), we can see that \(\mathbf{f}'(\mathbf{u})\) is similar to a symmetric matrix, i.e.

\[
\mathbf{H}^{-\frac{1}{2}} \mathbf{f}' \mathbf{H}^{-\frac{1}{2}} = \mathbf{H}^{-\frac{1}{2}} \mathbf{J} \mathbf{H}^{-\frac{1}{2}} = \mathbf{H}^{-\frac{1}{2}} \mathbf{J} \mathbf{H}^{-\frac{1}{2}}
\]

and thus has real eigenvalues. In our implementation, we explicitly compute the matrix representation and then use an eigensolver for non-symmetric matrices (LAPACK's \texttt{dgeevx}) to obtain the eigen decomposition. Unfortunately, though the Jacobian is a real matrix with real eigenvalues and thus also admits a set of real eigenvectors, the standard solvers for non-symmetric eigen problems (apart from \texttt{dgeevx}, we also tested the \texttt{EigenSolver} of the \texttt{Eigen} library [24]) often return complex eigenvectors. We thus add a step to compute real eigenvectors from the complex ones. Note that if

\[
\{ \mathbf{z}_l = \mathbf{y}_{2l} + i\mathbf{y}_{2l+1} \mid l = 0, \ldots, k - 1 \}
\]

is a set of linearly independent complex eigenvectors to the same eigenvalue \(\lambda\) for the Jacobian \(\mathbf{f}'\), where \(i\) is the imaginary unit and \(\mathbf{y}_m \in \mathbb{R}^n\) are real vectors, then

\[
\{ \mathbf{y}_m \mid m = 0, \ldots, 2k - 1 \}
\]

is a set of \(2k\) real eigenvectors for \(\mathbf{f}'\). Moreover, there are at least \(k\) linearly independent vectors in this set. To see that, assume the opposite, i.e. that any \(k\) vectors from the set (5.11) are linearly dependent. Without loss of generality, we assume that every vector in (5.11) can be written as a linear combination of the first \(k - 1\) vectors, i.e.

\[
\mathbf{y}_m = \sum_{r=0}^{k-2} a_{m,r} \mathbf{y}_r, \quad m = 0, \ldots, 2k - 1,
\]

with coefficients \(a_{m,r} \in \mathbb{R}\). Then, the \(k\) vectors \(\mathbf{z}_m\) can also be written as (complex) linear combinations of these \(k - 1\) real vectors

\[
\mathbf{z}_l = \mathbf{y}_{2l} + i\mathbf{y}_{2l+1} = \sum_{r=0}^{k-2} (a_{2l,r} + i a_{2l+1,r}) \mathbf{y}_r, \quad l = 0, \ldots, k - 1,
\]
and thus cannot be linearly independent. Consequently, to get real eigenvectors for $f'$ from the complex ones computed by the eigensolver, we first sort the eigenvectors into sets belonging to the same eigenvalue and then perform a Gram-Schmidt process with the real and imaginary parts for each of these sets.

**Remark 5.1.** While this procedure works reasonably well, a better approach would probably be to use the structure of the Jacobian and, instead of solving the non-symmetric eigenvalue problem

$$(f')z = JH^{-1}z = \lambda z,$$ (5.14)

solve the symmetric generalized eigenvalue problem [12, 37]

$$J\tilde{z} = \lambda H\tilde{z}$$ (5.15)

and then get the eigenvectors as $z = H\tilde{z}$. Since the matrices $J$ and $H$ are both symmetric and $H$ is positive definite, we can use a specialized algorithm like LAPACK’s dsygv and directly obtain real eigenvectors. Moreover, we can take advantage of the sparsity of these matrices and, e.g., use a generalized eigenvalue algorithm aimed at band matrices like dsbgv. In contrast, explicit assembly of the term $JH^{-1}$ might destroy the structure and result in a dense matrix even if the two factor matrices are sparse.

For the partial-moment models, the eigen decomposition can be done block-wise on the $2 \times 2$ or $4 \times 4$ matrix blocks which reduces the cubic complexity of the eigen decomposition [43] to a linear complexity for increasing number of moments and thus greatly accelerates computations for large problems.

### 5.3. Realizability limiting

The linear reconstruction process in the finite volume scheme does not guarantee preservation of realizability. Thus, we need an additional limiting step (4.17) to ensure that we are able to solve the optimization problem (2.5) for the reconstructed values. Since, in general, we cannot solve the integrals occurring in the optimization problem analytically and have to approximate them by a numerical quadrature $Q$, the admissible moment vectors are further restricted to the numerically realizable set ($Q$-realizable set)

$$R^Q_b = \{ u : \exists \psi(\Omega) \geq 0, \rho = \langle \psi \rangle_Q > 0, \text{such that } u = \langle b\psi \rangle_Q \} \subset \text{cl} \left( R_b \right),$$ (5.16)

where for an integrable function $f$, $\langle f \rangle_Q = \sum_{i=0}^{\sigma-1} w_i f(\Omega_i) \approx \langle f \rangle$ is the approximation of the corresponding integral $\langle \cdot \rangle$ with the quadrature rule $Q$. In general, the numerically realizable set is a strict subset of the analytically realizable set.

The numerically realizable set can be described as the convex hull of the basis function values at the quadrature nodes (see [3] for the Legendre basis, the proof can be easily adapted for the other bases)

$$R^Q_b|_{\rho=1} = \text{int} \left( \text{conv} \left( \{ b(\Omega_i) \}_{i=0}^{\sigma-1} \} \right) \right).$$ (5.17)

If $\rho$ depends linearly on $u$ it follows

$$R^Q_b|_{\rho<1} = \text{int} \left( \text{conv} \left( 0, \{ b(\Omega_i) \}_{i=0}^{\sigma-1} \} \right) \right).$$ (5.18)

We do not want the limited moments to be too close to to the boundary of the numerically realizable set as we are not able to solve the optimization problem (2.5) in that case (see [4]). Moving the limited value away from the boundary can be done in several ways. A simple but often sufficient method can be employed for all limiters presented in this section. We simply add a small parameter $\bar{\varepsilon}$ to the final limiter variable $\theta$ [52]. A problem with this approach is that the connecting line between $u$ and $\overline{u}$ might be almost parallel.
to the boundary which possibly results in a limited moment that is still too close to the boundary. Another approach is to require a fixed distance $\varepsilon_R$ to the boundary of $\mathcal{R}_b^Q$, i.e., to limit to the $(Q, \varepsilon_R)$-realizable set

$$\mathcal{R}_{b, \varepsilon_R}^Q = \{ \mathbf{u} \in \mathcal{R}_b^Q \text{ such that } d(\mathbf{u}, \partial \mathcal{R}_b^Q) \geq \varepsilon_R \},$$

(5.19)

where $d(\cdot, \partial \mathcal{R}_b^Q)$ is the Euclidean distance to $\partial \mathcal{R}_b^Q$. Limiting to this set is possible whenever $\mathbf{u}$ is farther than $\varepsilon_R$ away from the boundary. If $\mathbf{u}$ is already in the $\varepsilon_R$-range of the boundary, we disable reconstruction in that cell.

Unfortunately, checking whether a reconstructed value lies within the numerically realizable set is not trivial in general. In the following, we detail the limiting procedure for the different models. For the remainder of this section, let $\mathbf{u}$ be the moment vector before reconstruction and $\mathbf{u}$ a reconstructed moment vector. Let further $u_l$ and $\bar{u}_l$ be the $l$-th component of $\mathbf{u}$ and $\bar{\mathbf{u}}$, respectively.

### 5.3.1. $M_N$ models

In [2, 52], the half space representation for the convex hull (5.18) was explicitly calculated before starting the time stepping, yielding

$$\mathcal{R}_b^Q |_{\rho \leq 1} = \{ \mathbf{u} \in \mathbb{R}^n \mid \mathbf{c}_i \cdot \mathbf{u} < d_i, \ i \in \{0, \ldots, n_f - 1\} \},$$

(5.20)

where $n_f$ is the number of facets of the convex hull. During the time stepping, the intersection of the connecting line between $\mathbf{u}$ and $\bar{\mathbf{u}}$ and each facet can then be computed efficiently by solving

$$\mathbf{c}_i \cdot \mathbf{u}^\theta = d_i$$

(compare (4.17)) for the limiter variable. We thus obtain

$$\theta = \max_{i=0, \ldots, n_f-1} \theta_i,$$

$$\theta_i = \begin{cases} \frac{d_i - \mathbf{c}_i \cdot \mathbf{u}}{\mathbf{c}_i \cdot (\bar{\mathbf{u}} - \mathbf{u})} & \text{if } \frac{d_i - \mathbf{c}_i \cdot \mathbf{u}}{\mathbf{c}_i \cdot (\bar{\mathbf{u}} - \mathbf{u})} \in [0, 1], \\ 0 & \text{else}. \end{cases}$$

(5.21)

If $\rho(\mathbf{u}) \geq 1$ or $\rho(\bar{\mathbf{u}}) \geq 1$, the moments can simply be rescaled before applying the limiter [2, 52]. Alternatively, we can ignore the facet corresponding to the condition $\rho = \mathbf{a}_b^1 \cdot \mathbf{u} \leq 1$ which gives the half-space description for the full numerically realizable set (5.17). In that case, no rescaling is necessary and we can easily ensure a minimum distance of $\varepsilon_R$ to the realizable boundary by moving each facet in normal direction before calculating the intersections, resulting in

$$\tilde{d}_i = d_i - \varepsilon_R \|\mathbf{c}_i\|_2$$

(5.22)

instead of $d_i$ in (5.21). As for the other limiters, we disable reconstruction if $\bar{\mathbf{u}}$ does not lie within the $\varepsilon_R$-realizable set, i.e. if

$$\exists \ i \text{ s.t. } \mathbf{c}_i \cdot \bar{\mathbf{u}} \geq \tilde{d}_i.$$  

(5.23)

However, explicit calculation of the convex hull is only viable for a relatively small number of moments (such that the convex hull has to be calculated in a low-dimensional space) or very sparse quadratures (such that the convex hull has to be calculated from a small number of points). For a larger number of moments and a reasonable fine quadrature, the construction of the convex hull takes excessively long. Moreover, even when the convex hull is available, the performance of this approach might be unacceptable as the number of facets

---

4In one dimension, there are always two reconstructed values per grid cell (one at each interface), so each of the limiters described in the following is applied to both values and the larger $\theta$ is used in (4.17). In several dimensions, both reconstruction and limiting are performed independently for each coordinate direction.
grows rapidly with both the number of moments and the number of quadrature points [52]. We thus use this approach only for the partial moments (see Section 5.3.3) where we only have to calculate low-dimensional convex hulls.

For the $M_N$ models, as proposed in [52, Section 3.62], we instead utilize the quadrature description (5.16) of the numerically realizable set and limit by solving the linear program (LP)

$$
\min \theta
\quad \text{s.t.} \quad \sum_{i=0}^{n_g} \tilde{w}_i b(\Omega_i) = (1 - \theta)u + \theta \bar{u}
\quad \theta \geq 0, \quad \tilde{w}_i > 0,
$$

where $\tilde{w}_i$ should not be confused with $w_i$ but rather represents $w_i \psi(\Omega_i)$ for the sought representing distribution $\psi$. This approach removes the prohibitively costly explicit calculation of the convex hull. However, the runtime cost during the time stepping algorithm might be considerably higher as a linear program has to be solved for every reconstructed value.

Instead of using a single limiter variable $\theta$, principally, we can limit each component of $u$ independently. This has been done, e.g., in the context of the Euler equations in [62]. However, if the limiting is naively performed in ordinary coordinates, spurious oscillations may occur, as the limiting in ordinary coordinates may actually increase the slope in one of the characteristic components. In our implementation, we thus limit each of the characteristic components independently. Let $V$ be the matrix of eigenvectors of the Jacobian $f'(\bar{u})$ and let $u^c = V^{-1}u$, be the respective moment vectors in characteristic coordinates. Then we can find limiter variables $\theta = (\theta_1, \ldots, \theta_n)$ for each characteristic component by solving the LP

$$
\min 1 \cdot \theta
\quad \text{s.t.} \quad (B \tilde{V}) \begin{pmatrix} w \\ \theta \end{pmatrix} = Vu^c = u
\quad w \geq 0, \quad \tilde{w}_i > 0
$$

where the matrix $\tilde{V}$ is defined as $\tilde{V}_{ij} = V_{ij} \left( \psi_j^c - \psi_i^c \right)$ and the $i$-th column of $B$ is $b(\Omega_i)$.

For the LP-based limiter, it is not clear how to ensure a fixed distance $\varepsilon R$ to the boundary. We thus use the method of adding a small parameter $\tilde{\varepsilon}$ to the final limiter variable by replacing $\theta \geq 0$ by $\theta \geq -\tilde{\varepsilon}$ in (5.24c) and using $\theta + \tilde{\varepsilon}$ instead of $\theta$ if it is in the interval $[-\tilde{\varepsilon}, 1 - \tilde{\varepsilon}]$.

For checking realizability of $u$, which is needed for the stopping criterion (4.25b) in the optimization algorithm, we solve the simpler LP

$$
\min 0
\quad \text{s.t.} \quad Bw = u
\quad w \geq 0.
$$

Note that the limiters presented thus far are not restricted to the $M_N$ models but could be used for any basis $b$. However, for the $HFM_n$ and $PMM_n$ models, due to the simpler realizability conditions, limiters that are both faster and easier to implement can be used.

### 5.3.2 HFM$_n$ models

For the hat functions the numerically realizable set and the realizable set agree for suitable quadratures [56]. As a consequence, we can use a limiter based on the analytical realizability conditions which only require
component-wise positivity (see Section 3.3). We thus calculate the limiter variable \( \theta \) (limiting to \( \mathcal{R}_{\text{Q},R}^{Q,R} \)) as

\[
\theta_I = \begin{cases} 
1 & \text{if } \epsilon_R < \epsilon_R \\
\frac{\epsilon_R - u_0}{\epsilon_R - u_i} & \text{else if } \frac{\epsilon_R - u_0}{\epsilon_R - u_i} \in [0, 1] \\
0 & \text{else}
\end{cases}
\]

\( \theta = \max_i \theta_i. \) \hfill (5.27)

5.3.3. PMM\(_n\) models

In one dimension, \( \mathcal{R}_{\text{P},n}^{Q,R} = \mathcal{R}_{\text{P},n}^{Q,R} \) for suitable quadratures, so a limiter based on the analytical realizability conditions (3.3) can be used. We use a limiter variable \( \theta_i \) per interval \( I_i = [\mu_i, \mu_{i+1}] \). If we require a distance of at least \( \epsilon_R \) to the boundary, the realizability conditions (3.3) become

\[
u_{0,i} \geq \epsilon_R \quad \text{and} \quad \mu_i u_{0,i} + \epsilon_R \sqrt{\mu_i^2 + 1} \leq u_{1,i} \leq \mu_{i+1} u_{0,i} - \epsilon_R \sqrt{\mu_{i+1}^2 + 1}. \] \hfill (5.28)

If \( \mathbf{\mathbf{I}}_{\mathbf{I}} \) is already not \( \epsilon_R \)-realizable, we disable reconstruction for that interval. This results in the following limiter for one-dimensional partial moments

\[
\theta_{\mathbf{I}} = \begin{cases} 
1 & \text{if } \mathbf{\mathbf{I}}_{\mathbf{I}} \text{ does not fulfill (5.28)} \\
\max \left( \theta_{I_1}^0, \theta_{I_1}^1, \theta_{I_1}^2 \right) & \text{else}
\end{cases}
\]

where

\[
\theta_{I_1}^0 = \begin{cases} 
\frac{\epsilon_R - u_{0,i}}{\epsilon_R - u_i} & \text{if } \frac{\epsilon_R - u_{0,i}}{\epsilon_R - u_i} \in [0, 1] \\
0 & \text{else}
\end{cases}
\]

\[
\theta_{I_1}^1 = \begin{cases} 
\frac{u_{0,i} - u_{1,i} + \epsilon_R \sqrt{\mu_i^2 + 1}}{(\epsilon_R - u_{1,i}) - (\epsilon_R - u_{0,i}) \mu_i} & \text{if } \frac{u_{0,i} - u_{1,i} + \epsilon_R \sqrt{\mu_i^2 + 1}}{(\epsilon_R - u_{1,i}) - (\epsilon_R - u_{0,i}) \mu_i} \in [0, 1] \\
0 & \text{else}
\end{cases}
\]

\[
\theta_{I_1}^2 = \begin{cases} 
\frac{u_{0,i} - u_{1,i} - \epsilon_R \sqrt{\mu_i^2 + 1}}{(\epsilon_R - u_{1,i}) - (\epsilon_R - u_{0,i}) \mu_{i+1}} & \text{if } \frac{u_{0,i} - u_{1,i} - \epsilon_R \sqrt{\mu_i^2 + 1}}{(\epsilon_R - u_{1,i}) - (\epsilon_R - u_{0,i}) \mu_{i+1}} \in [0, 1] \\
0 & \text{else}
\end{cases}
\]

For the partial moment basis in three dimensions, the analytical and numerical realizable set differ. However, note that (5.18) holds separately for each spherical triangle (see [56, Lemma 5.13]), so we can explicitly calculate the half space representation (5.20) for each spherical triangle. Instead of calculating a convex hull in \( n \) dimensions, as would be needed for the full moment models, we only have to calculate \( \frac{n}{2} \) convex hulls in 4 dimensions, which is considerably faster and usually finished within a few seconds in our implementation (remember that this calculation has to be done only once before the time stepping).

5.4. Implementation of quadrature rules

In one dimension, we use Gauss-Lobatto quadratures on each interval. These quadratures include the endpoints of the interval, which ensures that the numerically realizable set (see (5.16)) equals the analytically realizable set for hat functions and partial moments, see [56]. To choose a suitable quadrature order, we solved some of our numerical test cases for different quadrature orders and calculated the errors with respect to the reference solution (see Section S2 in the supplementary materials). As suggested by this analysis, for the first-order models, we use a quadrature of order 15 per interval of the partition \( \mathcal{P} \). For the full moment \( \text{M}_N \) models, we split the domain in the two intervals \([-1, 0]\) and \([0, 1]\) that are needed for calculation of the kinetic flux and use a quadrature of order \( 2N + 40 \) on each interval.

In three dimensions, for partial moments and hatfunctions, we are using Fekete quadratures [59] (from the \textsc{Triangle_Fekete_Rule} library [13]) mapped to the spherical triangles. The library provides seven Fekete
quadratures of order 3, 6, 9, 12, 15 and 18, using 10, 28, 55, 91, 136, 190 quadrature points, respectively. The second rule of order 12 contains some negative quadrature weights, so we do not use that quadrature. If we want to improve the approximation, we subdivide each spherical triangle in several smaller ones as in [10] and use the mapped Fekete quadrature on each subtriangle. The Fekete rules correspond to Gauss-Lobatto rules on the triangle edges and thus also include the vertices of each triangle [59], which simplifies the realizability preservation (see [56]). As suggested by our quadrature sensitivity analysis (see Section S2), we use a quadrature order of 15 for the HFM and PMM models and a quadrature order of 9 for the other HFM and PMM models. For the MN models, we use tensor-product quadrature rules of order $2N + 8$ on the octants of the sphere.

For the hat function basis in one dimension, we alternatively explicitly calculate all integrals needed in the Newton algorithm using the analytical formulas and Taylor expansion at the numerical singularities of the analytical formulas (see [32, Appendix A1] for the explicit formulas). The Taylor expansion is performed in a neighborhood of radius 0.1 around the singularity, up to vanishing remainder or a maximal order of 200. This completely removes the need for quadrature rules. Note that the same approach could be used for the partial moments in one dimension. However, as the quadrature-based adaptive-change-of-basis algorithm is very efficient for partial moments, we did not implement the analytical formulas for partial moments.

For the hat function basis in three dimensions, integrals cannot be evaluated analytically anymore. We experimented with an approach where the integrals are expanded in a Taylor series representation (see Section S1 in the supplementary materials). However, it turned out that the Taylor series had to be computed up to a prohibitively high order. For this reason, we dismissed that approach and use the Fekete quadrature approach described above also for the hat functions in three dimensions.

5.5. Implementation of initial and boundary conditions

The initial values for the finite volume scheme are computed by integration of the kinetic equation’s initial values (2.1c).

$$u_i^0 = \frac{1}{\Delta z} \int_{z_i - \frac{1}{2}}^{z_i + \frac{1}{2}} (\psi_{t=0}(x, \Omega) b) dx$$

Since the initial values in our test cases are isotropic (see Section 6), i.e. $\psi_{t=0}(x, \Omega) = \psi_{t=0}(x)$, we only have to compute the velocity integral of the basis $b$. For this integral, we use the same quadratures as in Section 5.4 to ensure that the result is numerically realizable. Except for the plane-source and point-source tests, the initial values are constant in each grid cell, so we use the midpoint quadrature to evaluate the spatial integral. For the plane-source test, we always use an even number of grid cells and distribute the Dirac delta at $x = 0$ into the two adjacent grid cells, i.e. the initial value in these grid cells is set to the constant $\psi_{t=0}(x) = \psi_{\text{vac}} + \frac{1}{2\Delta z}$. For the point-source test, we use a Gauss-Legendre tensor product quadrature of order 20 to evaluate the spatial integrals for the initial values.

Boundary conditions for the moment equations are implemented by replacing the ansatz function $\hat{\psi}_m$ belonging to a grid cell $j$ outside of the computational domain (such cells often called “ghost cells”) by the boundary condition $\psi_b$ of the kinetic equation (2.1d) in the computation of the kinetic flux (4.15).

6. Numerical results

We want to apply our moment models to several test cases in the one- and three-dimensional setting. We follow the FAIR guiding principles for scientific research [61] and publish the code that generates the following results in [34]. As already mentioned (see Section 5), the scheme was implemented in the DUNE generic discretization toolbox dune-gdt [47]. The computations were done on a varying number of nodes.
of a distributed memory computer cluster\(^5\). Communication between nodes was done via MPI (Message Passing Interface) [38]. On each node, we used a work-stealing task-based shared-memory parallelization which was implemented using Intel TBB [27].

6.1. Slab geometry (1D)

6.1.1. Plane source

In this test case an isotropic distribution with all mass concentrated in the middle of an infinite domain \(z \in (-\infty, \infty)\) is defined as initial condition, i.e.

\[
\psi_{t=0}(z, \mu) = \psi_{\text{vac}} + \delta(z),
\]

where the small parameter \(\psi_{\text{vac}} = 0.5 \cdot 10^{-8}\) is used to approximate a vacuum. In practice, a bounded domain must be used which is large enough that the boundary should have only negligible effects on the solution. For the final time \(t_f = 1\), the domain is set to \(X = [-1.2, 1.2]\) (recall that for all presented models the maximal speed of propagation is bounded in absolute value by one).

At the boundary the vacuum approximation \(\psi(t, z_L, \mu) \equiv \psi_{\text{vac}}\) and \(\psi(t, z_R, \mu) \equiv \psi_{\text{vac}}\) is used again. Furthermore, the physical coefficients are set to \(\sigma_s \equiv 1\), \(\sigma_a \equiv 0\) and \(Q \equiv 0\).

All solutions are computed with an even number of cells, so the initial Dirac delta lies on a cell boundary. Therefore it is approximated by splitting it into the cells immediately to the left and right. In all figures below, only positive \(z\) are shown since the solutions are always symmetric around \(z = 0\).

Note that since the method of moments is indeed a type of spectral method, it can be expected that due to the non-smoothness of the initial condition the convergence towards the kinetic solution of this test case is slow (note that \(\psi_{t=0}(\cdot, \mu) \notin L^p\) for any \(p\)). Nevertheless, it is an often-used benchmark revealing many properties of a moment model (see e.g. [22]).

Some exemplary solutions at the final time are shown in Figures 1 to 3. Remember that the full-moment models are indexed by the basis order \(N\) while the piecewise linear bases are indexed by the number of moments \(n\). Since, in one dimension, a basis of the space of polynomials up to order \(N\) has \(n = N + 1\) elements, we compare the \(P_N\) and \(M_N\) models to the piecewise linear models with \(N + 1\) moments.

As expected, there are strong oscillations about the reference solution (the analytical solution from [21]) for all tested models. With increasing number of moments, the number of peaks increases while their height decreases. The oscillations are considerably stronger for the linear models than for the corresponding minimum-entropy models. The \(M_N\) models are closest to the reference solution, particularly for the low-order models.

This is also reflected in the convergence results, which can be found in Figure 4. Depicted are the \(L^1\) and \(L^\infty\) errors between the local particle densities \(\rho(u)\) of the moment models and the analytic reference solution from [21] at the final time \(t_f\). As expected, overall convergence is slow. The \(\text{HFP}_n\), \(\text{PMP}_n\) and \(P_N\) models show very similar \(L^1\) errors at all orders. With respect to \(L^\infty\) norm, the \(\text{PMP}_n\) models are slightly better than the \(P_N\) models. As observed before for the \(P_N\) models [50, 52], \(\text{HFP}_n\) models with odd \(n\) show a higher \(L^\infty\) error than models with even \(n\) due to a zero eigenvalue of the flux Jacobian. A similar but much less pronounced behaviour can be seen for the \(\text{PMP}_n\) models (for odd and even number of intervals \(2n\)). For odd \(n\), \(\text{HFP}_n\) \(L^\infty\) errors are close to the corresponding \(P_N\) error. For even \(n\) the \(\text{HFP}_n\) models perform better than the \(P_N\) models and similar to the \(\text{PMP}_n\) model.

---

\(^5\)Each node encloses two Intel Skylake Xeon Gold 6140 CPUs (2 \(\times\) 18 cores) and 92GB RAM.
Figure 1: Local particle density $\rho$ in the plane-source test case at time $t_f = 1$ for different orders of the full-moment models.

Figure 2: Local particle density $\rho$ in the plane-source test case at time $t_f = 1$ for different orders of the hat function moment models.
Figure 3: Local particle density $\rho$ in the plane-source test case at time $t_f = 1$ for different orders of the partial-moment models.

Figure 4: Convergence of the local particle density $\rho$ in the plane-source test case for different models.
The entropy-based HFM\(_n\), PMM\(_n\) and MN models show lower errors than their linear counterparts both in \(L^1\) and \(L^\infty\) norm. The PMM\(_n\) models perform slightly better than HFM\(_n\) models of the same order. The difference between odd and even orders/number of intervals is much more pronounced than for the HFP\(_n\) and PMP\(_n\) models. The MN models give the lowest errors of all tested models. However, the errors are still high and the rate of convergence is equally bad for all models although the convergence rate seems to improve with higher orders, especially for the minimum-entropy-based models.

6.1.2. Source beam

The discontinuous version of the source-beam problem from [26] is presented. The spatial domain is \(X = [0,3]\), and

\[
\sigma_a(z) = \begin{cases} 
1 & \text{if } z \leq 2, \\
0 & \text{else},
\end{cases} \\
\sigma_s(z) = \begin{cases} 
0 & \text{if } z \leq 1, \\
2 & \text{if } 1 < z \leq 2, \\
10 & \text{else},
\end{cases} \\
Q(z) = \begin{cases} 
\frac{1}{2} & \text{if } 1 \leq z \leq 1.5, \\
0 & \text{else},
\end{cases}
\]

with initial and boundary conditions

\[
\psi_{t=0}(z,\mu) \equiv \psi_{\text{vac}}, \\
\psi_b(t, z_L, \mu) = \frac{e^{-10^5(\mu-1)^2}}{(e^{-10^5(\mu-1)^2})^2} 	ext{ and } \psi_b(t, z_R, \mu) \equiv \psi_{\text{vac}}.
\]

The final time is \(t_f = 2.5\) and the same vacuum approximation \(\psi_{\text{vac}}\) as in the plane-source problem is used.

![Convergence of the local particle density \(\rho\) in the source beam test case for different models.](image)

Convergence results can be found in Figure 5. The reference solution for this test case is computed from a direct finite difference discretization of the kinetic equation on a grid with 21000 \(\times\) 14000 elements. As expected due to the higher regularity of the test case\(^6\), the convergence for all tested models is much better

\(^6\psi\) is "only" a discontinuous function compared the distributional setting in the plane-source test.
than in the plane-source test. As a consequence, most of the models are relatively close to the reference solution which is why we do not show exemplary solutions plots. The piecewise linear models and the \( M_N \) models converge with second and first order in \( L^1 \) and \( L^\infty \) norm, respectively. The \( P_N \) models show spectral convergence which is reflected in comparatively high errors and a slow rate of convergence for the lower-order models and then a rapid convergence for the high-order models. Despite the eventual high rate of convergence, for the maximal moment number \( n = 50 \) regarded here the \( P_N \) models are mostly outperformed by the other models. Again, the minimum-entropy-based models perform better than their linear counterparts although the rate of convergence is the same. In \( L^\infty \) norm, the \( M_N \), partial-moment and HFP\(_n\) models again show a zig-zag pattern where, e.g., the HFP\(_n\) models using an odd number of intervals (even number of moments \( n \)) perform better than the ones using an even interval number (odd \( n \)). For the partial-moment models, an even number of intervals gives better results. Notably, the HFM\(_n\) do not show such an alternating behaviour.

6.2. Three dimensions

We now consider numerical results in three spatial dimensions with velocities on the unit sphere.

6.2.1. Point source

The point-source test is the three-dimensional analogue of the plane-source test (Section 6.1.1) in slab geometry. Due to the limitations in the resolution we use a smoothed version of the initial Dirac delta:

\[
\psi_{t=0}(x, \Omega) = \psi_{\text{vac}} + \frac{1}{4\pi^4\sigma^3} \exp\left(-\frac{|x|^2}{\pi \sigma^2}\right),
\]

where \( \sigma = 0.03 \), \( \psi_{\text{vac}} = \frac{10^{-8}}{4\pi\sigma^3} \). As before, we choose \( \sigma_s \equiv 1 \), \( \sigma_a \equiv 0 \) and \( Q \equiv 0 \). All models are calculated on \( X = [-1,1]^3 \) to the final time \( t_f = 0.75 \). The grid size is chosen to be \( \Delta x = \Delta y = \Delta z = 0.02 \). The point-source test is well-suited to demonstrate symmetries (or symmetry breaks) appearing in the solution. We show some selected models in Figures 6 and 7, where we use the endcap geometry \([0,1] \times [-1,1] \times [-1,1]\) for the isosurfaces.

The reference solution itself is rotationally symmetric and can be computed analytically using the formulas by Ganapol [21]. It can be observed that the hat functions have a preferred directions of propagation, directly related to the position of the vertices in the spherical triangulation (e.g., the octahedron that defines the HFM\(_6\) basis can be easily identified in Figure 6). Similar effects occur for the partial moments. However, the discontinuity of their basis is also reflected in the peaks along the boundaries of the spherical triangles (compare PMM\(_{32}\)). In contrast to this, the full-moment models preserve the rotational symmetry (compare M\(_3\), where small irregularities in the solution arise due to the spherical quadrature rule) but adding more waves to the solution.

Finally, we show error plots for our models in Figure 8. The models show the expected slow convergence in the \( L^1 \)-norm, similar to the plane-source test (Section 6.1.1). All first-order models show roughly order \( \frac{1}{2} \), whereas the full-moment models have varying convergence rates. In the \( L^\infty \)-norm, the first-order models show order 1 convergence in the beginning, which then slows down to order \( \frac{1}{2} \) as well. The full-moment models are showing no (or very slow) convergence, which is the well-known Gibbs phenomenon.

Note that the PMM\(_n\) models clearly outperform the other methods. In particular, they are as good as or even slightly better than the \( M_N \) models whose calculation is significantly more expensive for the same degrees of freedom (see Section 6.3).

6.2.2. Checkerboard

The checkerboard test case is a lattice problem which is loosely based on a part of a reactor core [11]. We extend it in a straightforward manner to the three-dimensional case. The used geometry is shown in...
Figure 6: Two-dimensional cuts and selected isosurfaces for some models in the point-source test.
Figure 7: Two-dimensional cuts and selected isosurfaces for some models in the point-source test.
Figure 8: Convergence of the local particle density $\rho$ in the point-source test for different models.

Figure 9. There are scattering (orange and green) and highly absorbing (black) regions. The parameters are chosen to be the following.

- **Domain:** $X = [0, 7]^3$, subdivided into the three regimes

  $$X_a = \left\{ x = (x, y, z)^T \in [1, 6]^3 \mid ([x] + [y] + [z]) \mod 2 = 1, \quad x \notin [3, 4]^3 \cup [3, 4] \times [5, 6] \times [3, 4] \right\},$$

  $$X_s = X \setminus X_a,$$

  $$X_Q = [3, 4]^3,$$

- **Final time:** $t_f = 3.2$,

- **Parameters** (compare Figure 9):

  $$\sigma_s(x) = \begin{cases} 1 & \text{if } x \in X_s, \\ 0 & \text{else,} \end{cases}, \quad \sigma_a(x) = \begin{cases} 0 & \text{if } x \in X_s, \\ 10 & \text{else}, \end{cases}, \quad Q(x) = \begin{cases} \frac{1}{4\pi} & \text{if } x \in X_Q, \\ 0 & \text{else.} \end{cases}$$

- **Initial condition:** $\psi_{t=0}(x, \Omega) = \psi_{\text{vac}} := \frac{10^{-8}}{4\pi}$ (approx. vacuum),

- **Boundary conditions:** $\psi_0(t, x, \Omega) = \psi_{\text{vac}}$.

Due to the discontinuous nature of the physical parameters, this test case is a challenging task for a numerical solver. We align our grid with the discontinuities of the parameters by using a multiple of 7 (usually 70) regularly spaced grid points in each direction.

Solution plots for selected models can be found in the supplementary materials (Figures S3.1–S3.3). The PMP$_{32}$ model and the $P_N$ models of order $N \in \{2, \ldots, 9\}$ (only $N = 9$ shown) have negative particle densities $\rho$. Surprisingly, the hat function basis HFP$_n$ has positive densities for all $n$ that we calculated.
We compare our models to a discrete ordinate implementation [22, 31] of second order. $L^1$- and $L^\infty$-errors of the local particle density $\rho$ can be found in Figure 10. All tested models converge with about first order both in $L^1$ and $L^\infty$ norm. Again, PMM$_n$ and HFM$_n$ models are comparable to the M$_N$ models.

6.2.3. Shadow

The shadow test case represents a particle stream that is partially blocked by an absorber, resulting in a shadowed region behind the absorber. The used geometry is shown in Figure 11. The parameters are chosen to be the following.

- Domain: $X = [0, 12] \times [0, 4] \times [0, 3]$
- Final time: $t_f = 20$,
- Parameters:

\[
\begin{align*}
\sigma_s(x) &= Q(x) = 0 \\
\sigma_a(x) &= \begin{cases} 
50 & \text{if } x \in [2, 3] \times [1, 3] \times [0, 2] \\
0 & \text{else,}
\end{cases}
\end{align*}
\]
- Initial condition: $\psi_{t=0}(x, \Omega) = \psi_{\text{vac}} := \frac{10^{-8}}{4\pi}$ (approx. vacuum),
- The isotropic particle stream with density $\rho = 2$ enters the region via the boundary condition at $x = 0$. At all other boundaries, vacuum boundary conditions are used.

\[
\psi_b(t, x, \Omega) = \begin{cases} 
\frac{2}{4\pi} & \text{if } x = 0 \\
\psi_{\text{vac}} & \text{else}.
\end{cases}
\]

We show slices and isovalues of several models at the final time in the supplementary materials (Figures S.4–S.6). Again, several of the linear models (e.g. PMP$_{32}$ or $P_{22}$) show negative values (depicted in red). As in the previous test case, the partial moments perform very well. Compare, for example, the linear
Figure 10: Convergence of the local particle density $\rho$ in the checkerboard test case for different models.

Figure 11: Setup of the shadow test. The absorbing region is depicted in black.
PMP_{512} model to P_{22} (which has roughly the same number of degrees of freedom). The partial-moment model approximates the reference much better, especially in the far field where also the small oscillations are captured accurately. A similar tendency can be observed for the hat function model HFP_{258}.

Both hat function and discontinuous minimum-entropy models show a good approximation of the absorber (compare PMM_{32} and HFM_{6}). However, they are not able to provide a reasonable approximation in the far field. Further repartitioning of the sphere yields much better results in this case.

Investigating again the convergence towards the reference solution (see Figure 12), we see that the full-moment models are slightly superior in the beginning, but convergence slows down for higher \( n \). Both HFM_{n} as well as PMM_{n} show a similar convergence behaviour. Again, taking running time into account, both models outperform the classical M_{N} model in terms of efficiency (see Section 6.3). Note that we only computed M_{N} models up to a order of \( N = 4 \) since the higher-order models did not finish in the available computation time. The same is true for the HFM_{n} models with \( n > 66 \). Distributing the workload among more nodes of the distributed cluster did not improve computation times for these models, which is probably due to load-balancing issues. While the task-stealing algorithm (see above) ensures that the work is distributed evenly on each node, there is no load-balancing between nodes. In our implementation, the grid is distributed to the nodes using a decomposition of the domain in connected parts. Since the optimization problems are particularly challenging near the absorber (due to low particle densities and very anisotropic distributions), the node(s) containing the absorber often need much longer than the other nodes to solve the optimization problems. Here, an MPI-based load-balancing implementation may be required which would, however, significantly increase the communication overhead. As an alternative, in [33], we investigate a numerical scheme that avoids the non-linear optimization problems and thus does not show the same load-balancing issues.

In conclusion, moment models based on piecewise first-order continuous (HFP_{n}, HFM_{n}) or discontinuous (PMP_{n}, PMM_{n}) basis functions often approximate the true solution as good as or even better than the standard models (P_{N}, M_{N}) using polynomial bases. In contrast to the standard models, however, these models can be implemented very efficiently. This is especially true for the entropy-based models since the necessary realizability limiting can be based on the analytical realizability conditions (see Section 5.4). In
addition, in case of the discontinuous models, all matrix operations can be performed on small matrix blocks which provides further performance advantages (also for the PMP\textsubscript{n} models, see Section 6.3).

6.3. Timings

Performance measurements can be found in Figure 13. The times were measured without parallelization. Displayed times are the minimum of three runs. Quadratures were chosen as described in Section 5.4. Measurements were done both for the first-order scheme without linear reconstruction and for the realizability-preserving second-order scheme (see Section 4). Profiling shows that the first-order scheme spends most of the time solving the optimization problems. For the second-order scheme, solving the eigen problems for the reconstruction in characteristic coordinates also has a large impact on the execution time. Here, computation times could probably be improved by using a generalized eigen solver which takes the structure of the Jacobians into account (see Section 5.2). Both the adaptive-change-of-basis scheme and the eigensolver have third-order complexity. We thus asymptotically expect third-order complexity in \( n \) for both the first-order and the second-order scheme for the M\( _N \) models. For the PMM\textsubscript{n} models, all operations (including the solution of the eigen problems) can be done block-wise, so we expect first-order complexity in that case. Regarding only the optimization problem, the same is true for HFM\textsubscript{n} models as all matrices involved are tridiagonal (in slab geometry) or very sparse (in three dimensions). However, the Jacobian of the flux function is not sparse in general for the HFM\textsubscript{n} models, so the eigen problems are currently solved with a standard third-order-complex eigensolver. These models would particularly benefit from an improved implementation of the eigensolver which exploits the fact that the Jacobians are products of two sparse symmetric matrices (see Section 5.2).

In slab geometry, we used a reduced version of the plane-source test case (1000 grid cells, final time \( t_f = 0.1 \)). For the HFM\textsubscript{n} models, two different implementations were tested: the backtracking Newton solver without change of basis (see Section 5.1.1) using quadratures to calculate the integrals and the same backtracking Newton solver where all needed integrals were solved using the analytical formulas (and Taylor expansion at the singularities, see Section 5.4). In three dimensions, we used a reduced version of the point-source test case (\( I = 10^3 \) grid cells, single Runge-Kutta step).

As can be seen in Figure 13, for the first-order scheme, results are as expected except that the M\( _N \) models show second-order complexity in slab geometry, probably because the matrices are relatively small here and thus the third-order matrix operations do not dominate the execution time. The HFM\textsubscript{n} implementation using analytic integrals is faster than the quadrature version but the difference is negligible in practice. Given that the implementation using analytic integrals is considerably more complex and that analytic realizability conditions can also be used for the quadrature-based version, we suggest to generally use a quadrature-based implementation also for the HFM\textsubscript{n} models.

For the second-order scheme, as expected, the PPM\textsubscript{n} models show first-order complexity both in slab geometry and in three dimension and thus are several orders of magnitudes faster than the other models. Curiously, the HFM\textsubscript{n} models are close to second-order complexity also in three dimensions. For even larger \( n \), we expect the HFM\textsubscript{n} models to also increase with third-order due to the eigensolver but the results show that the HFM\textsubscript{n} models are much faster than the M\( _N \) models for a long time.

Note that though the PPM\textsubscript{2} and M\textsubscript{2} models are equivalent, the measured times are different as the M\textsubscript{2} model uses the convex-hull based realizability limiter while the PPM\textsubscript{2} model uses the limiter based on the analytical realizability conditions to be consistent with the models with higher \( n \).

7. Conclusions and outlook

We derived two classes of minimum-entropy moment models based on a continuous finite element basis as well as a discontinuous piece-wise linear basis. Both types of models are realizable, i.e., generated by a non-negative ansatz, such that important physical properties like positivity of mass are preserved.
Figure 13: Execution wall time for the minimum-entropy models in one and three dimensions. Times were measured in serial computations (no parallelization). Plotted is the minimum of three runs. 
a) Plane-source test case (1000 grid cells, \( t_f = 0.1 \)).
b) Point-source test case (10^3 grid cells, single Runge-Kutta step).

---

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure13}
\caption{Execution wall time for the minimum-entropy models in one and three dimensions. Times were measured in serial computations (no parallelization). Plotted is the minimum of three runs. a) Plane-source test case (1000 grid cells, \( t_f = 0.1 \)). b) Point-source test case (10^3 grid cells, single Runge-Kutta step).}
\end{figure}
We demonstrated in various numerical tests in one and three dimensional geometry that those models are qualitatively competitive with the classical full-moment $M_N$ models of the same number of degrees of freedom if the solution of the kinetic equation only has a limited smoothness (since otherwise the $M_N$ models typically show spectral convergence). Additionally, the new models are much cheaper (with respect to running time) than the full moment models since the non-linear problems that have to be solved locally are much smaller and typically much easier to solve as well. Consequently, the new models are considerably more efficient in the sense that they reach the same approximation error with much less computation effort. In particular, the partial moments $PM_{M_n}$ show a linear relation between wall time and number of moments, which is also true for the hat function basis if only a first-order scheme is used. If a higher-order discretization in space and time is required, the partial moments appear to be the model of choice. However, in some cases the discontinuity in the basis functions may lead to severe problems, for example when collision is modeled with the Laplace-Beltrami operator [53]. In such cases, $HFM_n$ might be favorable.

We provided a second-order realizability-preserving scheme by using a splitting technique and analytic solutions of the stiff part, combined with a realizability-preserving reconstruction scheme. Higher-order variants of this scheme can in principle be derived similarly, but we emphasize that we strictly focused on non-smooth problems, where the sense in applying schemes with (much) more than second order is questionable.

If the underlying problem admits more smoothness (especially in the velocity domain), higher-order moment models might be more appropriate to enhance the speed of convergence towards the kinetic solution. While this is rather straightforward to define both in slab as well as three-dimensional geometry (partial moments can be constructed immediately while the hat-function basis can be extended to higher-order splines on the unit interval/unit sphere, respectively [1]), special care is required since the realizability conditions are needed in order to use our realizability-preserving scheme. Up to our knowledge, the corresponding realizability problems are only solved for partial moments (of arbitrary order) in slab geometry [17], while first approaches are given for second-order partial moments on quadrants/octants of the sphere [57].

References

[1] P. Alfeld, M. Neamtu, and L. L. Schumaker, Bernstein–Bézier polynomials on spheres and sphere-like surfaces, Computer Aided Geometric Design, 13 (1996), pp. 333–349.
[2] G. Alldredge and F. Schneider, A realizability-preserving discontinuous Galerkin scheme for entropy-based moment closures for linear kinetic equations in one space dimension, Journal of Computational Physics, 295 (2015), pp. 665–684.
[3] G. W. Alldredge, C. D. Hauck, D. P. O’Leary, and A. L. Tits, Adaptive change of basis in entropy-based moment closures for linear kinetic equations, Journal of Computational Physics, 258 (2014), pp. 489–508.
[4] G. W. Alldredge, C. D. Hauck, and A. L. Tits, High-order entropy-based closures for linear transport in slab geometry II: A computational study of the optimization problem, SIAM Journal on Scientific Computing, 34 (2012), pp. B361–B391.
[5] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen, LAPACK Users’ Guide, Society for Industrial and Applied Mathematics, Philadelphia, PA, third ed., 1999.
[6] I. Babuška and B. Guo, The $h$, $p$ and $h$-$p$ version of the finite element method: basis theory and applications, Advances in Engineering Software, 15 (1992), pp. 159–174.
[7] P. Bastian, M. Blatt, A. Dedner, C. Engwer, R. Klöfkorn, R. Kornhuber, M. Ohlberger, and O. Sander, A Generic Grid Interface for Parallel and Adaptive Scientific Computing. Part II: Implementation and Tests in DUNE, Computing, 82 (2008), pp. 121–138.
[8] P. Bastian, M. Blatt, A. Dedner, C. Engwer, R. Klöfkorn, M. Ohlberger, and O. Sander, A Generic Grid Interface for Parallel and Adaptive Scientific Computing. Part I: Abstract Framework, Computing, 82 (2008), pp. 103–119.
[9] M. A. Blanco, M. Flórez, and M. Bermejo, Evaluation of the rotation matrices in the basis of real spherical harmonics, Journal of Molecular Structure, 419 (1997), pp. 19–27.
[10] N. Boal and F.–J. Sayas, Adaptive numerical integration on spherical triangles, in Proceedings of VIII International Zaragoza-Pau Conference on Applied Mathematics and Statistics (MC Lépez de Silanes et al, eds). Monografías Sem. Mat. G Galdeano, vol. 31, 2004, pp. 61–69.
[11] T. A. Brunner and J. P. Holloway, Two-dimensional time dependent Riemann solvers for neutron transport, Journal of Computational Physics, 210 (2005), pp. 386–399.
[12] A. Bunse-Gerstner, An algorithm for the symmetric generalized eigenvalue problem, Linear Algebra and its Applications, 58 (1984), pp. 43–68.
[13] J. Burkardt, TRIANGLE_FEKETE_RULE. https://people.sc.fsu.edu/~jburkardt/cpp_src/triangle_fekete_rule/triangle_fekete_rule.html, 2014.
[14] S. R. Buss and J. P. Fillmore, Spherical averages and applications to spherical splines and interpolation, ACM Transactions on Graphics, 20 (2001), pp. 95–126.
[15] P. Chidyagwai, M. Frank, F. Schneider, and B. Seibold, A Comparative Study of Limiting Strategies in Discontinuous Galerkin Schemes for the $M_1$ Model of Radiation Transport, Journal of Computational and Applied Mathematics, 342 (2018), pp. 399–418.
[16] I. Cravero, G. Puppo, M. Semplice, and G. Visconti, Cool weno schemes, Computers & Fluids, 169 (2018), pp. 71–86.
[17] R. E. Curtu and L. A. Fialkow, Recursiveness, positivity, and truncated moment problems, Houston Journal of Mathematics, 17 (1991), pp. 603–635.
[18] B. Dubroca and J.-L. Feugeas, Entropic Moment Closure Hierarchy for the Radiative Transfer Equation, C. R. Acad. Sci. Paris Ser. I, 329 (1999), pp. 915–920.
[19] B. Dubroca and A. Klar, Half-moment closure for radiative transfer equations, Journal of Computational Physics, 180 (2002), pp. 584–596.
[20] M. Frank, B. Dubroca, and A. Klar, Partial moment entropy approximation to radiative heat transfer, Journal of Computational Physics, 218 (2006), pp. 1–18.
[21] B. D. Ganapol, R. S. Baker, J. A. Dahl, and R. E. Alcouffe, Homogeneous infinite medium time-dependent analytical benchmarks, tech. rep., Tech. Rep. LA-UR-01-1854. Los Alamos National Laboratory, 2001.
[22] C. K. Garrett and C. D. Hauck, A Comparison of Moment Closures for Linear Kinetic Transport Equations: The Line Source Benchmark, Transport Theory and Statistical Physics, (2013).
[23] S. Gottlieb, On High Order Strong Stability Preserving Runge–Kutta and Multi Step Time Discretizations, Journal of Scientific Computing, 25 (2005), pp. 105–129.
[24] G. Guennebaud, B. Jacob, et al., Eigen v3. http://eigen.tuxfamily.org, 2010.
[25] C. D. Hauck, High-order entropy-based closures for linear transport in slab geometry, Commun. Math. Sci. v9, (2010).
[26] C. D. Hauck, M. Frank, and E. Olbrant, Perturbed, entropy-based closure for radiative transfer, SIAM Journal on Applied Mathematics, 6 (2013).
[27] Intel, Threading building blocks.
[28] G. S. Jiang and C.-W. Shu, Efficient implementation of weighted ENO schemes., Journal of Computational Physics, 228 (1995), pp. 202–228.
[29] D. I. Ketcheson, Highly efficient strong stability-preserving Runge-Kutta methods with low-storage implementations, SIAM Journal on Scientific Computing, 30 (2008), pp. 2113–2136.
[30] T. Langer, A. Belyaev, and H.-P. Seidel, Spherical barycentric coordinates, Proceedings of the fourth Eurographics symposium on Geometry processing, (2006), pp. 81–88.
[31] E. W. Larsen and J. E. Morel, Advances in discrete-ordinates methodology, in Nuclear Computational Science, Springer, 2010, pp. 1–84.
[32] T. Leinert, Model reduction for kinetic equations: moment approximations and hierarchical approximate proper orthogonal decomposition, PhD thesis, WWU Münster, 2021.
[33] T. Leinert and M. Ohlberger, A new entropy-variable-based discretization scheme for minimum entropy moment models for a linear kinetic equation, arXiv, (2020).
[34] T. Leinert and F. Schneider, Replication Data for: First-order continuous and discontinuous Galerkin moment models for a linear kinetic equation: realizability-preserving splitting scheme and numerical analysis, Harvard Dataverse, (2019).
[35] C. D. Levermore, Moment closure hierarchies for kinetic theories, Journal of Statistical Physics, 83 (1996), pp. 1021–1065.
[36] E. E. Lewis and W. F. Miller, Jr., Computational Methods in Neutron Transport, John Wiley and Sons, New York, 1984.
[37] R. S. Martin and J. H. Wilkinson, Reduction of the Symmetric Eigenproblem $Ax=\lambda Bx$ and Related Problems to Standard Form, Springer Berlin Heidelberg, Berlin, 1971, pp. 303–314.
[38] Message Passing Interface Forum, MPI: A message-passing interface standard (version 3.1). http://www.mpi-forum.org/docs/mpi-3.1/spi31-report.pdf, 2015.
[39] R. Milk, F. Schindler, and T. Leinert, dune-gdt. http://github.com/dune-community/dune-gdt, 2017.
[48] F. Schneider, First-order quarter- and mixed-moment realizability theory and Kershaw closures for a Fokker-Planck equation in two space dimensions: Code, 2016.

[49] ———, Implicit-explicit, realizability-preserving first-order scheme for moment models with lipschitz-continuous source terms, arXiv:1611.01314, (2016).

[50] ———, Kershaw closures for linear transport equations in slab geometry I: Model derivation, Journal of Computational Physics, 322 (2016), pp. 905–919.

[51] ———, Kershaw closures for linear transport equations in slab geometry II: high-order realizability-preserving discontinuous-Galerkin schemes, Journal of Computational Physics, 322 (2016), pp. 920–935.

[52] ———, Moment models in radiation transport equations, Verlag Dr. Hut, 2016.

[53] F. Schneider, G. W. Alldredge, M. Frank, and A. Klar, Higher Order Mixed-Moment Approximations for the Fokker–Planck Equation in One Space Dimension, SIAM Journal on Applied Mathematics, 74 (2014), pp. 1087–1114.

[54] F. Schneider, G. W. Alldredge, and J. Kall, A realizability-preserving high-order kinetic scheme using weno reconstruction for entropy-based moment closures of linear kinetic equations in slab geometry, Kinetic & Related Models, 9 (2016), p. 193.

[55] ———, A realizability-preserving high-order kinetic scheme using weno reconstruction for entropy-based moment closures of linear kinetic equations in slab geometry, Kinetic & Related Models, 9 (2016), p. 193.

[56] F. Schneider and T. Leibner, First-order continuous- and discontinuous-galerkin moment models for a linear kinetic equation: Model derivation and realizability theory, Journal of Computational Physics, 416 (2020), p. 109547.

[57] F. Schneider, A. Roth, and J. Kall, First-order quarter- and mixed-moment realizability theory and Kershaw closures for a Fokker-Planck equation in two space dimensions, Kinetic and Related Models, 10 (2017), pp. 1127–1161.

[58] B. Seibold and M. Frank, StaRMAP—A Second Order Staggered Grid Method for Spherical Harmonics Moment Equations of Radiative Transfer, ACM Transactions on Mathematical Software, 41 (2014), pp. 1–28.

[59] M. A. Taylor, B. A. Wingate, and R. E. Vincent, An Algorithm for Computing Fekete Points in the Triangle, SIAM J. Numer. Anal., 38 (2000), pp. 1707–1720.

[60] V. Titarev and E. Toro, Finite-volume weno schemes for three-dimensional conservation laws, Journal of Computational Physics, 201 (2004), pp. 238 – 260.

[61] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne, J. Bouwman, A. J. Brookes, T. Clark, M. Crosas, I. Dillo, O. Dumon, S. Edmunds, C. T. Evelo, R. Finkers, A. Gonzalez-Beltran, A. J. G. Gray, P. Groot, C. Grobe, J. S. Grethe, J. Heringa, P. A. C. ‘t Hoen, R. Hooft, T. Kuhn, R. Kok, J. Kok, S. J. Lusher, M. E. Martone, A. Mons, A. L. Packer, B. Persson, P. Rocca-Serra, M. Roos, R. van Schaik, S.-A. Sansone, E. Schultes, T. Sengstag, T. Slater, G. Strawn, M. A. Swertz, M. Thompson, J. van der Lei, E. van Mulligen, J. Velterop, A. Waagmeester, P. Wittenburg, K. Wolstencroft, J. Zhao, and B. Mons, The fair guiding principles for scientific data management and stewardship, Scientific Data, 3 (2016).

[62] X. Zhang and C. W. Shu, On positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes, Journal of Computational Physics, 229 (2010), pp. 8918–8934.
Supplementary Material – First-order continuous- and discontinuous-Galerkin moment models for a linear kinetic equation: realizability-preserving splitting scheme and numerical analysis

Florian Schneider, Tobias Leibner

S1. Evaluating the HFM\(_m\) integrals in three dimensions using Taylor series

For the hat function basis in three dimensions, integrals cannot be evaluated analytically anymore. However, the integrals can be expanded in a Taylor series representation. Though we dismissed that approach (since it turned out that the Taylor series had to be computed up to a prohibitively high order) we describe it here for future reference.

Let \( \mathbf{K} \) be a spherical triangle with vertices \( A, B \) and \( C \) on the unit sphere \( S_2 \). Let \( h_\mathbf{K} = (h_1, h_2, h_3)^T \) be the barycentric basis functions on \( \mathbf{K} \). We are interested in the integral

\[
\int_\mathbf{K} f(\Omega) \exp(h_\mathbf{K}(\Omega) \cdot \alpha) \, d\Omega
\]

for some function \( f \). Since we cannot evaluate this integral analytically, we write the exponential function in its Taylor series representation. Let \( \Omega = (\Omega_x, \Omega_y, \Omega_z)^T \) and \( \alpha = (\alpha_1, \ldots, \alpha_3)^T \). Using the partition of unity property of the barycentric basis functions, we get

\[
\exp(h_\mathbf{K}(\Omega) \cdot \alpha) = \exp(\sum_{i=1}^3 h_i \alpha_i) = \exp(h_1 \alpha_1 + h_2 \alpha_2 + (1 - h_1 - h_2) \alpha_3)
\]

Expanding the second exponential in a Taylor series representation gives

\[
\exp(h_\mathbf{K}(\Omega) \cdot \alpha) = \exp(\alpha_3) \sum_{k=0}^\infty \frac{\left(\sum_{i=1}^2 h_i (\alpha_i - \alpha_3)\right)^k}{k!} = \sum_{k=0}^\infty \sum_{k_1+k_2+k_3=k} \prod_{i=1}^2 \frac{(h_i (\alpha_i - \alpha_3))^{k_i}}{k_i!}
\]

where we used the multinomial theorem for the last equality. Interchanging summation and integration yields

\[
\int_\mathbf{K} f(\Omega) \exp(h_\mathbf{K}(\Omega) \cdot \alpha) \, d\Omega = \exp(h_3) \sum_{k=0}^\infty \sum_{k_1+k_2+k_3=k} \left(\prod_{i=1}^2 \frac{(\alpha_i - \alpha_3)^{k_i}}{k_i!}\right) \int_\mathbf{K} f(\Omega) \prod_{i=1}^2 h_i^{k_i} \, d\Omega
\]

The integrals \( \int_\mathbf{K} f(\Omega) \prod_{i=1}^2 h_i^{k_i} \, d\Omega \) are independent of \( \alpha \) can be precomputed once and for all (up to some maximal order), if the spherical triangle does not change. For the optimization algorithm, we need to calculate these integrals for

\[
f(\Omega) \in \{1, h_i(\Omega), \Omega_k h_i(\Omega), h_i(\Omega) h_j(\Omega), \Omega_k h_i(\Omega) h_j(\Omega)\} \text{ for } i, j, k \in \{1, 2, 3\}.
\]

Remark S1.1. Note that, if the spherical triangle is contained in an octant of the sphere, none of these possible choices for \( f(\Omega) \) changes sign over the domain of integration. Thus, if we order the multipliers such that \( \alpha_3 = \min_{i=1}^3 \alpha_i \), all terms of the Taylor expansion have the same sign. Hence, if we precalculate the integrals for all three possible choices of the basis function \( h_3 \), we can calculate the Taylor series without numerical cancelation.

This procedure allows for using high-order quadratures to precompute the integrals up to some maximal order (which is reasonably fast even for very fine quadratures). When solving the optimization problems, we only have to evaluate the Taylor series, which is considerably faster than using a quadrature of comparable order. We tested this procedure with a maximal order of 250 and it indeed worked quite well for the vast majority of optimization problems. However, for moments corresponding to anisotropic distributions, \( (\alpha_i - \alpha_3) \) may become arbitrarily large such that the Taylor series has to be evaluated up to a prohibitively high order. This leads to additional regularization for these moments which introduces additional errors.
S2. Quadrature sensitivity

For the minimum entropy models, we usually cannot calculate the integrals occurring in the minimum-entropy optimization problems analytically but have to use a quadrature. Choosing an appropriate quadrature is not trivial as it has a great influence on both realizability and performance. Due to realizability considerations, in one dimension, we chose Gauss-Lobatto quadratures. For the HFM and PMM models, we use one quadrature per interval $I_i$. The $\text{M}_N$ models do not use a subdivision of the quadrature domain in intervals. However, for the kinetic flux we integrate over the intervals $[-1,0]$ and $[0,1]$, so we use one quadrature on each of these two intervals. For the one-dimensional PMM and HFM models, the integrals can be solved analytically, which we did for the HFM models in our implementation. However, for the sake of completeness, we also tested the HFM models using quadratures instead of the analytical formulas.

To test which quadrature order to use and how sensitive the different models are with respect to the quadrature order, we solved some of our numerical test cases for different quadrature orders and calculated the errors with respect to the reference solution. For high quadrature orders, the integrals should be approximated very good such that the error with respect to the reference solution should only be due to the moment approximation, not due to errors in evaluating the integrals. We thus expect the errors to converge to a model-dependent limit value with increasing quadrature order.

The results for one dimension can be found in Figures S2.1, S2.2 and S2.3. The HFM and PMM models show similar behavior, which is expected as they are both first-order models and thus similar integrals have to be approximated. In the source-beam test case, both in $L^1$ and $L^\infty$ norm, the error mostly reaches its limit value already for a quadrature order of 5. The plane-source test case is more sensitive to badly approximated integrals and needs a quadrature order of about 11 to reach its limit. Unsurprisingly, models with fewer intervals are more sensitive to low-order quadratures. Given these results, we use a quadrature order of 15 for the HFM and PMM models in our numerical tests.

For the $\text{M}_N$ models, obviously, higher-order models need higher-order quadratures. For the plane-source test case, a quadrature order of $2N + 40$ seems appropriate. The source-beam test case is a special case due to the approximate Dirac boundary value. For the HFM and PMM models, the boundary value can be evaluated analytically, yielding a (numerically) realizable moment vector as the numerically realizable set equals the analytically realizable set. For the Legendre basis, however, we cannot evaluate the boundary integrals analytically and we cannot use an arbitrary high-order quadrature for the boundary-value only as the resulting moment vector might not be numerically realizable. We thus use the same quadrature to evaluate the boundary value as we use in the optimization problem. To fully resolve the boundary value, we have to use a much higher quadrature order than we need for the plane-source test case. To be on the safe side, we use a quadrature with 100 quadrature points per half interval (order 197) in this test case.

In three dimension, we use Fekete quadratures on each spherical triangle for the HFM and PMM models and tensor-product rules on the octants of the sphere for the $\text{M}_N$ models. To test the influence of the quadrature, we solve the pointsource problem (using $50^3$ grid cells) for each quadrature and calculate the error with respect to the analytical solution of the kinetic equation. The results for the HFM and PMM can be found in Figure S2.4. The models with 8 spherical triangles (HFM, PMM) give significantly different results when a low-order quadrature is used. A quadrature order of about 12 is needed to fully resolve the structure in these models. In contrast, the error graphs for the higher-order HFM and PMM are mostly flat, so these models do not profit from quadratures with degree larger than 6. Apparently, the finer triangulation of the quadrature domain in these models is sufficient to properly approximate the integrals even with low-order quadratures on each triangle. For the following numerical experiments, we thus use a quadrature order of 15 for HFM and PMM and order 9 for the other models.

For the $\text{M}_N$ models, the results can be found in Figure S2.5). Obviously, higher-order models need higher-order quadratures. We use a quadrature order of $2N + 8$ in the numerical experiments.
Figure S2.1: Analysis of the quadrature dependency of the PMM\(_n\) models.
Figure S2.2: Analysis of the quadrature dependency of the HFM_n models.
Figure S2.3: Analysis of the quadrature dependency of the $M_N$ models.
Figure S2.4: $L^1$ and $L^\infty$ errors of PMM$_n$ and HFM$_n$ models for different quadratures. A Fekete quadrature rule of the respective order is used on each spherical triangle.

Figure S2.5: Analysis of the quadrature dependency of the $M_N$ models.
S3. Supplementary figures

We here include some plots of the solutions of the Checkerboard and Shadow test cases. All models are shown as two-dimensional slices through the spatial domain, as well as isosurfaces in an endcap geometry (i.e., some portion of the surfaces is removed to get some insight into the interior of the solution).

Figure S3.1: Two-dimensional cuts and selected isosurfaces for some models in the checkerboard test, logarithmic scale.
Figure S3.2: Two-dimensional cuts and selected isosurfaces for some models in the checkerboard test, logarithmic scale. Negative values are shown in red.
Figure S3.3: Two-dimensional cuts and selected isosurfaces for some models in the checkerboard test, logarithmic scale. Negative values are shown in red.
Figure S3.4: Two-dimensional cuts and selected isosurfaces for some models in the shadow test, logarithmic scale. Negative values are shown in red.
Figure S3.5: Two-dimensional cuts and selected isosurfaces for some models in the shadow test, logarithmic scale. Negative values are shown in red.
Figure S3.6: Two-dimensional cuts and selected isosurfaces for some models in the shadow test, logarithmic scale. Negative values are shown in red.