BIT

Dynamical low-rank approximation of the Vlasov–Poisson equation with piecewise linear spatial boundary

André Uschmajew[1] · Andreas Zeiser[2]

Abstract

Dynamical low-rank approximation (DLRA) for the numerical simulation of Vlasov–Poisson equations is based on separation of space and velocity variables, as proposed in several recent works. The standard approach for the time integration in the DLRA model uses a splitting of the tangent space projector for the low-rank manifold according to the separated variables. It can also be modified to allow for rank-adaptivity. A less studied aspect is the incorporation of boundary conditions in the DLRA model. In this work, a variational formulation of the projector splitting is proposed which allows to handle inflow boundary conditions on spatial domains with piecewise linear boundary. Numerical experiments demonstrate the principle feasibility of this approach.

Communicated by David Cohen.

✉ Andreas Zeiser
andreas.zeiser@htw-berlin.de

André Uschmajew
andre.uschmajew@uni-a.de

[1] Institute of Mathematics & Centre for Advanced Analytics and Predictive Sciences, University of Augsburg, 86159 Augsburg, Germany

[2] Faculty 1: School of Engineering - Energy and Information, HTW Berlin - University of Applied Sciences, 12459 Berlin, Germany

⧜ Springer

1 Introduction

In this work we consider the dynamical low-rank approximation of the Vlasov–Poisson equation in $d \leq 3$ spatial dimensions:

$$\partial_t f + \boldsymbol{v} \cdot \nabla_{\boldsymbol{x}} f - \boldsymbol{E}(t, \boldsymbol{x}) \cdot \nabla_{\boldsymbol{v}} f = 0 \quad \text{in } \Omega = \Omega_x \times \Omega_v, \quad t \in (0, T), \qquad (1)$$

with a bounded domain $\Omega_x \subset \mathbb{R}^d$, and $\Omega_v = \mathbb{R}^d$. This equation models the time evolution of an electron density $f = f(t, \boldsymbol{x}, \boldsymbol{v})$ of a collisionless plasma as a function of space and velocity in the presence of an electrical field $\boldsymbol{E}$. We assume the initial condition

$$f(0, \boldsymbol{x}, \boldsymbol{v}) = f_0(\boldsymbol{x}, \boldsymbol{v})$$

and inflow boundary conditions of the form

$$f(t, \cdot, \cdot) = g(t, \cdot, \cdot), \quad \text{on } \Gamma^- = \{(\boldsymbol{x}, \boldsymbol{v}) \in \partial\Omega_x \times \Omega_v \,|\, \boldsymbol{v} \cdot \boldsymbol{n}_x < 0\}. \qquad (2)$$

Here $\boldsymbol{n}_x$ denote the outward normal vectors of the spatial domain Ω_x. The electrical field $\boldsymbol{E}$ can either be fixed or dependent on the density f via a Poisson equation:

$$\boldsymbol{E}(t, \boldsymbol{x}) = -\nabla_{\boldsymbol{x}} \Phi(t, \boldsymbol{x}), \quad -\Delta\Phi = \rho, \quad \rho(t, \boldsymbol{x}) = \rho_b - \int_{\Omega_v} f(t, \boldsymbol{x}, \boldsymbol{v}) \, \mathrm{d}\boldsymbol{v}, \qquad (3)$$

supplemented with appropriate boundary conditions. Here ρ_b is a background charge.

Simulations of such systems are computationally demanding since the time evolution of an $2d$-dimensional, i.e. up to six-dimensional, function has to be calculated. Applying standard discretization schemes thus leads to an evolution equation in $\mathcal{O}(n^{2d})$ degrees of freedom, where n is the number of grid points in one dimension and a corresponding high computational effort. To tackle the problem, methods such as particle methods [24], adaptive multiscale methods [5], and sparse grids [17] have been used.

In the seminal paper [7] dynamical low-rank approximation (DLRA) has been proposed for solving (1). DLRA is a general concept of approximating the evolution of time-dependent multivariate functions using a low-rank model, typically based on separation of variables. It originates in classic areas of mathematical physics, such as molecular dynamics [21], and has been proposed in the works [18, 19] as a general numerical tool for the time-integration of ODEs on fixed-rank matrix or tensor manifolds. An overview from the perspective of numerical analysis and further references can be found in [23]. The approach can also be made rigorous for low-rank functions in L_2-spaces thanks to their tensor product Hilbert space structure; see, e.g., [2].

For the simulation of (1) using DLRA, a rather natural separation of space and velocity variables is applied. After discretizing the problem in a corresponding tensor product discretization space, one then seeks an approximate solution curve for equation (1) of the form

$$f_r(t, \boldsymbol{x}, \boldsymbol{v}) = \sum_{i=1}^{r} \sum_{j=1}^{r} X_i(t, \boldsymbol{x}) S_{ij}(t) V_j(t, \boldsymbol{v}), \tag{4}$$

that is, $f_r(t, \cdot, \cdot)$ is a rank-r function for every time point t. For the numerical time-integration of the low-rank factors in this representation, [7] adopted the so-called projector-splitting approach from [22], which is one of the work-horse algorithms for DLRA, to the case of the transport Eq. (1).

In the present work, we wish to study in more detail how to incorporate the inflow boundary condition (2) into such a scheme. In [7] this question was somewhat circumvented by considering periodic boundary conditions, which, however, is not always applicable in real world problems. Enforcing boundary conditions directly on low-rank representations like (4) appears to be rather impractical, or at least poses some difficulties. For the nonlinear Boltzmann equation such an approach has been considered in [16]. Here, we will instead start out from a weak formulation of the transport problem (1) including a weak formulation of the boundary condition according to [11, Sec. 76.3]. We then use the projector splitting approach for the time-stepping in this weak formulation, where the problem is iteratively reduced to subspaces belonging to variations of space or velocity variables only. We note that in [15] boundary conditions for a two-dimensional product domain (corresponding to $d = 1$ in our notation) were included in this way through discretization by a discontinuous Galerkin method.

It is important to note already here that in order to still benefit from the seperation of variables, our approach assumes the boundary function g to be a finite sum of tensor products, or at least be approximable by such, see Eq. (7) below. Moreover, we require the spatial domain to have a piecewise linear boundary so that the outer normal vectors are piecewise constant, see Eq. (6).

As will be derived in Sect. 2, the resulting effective equations for the low-rank factors in our weak formulation of the projector splitting scheme take the form of Friedrichs' systems, that is, systems of hyperbolic equations in weak formulation, that respect the boundary conditions without violating the tensor product structure. These systems can hence be solved by established PDE solvers. We remark that our derivation of these equations remains more or less formal, and the existence of weak solutions (in the continuous setting) will not be studied. Let us also mention the work [20] which is related to our work insofar as it shows that for hyperbolic problems a continuous formulation of the projector splitting integrator prior to discretization has favorable numerical stability properties.

We then proceed by deriving corresponding discrete equations that allow to solve the system numerically. Specifically, in our implementation we will use stabilized finite elements [11] for discretization. We then apply the method to solve the weak Landau damping on a periodic domain (in order to verify the discretization) as well as a linear equation involving a piecewise linear boundary. Here, in addition to the standard projector-splitting approach we will also consider a weak formulation of the unconventional low-rank integrator proposed in [3]. It consists in modifications regarding the update strategy for the low-rank factors and allows to make the whole scheme rank-adaptive [4], which is important in practice, based on subspace augmentation. Similar strategies are considered in [13] and [15].

Several recent works have been focussing on the conservation properties in dynamical low-rank integrators [6, 8–10, 12, 14], such as mass, momentum and energy, based on modified Galerkin conditions. This important aspect is not yet addressed in the present paper and will remain for future work.

The paper is organized as follows. In Sect. 2 we detail our idea of a weak formulation of the projector-splitting integrator for the Vlasov–Poisson equation which is capable of handling the inflow boundary conditions. From this, in Sect. 3 we obtain discrete equations by restricting to finite-dimensional spaces according to the Galerkin principle, resulting in Algorithm 1. The rank-adaptive unconventional integrator is also considered (Algorithm 2). In Sect. 4 results of numerical experiments are presented to illustrate the principle feasibility of our algorithms.

2 Weak formulation of the projector splitting integrator

Our goal is to develop a projector splitting scheme for DLRA of the transport problem (1) that is capable of handling the inflow boundary conditions (2). To achieve this, we start out from a weak formulation of (1) including a weak formulation of the boundary condition according to [11, Sec. 76.3]. Assume for now that the electrical field is fixed. Let W be an appropriate closed subspace of $L_2(\Omega)$ (e.g. $W \subseteq H^1(\Omega)$ is sufficient) and denote by P_W the L_2-orthogonal projector onto W. Then the goal in the weak formulation is to find $f \in C^1([0, T], W)$ with $f(0) = P_W f_0$ and such that at every $t \in (0, T)$ it holds

$$\int_\Omega \partial_t f(t) w \, \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{v} + a(t, f(t), w) = \ell(w) \text{ for all } w \in W, \tag{5}$$

where

$$a(t, u, w) = \int_\Omega \boldsymbol{v} \cdot \nabla_{\boldsymbol{x}} u \, w - \boldsymbol{E}(t, \boldsymbol{x}) \cdot \nabla_{\boldsymbol{v}} u \, w \, \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{v} - \int_{\Gamma^-} \boldsymbol{v} \cdot \boldsymbol{n}_x \, u \, w \, \mathrm{d}s,$$

$$\ell(w) = - \int_{\Gamma^-} \boldsymbol{v} \cdot \boldsymbol{n}_x \, g \, w \, \mathrm{d}s.$$

Here we have taken into account that $\Omega_v = \mathbb{R}^d$ is unbounded and hence the normal vectors for Ω read $\boldsymbol{n} = (\boldsymbol{n}_x, \boldsymbol{0})$. While our derivations are based on the above formulation with fixed electric field $\boldsymbol{E}$, the case where $\boldsymbol{E}$ depends on f will be discussed in Sect. 2.5. The variational formulation (5) has the advantage that the boundary conditions are integrated in the bilinear form and right hand side. It therefore can be easily combined with the so-called time-dependent variational principle, also called Dirac–Frenkel principle, underlying DLRA.

In order to benefit from the separation of variables in DLRA, it will be necessary to impose some restrictions on the boundary and the function g in the inflow condition (2). Specifically, we assume that the spatial domain has a piecewise linear boundary, so that one has a decomposition

$$\Gamma_x := \partial\Omega_x = \bigcup_\nu \Gamma_x^{(\nu)}, \tag{6}$$

where each part $\Gamma_x^{(\nu)}$ has a *constant* outer normal vector $\boldsymbol{n}_x^{(\nu)}$. In addition, we require that g itself admits a separation into space and velocity variables by a finite sum of tensor products,

$$g(t, \boldsymbol{x}, \boldsymbol{v}) = \sum_\mu g_x^{(\mu)}(t, \boldsymbol{x}) \cdot g_v^{(\mu)}(t, \boldsymbol{v}), \tag{7}$$

or can at least be approximated in this form.

In the following sections we derive the projector splitting approach of DLRA in the weak formulation and derive effective equations for the low-rank factors in (4) in the form of Friedrichs' systems (systems of hyperbolic equations in weak formulation) that are amenable to established PDE solvers. In this way the boundary conditions will be incorporated without sacrificing the tensor product structure.

2.1 DLRA and projector splitting

We first present the basic idea of the projector splitting approach for the dynamical low-rank solution of the weak formulation (5). As required for DLRA, we consider a possibly infinite-dimensional tensor product subspace

$$W = W_x \otimes W_v \subseteq L_2(\Omega_x) \otimes L_2(\Omega_v) = L_2(\Omega)$$

where we assume that $W_x \subseteq L_2(\Omega_x)$ and $W_v \subseteq L_2(\Omega_v)$ are suitable subspaces to ensure that (5) is well-defined on W. For example, W_x and W_v could be subspaces of H^1, which would correspond to a mixed regularity with respect to space and velocity.

The corresponding manifold $\mathcal{M}_r$ of low-rank functions in W is

$$\mathcal{M}_r = \left\{ \varphi(\boldsymbol{x}, \boldsymbol{v}) = \sum_{i=1}^r \sum_{j=1}^r X_i(\boldsymbol{x}) S_{ij} V_j(\boldsymbol{v}) \,\middle|\, X_i \in W_x, \ V_j \in W_v, \ S_{ij} \in \mathbb{R} \right\}, \tag{8}$$

with the additional conditions that the $X_1, \ldots, X_r$ and $V_1, \ldots, V_r$ are orthonormal systems in W_x and W_v, respectively, and the matrix $S = [S_{ij}]$ has rank r. Following the Dirac-Frenkel variational principle, a dynamical low-rank approximation for the weak formulation (5) asks for a function $f_r \in C^1([0, T], W)$ such that $f_r(t) \in \mathcal{M}_r$ and

$$\int_\Omega \partial_t f_r(t) w \, d\boldsymbol{x} d\boldsymbol{v} + a(t, f_r(t), w) = \ell(w) \quad \text{for all } w \in T_{f_r(t)}\mathcal{M}_r, \tag{9}$$

for all $t \in (0, T)$, where $T_{f_r(t)}\mathcal{M}_r$ is the tangent space of $\mathcal{M}_r$ at $f_r(t)$ specified below. For the initial value one may take $f_r(0) = \tilde{f}_0$, where $\tilde{f}_0$ is a rank-r approximation of f_0.

The DLRA formulation (9) can be interpreted as a projection of the time derivative of the solution onto the tangent space, which implicitly restricts it to the manifold. The projector splitting integrator from [22] is now based on the rather peculiar fact that the tangent spaces of $\mathcal{M}_r$ can be decomposed into two smaller subspaces corresponding to variations in the X and V component only. Specifically, let

$$f_r(t, \boldsymbol{x}, \boldsymbol{v}) = \sum_{i=1}^{r} \sum_{j=1}^{r} X_i(t, \boldsymbol{x}) S_{ij}(t) V_j(t, \boldsymbol{v})$$

be in $\mathcal{M}_r$ at time t, then it holds that

$$T_{f_r(t)} \mathcal{M}_r = T_{V(t)} + T_{X(t)},$$

where

$$T_{V(t)} = \left\{ \sum_{i=1}^{r} K_i(\boldsymbol{x}) V_i(t, \boldsymbol{v}) \,\middle|\, K_i \in W_x \right\}, \quad T_{X(t)} = \left\{ \sum_{i=1}^{r} X_i(t, \boldsymbol{x}) L_i(\boldsymbol{v}) \,\middle|\, L_i \in W_v \right\}.$$

Note that these two subspaces are not complementary as they intersect in the space

$$T_{X(t), V(t)} = T_{X(t)} \cap T_{V(t)} = \left\{ \sum_{i=1}^{r} \sum_{j=1}^{r} X_i(t, \boldsymbol{x}) \tilde{S}_{ij} V_j(t, \boldsymbol{v}) \,\middle|\, \tilde{S}_{ij} \in \mathbb{R} \right\}.$$

Correspondingly, the L_2-orthogonal projection onto the full tangent space $T_{f_r(t)} \mathcal{M}_r$ can be decomposed as

$$P_{f_r(t)} = P_{V(t)} - P_{X(t), V(t)} + P_{X(t)}, \tag{10}$$

where $P_{V(t)}$, $P_{X(t), V(t)}$, and $P_{X(t)}$ denote the L_2-orthogonal projections onto the subspaces $T_{X(t)}$, $T_{X(t), V(t)}$, and $T_{V(t)}$, respectively.

The projector-splitting integrator performs the time integration of the system according to the decomposition (10) of the tangent space projector. Sticking to the weak formulation (9), one time step from some point t_0 with

$$f_r(t_0) = \sum_{i=1}^{r} \sum_{j=1}^{r} X_i^0(\boldsymbol{x}) S_{ij}^0 V_j^0(\boldsymbol{v}) \in \mathcal{M}_r \tag{11}$$

to a point $t_1 = t_0 + \Delta t$ then consists of the following three steps:

1. Solve the system (9) restricted to the subspace T_{V^0} on the time interval $[t_0, t_1]$ with initial condition $f_r(t_0)$. At time t_1 one obtains

$$\hat{f}(\boldsymbol{x}, \boldsymbol{v}) = \sum_{j=1}^{r} K_j(\boldsymbol{x}) V_j^0(\boldsymbol{v}) \in T_{V^0}.$$

Find an orthonormal system $X_1^1, \ldots, X_r^1$ for $K_1, \ldots, K_r$ such that

$$K_j = \sum_{i=1}^{r} X_i^1 \hat{S}_{ij}.$$

2. Solve the system (9) restricted to the subspace T_{X^1, V^0} on the time interval $[t_0, t_1]$ with initial condition $\hat{f}$, and taking into account the minus sign in the projector. At time t_1 one obtains

$$\tilde{f}(x, v) = \sum_{i=1}^{r} \sum_{j=1}^{r} X_i^1(x) \tilde{S}_{ij} V_j^0(v) \in T_{X^1, V^0}.$$

This step is often interpreted as a backward in time integration step, which is possible if there is no explicit time dependence in the coefficients or boundary conditions. However, in our case the inflow g is in general time dependent which then does not admit for such an interpretation in general.

3. Solve the system (9) restricted to the subspace T_{X^1} on the time interval $[t_0, t_1]$ with initial condition $\tilde{f}$. At time t_1 one obtains

$$\bar{f}(x, v) = \sum_{i=1}^{r} X_i^1(x) L_i(v) \in T_{X^1}.$$

Find an orthonormal system $V_1^1, \ldots, V_r^1$ for $L_1, \ldots, L_r$ such that

$$L_i = \sum_{j=1}^{r} S_{ij}^1 V_j^1.$$

As the final solution at time point t_1 one then takes

$$f_r(t_1, x, v) \approx \bar{f}(x, v) = \sum_{i=1}^{r} \sum_{j=1}^{r} X_i^1(x) S_{ij}^1 V_j^1(v).$$

A modification of this scheme is the so called unconventional integrator proposed in [3]. There, the K-step and L-step are performed independently using the same initial data from $f_r(t_0)$ (i.e. the K-step is identical with the one above, but the L-step is performed on T_{X^0} with initial value $f_r(t_0)$). As a result, one obtains two new orthonormal sets of component functions $X_1^1, \ldots, X_r^1$ and $V_1^1, \ldots, V_r^1$ for the spatial and velocity domains. The S-step is then performed afterwards in a "forward" way (i.e. without flipping signs) using these new bases. Compared to the standard scheme, this decoupling the S-step from the K- and L-steps also offers a somewhat more natural way of making the scheme rank-adaptive, which is important in practice since a good guess of r might not be known. Specifically, in [4] it is proposed to first augment the new bases to $\hat{X} = \{X_1^0, \ldots, X_r^0, K_1^1, \ldots, K_r^1\}$ and $\hat{V} = \{V_1^0, \ldots, V_r^0, L_1^1, \ldots, L_r^1\}$

(i.e. including the old bases), orthonormalize these augmented bases, and perform the S-step in the space

$$T_{\hat{X}, \hat{V}} = \operatorname{span} \hat{X} \otimes \operatorname{span} \hat{V}.$$

Note that this space contains functions of rank $2r$ (in general) and still contains $f_r(t_0)$ which one takes as initial condition for the S-step. The solution at time point t_1 can then be truncated back to rank r or any other rank (less or equal to $2r$) depending on a chosen truncation threshold using singular value decomposition.

Regardless of whether the projector splitting or the modified unconventional integrator is chosen, the efficient realization of the above steps is based on their formulation in terms of the component functions K, S and L only, which in turn requires a certain separability of space and velocity variables in the system (9). We therefore next investigate the single steps in detail, focusing only on the projector splitting integrator and omitting the required modifications for the (rank-adaptive) unconventional integrator (it will be discussed in the discrete setting in Sect. 3.3). A particular focus is how to ensure the required separability for the boundary value terms, which will lead to the Assumptions (7) and (6) that g is separable and the boundary is piecewise linear. As we will see, this gives effective equations in the form of Friedrichs' systems for the factors K and L, and a matrix ODE for S. Their discrete counter-parts as well as the algorithmic schemes for both the projector splitting integrator and the unconventional integrator will be presented in Sect. 3.

2.2 Weak formulation of the *K*-step

The time dependent solution of the first step, i.e. (9) restricted to T_{V^0}, is a function

$$\hat{f}(t, \boldsymbol{x}, \boldsymbol{v}) = \sum_{j=1}^{r} K_j(t, \boldsymbol{x}) \cdot V_j^0(\boldsymbol{v})$$

on the time interval $[t_0, t_1]$ with initial condition $\hat{f}(t_0, \boldsymbol{x}, \boldsymbol{v}) = f_r(t_0, \boldsymbol{x}, \boldsymbol{v})$ from (11). Here the V_j^0 are fixed and the functions $K_1, \ldots, K_r$ need to be determined. Their dynamics are governed by a system of first order partial differential equations which we derive in the following.

The test functions $w \in T_{V^0}$ have the form

$$w(\boldsymbol{x}, \boldsymbol{v}) = \sum_{j=1}^{r} \psi_j^{(x)}(\boldsymbol{x}) \cdot V_j^0(\boldsymbol{v}), \qquad \psi_j^{(x)} \in W_x,$$

for all $j = 1, \ldots, r$. Testing with w in (9) specifically means

$$
\begin{aligned}
\left(\partial_t \hat{f}(t) + \boldsymbol{v} \cdot \nabla_{\boldsymbol{x}} \hat{f}(t) - \boldsymbol{E}(t, \boldsymbol{x}) \cdot \nabla_{\boldsymbol{v}} \hat{f}(t), w \right)_{L_2(\Omega, \mathbb{R})} & \\
- \left(\boldsymbol{n}_x \cdot \boldsymbol{v} \, \hat{f}(t), w \right)_{L_2(\Gamma^-, \mathbb{R})} &= - \left(\boldsymbol{n}_x \cdot \boldsymbol{v} \, g, w \right)_{L_2(\Gamma^-, \mathbb{R})}. \quad (12)
\end{aligned}
$$

We will write this equation in vector form and therefore define

$$K = \left[K_j\right]_{j=1,\ldots,r}, \qquad \boldsymbol{\psi}^{(x)} = \left[\psi_j^{(x)}\right]_{j=1,\ldots,r},$$

which can be regarded as vector-valued functions in $(W_x)^r$.

We investigate the terms in (12) separately. The term regarding the inner product in $L_2(\Omega, \mathbb{R})$ has three parts. The first involves the time derivative:

$$
\begin{aligned}
\left(\partial_t \hat{f}(t), w\right)_{L_2(\Omega, \mathbb{R})} &= \sum_{i=1}^r \sum_{j=1}^r \int_{\Omega_x} \partial_t K_i(t, \boldsymbol{x}) \cdot \psi_j^{(x)}(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \cdot \int_{\Omega_v} V_i^0(\boldsymbol{v}) \cdot V_j^0(\boldsymbol{v}) \, \mathrm{d}\boldsymbol{v} \\
&= \sum_{j=1}^r \int_{\Omega_x} \partial_t K_j(t, \boldsymbol{x}) \cdot \psi_j^{(x)}(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \\
&= \left(\partial_t \boldsymbol{K}(t), \boldsymbol{\psi}^{(x)}\right)_{L_2(\Omega_x, \mathbb{R}^r)},
\end{aligned}
$$

where we have used the pairwise orthonormality of $V_1, \ldots, V_r$. The second part reads

$$
\begin{aligned}
\int_{\Omega} \boldsymbol{v} \cdot \nabla_{\boldsymbol{x}} \hat{f}(t, \boldsymbol{x}, \boldsymbol{v}) \cdot w(\boldsymbol{x}, \boldsymbol{v}) \, \mathrm{d}(\boldsymbol{x}, \boldsymbol{v}) \\
= \sum_{k=1}^d \sum_{i=1}^r \sum_{j=1}^r \int_{\Omega_x} \partial_{x_k} K_i(t, \boldsymbol{x}) \cdot \psi_j^{(x)}(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \cdot \int_{\Omega_v} v_k \cdot V_i^0(\boldsymbol{v}) \cdot V_j^0(\boldsymbol{v}) \, \mathrm{d}\boldsymbol{v} \\
= \sum_{k=1}^d \left(\mathcal{A}_x^k \cdot \partial_{x_k} \boldsymbol{K}(t), \boldsymbol{\psi}^{(x)}\right)_{L_2(\Omega_x, \mathbb{R}^r)}
\end{aligned}
$$

with the symmetric $r \times r$ matrix

$$
\mathcal{A}_x^k = \left[\int_{\Omega_v} v_k \cdot V_j^0(\boldsymbol{v}) \cdot V_i^0(\boldsymbol{v}) \, \mathrm{d}\boldsymbol{v}\right]_{i,j=1,\ldots,r}.
$$

Finally, the part involving the electrical field can be written as

$$
\begin{aligned}
\int_{\Omega} \boldsymbol{E}(t, \boldsymbol{x}) \cdot \nabla_{\boldsymbol{v}} \hat{f}(t, \boldsymbol{x}, \boldsymbol{v}) \cdot w(\boldsymbol{x}, \boldsymbol{v}) \, \mathrm{d}(\boldsymbol{x}, \boldsymbol{v}) \\
= \sum_{i=1}^r \sum_{j=1}^r \int_{\Omega_x} \sum_{k=1}^d \left[E_k(t, \boldsymbol{x}) \cdot \int_{\Omega_v} \partial_{v_k} V_i^0(\boldsymbol{v}) \cdot V_j^0(\boldsymbol{v}) \, \mathrm{d}\boldsymbol{v}\right] K_i(t, \boldsymbol{x}) \cdot \psi_j^{(x)}(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \\
= \left(\mathcal{K}_x(t, \cdot) \cdot \boldsymbol{K}(t), \boldsymbol{\psi}^{(x)}\right)_{L_2(\Omega_x, \mathbb{R}^r)}
\end{aligned}
$$

where

$$
\mathcal{K}_x(t, \boldsymbol{x}) = \left[\sum_{k=1}^d E_k(t, \boldsymbol{x}) \int_{\Omega_v} \partial_{v_k} V_j^0(\boldsymbol{v}) \cdot V_i^0(\boldsymbol{v}) \, \mathrm{d}\boldsymbol{v}\right]_{i,j=1,\ldots,r}. \tag{13}
$$

For the terms in (12) involving the inflow boundary Γ^- we make use of our assumption that the spatial boundary $\partial\Omega_x$ can be decomposed as in (6) with a constant normal vector $\boldsymbol{n}_x^{(v)}$ on each part $\Gamma_x^{(v)}$ of the boundary. We decompose Γ^- accordingly into

$$\Gamma^- = \bigcup_v \Gamma_x^{(v)} \times \Omega_v^{(v)}, \qquad \Omega_v^{(v)} = \{\boldsymbol{v} \in \Omega_v \mid \boldsymbol{n}_x^{(v)} \cdot \boldsymbol{v} < 0\}.$$

Then the boundary term from the bilinear form can be written as

$$\int_{\Gamma^-} \boldsymbol{n}_x \cdot \boldsymbol{v} \, \hat{f}(t, \boldsymbol{x}, \boldsymbol{v})) \cdot w(\boldsymbol{x}, \boldsymbol{v}) \, \mathrm{d}s$$

$$= \sum_{i=1}^{r} \sum_{j=1}^{r} \sum_v \int_{\Gamma_x^{(v)}} \Big[\int_{\Omega_v^{(v)}} \boldsymbol{n}_x^{(v)} \cdot \boldsymbol{v} \, V_i^0(\boldsymbol{v}) \, V_j^0(\boldsymbol{v}) \, \mathrm{d}\boldsymbol{v} \Big] K_i(t, \boldsymbol{x}) \cdot \psi_j^{(x)}(\boldsymbol{x}) \, \mathrm{d}s$$

$$= \big(\mathcal{B}_x(\cdot) \, \boldsymbol{K}(t), \boldsymbol{\psi}^{(x)} \big)_{L_2(\Gamma_x, \mathbb{R}^r)}$$

where

$$\mathcal{B}_x(\boldsymbol{x}) = \Big[\sum_v \chi_{\Gamma_x^{(v)}}(\boldsymbol{x}) \int_{\Omega_v^{(v)}} \boldsymbol{n}_x^{(v)} \cdot \boldsymbol{v} \, V_j^0(\boldsymbol{v}) \, V_i^0(\boldsymbol{v}) \, \mathrm{d}\boldsymbol{v} \Big]_{i,j=1,\dots,r}$$

with characteristic function χ.

For the boundary term on the right hand side we recall the decomposition (7) of the function g into a sum of tensor products. Proceeding as above we obtain

$$\int_{\Gamma^-} \boldsymbol{n}_x \cdot \boldsymbol{v} \, g(t, \boldsymbol{x}, \boldsymbol{v}) \cdot w(\boldsymbol{x}, \boldsymbol{v}) \, \mathrm{d}s$$

$$= \sum_{j=1}^{r} \sum_{v,\mu} \int_{\Gamma_x^{(v)}} \Big[\int_{\Omega_v^{(v)}} \boldsymbol{n}_x^{(v)} \cdot \boldsymbol{v} \, g_v^{(\mu)}(t, \boldsymbol{v}) \, V_j^0(\boldsymbol{v}) \, \mathrm{d}\boldsymbol{v} \Big] g_x^{(\mu)}(t, \boldsymbol{x}) \cdot \psi_j^{(x)}(\boldsymbol{x}) \, \mathrm{d}s$$

$$= \big(\boldsymbol{G}_x(t, \cdot), \boldsymbol{\psi}^{(x)} \big)_{L_2(\Gamma_x, \mathbb{R}^r)}$$

with

$$\boldsymbol{G}_x(t, \boldsymbol{x}) = \Big[\sum_{\mu,v} \chi_{\Gamma_x^{(v)}}(\boldsymbol{x}) \, g_x^{(\mu)}(t, \boldsymbol{x}) \int_{\Omega_v^{(v)}} \boldsymbol{n}_x^{(v)} \cdot \boldsymbol{v} \, g_v^{(\mu)}(\boldsymbol{v}) \, V_j^0(\boldsymbol{v}) \, \mathrm{d}\boldsymbol{v} \Big]_{j=1,\dots,r}.$$

In summary (12) is equivalent to the following system of first order partial differential equations for $\boldsymbol{K}(t)$ in weak formulation with boundary penalty,

$$\Big(\partial_t \boldsymbol{K}(t) + \sum_{k=1}^{d} \mathcal{A}_x^k \cdot \partial_k \boldsymbol{K}(t) - \mathcal{K}_x(t, \cdot) \, \boldsymbol{K}(t), \boldsymbol{\psi}^{(x)} \Big)_{L_2(\Omega_x, \mathbb{R}^r)}$$

$$- \big(\mathcal{B}_x(\cdot) \, \boldsymbol{K}, \boldsymbol{\psi}^{(x)} \big)_{L_2(\Gamma_x, \mathbb{R}^r)} = - \big(\boldsymbol{G}_x(t, \cdot), \boldsymbol{\psi}^{(x)} \big)_{L_2(\Gamma_x, \mathbb{R}^r)} \quad \text{for all } \boldsymbol{\psi}^{(x)} \in (W_x)^r,$$

$$\tag{14}$$

which needs to be solved for $K(t_1)$ with the initial values

$$K(t_0) = \Big[\sum_{i=1}^{r} X_i^0(\boldsymbol{x}) S_{ij}^0 \Big]_{j=1,\dots,r}$$

from the previous approximation (11).

2.3 Formulation of the *S*-step

The solution of the second step of the splitting integrator is a time-dependent function

$$\tilde{f}(t, \boldsymbol{x}, \boldsymbol{v}) = \sum_{i=1}^{r} \sum_{j=1}^{r} X_i^1(\boldsymbol{x}) S_{ij}(t) V_j^0(\boldsymbol{v})$$

in the subspace space T_{X^1, V^0}, where the evolution for S on the interval $[t_0, t_1]$ is governed by (9) restricted to test functions $w \in T_{X^1, V^0}$, but taking into account the minus sign of P_{X^1, V^0} in the corresponding projector splitting (10). The initial condition reads $\tilde{f}(t_0, \boldsymbol{x}, \boldsymbol{v}) = \hat{f}(t_1, \boldsymbol{x}, \boldsymbol{v}) = \sum_{i,j=1}^{r} X_i^1(\boldsymbol{x}) \hat{S}_{ij} V_j^0(\boldsymbol{v})$. The test function in $w \in T_{X^1, V^0}$ will be written as

$$w(\boldsymbol{x}, \boldsymbol{v}) = \sum_{m=1}^{r} \sum_{n=1}^{r} X_m^1(\boldsymbol{x}) \Sigma_{mn} V_n^0(\boldsymbol{v}), \qquad \Sigma = [\Sigma_{mn}] \in \mathbb{R}^{r \times r}.$$

For such test functions, we again consider the different contributions as in (12) (with $\tilde{f}$ instead of $\hat{f}$). For the time derivative we get

$$
\begin{aligned}
\big(\partial_t \tilde{f}(t), w\big)_{L_2(\Omega, \mathbb{R})} &= \sum_{ijmn} \dot{S}_{ij} \Sigma_{mn} \int_{\Omega_x} X_i^1(\boldsymbol{x}) X_m^1(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \cdot \int_{\Omega_v} V_j^0(\boldsymbol{v}) V_n^0(\boldsymbol{v}) \, \mathrm{d}\boldsymbol{v} \\
&= \big(\dot{S}, \Sigma\big)_F,
\end{aligned}
$$

where $(\cdot, \cdot)_F$ is the Frobenius inner product on $\mathbb{R}^{r \times r}$. The first two contributions of the bilinear form read

$$
\begin{aligned}
\int_{\Omega} &\boldsymbol{v} \cdot \nabla_{\boldsymbol{x}} \tilde{f}(t, \boldsymbol{x}, \boldsymbol{v}) \cdot w(\boldsymbol{x}, \boldsymbol{v}) \, \mathrm{d}(\boldsymbol{x}, \boldsymbol{v}) \\
&= \sum_{k=1}^{d} \sum_{ijmn} \underbrace{\int_{\Omega_x} \partial_{x_k} X_i^1(\boldsymbol{x}) \cdot X_m^1(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x}}_{:=[D^{(k,1)}]_{mi}} \cdot \underbrace{\int_{\mathbb{R}^d} v_k \cdot V_j^0(\boldsymbol{v}) \cdot V_n^0(\boldsymbol{v}) \, \mathrm{d}\boldsymbol{v}}_{:=[C^{(k,1)}]_{nj}} S_{ij} \Sigma_{mn} \\
&= \Big(\sum_{k=1}^{d} D^{(k,1)} S (C^{(k,1)})^T, \Sigma \Big)_F
\end{aligned}
$$

and

$$\int_{\Omega} \boldsymbol{E}(t, \boldsymbol{x}) \cdot \nabla_{\boldsymbol{v}} f(t, \boldsymbol{x}, \boldsymbol{v}) \cdot w(\boldsymbol{x}, \boldsymbol{v})\, \mathrm{d}(\boldsymbol{x}, \boldsymbol{v})$$

$$= \sum_{k=1}^{d} \sum_{ijmn} \underbrace{\int_{\Omega_x} E_k(t, \boldsymbol{x}) X_i^1(\boldsymbol{x}) \cdot X_m^1(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x}}_{:=[D^{(k,2)}(t)]_{mi}} \cdot \underbrace{\int_{\mathbb{R}^d} \partial_{v_k} \cdot V_j^0(\boldsymbol{v}) \cdot V_n^0(\boldsymbol{v})\, \mathrm{d}\boldsymbol{v}}_{:=[C^{(k,2)}]_{nj}}\, S_{ij}\,\Sigma_{mn}$$

$$= \left(\sum_{k=1}^{d} D^{(k,2)} S (C^{(k,2)})^T, \Sigma \right)_F .$$

The terms stemming from the boundary condition take the form

$$\int_{\Gamma^-} \boldsymbol{n}_x \cdot \boldsymbol{v}\, f(t, \boldsymbol{x}, \boldsymbol{v}) \cdot w(\boldsymbol{x}, \boldsymbol{v})\, \mathrm{d}s$$

$$= \sum_{\nu} \sum_{ijmn} \underbrace{\int_{\Gamma_x^{(\nu)}} X_i^1(\boldsymbol{x}) \cdot X_m^1(\boldsymbol{x})\, \mathrm{d}s}_{:=[B^{(\nu,x)}]_{mi}} \cdot \underbrace{\int_{\Omega_v^{(\nu)}} \boldsymbol{n}_x^{(\nu)} \cdot \boldsymbol{v}\, V_j^0(\boldsymbol{v}) V_n^0(\boldsymbol{v})\, \mathrm{d}\boldsymbol{v}}_{:=[B^{(\nu,v)}]_{nj}}\, S_{ij}\,\Sigma_{mn}$$

$$= \sum_{\nu} \left(B^{(\nu,x)} S (B^{(\nu,v)})^T, \Sigma \right)_F ,$$

and

$$\int_{\Gamma^-} \boldsymbol{n}_x \cdot \boldsymbol{v}\, g(t, \boldsymbol{x}, \boldsymbol{v}) \cdot w(\boldsymbol{x}, \boldsymbol{v})\, \mathrm{d}s$$

$$= \sum_{\nu\mu} \sum_{mn} \underbrace{\int_{\Gamma_x^{(\nu)}} g_x^{(\mu)}(t, \boldsymbol{x}) \cdot X_m^1(\boldsymbol{x})\, \mathrm{d}s}_{:=[g^{(\nu,\mu,x)}(t)]_m} \cdot \underbrace{\int_{\Omega_v^{(\nu)}} \boldsymbol{n}_x^{(\nu)} \cdot \boldsymbol{v}\, g_v^{(\mu)}(\boldsymbol{v})\, V_n^0(\boldsymbol{v})\, \mathrm{d}\boldsymbol{v}}_{:=[g^{(\nu,\mu,v)}(t)]_n}\, \Sigma_{mn}$$

$$= \left(G_S, \Sigma \right)_F ,$$

with

$$G_S = \sum_{\nu\mu} \boldsymbol{g}^{(\nu,\mu,x)} \cdot \left(\boldsymbol{g}^{(\nu,\mu,v)} \right)^T .$$

Putting everything together and testing with all $\Sigma = [\Sigma_{mn}] \in \mathbb{R}^{r \times r}$ yields the ODE

$$\dot{S} - \sum_{k=1}^{d} \left[D^{(k,1)} S (C^{(k,1)})^T - D^{(k,2)} S (C^{(k,2)})^T \right] + \sum_{\nu} B^{(\nu,x)} S (B^{(\nu,v)})^T = G_S$$

(15)

(in strong form) for obtaining $S(t_1) = \tilde{S}$ from the initial condition $S(t_0) = \hat{S}$. Note again that compared to the original problem (9) all signs (except the one of $\dot{S}$) have been flipped due to the negative sign of the corresponding projector in the splitting (10).

2.4 Weak formulation of the *L*-step

The equations for the *L*-step are derived in a similar way as the K-step in Sect. 2.2. Here we seek

$$\bar{f}(t, \boldsymbol{x}, \boldsymbol{v}) = \sum_{i=1}^{r} X_i^1(\boldsymbol{x}) \cdot L_i(t, \boldsymbol{v})$$

in the subspace T_{X^1} with fixed $X_1^1, \ldots, X_r^1$ from the initial value $\bar{f}(t_0, \boldsymbol{x}, \boldsymbol{v}) = \tilde{f}(t_1, \boldsymbol{x}, \boldsymbol{v})$. The test functions in T_{X^1} are of the form

$$w(\boldsymbol{x}, \boldsymbol{v}) = \sum_{i=1}^{r} X_i^1(\boldsymbol{x}) \cdot \psi_i^{(v)}(\boldsymbol{v}), \qquad \psi_i^{(v)} \in W_v.$$

Let again

$$\boldsymbol{L} = \big[L_i\big]_{i=1,\ldots,r}, \qquad \boldsymbol{\psi}^{(v)} = \big[\psi_i^{(v)}\big]_{i=1,\ldots,r}.$$

be vector valued functions in $(W_v)^r$.

When restricting the weak formulation as in (12) (with $\bar{f}$ instead of $\hat{f}$) to test functions in T_{X^1}, the first three resulting contributions take a similar form as in the K-step,

$$\big(\partial_t \boldsymbol{L}(t) - \sum_{k=1}^{d} \mathcal{A}_v^k(t) \cdot \partial_k \boldsymbol{L}(t) + \mathcal{K}_v(\cdot) \cdot \boldsymbol{L}(t), \boldsymbol{\psi}^{(v)}\big)_{L_2(\Omega_v, \mathbb{R}^r)}$$

but this time with

$$\mathcal{A}_v^k(t) = \Big[\int_{\Omega_x} E_k(t, \boldsymbol{x}) \, X_j^1(\boldsymbol{x}) \cdot X_i^1(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \Big]_{i,j=1,\ldots,r},$$

$$\mathcal{K}_v(\boldsymbol{v}) = \Big[\sum_{k=1}^{d} v_k \int_{\Omega_x} \partial_{x_k} X_j^1(\boldsymbol{x}) \cdot X_i^1(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \Big]_{i,j=1,\ldots,r} \tag{16}$$

(note that in the K-step $\mathcal{A}_x^k$ featured the velocity and $\mathcal{K}_x$ the electric field).

The boundary term in the bilinear form gives

$$
\int_{\Gamma^-} \boldsymbol{n}_x \cdot \boldsymbol{v} \, f(t, \boldsymbol{x}, \boldsymbol{v})) \cdot w(\boldsymbol{x}, \boldsymbol{v}) \, \mathrm{d}s
$$

$$
= \sum_{i=1}^{r} \sum_{j=1}^{r} \sum_{v} \int_{\Omega_v^{(v)}} \boldsymbol{n}_x^{(v)} \cdot \boldsymbol{v} \left[\int_{\Gamma_x^{(v)}} X_i^1(\boldsymbol{x}) \, X_j^1(\boldsymbol{x}) \, \mathrm{d}s \right] L_i(t, \boldsymbol{v}) \cdot \psi_j^{(v)}(\boldsymbol{v}) \, \mathrm{d}\boldsymbol{v}
$$

$$
= \left(\mathcal{B}_v(\cdot) \, \boldsymbol{L}, \, \boldsymbol{\psi}^{(v)} \right)_{L_2(\Omega_v, \mathbb{R}^r)},
$$

where

$$
\mathcal{B}_v(\boldsymbol{v}) = \left[\sum_v \chi_{\Omega_v^{(v)}}(\boldsymbol{v}) \boldsymbol{n}_x^{(v)} \cdot \boldsymbol{v} \int_{\Gamma_x^{(v)}} X_j^1(\boldsymbol{x}) \, X_i^1(\boldsymbol{x}) \, \mathrm{d}s \right]_{i,j=1,\dots,r}.
$$

Hence, the boundary condition on the spatial domain Ω_x leads to the additional multiplicative term on the whole domain Ω_v. Boundary terms for the velocity variable are not present, since since we assume $\Omega_v = \mathbb{R}^d$ to be unbounded.

For the right hand side, we have

$$
\int_{\Gamma^-} \boldsymbol{n}_x \cdot \boldsymbol{v} \, g(t, \boldsymbol{x}, \boldsymbol{v}) \cdot w(\boldsymbol{x}, \boldsymbol{v}) \, \mathrm{d}s
$$

$$
= \sum_{i=1}^{r} \sum_{\mu v} \int_{\Omega_v} \chi_{\Omega_v^{(v)}}(\boldsymbol{v}) \boldsymbol{n}_x^{(v)} \cdot \boldsymbol{v} \left[\int_{\Gamma_x^{(v)}} g_x^{(\mu)}(t, \boldsymbol{x}) \, X_i^1(\boldsymbol{x}) \, \mathrm{d}s \right] g_v^{(\mu)}(t, \boldsymbol{v}) \cdot \psi_i^{(v)}(\boldsymbol{v}) \, \mathrm{d}\boldsymbol{v}
$$

$$
= \left(\boldsymbol{G}_v(t, \cdot), \, \boldsymbol{\psi}^{(v)} \right)_{L_2(\Omega_v, \mathbb{R}^r)}
$$

with

$$
\boldsymbol{G}_v(t, \boldsymbol{v}) = \left[\sum_{\mu v} g_v^{(\mu)}(t, \boldsymbol{v}) \chi_{\Omega_v^{(v)}}(\boldsymbol{v}) \boldsymbol{n}_x^{(v)} \cdot \boldsymbol{v} \int_{\Gamma_x^{(v)}} g_x^{(\mu)}(t, \boldsymbol{x}) \, X_i^1(\boldsymbol{x}) \, \mathrm{d}s \right]_{i=1,\dots,r}.
$$

The resulting equation for the L-step reads: Find $\boldsymbol{L}(t) \in (W_v)^r$ for $t \in [t_0, t_1]$ such that

$$
\left(\partial_t \boldsymbol{L}(t) - \sum_{k=1}^{d} \mathcal{A}_v^k(t) \cdot \partial_k \boldsymbol{L}(t) + \mathcal{K}_v(\cdot) \, \boldsymbol{L}(t) - \mathcal{B}_v(\cdot) \boldsymbol{L}, \, \boldsymbol{\psi}^{(v)} \right)_{L_2(\Omega_v, \mathbb{R}^r)}
$$

$$
= -\left(\boldsymbol{G}_v(t, \cdot), \, \boldsymbol{\psi}^{(v)} \right)_{L_2(\Omega_v, \mathbb{R}^r)} \quad \text{for all } \boldsymbol{\psi}^{(v)} \in (W_v)^r. \tag{17}
$$

The initial condition is

$$
\boldsymbol{L}(t_0) = \left[\sum_{i=1}^{r} \tilde{S}_{ij}^0 \, V_j^0(\boldsymbol{v}) \right]_{j=1,\dots,r},
$$

where the V_j^0 are from $f_r(t_0)$ in the previous time step (11), and $\tilde{S}$ has been determined in the S-step.

2.5 Electrical field

So far we have assumed that the electrical field does not depend on the density f. However, in the setting of a Lie splitting the other case can be included quite easily as well. At the beginning of the time step from t_0 to t_1, as discussed in Sects. 2.2–2.4, the electrical field can be computed via the Poisson equation (3) and then simply held constant during that step. This will only introduce an error of first order in time, as does the projector splitting itself.

3 Discrete equations

In Sects. 2.2–2.4 the governing equations for the projector splitting integrator have been formulated in a continuous setting. They consist of the two Friedrichs' systems (14) and (17) for K and L, and the finite-dimensional system of ODEs (15) for S. In this section we will introduce a finite element discretization for the approximate numerical solution of the hyperbolic systems and derive the governing discrete equations that need to be solved in the practical computation.

A natural choice for the discretization is to use the discontinuous Galerkin method. However, the resulting equations for K, L, and S (Sect. 2) include derivatives of the bases functions X_i and V_j, see for example Eqs. (16) and (13). Hence, for a first approach we will use continuous finite element method, which of course has to be stabilized. It remains to future work to investigate the use of DG methods.

3.1 Discretization

Instead of an unbounded domain we will now use a finite domain $\Omega_v = [-v_{\max}, v_{\max}]^d$ for the velocity variable with $v_{\max}$ big enough, and impose periodic boundary conditions.

Let meshes $\mathcal{T}_{x/v}$ on the domains $\Omega_{x/v}$ be given and define H^1-conforming discretization spaces

$$W_x^h = \operatorname{span}\{\varphi_\alpha^{(x)} \mid \alpha = 1, \ldots, n_x\}, \qquad W_v^h = \operatorname{span}\{\varphi_\beta^{(v)} \mid \beta = 1, \ldots, n_v\}.$$

We are now looking for an approximate solution of the Vlasov–Poisson equation (1) in the manifold

$$\mathcal{M}_r^h = \left\{\varphi^h(\boldsymbol{x}, \boldsymbol{v}) = \sum_{i=1}^r \sum_{j=1}^r X_i^h(\boldsymbol{x}) S_{ij} V_j^h(\boldsymbol{v}) \,\middle|\, X_i^h \in W_x^h, \ V_j^h \in W_v^h, \ S_{ij} \in \mathbb{R}\right\},$$

of rank-r finite element functions, where as before the systems $X_1^h, \ldots, X_r^h$ and $V_1^h, \ldots, V_r^h$ are assumed to be orthonormal, and $S = [S_{ij}]$ has rank r. To represent elements in $\mathcal{M}_r^h$, we introduce coefficient matrices

$$\mathsf{X} = [\mathsf{X}_{\alpha i}] \in \mathbb{R}^{n_x \times r}, \qquad \mathsf{V} = [\mathsf{V}_{\beta j}] \in \mathbb{R}^{n_v \times r}$$

such that

$$X_i^h(\boldsymbol{x}) = \sum_{\alpha=1}^{n_x} \mathsf{X}_{\alpha i} \varphi_\alpha^{(x)}(\boldsymbol{x}), \qquad V_j^h(t, \boldsymbol{v}) = \sum_{\beta=1}^{n_v} \mathsf{V}_{\beta j} \varphi_\beta^{(v)}(\boldsymbol{v}). \tag{18}$$

The goal is then to find the solution in the form

$$f_r^h(t, \boldsymbol{x}, \boldsymbol{v}) = \sum_{\alpha=1}^{n_x} \sum_{\beta=1}^{n_v} \mathsf{f}_{\alpha\beta}(t) \varphi_\alpha^{(x)}(\boldsymbol{x}) \varphi_\beta^{(v)}(\boldsymbol{v})$$

with coefficients

$$\mathsf{f}(t) = [\mathsf{f}_{\alpha\beta}(t)] = \mathsf{X}(t) \cdot \mathsf{S}(t) \cdot \mathsf{V}(t)^T \in \mathbb{R}^{n_x \times n_v}$$

at prescribed time points t. Note that in order to have the $X_1^h, \ldots, X_k^h$ orthonormal in L_2 as was always assumed above, the matrix X needs to be orthogonal with respect to the inner product of the mass matrix of the basis functions $\varphi_\alpha^{(x)}$ (the matrix M_x below). A similar remark applies to V.

Following the Galerkin principle, the steps of the projector splitting integrator can be formulated in terms of the matrices X, S, and V by solving the Eqs. (14), (15), and (17) with the finite element spaces $W_{x/v}^h$ instead of $W_{x/v}$. This means that we use the finite dimensional spaces $(W_{x/v}^h)^r$ both as ansatz and test spaces. Specifically, for the Friedrichs' systems (14) and (17) our aim is to find factors K_j^h and L_i^h of the form

$$K_j^h(t, \boldsymbol{x}) = \sum_{\alpha=1}^{n_x} \mathsf{K}_{\alpha j}(t) \varphi_\alpha^{(x)}(\boldsymbol{x}), \qquad L_i^h(t, \boldsymbol{v}) = \sum_{\beta=1}^{n_v} \mathsf{L}_{\beta i}(t) \varphi_i^{(v)}(\boldsymbol{v}) \tag{19}$$

for $i, j = 1, \ldots, r$. We gather the coefficients in the matrices

$$\mathsf{K}(t) = \left[\mathsf{K}_{\alpha j}(t) \right]_{\alpha, j} \in \mathbb{R}^{n_x \times r}, \quad \mathsf{L}(t) = \left[\mathsf{L}_{\beta i}(t) \right]_{\beta, i} \in \mathbb{R}^{n_v \times r}.$$

The governing discrete equations are obtained by inserting the expressions (18) for X_i and V_j as well as (19) for K_i and L_j into (14), (15), and (17). In order to formulate them conveniently we will use the following additional discretization matrices:

$$\mathsf{M}_x = \left[\langle \varphi_\alpha^{(x)}, \varphi_{\alpha'}^{(x)} \rangle_{L_2(\Omega_x)} \right]_{\alpha, \alpha'}, \qquad \mathsf{M}_{x, E_k} = \left[\langle \varphi_\alpha^{(x)}, E_k \varphi_{\alpha'}^{(x)} \rangle_{L_2(\Omega_x)} \right]_{\alpha, \alpha'},$$

$$\mathsf{T}_{x, k} = \left[\langle \varphi_\alpha^{(x)}, \partial_{x_k} \varphi_{\alpha'}^{(x)} \rangle_{L_2(\Omega_x)} \right]_{\alpha, \alpha'}, \qquad \mathsf{M}_{x, \Gamma_x^{(v)}} = \left[\langle \varphi_\alpha^{(x)}, \varphi_{\alpha'}^{(x)} \rangle_{L_2(\Gamma^v)} \right]_{\alpha, \alpha'},$$

$$\mathsf{M}_v = \left[\langle \varphi_\beta^{(v)}, \varphi_{\beta'}^{(v)} \rangle_{L_2(\Omega_v)} \right]_{\beta, \beta'}, \qquad \mathsf{M}_{v, k} = \left[\langle \varphi_\beta^{(v)}, v_k \varphi_{\beta'}^{(v)} \rangle_{L_2(\Omega_v)} \right]_{\beta, \beta'},$$

$$\mathsf{T}_{v, k} = \left[\langle \varphi_\beta^{(v)}, \partial_{v_k} \varphi_{\beta'}^{(v)} \rangle_{L_2(\Omega_v)} \right]_{\beta, \beta'}, \qquad \mathsf{M}_{v, \Omega^{(v)}} = \left[\langle \varphi_\beta^{(v)}, n^{(v)} \cdot v \, \varphi_{\beta'}^{(v)} \rangle_{L_2(\Omega_v^{(v)})} \right]_{\beta, \beta'},$$

$$\mathsf{G}_x(t) = \left[\langle \varphi_\alpha^{(x)}, G_{x, j}(t, \cdot) \rangle_{L_2(\Gamma_x, \mathbb{R}^r)} \right]_{\alpha, j}, \qquad \mathsf{G}_v(t) = \left[\langle \varphi_\beta^{(v)}, G_{v, i}(t, \cdot) \rangle_{L_2(\Omega_v, \mathbb{R}^r)} \right]_{\beta, i}.$$

In addition, we will take into account that finite element solutions of hyperbolic partial differential equations have to be stabilized. For that purpose we use the continuous interior penalty (CIP) stabilization [11]. For a mesh $\mathcal{T}$, its bilinear form is given by

$$s_{\mathcal{T}}^{\text{CIP}}(v_h, w_h) = \sum_{F \in \mathcal{F}_0^h} h_F^2 \, ([\![\nabla v_h]\!], [\![\nabla w_h]\!])_{L_2(F)}$$

where $\mathcal{F}_0^h$ is the set of all interior interfaces of $\mathcal{T}$ and $[\![\cdot]\!]$ is the jump term with respect to the interface F. The corresponding discretization matrices are denoted by

$$\mathsf{C}_x = \big[s_{\mathcal{T}_x}^{\text{CIP}}(\varphi_j^{(x)}, \varphi_i^{(x)})\big]_{i,j}, \qquad \mathsf{C}_v = \big[s_{\mathcal{T}_v}^{\text{CIP}}(\varphi_j^{(v)}, \varphi_i^{(v)})\big]_{i,j}.$$

Eventually, the discrete versions of Eqs. (14), (15), and (17) read as follows:

$$\mathsf{M}_x \dot{\mathsf{K}} = -\sum_{k=1}^{d} \Big(\mathsf{T}_{x,k} \cdot \mathsf{K} \cdot \langle \mathsf{M}_{v,k}\rangle_\mathsf{V}^T - \mathsf{M}_{x,E_k} \cdot \mathsf{K} \cdot \langle \mathsf{T}_{v,k}\rangle_\mathsf{V}^T\Big) - \delta \, \mathsf{C}_x \cdot \mathsf{K}$$
$$+ \sum_v \mathsf{M}_{x,\Gamma_x^{(v)}} \cdot \mathsf{K} \cdot \langle \mathsf{M}_{v,\Omega_v^{(v)}}\rangle_\mathsf{V}^T - \mathsf{G}_x(t), \tag{20a}$$

$$\dot{S} = \sum_{k=1}^{d} \Big(\langle \mathsf{T}_{x,k}\rangle_\mathsf{X} \cdot S \cdot \langle \mathsf{M}_{v,k}\rangle_\mathsf{V}^T - \langle \mathsf{M}_{x,E_k}\rangle_\mathsf{X} \cdot S \cdot \langle \mathsf{T}_{v,k}\rangle_\mathsf{V}^T\Big)$$
$$- \sum_v \langle \mathsf{M}_{x,\Gamma_x^{(v)}}\rangle_\mathsf{X} \cdot S \cdot \langle \mathsf{M}_{v,\Omega_v^{(v)}}\rangle_\mathsf{V}^T + G_S, \tag{20b}$$

$$\mathsf{M}_v \dot{\mathsf{L}} = -\sum_{k=1}^{d} \Big(\mathsf{M}_{v,k} \cdot \mathsf{L} \cdot \langle \mathsf{T}_{x,k}\rangle_\mathsf{X}^T - \mathsf{T}_{v,k} \cdot \mathsf{L} \cdot \langle \mathsf{M}_{x,E_k}\rangle_\mathsf{X}^T\Big) - \delta \, \mathsf{C}_v \cdot \mathsf{L}$$
$$+ \sum_v \mathsf{M}_{v,\Omega_v^{(v)}} \cdot \mathsf{L} \cdot \langle \mathsf{M}_{x,\Gamma_x^{(v)}}\rangle_\mathsf{X}^T - \mathsf{G}_v(t). \tag{20c}$$

Here we use the abbreviations

$$\langle \mathsf{M}\rangle_\mathsf{X} = \mathsf{X}^T \cdot \mathsf{M} \cdot \mathsf{X}, \quad \langle \mathsf{M}\rangle_\mathsf{V} = \mathsf{V}^T \cdot \mathsf{M} \cdot \mathsf{V}$$

for discretization matrices M. The parameter $\delta \geq 0$ controls the stabilization.

3.2 Discrete projector splitting integrator

Based on the above equations, the complete realization of the projector splitting scheme, including the appropriate initial conditions and the orthogonalization steps, is outlined in Algorithm 1.

It also includes in lines 1 and 2 the computation of the electric field as described in Sect. 2.5. For that purpose the density ρ has to be calculated given by $f_r^h(t_0, \cdot, \cdot) \in \mathcal{M}_r^h$

by integrating out the v variable. The result is a finite element function $\rho^h \in W_x^h$ which is used on the right hand side of the Poisson equation (3). We used ansatz functions of order two to solve the resulting linear equation directly for the potential Φ^h. The electrical field $\boldsymbol{E}^h$ can be computed as the negative gradient of Φ^h and is a discontinuous finite element function on the same mesh. Now, updating the discretization matrices M_{x,E_k} involving the electrical field, the projector splitting steps from t_0 and t_1 can be performed.

Algorithm 1 First order dynamical low rank integrator with projector splitting

---

Data: X^0, S^0, V^0
    ▷ update electrical field
1: Solve Poisson equation (3) for density resulting from X^0, S^0, V^0
2: Compute $\boldsymbol{E}$ and update M_{x,E_k}
    ▷ K step
3: $K^0 = X^0 S^0$
4: Solve (20a) with $V = V^0$ on interval $[t_0, t_1]$ with initial condition K^0 to obtain K^1
5: Compute orthonormal basis X^1 with respect to M_x such that $X^1 \cdot \hat{S} = K^1$
    ▷ S step
6: Solve (20b) with $X = X^1$, $V = V^0$ on interval $[t_0, t_1]$ with initial condition $\hat{S}$ to obtain $\tilde{S}$
    ▷ L step
7: $L^0 = V^0 \tilde{S}^T$
8: Solve (20c) on interval $[t_0, t_1]$ with initial condition L^0 to obtain L^1
9: Compute orthonormal basis V^1 with respect to M_v such that $V^1 \cdot (S^1)^T = L^1$
10: **return** X^1, S^1, V^1

3.3 Rank adaptive algorithm

The rank-adaptive algorithm is based on the unconventional low-rank integrator proposed in [3] and its rank-adaptive extension in [4]. It is displayed in Algorithm 2. Starting with a rank-r_0 function, it first performs independent K-steps and L-steps from the initial data X^0 and V^0. The computed factors K^1 and L^1 are used to enlarge the previous bases for X^0 and V^0 to dimension $2r_0$ each. This is achieved by computing orthonormal bases $\hat{X}$ and $\hat{V}$ of the augmented matrices $[X^0, K^1] \in \mathbb{R}^{n_x \times 2r_0}$ and $[V^0, L^1] \in \mathbb{R}^{n_v \times 2r_0}$. Then a new coefficient matrix of the form

$$[f_{\alpha\beta}] = \hat{X} S \hat{V}^T$$

is sought by performing a 'forward' S-step

$$\dot{S} = -\sum_{k=1}^{d} \left(\langle T_{x,k} \rangle_{\hat{X}} \cdot S \cdot \langle M_{v,k} \rangle_{\hat{V}}^T - \langle M_{x,E_k} \rangle_{\hat{X}} \cdot S \cdot \langle T_{v,k} \rangle_{\hat{V}}^T \right)$$

$$+ \sum_{v} \langle M_{x,\Gamma_x^{(v)}} \rangle_{\hat{X}} \cdot S \cdot \langle M_{v,\Omega_v^{(v)}} \rangle_{\hat{V}}^T - G_S, \tag{21}$$

which differs from (20b) in the signs of the right-hand side. For the initial condition $S(t_0) = \hat{S}$, $\mathsf{X}^0 S^0 \mathsf{V}^0$ has to be expressed with respect to the larger bases $\hat{\mathsf{X}}$ and $\hat{\mathsf{V}}$, i.e. as $\hat{\mathsf{X}} \hat{S} \hat{\mathsf{V}}$. The new solution then has rank $2r_0$ (in general) and can be truncated to a lower rank r_1 according to a tolerance, which yield the actual new factors $\mathsf{X}^1 \in \mathbb{R}^{n_x \times r_1}$, $S^1 \in \mathbb{R}^{r_1 \times r_1}$, and $\mathsf{V}^1 \in \mathbb{R}^{n_v \times r_1}$.

Algorithm 2 First order rank adaptive unconventional integrator (RAUC)

Data: X^0, S^0, V^0, with $S^0 \in \mathbb{R}^{r_0 \times r_0}$, tolerance ϵ
  ▷ update electrical field
1: Solve Poisson equation (3) for density resulting from X^0, S^0, V^0
2: Compute $\boldsymbol{E}$ and update M_{x, E_k}
  ▷ Compute augmented X basis
3: $\mathsf{K}^0 = \mathsf{X}^0 S^0$
4: Solve (20a) on interval $[t_0, t_1]$ with initial condition K^0 to obtain K^1
5: Compute orthonormal basis $\hat{\mathsf{X}}$ with respect to M_x such that $\hat{\mathsf{X}} \cdot [R_x, \tilde{R}_x] = [\mathsf{X}^0, \mathsf{K}^1]$
  ▷ Compute augmented V basis
6: $\mathsf{L}^0 = \mathsf{V}^0 (S^0)^T$
7: Solve (20c) on interval $[t_0, t_1]$ with initial condition L^0 to obtain L^1
8: Compute orthonormal basis $\hat{\mathsf{V}}$ with respect to M_v such that $\hat{\mathsf{V}} \cdot [R_v, \tilde{R}_v] = [\mathsf{V}^0, \mathsf{L}^1]$
  ▷ S step
9: Compute initial condition $\hat{S} = R_x \cdot S^0 \cdot R_v^T$
10: Solve modified equation (21) on interval $[t_0, t_1]$ with initial condition $\hat{S}$ to obtain $\tilde{S}$
  ▷ truncation
11: Compute SVD $Q_x \cdot \Sigma \cdot Q_v^T = \tilde{S}$ with $\Sigma = \mathrm{diag}(\sigma_i)$ and monotonically decreasing σ_i
12: For tolerance ϵ compute $r_1 \leq 2r_0$ as the minimal number such that

$$\sum_{i=r_1+1}^{2r_0} \sigma_i^2 < \epsilon^2.$$

13: Set $\mathsf{X}^1 = \hat{\mathsf{X}} \cdot Q_x(:, 1 : r_1)$, $\mathsf{V}^1 = \hat{\mathsf{V}} \cdot Q_v(:, 1 : r_1)$, $S^1 = \mathrm{diag}(\sigma_i)_{i=1,\dots,r_1}$
14: **return** X^1, S^1, V^1

---

4 Numerical experiments

We present numerical results of two experiments for testing the methods described in Sect. 3. Since to our knowledge there does not exist an analytic solution of the Vlasov–Poisson equation on bounded domains to which our numerical results could be compared, we chose to consider the following two setups. The first experiment will be the classical Landau damping where analytic results on the decay of the electric energy are known. However, the domain is assumed to be periodic so that no boundary conditions are needed. This scenario has been treated in several previous works. Our second experiment will then include boundary conditions on a polygonal spatial domain to actually test our proposed approach for handling these.

In our tests linear finite elements are used and the matrix ODEs (lines 4, 6, and 8 in Algorithm 1, and line 11 in Algorithm 2) are solved using an explicit Runge-Kutta method of third order. The implementation is based on the finite element library MFEM

[1] and uses its Python wrapper PyMFEM.[1] The computations are carried out using standard numerical routines from numpy and scipy. All experiments have been performed on a desktop computer (i9 7900, 128 GB RAM), the code is available online.[2]

4.1 Landau damping

As a first numerical test we consider the classical Landau damping in two spatial and velocity dimensions. We use periodic domains $\Omega_x = [0, 4\pi]^2$ and $\Omega_v = [-6, 6]^2$ and a background density of $\rho_b = 1$. As initial condition we choose

$$f(0, \boldsymbol{x}, \boldsymbol{v}) = \frac{1}{2\pi} e^{-|\boldsymbol{v}|^2/2} \left(1 + \alpha \cos(kx_1) + \alpha \cos(kx_2)\right), \quad \alpha = 10^{-2}, \ k = \frac{1}{2}.$$

The same setup was investigated in [7]. For this case linear analysis shows that the electric field decays with a rate of $\gamma \approx 0.153$. For the discretization we use regular grids with $n_x = 64^2$ and $n_v = 256^2$ degrees of freedom (level 0).

Using Algorithm 1 the simulation was carried out fixed ranks $r = 5, 10, 15$ and a time step of $\Delta t = 0.005$. The results are shown in Fig. 1. As can be seen in the top left plot, for rank $r = 5$ the computed electric energy $\frac{1}{2} \int_{\Omega_x} |\boldsymbol{E}(t, \boldsymbol{x})|^2 \, d\boldsymbol{x}$ exhibits the analytical decay rate approximately up to $t = 35$. For rank $r = 10$ the electrical energy starts to deviate at the end of the time interval, while the solution with rank $r = 15$ shows the correct rate within the full simulation time.

Furthermore, it is known that the particle number, the total energy and the entropy, that is, the quantities

$$\int_\Omega f(t, \boldsymbol{x}, \boldsymbol{v}) \, d\boldsymbol{x} \, d\boldsymbol{v}, \quad \frac{1}{2} \int_\Omega |\boldsymbol{v}|^2 \, f(t, \boldsymbol{x}, \boldsymbol{v}) \, d\boldsymbol{x} \, d\boldsymbol{v} + \frac{1}{2} \int_{\Omega^{(x)}} |\boldsymbol{E}(t, \boldsymbol{x})|^2 \, d\boldsymbol{x},$$
$$\int_\Omega |f(t, \boldsymbol{x}, \boldsymbol{v})|^2 \, d\boldsymbol{x} \, d\boldsymbol{v}, \tag{22}$$

are invariants of the exact solution. In Fig. 1 we see that the mass and the total energy are almost conserved, whereas the entropy is only preserved up to a small error.

For the rank adaptive case (Algorithm 2) the system was simulated for the same discretization (level 0) and different tolerances ϵ. The corresponding results in Fig. 2 show that the electric energy deteriorates at around $t = 25$ for both tolerances $\epsilon = 10^{-5}, 10^{-6}$. Although the rank increases up to 40 (the maximal rank allowed) for the case of $\epsilon = 10^{-6}$ the accuracy in the electric energy does not improve. To improve accuracy the simulation is carried out on a uniformly refined spatial and velocity mesh (level 1). The results for a time step of $\Delta t = 0.00125$ and a tolerance of $\epsilon = 10^{-6}$ are shown in Fig. 2. The electric energy shows the correct decay in electric energy up to approximately $t = 40$.

Investigating the computed bases X and V in more detail shows that in the rank adaptive case spurious oscillations are present. In Fig. 3 two exemplary basis functions

---

[1] https://github.com/mfem/PyMFEM.
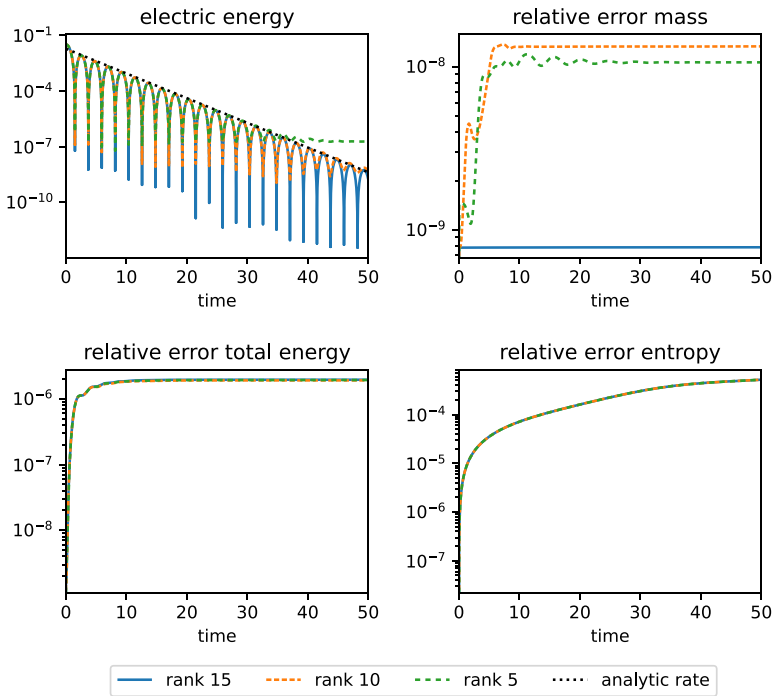
[2] https://github.com/azeiser/dlra-bc.

Fig. 1 Simulation results of the 2+2-dimensional Landau damping using fixed ranks (Alg. 1); electric energy including the analytical decay rate (upper left) and relative error of the invariants (22). For the total energy and entropy the lines for all three ranks almost overlap
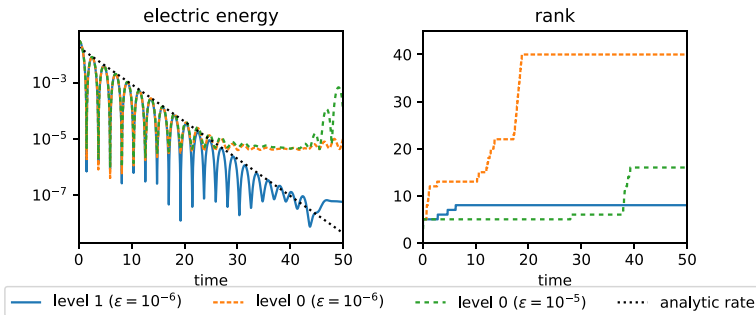

Fig. 2 Simulation results of the 2+2-dimensional Landau damping using the rank adaptive algorithm (Algorithm 2) for different tolerances ϵ and discretizations; electric energy including the analytical decay rate (left) and ranks (right)

$X_i(\boldsymbol{x})$ of level 0 at time $t = 50$ are displayed. In the fixed rank case ($r = 15$) the basis function is much smoother than in the rank adaptive simulation ($\epsilon = 10^{-5}$).

In summary spurious modes may enter in the course of the simulation for the rank adaptive simulation. However, the accuracy can be improved by refining the discretization. In contrast, the algorithm using a fixed rank seems to have a regularizing effect. It remains to future work to investigate this effect more closely.
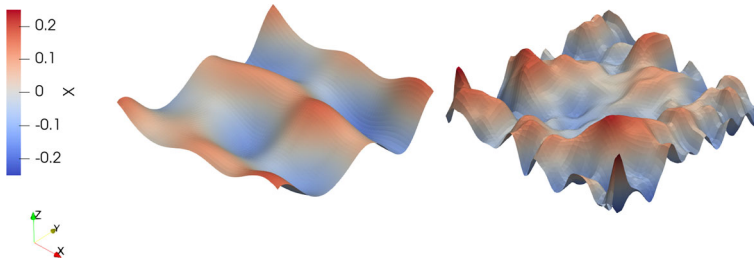
Fig. 3 Spatial basis functions $X_i(x)$ at time $t = 50$ for fixed rank ($r = 15$, left) and rank adaptive (level 0, $\epsilon = 10^{-5}$, right) simulation

4.2 Inflow boundary condition with constant electrical field

In the second example we focus on the boundary condition and compare the numerical to an analytical solution. In this setting, we are solving the transport equation (1) in $2 + 2$ dimensions with a constant electrical field $\boldsymbol{E} = [0, 4]^T$ on a triangular spatial domain

$$\Omega_x = \{(x_1, x_2) \mid -0.5 < x_1 < 0.5, \ -x_1/2 + 1/4 < x_2 < x_1/2 - 1/4\}$$

with initial condition $f(0, \boldsymbol{x}, \boldsymbol{v}) = 0$ and $\Omega_v = \mathbb{R}^2$. The inflow

$$f(t, \boldsymbol{x}, \boldsymbol{v}) = \bar{f}(t, \boldsymbol{x}, \boldsymbol{v}) \quad \text{on } \Gamma^-. \tag{23}$$

on the boundary of Ω_x will be determined by a function $\bar{f}$, which is a solution of the same equation as for f, but on the whole domain $\mathbb{R}^2$ and with initial condition

$$\bar{f}(0, \boldsymbol{x}, \boldsymbol{v}) = \bar{f}_0(\boldsymbol{x}, \boldsymbol{v}).$$

Here, $\bar{f}_0$ has compact spatial support just outside the triangular domain Ω_x and a compactly supported velocity distribution centred around $\boldsymbol{v} = [2, 0]^T$. More precisely, we set

$$\bar{f}_0(\boldsymbol{x}, \boldsymbol{v}) = \phi\Big(\frac{x_1 - 0.5 - \sigma_x}{\sigma_x}\Big) \cdot \phi\Big(\frac{x_2 - 0.1}{\sigma_x}\Big) \cdot \phi\Big(\frac{v_1 - 2}{\sigma_v}\Big) \cdot \phi\Big(\frac{v_2}{\sigma_v}\Big), \tag{24}$$

where $\sigma_x = 0.2$, $\sigma_v = 0.5$, and

$$\phi(z) = \begin{cases} z^2 \cdot (2|z| - 3) + 1 & |z| \leq 1 \\ 0 & |z| > 1 \end{cases}$$

is a C^1 function supported in $[-1, 1]$ and centered around 0. In consequence $\bar{f}_0$ is a product function supported in a four-dimensional cube with side lengths controlled by σ_x and σ_v.

By the method of characteristics we obtain

$$\bar{f}(t, \boldsymbol{x}, \boldsymbol{v}) = \bar{f}_0(\boldsymbol{x} - \boldsymbol{v} \cdot t - \boldsymbol{E} \cdot t^2/2, \boldsymbol{v} + \boldsymbol{E} \cdot t). \tag{25}$$

Restricting $\bar{f}$ on $\Omega_x \times \mathbb{R}^2$ solves the original problem and can be used to assess the quality of its numerical solution f_r^h.

In order to use $\bar{f}$ as the inflow function in our dynamical low-rank approach, we have to work with an approximation in separable form instead. For that purpose, we use the fact that, by (25) and our choice (24), $\bar{f}$ can be written as

$$\bar{f}(t, \boldsymbol{x}, \boldsymbol{v}) = g_1(t, x_1, v_1) \cdot g_2(t, x_2, v_2).$$

The factors g_1 and g_2 are then evaluated for a given time on a regular fine grid (x_i, v_i) and singular value decompositions are computed. The product of the truncated SVDs is then used for computing the inflow (23) as well as the error of the numerical solution. In our experiments we used the truncation ranks of 25.

The transport equation is solved numerically on the time domain $[0, 0.5]$. On the coarsest scale (level 0) we use a conforming triangulation of Ω_x with 339 vertices. The velocity domain is chosen as $\Omega_v = [-4, 4]^2$ with periodic boundary conditions and a regular triangulation with 4096 vertices. For the numerical solution the rank adaptive algorithm (Algorithm 2) is used with a time step of $\Delta t = 0.005$, a truncation threshold of $\epsilon = 10^{-3}$, and a stabilization parameter $\delta = 10^{-2}$. Higher levels $\ell = 1, 2$ are obtained by uniformly refining the meshes in Ω_x as well as Ω_v. For these levels the parameters Δt and ϵ are scaled by $2^{-2\ell}$.

Figure 4 shows the evolution of the computed spatial density

$$\rho^h(t, \boldsymbol{x}) = \int_{\Omega_v} f_r^h(t, \boldsymbol{x}, \boldsymbol{v}) \, d\boldsymbol{v} \tag{26}$$

for level $\ell = 1$ at different time steps together with its error computed using the analytical solution $\bar{f}$. At the beginning of the simulation, the density flows into the domain, is transported and finally leaves the domain. The error remains reasonably small during the whole process.

A more detailed investigation of the L_2 error is depicted in the upper part of Fig. 5. It is obtained by comparing the numerical solutions f_r^h to the analytical solution $\bar{f}$ on a once uniformly refined grid. As one can see, in the beginning the error increases for all levels but decays again for $t \geq 0.35$. At about that time a significant fraction of the density has left the domain already (see Fig. 4). The maximal error decays as the level increases.

The lower part of the figure shows the ranks which were used by the rank adaptive algorithm. As more particles enter the domain and the distribution spreads out, the ranks increase. For higher levels of discretization a smaller truncation parameter is used in order to balance the low-rank approximation error with the discretization error, leading to higher ranks.
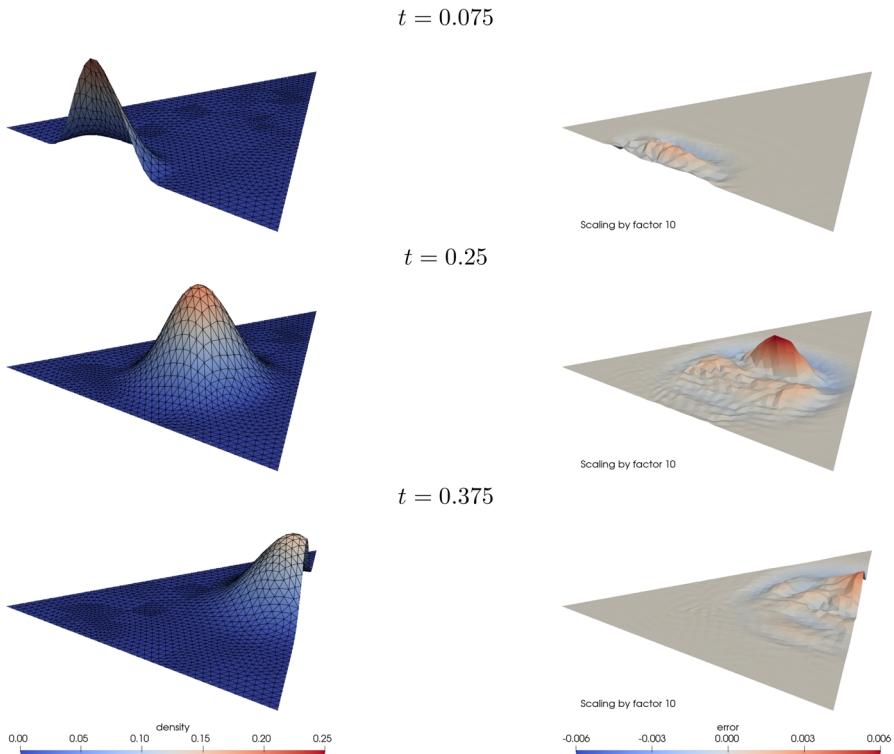
$$t = 0.075$$

$$t = 0.25$$

$$t = 0.375$$

Fig. 4 Spatial density $\rho^h(t, \boldsymbol{x})$ (left), see (26), and error scaled by a factor of 10 (right) of the numerical solution of (1) with constant electrical field for level $\ell = 1$ at times $t = 0.075,\ 0.25,\ 0.375$

5 Conclusion and outlook

In this paper we have studied how to incorporate inflow boundary conditions in the dynamical low-rank approximation (DLRA) for the Vlasov–Poisson equation based on its weak formulation. The single steps in the projector splitting integrator, or the rank-adaptive unconventional integrator, can be interpreted as restrictions of the weak formulation to certain subspaces of the tangent space. The efficient solution of these sub-steps requires the separability of the boundary integrals, which is ensured for piecewise linear boundaries together with a separable inflow function. The resulting equations can be solved using FEM solvers based on Galerkin discretization. We confirmed the feasibility of our approach in numerical experiments.

As a next step, the conservation of physical invariants such as mass and momentum in the numerical schemes could be addressed, perhaps by extending methods from [6, 8, 10, 12, 14]. Enforcing nonnegativity of the density function in the DLRA approach is another open issue.

A potential advantage of the weak formulation for the sub-problems in the projector splitting integrator is that in principle it should allow for a great flexibility regarding the discretization spaces. In particular, they do not need to be fixed in advance and mesh-adaptive methods could be used for solving the sub-steps, provided suitable
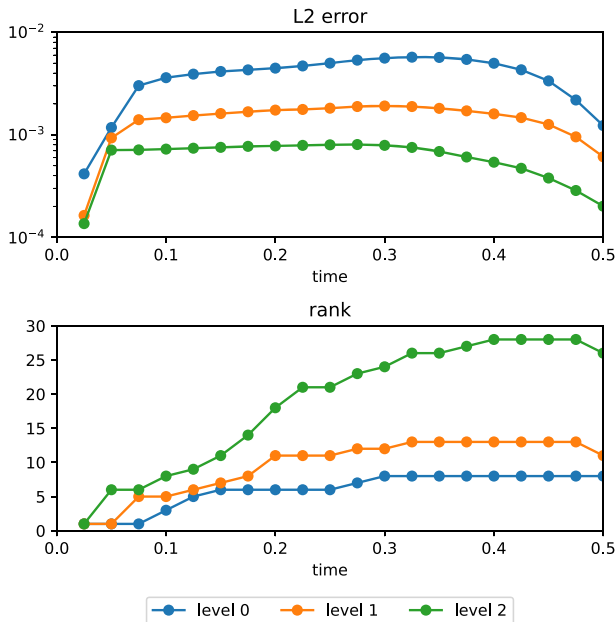
Fig. 5 Numerical solution of (1) for different levels of discretization. The upper graph shows the L_2 error computed with respect to the analytical solution $\tilde{f}$, see (25), on a uniformly refined grid. The lower graph shows the ranks used by the rank adaptive integrator

interpolation and prolongation operations are available, as well as adaptive solvers for Friedrichs' systems. We leave this as possible future work. Other elaborate adaptive integrators have been proposed in [15], which also could be combined with our approach.

Code Availability The code is available at https://github.com/azeiser/dlra-bc.

Declarations

Conflict of interest The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Anderson, R., Andrej, J., Barker, A., et al.: MFEM: A modular finite element methods library. Comput. Math. Appl. **81**, 42–74 (2021). https://doi.org/10.1016/j.camwa.2020.06.009

2. Bachmayr, M., Eisenmann, H., Kieri, E., Uschmajew, A.: Existence of dynamical low-rank approximations to parabolic problems. Math. Comp. **90**(330), 1799–1830 (2021). https://doi.org/10.1090/mcom/3626

3. Ceruti, G., Lubich, C.: An unconventional robust integrator for dynamical low-rank approximation. BIT **62**(1), 23–44 (2022). https://doi.org/10.1007/s10543-021-00873-0

4. Ceruti, G., Kusch, J., Lubich, C.: A rank-adaptive robust integrator for dynamical low-rank approximation. BIT **62**(4), 1149–1174 (2022). https://doi.org/10.1007/s10543-021-00907-7

5. Deriaz, E., Peirani, S.: Six-dimensional adaptive simulation of the Vlasov equations using a hierarchical basis. Multiscale Model. Simul. **16**(2), 583–614 (2018). https://doi.org/10.1137/16M1108649

6. Einkemmer, L., Joseph, I.: A mass, momentum, and energy conservative dynamical low-rank scheme for the Vlasov equation. J. Comput. Phys. **443**, 110495 (2021). https://doi.org/10.1016/j.jcp.2021.110495

7. Einkemmer, L., Lubich, C.: A low-rank projector-splitting integrator for the Vlasov–Poisson equation. SIAM J. Sci. Comput. **40**(5), 1330–1360 (2018). https://doi.org/10.1137/18M116383X

8. Einkemmer, L., Lubich, C.: A quasi-conservative dynamical low-rank algorithm for the Vlasov equation. SIAM J. Sci. Comput. **41**(5), 1061–1081 (2019). https://doi.org/10.1137/18M1218686

9. Einkemmer, L., Ostermann, A., Piazzola, C.: A low-rank projector-splitting integrator for the Vlasov–Maxwell equations with divergence correction. J. Comput. Phys. **403**, 109063 (2020). https://doi.org/10.1016/j.jcp.2019.109063

10. Einkemmer, L., Ostermann, A., Scalone, C.: A robust and conservative dynamical low-rank algorithm. arXiv:2206.09374 (2022) https://doi.org/10.48550/ARXIV.2206.09374

11. Ern, A., Guermond, J.-L.: Finite Elements III—first-order and Time-dependent PDEs. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-57348-5

12. Guo, W., Qiu, J.-M.: A conservative low rank tensor method for the Vlasov dynamics. arXiv:2201.10397 (2022a) https://doi.org/10.48550/ARXIV.2201.10397

13. Guo, W., Qiu, J.-M.: A low rank tensor representation of linear transport and nonlinear Vlasov solutions and their associated flow maps. J. Comput. Phys. **458**, 111089 (2022). https://doi.org/10.1016/j.jcp.2022.111089

14. Guo, W., Ema, J.F., Qiu, J.-M.: A local macroscopic conservative (LoMaC) low rank tensor method with the discontinuous Galerkin method for the Vlasov dynamics. arXiv:2210.07208 (2022) https://doi.org/10.48550/arXiv.2210.07208

15. Hauck, C., Schnake, S.: A predictor-corrector strategy for adaptivity in dynamical low-rank approximations. arXiv:2209.00550 (2022) https://doi.org/10.48550/arXiv.2209.00550

16. Hu, J., Wang, Y.: An adaptive dynamical low rank method for the nonlinear Boltzmann equation. J. Sci. Comput. **92**(2), 75–24 (2022). https://doi.org/10.1007/s10915-022-01934-4

17. Huang, J., Guo, W., Cheng, Y.: Adaptive sparse grid discontinuous Galerkin method: review and software implementation. Commun. Appl. Math. Comput. (2023). https://doi.org/10.1007/s42967-023-00268-8

18. Koch, O., Lubich, C.: Dynamical low-rank approximation. SIAM J. Matrix Anal. Appl. **29**(2), 434–454 (2007). https://doi.org/10.1137/050639703

19. Koch, O., Lubich, C.: Dynamical tensor approximation. SIAM J. Matrix Anal. Appl. **31**(5), 2360–2375 (2010). https://doi.org/10.1137/09076578X

20. Kusch, J., Einkemmer, L., Ceruti, G.: On the stability of robust dynamical low-rank approximations for hyperbolic problems. SIAM J. Sci. Comput. **45**(1), 1–24 (2023). https://doi.org/10.1137/21M1446289

21. Lubich, C.: From Quantum to Classical Molecular Dynamics: Reduced Models and Numerical Analysis. European Mathematical Society (EMS), Zürich (2008). https://doi.org/10.4171/067

22. Lubich, C., Oseledets, I.V.: A projector-splitting integrator for dynamical low-rank approximation. BIT **54**(1), 171–188 (2014). https://doi.org/10.1007/s10543-013-0454-0

23. Uschmajew, A., Vandereycken, B.: Geometric methods on low-rank matrix and tensor manifolds. In: Grohs, P., Holler, M., Weinmann, A. (eds.) Handbook of Variational Methods for Nonlinear Geometric Data, pp. 261–313. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-31351-7_9

24. Verboncoeur, J.P.: Particle simulation of plasmas: review and advances. Plasma Phys. Controlled Fusion **47**, 231–260 (2005). https://doi.org/10.1088/0741-3335/47/5A/017