A CONSERVATIVE LOW RANK TENSOR METHOD FOR THE VLASOV DYNAMICS

WEI GUO * AND JING-MEI QIU †

Abstract. In this paper, we propose a conservative low rank tensor method to approximate nonlinear Vlasov solutions. The low rank approach is based on our earlier work [17]. It takes advantage of the fact that the differential operators in the Vlasov equation are tensor friendly, based on which we propose to dynamically and adaptively build up low rank solution basis by adding new basis functions from discretization of the differential equation, and removing basis from a singular value decomposition (SVD)-type truncation procedure. For the discretization, we adopt a high order finite difference spatial discretization together with a second order strong stability preserving multi-step time discretization.

While the SVD truncation will remove the redundancy in representing the high dimensional Vlasov solution, it will destroy the conservation properties of the associated full conservative scheme. In this paper, we develop a conservative truncation procedure with conservation of mass, momentum and kinetic energy densities. The conservative truncation is achieved by an orthogonal projection onto a subspace spanned by 1, v and v^2 in the velocity space associated with a weighted inner product. Then the algorithm performs a weighted SVD truncation of the remainder, which involves a scaling, followed by the standard SVD truncation and rescaling back. The algorithm is further developed in high dimensions with hierarchical Tucker tensor decomposition of high dimensional Vlasov solutions, mitigating the curse of dimensionality. An extensive set of nonlinear Vlasov examples are performed to show the effectiveness and conservation property of proposed conservative low rank approach. Comparison is performed against the non-conservative low rank tensor approach on conservation history of mass, momentum and energy.

Key words. Low rank; high order SVD; Conservative truncation; Hierarchical Tuck decomposition of tensors; Mass and momentum conservation; Vlasov Dynamics.

1. Introduction.

The Vlasov-Poisson (VP) system is known as a fundamental model in plasma physics which describes the dynamics of dilute charged particles due to self-induced electrostatic forces from a statistical viewpoint. The numerical challenges of realistic Vlasov simulations include the high-dimensionality of the phase space, features with multiple scales in time and in phase space, preservation of physical invariants, among many others.

The Particle-In-Cell method employs a collection of sampled macro particles to represent the distribution function and is widely used for the plasma simulations [4, 2]. On the other hand, such a method suffers the inherent statistical noise, and hence may fail to accurately capture the physics of interest. For example, if the resolution of the tail of the distribution function is desired, a noise-free grid-based deterministic method is a better choice. Over the past two decades, the development of deterministic Vlasov solvers has attracted lots of research interest, see e.g. [12]. Meanwhile, a conventional deterministic Vlasov simulation in a realistic and high-dimensional setting is prohibitively expensive because of the curse of the dimensionality and the associated huge computational and storage cost. Several dimension reduction techniques have been developed in the literature to alleviate the curse of dimensionality for the Vlasov equation and other high-dimensional partial differential equations (PDEs). One such example is the sparse grid approach [30, 33, 15], which can effectively reduce the computational complexity and is well-suited for the problems with moderately high dimensions. For the Vlasov simulations, we mention the sparse grid semi-Lagrangian (SL) method [24] and the sparse grid discontinuous Galerkin method [16, 31]. Recently, the tensor approach emerges as a promising tool for feasible high-dimensional simulations. Such an approach aims to extract the underlying low rank structure of the data with advanced tensor decomposition, potentially breaking the curse of dimensionality. The popular tensor formats include the canonical polyadic (CP) format [21, 3, 20, 22], Tucker format [32, 5], hierarchical Tucker (HT) format [19, 14], and tensor train (TT) format [28, 26, 27]. There are several pioneering works employing low rank tensor approach for nonlinear simulations, including the low rank SL method in the TT format [23], a low rank method with the CP format based on the underlying Hamiltonian formulation [7], a dynamical low rank method proposed in [9, 11] for which the dynamical low rank approximation of the Vlasov solution is evolved on the low rank manifold using a tangent space projection, and a dynamical tensor approximations for high dimensional linear and nonlinear PDEs based on functional tensor decomposition and dynamical tensor approximation [6].

*Department of Mathematics and Statistics, Texas Tech University, Lubbock, TX, 70409. E-mail: weimath.guo@ttu.edu. Research is supported by NSF grant NSF-DMS-1830838 and NSF-DMS-2111383, Air Force Office of Scientific Research FA9550-18-1-0257.

†Department of Mathematical Sciences, University of Delaware, Newark, DE, 19716. E-mail: jingqiu@udel.edu. Research supported by NSF grant NSF-DMS-1818924 and 2111253, Air Force Office of Scientific Research FA9550-18-1-0257.

With the existing understanding of the low rank solution structure for the Vlasov dynamics, as well as the observation that the differential operator in the Vlasov equation can be represented in the tensorized form, in [17], we developed a low rank tensor approach to dynamically and adaptively build up low rank solution basis, by adding new basis functions from discretization of the PDE and then removing basis from an singular value decomposition (SVD)-type truncation procedure. In particular, we start from a low rank solution in a tensor format and add additional basis by applying the well-established high order finite difference upwind method coupled with the strong-stability-preserving (SSP) multi-step time discretizations [13]; the solutions are being further updated by an SVD-type truncation to remove redundant bases. We further generalize the algorithm to high-dimensional cases with the HT decomposition, which attains a storage complexity that is linearly scaled with the dimension, overcoming the curse of dimensionality. An issue associated with the low rank algorithm in [17] is the loss of mass conservation in the SVD truncation step. In fact, it is well-known that the VP system conserves mass, momentum, and energy *locally* by respecting a set of macroscopic moment equations. The high order finite difference scheme with an SSP multi-step time integrator can be shown to locally preserve the macroscopic mass and momentum, i.e. when taking moments of the kinetic scheme, one will arrive at a consistent and locally conservative discretization of the corresponding macroscopic equations. Here "locally conservative" means that the scheme can be written in a flux difference form, where the fluxes represent the amount of macroscopic quantities transported between neighboring computational cells. In [25], it is pointed out that preserving local conservation laws of mass, momentum and energy is of much more practical significance than global conservation of these invariants and posing more restrictions on the numerical solution. However, the SVD truncation in the low rank algorithm would destroy local and global conservation property of the original numerical scheme.

There exist several techniques developed to correct the conservation errors for low rank methods in various settings. In [23], the low rank solution is rescaled so that the total mass is conserved, and a similar mass correction technique is proposed in [29] for a dynamical low rank method. In [1], moment fitting is applied to the low rank solution so that the corrected moments match those solved from the macroscopic fluid equations. In [10], a dynamical low rank method with Lagrangian multipliers is developed to improve conservation properties for the total mass and momentum as well as local projected moment equations. More recently, along the same line, the truly local conservation of mass, momentum, and energy is attained for the dynamical low rank method [8]. The idea is to fix certain basis functions in the dynamical low rank approximation and employ a modified Petrov–Galerkin formulation which is compatible with the remainder of the approximation.

Inspired by the recent work in [8], in this paper, we develop a novel locally mass, momentum and energy conservative truncation algorithm under the framework of the low rank tensor approach [17]. In particular, we apply an orthogonal projection operator to the low rank solution in a weighted inner product spaces of phase variables, to extract exactly the first few moments (i.e. the mass, momentum and kinetic energy density functions) of the low rank solution after the "add basis" step; a weighted SVD truncation is then applied to the remainder of the projection. Such a truncation is called "conservative truncation" throughout the paper. We further develop the conservative truncation algorithm for the 2D2V VP system with the HT tensor format using a dimension tree that separates the spatial and phase variables. For high order HT tensors, an additional projection step is needed, after the high order SVD (HOSVD) truncation of the remaining term, to ensure conservation of macroscopic variables from the first few moments. The low rank tensor algorithm with the conservative projection is theoretically proved and numerically verified to be locally mass and momentum conservative. The proposed methodology can be extended to the full 3D3V VP model with the HT format, and we leave it for the future investigation.

This paper is organized as follows. In Section 2, we introduce the kinetic Vlasov model and the corresponding macroscopic conservation laws. In Section 3, we first review the low rank tensor approach for the 1D1V Vlasov equation in Section 3.1; then we introduce the orthogonal projection at the continuous level to extract exactly the first few moments of Vlasov solution in Section 3.2; we further extend such an orthogonal projection to the discrete level and develop the main conservative truncation algorithm in Section 3.3. In Section 4, we develop the conservative truncation algorithm for the 2D2V Vlasov model, and show that the low rank algorithm locally conserves the mass and momentum at the macroscopic level. In Section 5, we present an extensive set of 1D1V and 2D2V numerical results to demonstrate the effectiveness and the conservation properties of the low rank tensor algorithm. We conclude the main contributions of the paper and comment on future directions in Section 6.

2. The kinetic Vlasov model and the corresponding macroscopic systems. We consider the VP system

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla_{\mathbf{x}} f + \mathbf{E}(\mathbf{x}, t) \cdot \nabla_{\mathbf{v}} f = 0, \tag{2.1}$$

$$\mathbf{E}(\mathbf{x}, t) = -\nabla_{\mathbf{x}} \phi(\mathbf{x}, t), \quad -\triangle_{\mathbf{x}} \phi(\mathbf{x}, t) = \rho(\mathbf{x}, t) - \rho_0, \tag{2.2}$$

which describes the probability distribution function $f(\mathbf{x}, \mathbf{v}, t)$ of electrons in collisionless plasma. Here $\mathbf{E}$ is the electric field, and ϕ is the self-consistent electrostatic potential. f couples to the long range fields via the charge density, $\rho(\mathbf{x}, t) = \int_{\Omega_{\mathbf{v}}} f(\mathbf{x}, \mathbf{v}, t) d\mathbf{v}$, where we take the limit of uniformly distributed infinitely massive ions in the background. The VP system describes the movement of electrons due to self-induced electric field $\mathbf{E}$ determined by the Poisson equation.

The Vlasov dynamics are well-known to conserve several physical invariants. In particular, let

$$\text{mass density:} \quad \rho(\mathbf{x}, t) = \int_{\Omega_{\mathbf{v}}} f(\mathbf{x}, \mathbf{v}, t) d\mathbf{v}, \tag{2.3}$$

$$\text{current density:} \quad \mathbf{J}(\mathbf{x}, t) = \int_{\Omega_{\mathbf{v}}} f(\mathbf{x}, \mathbf{v}, t) \mathbf{v} d\mathbf{v}, \tag{2.4}$$

$$\text{kinetic energy density:} \quad \kappa(\mathbf{x}, t) = \frac{1}{2} \int_{\Omega_{\mathbf{v}}} |\mathbf{v}|^2 f(\mathbf{x}, \mathbf{v}, t) d\mathbf{v}, \tag{2.5}$$

$$\text{energy density:} \quad e(\mathbf{x}, t) = \kappa(\mathbf{x}, t) + \frac{1}{2} \mathbf{E}(\mathbf{x})^2. \tag{2.6}$$

Then, by taking a few first moments of the Vlasov equation, the following conservation laws of mass, momentum and energy can be derived

$$\partial_t \rho + \nabla_{\mathbf{x}} \cdot \mathbf{J} = 0 \tag{2.7}$$

$$\partial_t \mathbf{J} + \nabla_{\mathbf{x}} \cdot \boldsymbol{\sigma} = \rho \mathbf{E} \tag{2.8}$$

$$\partial_t e + \nabla_{\mathbf{x}} \cdot \mathbf{Q} = 0, \tag{2.9}$$

where $\boldsymbol{\sigma}(\mathbf{x}, t) = \int_{\Omega_{\mathbf{v}}} (\mathbf{v} \otimes \mathbf{v}) f(\mathbf{x}, \mathbf{v}, t) d\mathbf{v}$ and $\mathbf{Q}(\mathbf{x}, t) = \frac{1}{2} \int_{\Omega_{\mathbf{v}}} \mathbf{v} |\mathbf{v}|^2 f(\mathbf{x}, \mathbf{v}, t) d\mathbf{v}$. It is well-known that the local conservation property of numerical schemes is critical to capture correct entropy solutions of hyperbolic systems such as macroscopic equations (2.7)-(2.9).

3. A low rank tensor approach for the Vlasov dynamics with local conservation. For simplicity of illustrating the basis idea, we only discuss a 1D1V example in this section. We first introduce the low rank method [17] together with a set of notations. Then a conservative truncation is developed which is the key component for the proposed algorithm. We also show that the discretization before truncation is locally conservative if a summation-by-parts condition is satisfied. Together with the conservative truncation, we obtain a low rank VP solver with local conservation.

3.1. Review of a low rank tensor approach for Vlasov dynamics [17]. The low rank tensor approach [17] is designed based on the assumption that our solution at time t has a low-rank representation in the form of

$$f(x, v, t) = \sum_{l=1}^{r} \left(C_l(t) \;\; U_l^{(1)}(x, t) U_l^{(2)}(v, t) \right), \tag{3.1}$$

where $\left\{ U_l^{(1)}(x, t) \right\}_{l=1}^{r}$ and $\left\{ U_l^{(2)}(v, t) \right\}_{l=1}^{r}$ are a set of time-dependent low rank unit length orthogonal basis in x and v directions, respectively. C_l is the coefficient for the basis $U_l^{(1)}(x, t) U_l^{(2)}(v, t)$, and r is the representation rank. (3.1) can be viewed as a Schmidt decomposition of functions in (x, v) by truncating small singular values up to rank r.

We assume a finite difference discretization of f on a truncated 1D1V domain of $[x_{\min}, x_{\max}] \times [-v_{\max}, v_{\max}]$ with uniform $N_x \times N_v$ grid points

$$x_{\text{grid}}: \quad x_{\min} = x_1 < \cdots < x_i < \cdots < x_{N_x} = x_{\max}, \tag{3.2}$$

$$v_{\text{grid}}: \quad -v_{\max} = v_1 < \cdots < v_j < \cdots < v_{N_v} = v_{\max}, \tag{3.3}$$

and denote h_x and h_v as the mesh sizes in x- and v-directions, respectively.

The numerical solution $\mathbf{f} \in \mathbb{R}^{N_x \times N_v}$, as an approximation to point values of the solution on the grids (3.2)-(3.3), has the corresponding low rank counterpart to (3.1) as

$$\mathbf{f} = \sum_{l=1}^{r} \left(C_l \, \mathbf{U}_l^{(1)} \otimes \mathbf{U}_l^{(2)} \right), \quad \text{(or element-wise:} \quad f_{ij} = \sum_{l=1}^{r} C_l \, U_{l,i}^{(1)} U_{l,j}^{(2)}), \tag{3.4}$$

where $\mathbf{U}_l^{(1)} \in \mathbb{R}^{N_x}$ and $\mathbf{U}_l^{(2)} \in \mathbb{R}^{N_v}$ can be viewed as approximations to corresponding basis functions in (3.1). (3.4) can also be viewed as an SVD of the matrix $\mathbf{f} \in \mathbb{R}^{N_x \times N_v}$. The associated storage cost is $\mathcal{O}(rN)$, where we assume $N = N_x = N_v$.

Our low rank tensor approach adaptively updates low-rank basis and associated coefficients by two steps: an adding basis step by conservative hyperbolic solvers and a removing basis step via an SVD-type truncation. We consider a simple first order forward Euler discretization of 1D1V Vlasov equation (2.1) to illustrate the main idea. We assume the solution in the form of (3.4) with superscript n for the solution at t^n.

1. *Add basis and obtain an intermediate solution* $\mathbf{f}^{n+1,*}$. A forward Euler discretization of (2.1) gives

$$f^{n+1,*} = f^n - \Delta t (v \partial_x (f^n) + E^n \partial_v (f^n)). \tag{3.5}$$

Here the electric field E^n is solved by a Poisson solver. Thanks to the tensor friendly form of the Vlasov equation, at the fully discrete level, $\mathbf{f}^{n+1,*}$ can be represented in the following low-rank format:

$$\mathbf{f}^{n+1,*} = \sum_{l=1}^{r^n} C_l^n \left[\left(\mathbf{U}_l^{(1),n} \otimes \mathbf{U}_l^{(2),n} \right) - \Delta t \left(D_x \mathbf{U}_l^{(1),n} \otimes \mathbf{v} \star \mathbf{U}_l^{(2),n} + \mathbf{E}^n \star \mathbf{U}_l^{(1),n} \otimes D_v \mathbf{U}_l^{(2),n} \right) \right]. \tag{3.6}$$

Here, with a slight abuse of notation, $\mathbf{v} \in \mathbb{R}^{N_v}$ denotes the coordinates of v_{grid} (3.3). D_x and D_v represent high order *locally conservative upwind discretization* of spatial differentiation terms, and $\star$ denotes an element-wise multiplication operation. For example, the discretization of $D_x \mathbf{U}_l^{(1),n} \otimes \mathbf{v} \star \mathbf{U}_l^{(2),n}$ follows

$$D_x^+ \mathbf{U}_l^{(1),n} \otimes v^+ \star \mathbf{U}_l^{(2),n} + D_x^- \mathbf{U}_l^{(1),n} \otimes \mathbf{v}^- \star \mathbf{U}_l^{(2),n}, \tag{3.7}$$

where $\mathbf{v}^+ = \max(\mathbf{v}, 0)$ and $\mathbf{v}^- = \min(\mathbf{v}, 0)$ and D_x^+ and D_x^- are the fifth order finite difference discretization of positive and negative velocities. In particular, for each basis vector $\mathbf{U}_l$ (omitted superscript for brevity), the numerical flux vectors $\hat{F}_l^{\pm}$ are computed via

$$\begin{aligned}
\hat{F}^-_{l,i+\frac{1}{2}} &= -\frac{1}{20} U_{l,i-1} + \frac{9}{20} U_{l,i} + \frac{47}{60} U_{l,i+1} - \frac{13}{60} U_{l,i+2} + \frac{1}{30} U_{l,i+3}, \\
\hat{F}^+_{l,i+\frac{1}{2}} &= \frac{1}{30} U_{l,i-2} - \frac{13}{60} U_{l,i-1} + \frac{47}{60} U_{l,i} + \frac{9}{20} U_{l,i+1} - \frac{1}{20} U_{l,i+2}
\end{aligned}$$

for $i = 1, \ldots, N_x$. Then, $D_x^- \mathbf{U}_l$, $D_x^+ \mathbf{U}_l \in \mathbb{R}^{N_x}$ are defined as

$$(D_x^- \mathbf{U}_l)_i = \frac{1}{h_x} (\hat{F}^-_{l,i+\frac{1}{2}} - \hat{F}^-_{l,i-\frac{1}{2}}), \quad (D_x^+ \mathbf{U}_l)_i = \frac{1}{h_x} (\hat{F}^+_{l,i+\frac{1}{2}} - \hat{F}^+_{l,i-\frac{1}{2}}),$$

which are in a finite difference form. Similarly, the discretization of $E^n \star \mathbf{U}_l^{(1),n} \otimes D_v \mathbf{U}_l^{(2),n}$ follows

$$(\mathbf{E}^{n,+} \star \mathbf{U}_j^{(1),n}) \otimes D_v^+ \mathbf{U}_j^{(2),n} + (\mathbf{E}^{n,-} \star \mathbf{U}_j^{(1),n}) \otimes D_v^- \mathbf{U}_j^{(2),n}, \tag{3.8}$$

where D_v^+ and D_v^- are the same fifth order upwind finite difference discretization of positive and negative velocities as D_x^+ and D_x^-, with $\mathbf{E}^+ = \max(\mathbf{E}, 0)$ and $\mathbf{E}^- = \min(\mathbf{E}, 0)$.

2. *Remove basis of* $\mathbf{f}^{n+1,*}$ *to update solution* $\mathbf{f}^{n+1}$. Since the number of basis has increased in a single step update, we perform an SVD-type truncation to remove redundant bases with a prescribed threshold ε. The truncation step has no guarantee of any mass, momentum or energy conservation property. The removing basis step costs $\mathcal{O}(r^2N + r^3)$, where r is the SVD rank of the numerical solution.

In this two-step process, both the basis and coefficients are updated. Extensions to schemes with high order spatial and temporal discretizations and to high dimensional problems, are developed in [17]. The low rank approach [17] is built upon the classical high order methods for conservation laws and kinetic equations, yet it optimizes the computational efficiency by dynamically building low rank global basis and updating the corresponding coefficients via a SVD truncation procedure. While the SVD truncation significantly reduces the computational storage and CPU time, it also destroys the conservation property in the truncation step.

3.2. An orthogonal projection with mass, momentum and energy conservation. To preserve the local mass, momentum and energy in the truncation step, following a similar idea in [8], we define a weighted inner product space and the associated norm for functions in v as follows,

$$L^2_w(\mathbb{R}) = \{f : \mathbb{R} \to \mathbb{R} : \|f\|_w < \infty\}.$$

where

$$\langle f, g\rangle_w = \int f(v)g(v)w(v)dv, \quad \|f\|_w = \sqrt{\langle f, f\rangle_w}. \tag{3.9}$$

The weight function $w(v)$ is chosen to have exponential decay so that $N \doteq \text{span}(1, v, v^2) \subset L^2_w(\mathbb{R})$. For example, we can take $w(v) = \exp(-v^2/2)$. Now we are ready to define an orthogonal decomposition of f with respect to the subspace N.

In the context of Vlasov dynamics, we consider $f(x, v)$ with a v-direction projection. We first scale function

$$\tilde{f} = \frac{1}{w(v)} f. \tag{3.10}$$

Then we find $P_N(\tilde{f}) \in N$ s.t.

$$\int P_N(\tilde{f}(x, v))g(v)w(v)dv = \int \tilde{f}(x, v)g(v)w(v)dv = \int f(x, v)g(v)dv, \quad \forall g \in N. \tag{3.11}$$

By taking $g = 1, v, v^2$, $P_N(\tilde{f}(x, v))w(v)$ preserves the mass density $\rho(x)$, current density $J(x)$ and kinetic energy density $\kappa(x)$ of f defined in (2.3)-(2.5). In fact, we have the *conservative* decomposition of f as in the following Proposition.

Proposition 3.1. *Let* $w(v) = \exp(-v^2/2)$ *and the weighted inner product defined as in* (3.9). $f \in L^2_w(\mathbb{R})$ *can be decomposed as*

$$f = w(v)(P_N(\tilde{f}) + (I - P_N)(\tilde{f})) \doteq w(v)(\tilde{f}_1 + \tilde{f}_2) \doteq f_1 + f_2, \tag{3.12}$$

where f_1 *admits an explicit representation*

$$f_1 = w(v)\tilde{f}_1 = w(v)(c_1 + c_2 v + c_3(v^2 - c)), \tag{3.13}$$

with $c = \frac{\langle 1, v^2\rangle_w}{\langle 1, 1\rangle_w}$, $c_1 = \frac{\rho(x)}{\|1\|^2_w}$, $c_2 = \frac{J(x)}{\|v\|^2_w}$, *and* $c_3 = \frac{2\kappa(x) - \rho(x)c}{\|v^2 - c\|^2_w}$, *and* $\rho(x)$, $J(x)$ *and* $\kappa(x)$ *are as defined in* (2.3)-(2.5). f_1 *preserves the mass, momentum and kinetic energy density of* f, *while* f_2 *has zero mass, current and kinetic energy density.*

Proof. We first compute c, so that $\{1, v, v^2 - c\}$ is a set of orthogonal basis. By taking $g = 1, v, v^2 - c$ in (3.11), we can determine the constants c_1, c_2, and c_3. By construction of the orthogonal projection, f_1 preserves the mass, momentum and kinetic energy density of f, while f_2 has zero of them.

3.3. Low rank truncation with mass, momentum and energy conservation. Similar to the weighted inner product (3.9), the projection operator (3.11) and the conservative decomposition of f in (3.12), we define their corresponding discrete analogue. Consider $\mathbf{f}, \mathbf{g} \in \mathbb{R}^{N_v}$, where $\mathbf{f}$ and $\mathbf{g}$ can be viewed as function values of f and g at v_{grid} (3.3), respectively. We define the standard l^2 inner product as

$$\langle \mathbf{f}, \mathbf{g} \rangle = h_v \sum_j f_j g_j, \tag{3.14}$$

where h_v serves as the quadrature weights for the uniform v_{grid}. We also define weighted inner product and the associated norm as

$$\langle \mathbf{f}, \mathbf{g} \rangle_{\mathbf{w}} = \sum_j f_j g_j w_j, \qquad \|\mathbf{f}\|_{\mathbf{w}} = \sqrt{\langle \mathbf{f}, \mathbf{f} \rangle_{\mathbf{w}}} \tag{3.15}$$

$\mathbf{w} \in \mathbb{R}^{N_v}$ with $w_j = w(v_j)h_v$ is the quadrature weights for v-integration with weight function $w(v)$. Correspondingly, we let

$$l^2_{\mathbf{w}} = \{\mathbf{f} \in \mathbb{R}^{N_v} : \|\mathbf{f}\|_{\mathbf{w}} < \infty\}.$$

We denote $\mathbf{1}_v \in \mathbb{R}^{N_v}$ the vector of all ones. $\mathbf{v}^2 \in \mathbb{R}^{N_v}$ is the element-wise square of $\mathbf{v}$. Consider the subspace $\mathcal{N} \doteq \text{span}\{\mathbf{1}_v, \mathbf{v}, \mathbf{v}^2\} \subset l^2_w$, a conservative low rank truncation of numerical solution $\mathbf{f} \in \mathbb{R}^{N_x \times N_v}$ written in the low rank form of (3.4) can be obtained from steps below.

1. **Compute macroscopic quantities of f.** We compute the discrete macroscopic charge, current and kinetic energy density $\boldsymbol{\rho}$, $\mathbf{J}$ and $\boldsymbol{\kappa} \in \mathbb{R}^{N_x}$ by quadrature

$$\begin{pmatrix} \boldsymbol{\rho} \\ \mathbf{J} \\ \boldsymbol{\kappa} \end{pmatrix} = \sum_{l=1}^{r} C_l \left\langle \mathbf{U}_l^{(2)}, \begin{pmatrix} \mathbf{1}_v \\ \mathbf{v} \\ \frac{1}{2}\mathbf{v}^2 \end{pmatrix} \right\rangle \mathbf{U}_l^{(1)}. \tag{3.16}$$

   They are the discrete analog of (2.3), (2.4) and (2.6), i.e.

$$\boldsymbol{\rho}(i) = \langle \mathbf{f}(i,:), \mathbf{1}_v \rangle, \quad \mathbf{J}(i) = \langle \mathbf{f}(i,:), \mathbf{v} \rangle, \quad \boldsymbol{\kappa}(i) = \langle \boldsymbol{f}(i,:), \frac{1}{2}\mathbf{v}^2 \rangle, \qquad i = 1, \cdots, N_x.$$

2. **Scale.** Similar to (3.10), we scale $\mathbf{f}$ as

$$\tilde{\mathbf{f}} = \frac{1}{\mathbf{w}} \star \mathbf{f} = \sum_{l=1}^{r} \left(C_l \;\; \mathbf{U}_l^{(1)} \otimes \left(\frac{1}{\mathbf{w}} \star \mathbf{U}_l^{(2)} \right) \right). \tag{3.17}$$

3. **Project.** We perform an orthogonal projection of $\tilde{\mathbf{f}}$ with respect to the inner product (3.15) onto subspace $\mathcal{N}$, i.e.

$$\langle P_{\mathcal{N}}(\tilde{\mathbf{f}}), \mathbf{g} \rangle_{\mathbf{w}} = \langle \tilde{\mathbf{f}}, \mathbf{g} \rangle_{\mathbf{w}}, \quad \forall \mathbf{g} \in \mathcal{N}. \tag{3.18}$$

   Similar to Proposition 3.1, $\mathbf{w} \star P_{\mathcal{N}}(\tilde{\mathbf{f}})$ preserves the mass, momentum and kinetic energy densities in the discrete sense. We show below in Proposition 3.2 a conservative decomposition of $\mathbf{f}$.

   Proposition 3.2. $\mathbf{f}$ *can be decomposed as*

$$\mathbf{f} = \mathbf{w} \star (P_{\mathcal{N}}(\tilde{\mathbf{f}}) + (I - P_{\mathcal{N}})(\tilde{\mathbf{f}})) \doteq \mathbf{w} \star (\tilde{\mathbf{f}}_1 + \tilde{\mathbf{f}}_2) \doteq \mathbf{f}_1 + \mathbf{f}_2 \tag{3.19}$$

   where $\mathbf{f}_1$ *can be represented as a rank three tensor*

$$\mathbf{f}_1 = \mathbf{w} \star \tilde{\mathbf{f}}_1 = \sum_{k=1}^{3} \mathcal{M}_k \otimes (\mathbf{w} \star \mathbf{V}_k), \tag{3.20}$$

   where $\mathbf{V}_1 = \mathbf{1}_v$, $\mathbf{V}_2 = \mathbf{v}$, *and* $\mathbf{V}_3 = \mathbf{v}^2 - c\mathbf{1}_v$, *and* $c = \frac{\langle \mathbf{1}_v, \mathbf{v}^2 \rangle_{\mathbf{w}}}{\|\mathbf{1}_v\|^2_{\mathbf{w}}}$,

$$\mathcal{M}_1 = \frac{\boldsymbol{\rho}}{\|\mathbf{1}_v\|^2_{\mathbf{w}}}, \quad \mathcal{M}_2 = \frac{\mathbf{J}}{\|\mathbf{v}\|^2_{\mathbf{w}}}, \quad \mathcal{M}_3 = \frac{2\boldsymbol{\kappa} - c\boldsymbol{\rho}}{\|\mathbf{v}^2 - c\mathbf{1}_v\|^2_{\mathbf{w}}}. \tag{3.21}$$

   $\boldsymbol{\rho}$, $\mathbf{J}$ *and* $\boldsymbol{\kappa}$ *are the discrete mass, momentum and kinetic energy density of* $\mathbf{f}$ *as in* (3.16). $\mathbf{f}_1$ *preserves the discrete mass, momentum and kinetic energy density of* $\mathbf{f}$, *while* $\mathbf{f}_2$ *has zero of them, i.e.* $\langle \mathbf{f}_2(i,:), \mathbf{1}_v \rangle = \langle \mathbf{f}_2(i,:), \mathbf{v} \rangle = \langle \mathbf{f}_2(i,:), \mathbf{v}^2 \rangle = 0$, $i = 1, \cdots, N_x$.

Proof. The proof follows in the same manner as Proposition 3.1, but considering the discrete weighted inner product and the associated norm (3.15).

4. **Truncate in $l^2_{\mathbf{w}}$.** To ensure the conservation in the truncation process, we propose to perform an SVD truncation of $\tilde{\mathbf{f}}_2$ in the decomposition (3.19) with respect to the weighted inner product (3.15). Because of the weight function, we scale $\tilde{\mathbf{f}}_2$ by $\sqrt{\mathbf{w}} \star \tilde{\mathbf{f}}_2$, where $\sqrt{\cdot}$ is in the element-wise sense and $\star$ is for the element-wise multiplication. Then, a standard SVD truncation procedure is applied, denoted as $\mathcal{T}_\varepsilon(\sqrt{\mathbf{w}} \star \tilde{\mathbf{f}}_2)$. Here and below, ε is truncation threshold. Lastly, we rescale $\mathcal{T}_\varepsilon(\sqrt{\mathbf{w}} \star \tilde{\mathbf{f}}_2)$ back and obtain the truncated $\tilde{\mathbf{f}}_2$. Such a weighted truncation writes

$$\mathcal{T}^{\mathbf{w}}_\varepsilon(\tilde{\mathbf{f}}_2) = \frac{1}{\sqrt{\mathbf{w}}} \star \mathcal{T}_\varepsilon(\sqrt{\mathbf{w}} \star \tilde{\mathbf{f}}_2). \tag{3.22}$$

In summary, $\mathbf{f}_2$ is truncated to

$$\mathbf{w} \star \mathcal{T}^{\mathbf{w}}_\varepsilon(\tilde{\mathbf{f}}_2) = \sqrt{\mathbf{w}} \star \mathcal{T}_\varepsilon(\sqrt{\mathbf{w}} \star \tilde{\mathbf{f}}_2) = \sqrt{\mathbf{w}} \star \mathcal{T}_\varepsilon(\frac{1}{\sqrt{\mathbf{w}}} \star \mathbf{f}_2).$$

5. **Update.** We obtain the low rank truncation of $\mathbf{f}$ with local mass, momentum and energy conservation, denoted as

$$T_c(\mathbf{f}) = \mathbf{f}_1 + \mathbf{w} \star \mathcal{T}^{\mathbf{w}}_\varepsilon(\tilde{\mathbf{f}}_2). \tag{3.23}$$

We call the proposed truncation (3.23) the conservative truncation. The following proposition guarantees that the local conservation of mass, momentum, and kinetic energy density of $\mathbf{f}$ is preserved in the proposed truncation procedure of $T_c(\mathbf{f})$.

Proposition 3.3. *$T_c(\mathbf{f})$ has the same discrete charge, current and kinetic energy density as $\mathbf{f}$.*

Proof. With Proposition 3.2, it is sufficient to show that $\mathbf{w} \star \mathcal{T}^{\mathbf{w}}_\varepsilon(\tilde{\mathbf{f}}_2)$ has zero charge, current and kinetic energy density. Denote by $\tilde{V} \subset \mathbb{R}^{N_v}$ the subspace spanned by the basis in representing $\tilde{\mathbf{f}}_2$ in the v-direction. By construction, $\tilde{V}$ is orthogonal to $\mathcal{N}$ with respect to the weighted inner product (3.15). With the weighted SVD truncation, $\tilde{\mathbf{f}}_2$ is projected onto the space spanned by the truncated singular vectors, which is a subspace of $\tilde{V}$. Hence, $\mathcal{T}^{\mathbf{w}}_\varepsilon(\tilde{\mathbf{f}}_2)$ must be orthogonal to $\mathcal{N}$, i.e.

$$\langle \mathcal{T}^{\mathbf{w}}_\varepsilon(\tilde{\mathbf{f}}_2), g\rangle_{\mathbf{w}} = 0, \quad g = 1,\, \mathbf{v},\, \mathbf{v}^2.$$

Then, $\mathbf{w} \star \mathcal{T}^{\mathbf{w}}_\varepsilon(\tilde{\mathbf{f}}_2)$ has zero discrete charge, current and kinetic energy density.

Next we establish the local conservation of mass and momentum in the low rank tensor approach with the conservative truncation (3.23). Since the full algorithm (without truncation) does not have energy conservation property, the low rank tensor scheme does not preserve energy conservation. If the associated full-rank method is able to locally preserve the energy density, so will the corresponding the low rank method with the proposed conservative truncation.

Proposition 3.4. *(Local mass and momentum conservation of the low rank tensor approach with conservative truncation.) If the discrete differential operators D_x, D_v employed are conservative, i.e., can be written in a flux difference form, and linear, i.e., can preserve linear relations, then the proposed low rank method with an SSP multi-step time integrator preserves the mass and momentum locally; that is the schemes for $\boldsymbol{\rho}^n$ and $\mathbf{J}^n$, from integrating the scheme on $\mathbf{f}^n$, are consistent and conservative discretization of the macroscopic moment equations* (2.7)-(2.8).

Proof. Without loss of generality, we prove the proposition for the low rank method with the forward Euler time integrator for simplicity. By taking discrete moments of (3.6) and from (3.16), we derive the discrete evolution equations for the charge density $\boldsymbol{\rho}$ and current density $\mathbf{J}$ in $\mathbb{R}^{N_x}$,

$$\boldsymbol{\rho}^{n+1,*} = \boldsymbol{\rho}^n - \Delta t D_x(\mathbf{J}^n), \tag{3.24}$$

$$\mathbf{J}^{n+1,*} = \mathbf{J}^n - \Delta t(D_x(\boldsymbol{\sigma}^n) - \boldsymbol{\rho}^n \star \mathbf{E}^n), \tag{3.25}$$

where we have used the fact that D_x and D_v are linear, and the summation by parts

$$\langle D_v \mathbf{U}^{(2)}_l, \mathbf{v}\rangle = -\langle \mathbf{U}^{(2)}_l, \mathbf{1}_v\rangle \tag{3.26}$$

holds by assuming the basis $\mathbf{U}_l^{(2)}$, $l = 1, \ldots, r^n$, vanishes at the boundary. Combining (3.24)-(3.25) with the property of the conservative truncation from Proposition 3.3, we have

$$\boldsymbol{\rho}^{n+1} = \boldsymbol{\rho}^{n+1,*} = \boldsymbol{\rho}^n - \Delta t D_x(\mathbf{J}^n), \tag{3.27}$$

$$\mathbf{J}^{n+1} = \mathbf{J}^{n+1,*} = \mathbf{J}^n - \Delta t (D_x(\boldsymbol{\sigma}^n) - \boldsymbol{\rho}^n \star \mathbf{E}^n), \tag{3.28}$$

which are consistent and conservative discretization of the macroscopic moment equations (2.7)-(2.8).

Remark 3.5. *If a WENO finite difference discretization is employed for D_v, then the local conservation of momentum cannot be guaranteed. This is because the WENO discretization does not satisfy the summation by parts property* (3.26)*. A WENO finite difference discretization can be employed for D_x for local conservation.*

Remark 3.6. *If only the mass conservation is desired, then we can modify the subspace $\mathcal{N} = span\{\mathbf{1}_v\}$ for projection; similarly, if only the mass and momentum conservation is desired, we let $\mathcal{N} = span\{\mathbf{1}_v, \mathbf{v}\}$. We denote the corresponding projections as P_1 and P_2, respectively, and denote the projection to $\mathcal{N} = span\{\mathbf{1}_v, \mathbf{v}, \mathbf{v}^2\}$ as P_3. Numerical performances of different projection operators will be assessed in the Section 5 for numerical experiments.*

Remark 3.7. *The choice of weight function $w(v)$ can affect the performance. In particular, if the weight function does not have sufficient decay, then the solution may not be close enough to zero at the boundary of the velocity domain, and large error could be incurred. On the other hand, when rescaling f_2 by dividing w, small weights at the boundary can introduce numerical instability. We will investigate the issue numerically in Section 5.*

Finally, the proposed conservative truncation algorithm is summarized in the Algorithm 1.

Algorithm 1: The conservative truncation procedure for the 1D1V VP system.

- Input: the pre-compressed low rank solution at time t^{n+1}:

$$\mathbf{f}^{n+1,*} = \sum_{l=1}^{r^*} C_l^* \ \mathbf{U}_l^{(1),*} \otimes \mathbf{U}_l^{(2),*}.$$

- Output: the compressed low rank solution f^{n+1} with the same density, current density, and kinetic energy functions as $f^{n+1,*}$.

1. Compute $\boldsymbol{\rho}^{n+1}$, $\mathbf{J}^{n+1}$, $\boldsymbol{\kappa}^{n+1}$ of $\mathbf{f}^{n+1,*}$ from (3.16).
2. Compute $\mathcal{M}^{n+1}$ by (3.21) from $\boldsymbol{\rho}^{n+1}$, $\mathbf{J}^{n+1}$, $\boldsymbol{\kappa}^{n+1}$.
3. Compute $\mathbf{f}_1$ from (3.20).
4. Perform the truncation of $\mathbf{f}_2$ by (3.22).
5. Update the compressed low rank solution by (3.23).

4. 2D2V Vlasov-Poisson system by the HT format. We extend the proposed conservative algorithm to the 2D2V case by the HT format. Below, we briefly review the fundamentals of the HT format for efficiently representing tensors in d dimensions, and the low rank tensor method with the HT format for solving the 2D2V VP system (2.1).

$$f_t + v_1 f_{x_1} + v_2 f_{x_2} + E_1 f_{v_1} + E_2 f_{v_2} = 0, \tag{4.1}$$

where the electric field (E_1, E_2) is solved from the coupled Poisson's equation. The macroscopic equations can be obtained from taking moments of (4.1) in the form of (2.7)-(2.9). Similar to the 1D1V case, we introduce a conservative truncation for the solution tensor in the HT format and further show that the proposed low rank tensor method is locally conservative.

4.1. HT format for high order tensors. We denote the dimension index $D = \{1, 2, \ldots, d\}$ and define a *dimension tree* $\mathcal{T}$ which is a binary tree containing a subset $\alpha \subset D$ at each node. Furthermore, $\mathcal{T}$ has D as the root node and $\{1\}$, $\{2\}, \ldots,$ $\{d\}$ as the leaf nodes. Each non-leaf node α has two children nodes denoted as α_l and α_r with $\alpha = \alpha_l \bigcup \alpha_r$ and $\alpha_l \bigcap \alpha_r = \emptyset$. For example, the dimension tree $\mathcal{T}$ given in Figure 4.1 can be used to approximate $f(x_1, x_2, v_1, v_2)$ in (4.1) in the HT format. The efficiency of the HT format lies in the nestedness property [19]: for a non-leaf node α with two children nodes α_l, α_r, then

$$\text{range}(\mathcal{M}^{(\alpha)}(\mathbf{a})) \subset \text{range}(\mathcal{M}^{(\alpha_l)}(\mathbf{a}) \otimes \mathcal{M}^{(\alpha_r)}(\mathbf{a})), \tag{4.2}$$

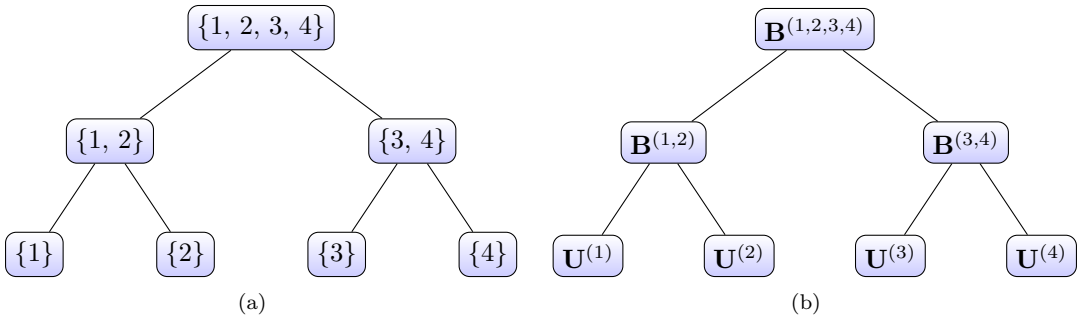


FIG. 4.1. *Dimension tree $\mathcal{T}$ to express fourth-order tensors in the HT format.*

which implies that there exists a third order tensor $\mathbf{B}^{(\alpha)} \in \mathbb{R}^{r_{\alpha_l} \times r_{\alpha_r} \times r_\alpha}$, known as the transfer tensor, such that

$$\mathbf{U}_{l_\alpha}^{(\alpha)} = \sum_{l_{\alpha_l}=1}^{r_{\alpha_l}} \sum_{l_{\alpha_r}=1}^{r_{\alpha_l}} \mathbf{B}_{l_{\alpha_l}, l_{\alpha_r}, l_\alpha}^{(\alpha)} \mathbf{U}_{l_{\alpha_l}}^{(\alpha_l)} \otimes \mathbf{U}_{l_{\alpha_r}}^{(\alpha_r)}, \quad l_\alpha = 1, \ldots, r_\alpha. \tag{4.3}$$

In other words, the frame vectors at the parent node can be recovered by those at the two children nodes α_l, α_r with the transfer tensor. By recursively making use of (4.3), a tensor in the HT format stores a frame at each leaf node and a third order transfer tensor at each non-leaf node based on a dimension tree. Denote $\mathbf{r} = \{r_\alpha\}_{\alpha \in \mathcal{T}}$ as the hierarchical ranks. The storage of the HT format scales as $\mathcal{O}(dr^3 + drN)$, where $r = \max \mathbf{r}$. If r is reasonably low, then the HT format avoids the curse of dimensionality. In summary, the HT format is fully characterized by the three key components, including a dimension tree, frames at leaf nodes and transfer tensors at non-leaf nodes, see Figure 4.1 for the data layout.

4.2. A conservative low rank tensor method in HT for the 2D2V VP system. We follow the low rank tensor method for updating the 2D2V VP solution in [17], but propose to perform a conservative hierarchical HOSVD truncation with preservation of charge, current and kinetic energy density. We assume at each time step, the solution $\mathbf{f}$ is expressed as the fourth order tensor in the HT format with dimension tree $\mathcal{T}$ together with frames $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \mathbf{U}^{(3)}, \mathbf{U}^{(4)}$ at four leaf nodes, corresponding to directions x_1, x_2, v_1, v_2, respectively, and transfer tensors $\mathbf{B}^{(1,2,3,4)}, \mathbf{B}^{(1,2)}, \mathbf{B}^{(3,4)}$, see Figure 4.1. In particular,

$$\mathbf{f} = \sum_{l_{12}=1}^{r_{12}} \sum_{l_{34}=1}^{r_{34}} \mathbf{B}_{l_{12}, l_{34}}^{(1,2,3,4)} \mathbf{U}_{l_{12}}^{(1,2)} \otimes \mathbf{U}_{l_{34}}^{(3,4)}, \tag{4.4}$$

with

$$\mathbf{U}_{l_{12}}^{(1,2)} = \sum_{l_1=1}^{r_1} \sum_{l_2=1}^{r_2} \mathbf{B}_{l_1, l_2, l_{12}}^{(1,2)} \mathbf{U}_{l_1}^{(1)} \otimes \mathbf{U}_{l_2}^{(2)}, \quad l_{12} = 1, \ldots, r_{12}, \tag{4.5}$$

and

$$\mathbf{U}_{l_{34}}^{(3,4)} = \sum_{l_3=1}^{r_3} \sum_{l_4=1}^{r_4} \mathbf{B}_{l_3, l_4, l_{34}}^{(3,4)} \mathbf{U}_{l_3}^{(3)} \otimes \mathbf{U}_{l_4}^{(4)}, \quad l_{34} = 1, \ldots, r_{34}. \tag{4.6}$$

Further, the electric field E_1 and E_2 are represented in the second order HT format.

Now we are ready to formulate a conservative HT truncation of VP solutions in a low rank tensor format. We denote pre-compressed solution $\mathbf{f}$ in the HT format as in (4.4). The idea of the conservative truncation is similar to the 1D1V case. We first introduce a weighted inner product space for $\mathbf{f}, \mathbf{g} \in \mathbb{R}^{N_{v_1} \times N_{v_2}}$

$$\langle \mathbf{f}, \mathbf{g} \rangle_{\mathbf{w}} = h_{v_1} h_{v_2} \sum_{j_1=1}^{N_{v_1}} \sum_{j_2=1}^{N_{v_2}} f_{j_1, j_2} g_{j_1, j_2} w_{j_1, j_2}, \tag{4.7}$$

where h_{v_1} and h_{v_2} are the mesh size for the v_1 and v_2 grids, respectively. Here

$$\mathbf{w} \doteq \mathbf{w}^{(1)} \otimes \mathbf{w}^{(2)} \in \mathbb{R}^{N_{v_1} \times N_{v_2}}, \quad \mathbf{w}^{(1)} \in \mathbb{R}^{N_{v_1}}, \mathbf{w}^{(2)} \in \mathbb{R}^{N_{v_2}}, \tag{4.8}$$

where $\mathbf{w}^{(1)}$, $\mathbf{w}^{(2)}$ are vectors as point values of the weight function $w(v) = \exp(-\frac{v^2}{2})$ at corresponding v_1 and v_2 grids. Then we seek a decomposition $\mathbf{f} = \mathbf{f}_1 + \mathbf{f}_2$, where $\mathbf{f}_1$ comes from the rescaling and orthogonal projection of $\mathbf{f}$ with respect to the weighted inner product (4.7) onto the subspace

$$\mathcal{N} \doteq \text{span}\{\mathbf{1}_{v_1 \otimes v_2}, \mathbf{v}_1 \otimes \mathbf{1}_{v_2}, \mathbf{1}_{v_1} \otimes \mathbf{v}_2, \mathbf{v}_1^2 \otimes \mathbf{1}_{v_2} + \mathbf{1}_{v_1} \otimes \mathbf{v}_2^2\} \tag{4.9}$$

for the conservation of mass, momentum and kinetic energy density, see Proposition 4.1 below for details. For further discussion, we also introduce a standard inner product for $\mathbf{f}, \mathbf{g} \in \mathbb{R}^{N_{v_1} \times N_{v_2}}$

$$\langle \mathbf{f}, \mathbf{g} \rangle = h_{v_1} h_{v_2} \sum_{j_1=1}^{N_{v_1}} \sum_{j_2=1}^{N_{v_2}} f_{j_1,j_2} g_{j_1,j_2}. \tag{4.10}$$

Proposition 4.1. *Let* $\mathbf{f}_1$ *come from the scaling and orthogonal projection of rescaled* $\mathbf{f}$ *with respect to the weighted inner product* (4.7) *onto the subspace* (4.9)*, in a similar spirit to the 1D1V case* (3.19)*. Assume* $\mathbf{f}$ *is written in the low rank HT format* (4.26)*.* $\mathbf{f}_1$ *can be represented in low rank HT format (consistently with the subscript* 1 *in the notations),*

$$\mathbf{f}_1 = \tilde{P}(\mathbf{f}) \quad \doteq \quad \sum_{l_{12}=1}^{4} \sum_{l_{34}=1}^{4} (\mathbf{B}_1^{(1,2,3,4)})_{l_{12},l_{34}} (\mathbf{U}_1^{(1,2)})_{l_{12}} \otimes (\mathbf{U}_1^{(3,4)})_{l_{34}}, \tag{4.11}$$

$$\overset{=}{(4.13)} \sum_{k=1}^{4} (\mathbf{U}_1^{(1,2)})_k \otimes (\mathbf{U}_1^{(3,4)})_k, \tag{4.12}$$

where we introduce the notation of $\tilde{P}$ *as the scaled orthogonal projection. The specifications of the frame vectors and transfer tensors are outlined below.*

- *For* $\mathbf{f}_1$*, the hierarchical ranks are* $r_{12} = r_{34} = 4$*,* $r_3 = r_4 = 3$*.* r_1 *and* r_2 *are the same as those for* $\mathbf{f}$*.*
- *The transfer tensor,* $\mathbf{B}_1^{(1,2,3,4)}$*, of size* 4×4*, is an identity matrix,*

$$\mathbf{B}_1^{(1,2,3,4)} = \mathbf{I}_{4\times 4}. \tag{4.13}$$

$(\mathbf{U}_1^{(3,4)})_k$*,* $k = 1, \ldots, 4$*, is constructed from an orthonormal set of basis* $\{\mathbf{V}_1, \cdots \mathbf{V}_4\}$ *in the* $v_1 - v_2$ *dimensions defined in* (4.27)*. We have* $r_3 = r_4 = 3$ *and*

$$(\mathbf{U}_1^{(3,4)})_k = \sum_{l_3=1}^{3} \sum_{l_4=1}^{3} (\mathbf{B}_1^{(3,4)})_{l_3,l_4,k} (\mathbf{w}^{(1)} \star (\mathbf{U}_1^{(3)})_{l_3}) \otimes (\mathbf{w}^{(2)} \star (\mathbf{U}_1^{(4)})_{l_4}) \tag{4.14}$$

with $\mathbf{w}^{(1)}$ *and* $\mathbf{w}^{(2)}$ *the same as in* (4.8)*. The frame vectors for node* 3 *are*

$$(\mathbf{U}_1^{(3)})_1 = \frac{1}{c_1} \mathbf{1}_{v_1}, \quad (\mathbf{U}_1^{(3)})_2 = \frac{1}{c_2} \mathbf{v}_1, \quad (\mathbf{U}_1^{(3)})_3 = \frac{1}{c_3} (\mathbf{v}_1^2 - c \mathbf{1}_{v_1}), \tag{4.15}$$

where $c = \frac{\langle \mathbf{1}_{v_1}, \mathbf{v}_1^2 \rangle_{\mathbf{w}}}{\langle \mathbf{1}_{v_1}, \mathbf{1}_{v_1} \rangle_{\mathbf{w}}}$ *is the orthogonalization constant of the basis, and* c_l*,* $l = 1, 2, 3$ *are normalization constants for the corresponding basis of* $\mathbf{1}_{v_1}$*,* $\mathbf{v}_1$ *and* $\mathbf{v}_1^2 - c\mathbf{1}_{v_1}$*. We have the same frame vectors for the node* 4 *but for* v_2*, assuming that the weight function and domain in* v_2 *is the same as* v_1*,*

$$(\mathbf{U}_1^{(4)})_1 = \frac{1}{c_1} \mathbf{1}_{v_2}, \quad (\mathbf{U}_1^{(4)})_2 = \frac{1}{c_2} \mathbf{v}_2, \quad (\mathbf{U}_1^{(4)})_3 = \frac{1}{c_3} (\mathbf{v}_2^2 - c \mathbf{1}_{v_2}) \tag{4.16}$$

The transfer tensor $\mathbf{B}_1^{(3,4)}$ *is a tensor of size* $3 \times 3 \times 4$*. It has zero elements, except the following specification for* $(\mathbf{B}_1^{(3,4)})_{l_3,l_4,l_{34}}$

$$(\mathbf{B}_1^{(3,4)})_{1,1,1} = (\mathbf{B}_1^{(3,4)})_{2,1,2} = (\mathbf{B}_1^{(3,4)})_{1,2,3} = 1, \quad (\mathbf{B}_1^{(3,4)})_{3,1,4} = (\mathbf{B}_1^{(3,4)})_{1,3,4} = \frac{1}{\sqrt{2}}. \tag{4.17}$$

- $(\mathbf{U}_1^{(1,2)})_k$, $k = 1, \cdots 4$, *is in the following form,*

$$(\mathbf{U}_1^{(1,2)})_k = \sum_{l_1=1}^{r_1} \sum_{l_2=1}^{r_2} (\mathbf{B}_1^{(1,2)})_{l_1,l_2,k} (\mathbf{U}_1^{(1)})_{l_1} \otimes (\mathbf{U}_1^{(2)})_{l_2}, \tag{4.18}$$

with the same frame vectors as $\mathbf{f}$ *on the leaf nodes* 1 *and* 2*, meaning that* $\mathbf{U}_1^{(1)}$ *and* $\mathbf{U}_1^{(2)}$ *are the same as* $\mathbf{U}^{(1)}$ *and* $\mathbf{U}^{(2)}$*, respectively.* $\mathbf{B}_1^{(1,2)}$*, of size* $r_1 \times r_2 \times 4$*, has its elements as,*

$$(\mathbf{B}_1^{(1,2)})_{l_1,l_2,k} = \sum_{l_{12}=1}^{r_{12}} \sum_{l_{34}=1}^{r_{34}} \mathbf{B}_{l_1,l_2,l_{12}}^{(1,2)} \mathbf{S}_{k,l_{34}} \mathbf{B}_{l_{12},l_{34}}^{(1,2,3,4)}, \quad k = 1, \cdots 4, \tag{4.19}$$

with r_{12}*,* r_{34} *and transfer tensors* $\mathbf{B}^{(1,2)}$ *and* $\mathbf{B}^{(1,2,3,4)}$ *from the HT representation of* $\mathbf{f}$*, and*

$$\mathbf{S}_{k,l_{34}} = \langle \mathbf{U}_{l_{34}}^{(3,4)}, \mathbf{V}_k \rangle, \quad k = 1, \cdots 4, \tag{4.20}$$

where $\mathbf{U}_{l_{34}}^{(3,4)}$*,* $l_{34} = 1, \ldots, r_{34}$*, is the frame tensor for node (3, 4) of* $\mathbf{f}$*. The inner product in the sense of* (4.10) *can be evaluated in a dimension-by-dimension manner.*

Finally $\mathbf{f}_1$ *has the same discrete macroscopic charge, current and kinetic energy density as with* $\mathbf{f}$*, which are denoted as* $\boldsymbol{\rho}$*,* $\mathbf{J}_1$*,* $\mathbf{J}_2$*, and* $\boldsymbol{\kappa}$*. The discrete macroscopic charge, current and kinetic energy density of* $\mathbf{f}_1$ *are*

$$\boldsymbol{\rho} = (c_1)^2 (\mathbf{U}_1^{(1,2)})_1 = (c_1)^2 \sum_{l_1,l_2} (\mathbf{B}_1^{(1,2)})_{l_1,l_2,1} \mathbf{U}_{l_1}^{(1)} \otimes \mathbf{U}_{l_2}^{(2)}, \tag{4.21}$$

$$\mathbf{J}_1 = c_1 c_2 (\mathbf{U}_1^{(1,2)})_2 = c_1 c_2 \sum_{l_1,l_2} (\mathbf{B}_1^{(1,2)})_{l_1,l_2,2} \mathbf{U}_{l_1}^{(1)} \otimes \mathbf{U}_{l_2}^{(2)}, \tag{4.22}$$

$$\mathbf{J}_2 = c_1 c_2 (\mathbf{U}_1^{(1,2)})_3 = c_1 c_2 \sum_{l_1,l_2} (\mathbf{B}_1^{(1,2)})_{l_1,l_2,3} \mathbf{U}_{l_1}^{(1)} \otimes \mathbf{U}_{l_2}^{(2)}, \tag{4.23}$$

$$\boldsymbol{\kappa} = \frac{1}{\sqrt{2}} c_1 c_3 (\mathbf{U}_1^{(1,2)})_4 + c\boldsymbol{\rho} = \frac{1}{\sqrt{2}} c_1 c_3 \sum_{l_1,l_2} (\mathbf{B}_1^{(1,2)})_{l_1,l_2,4} \mathbf{U}_{l_1}^{(1)} \otimes \mathbf{U}_{l_2}^{(2)} + c\boldsymbol{\rho}. \tag{4.24}$$

They are the same as those of $\mathbf{f}$*, which can be obtained by the HT tensor contraction*

$$\begin{aligned} \begin{pmatrix} \boldsymbol{\rho} \\ \mathbf{J}_1 \\ \mathbf{J}_2 \\ \boldsymbol{\kappa} \end{pmatrix} &= \sum_{l_{12}} \sum_{l_{34}} \mathbf{B}_{l_{12},l_{34}}^{(1,2,3,4)} \left\langle \mathbf{U}_{l_{34}}^{(3,4)}, \begin{pmatrix} \mathbf{1}_{v_1 \otimes v_2} \\ \mathbf{v}_1 \otimes \mathbf{1}_{v_2} \\ \mathbf{1}_{v_1} \otimes \mathbf{v}_2 \\ \frac{1}{2} \mathbf{v}_1^2 \otimes \mathbf{1}_{v_2} + \frac{1}{2} \mathbf{1}_{v_1} \otimes \mathbf{v}_2^2 \end{pmatrix} \right\rangle \mathbf{U}_{l_{12}}^{(1,2)}, \\ &= \sum_{l_{12}} \sum_{l_{34}} \mathbf{B}_{l_{12},l_{34}}^{(1,2,3,4)} \begin{pmatrix} S_{1,l_{34}} (c_1)^2 \\ S_{2,l_{34}} c_1 c_2 \\ S_{3,l_{34}} c_1 c_2 \\ S_{4,l_{34}} c_1 c_3 \frac{1}{\sqrt{2}} + S_{1,l_{34}} (c_1)^2 c \end{pmatrix} \mathbf{U}_{l_{12}}^{(1,2)}. \end{aligned} \tag{4.25}$$

Proof. We first scale $\mathbf{f}$ by $\mathbf{w}$ in (4.8)

$$\tilde{\mathbf{f}} = \sum_{l_{12}=1}^{r_{12}} \sum_{l_{34}=1}^{r_{34}} \mathbf{B}_{l_{12},l_{34}}^{(1,2,3,4)} \mathbf{U}_{l_{12}}^{(1,2)} \otimes \frac{1}{\mathbf{w}} \star \mathbf{U}_{l_{34}}^{(3,4)}. \tag{4.26}$$

Let $\tilde{\mathbf{U}}_{l_{34}}^{(3,4)} \doteq \frac{1}{\mathbf{w}} \star \mathbf{U}_{l_{34}}^{(3,4)}$, where $\star$ is the element-wise multiplication in the corresponding dimensions. Then we perform the orthogonal projection of $\tilde{\mathbf{f}}$ onto the subspace of (4.9) to obtain $\tilde{\mathbf{f}}_1$. Finally we rescale back to $\mathbf{f}_1 = \mathbf{w} \star \tilde{\mathbf{f}}_1$.

To perform the orthogonal projection, the orthogonalization of basis in (4.9) gives the frame vectors for the node 3 and 4 as specified in (4.15) and the transfer tensor $\mathbf{B}_1^{(3,4)}$ as specified in (4.17). This gives a set of orthonormal basis of $(\mathbf{U}_1^{(3,4)})_l$, $l = 1, 2, 3, 4$. We denote them as $\mathbf{V}_l$, $l = 1, 2, 3, 4$,

$$\begin{aligned}
&\mathbf{V}_1 = (\mathbf{U}_1^{(3)})_1 \otimes (\mathbf{U}_1^{(4)})_1, \quad \mathbf{V}_2 = (\mathbf{U}_1^{(3)})_2 \otimes (\mathbf{U}_1^{(4)})_1 \\
&\mathbf{V}_3 = (\mathbf{U}_1^{(3)})_1 \otimes (\mathbf{U}_1^{(4)})_2, \quad \mathbf{V}_4 = \frac{1}{\sqrt{2}}(\mathbf{U}_1^{(3)})_3 \otimes (\mathbf{U}_1^{(4)})_1 + \frac{1}{\sqrt{2}}(\mathbf{U}_1^{(3)})_1 \otimes (\mathbf{U}_1^{(4)})_3.
\end{aligned} \tag{4.27}$$

To perform the orthogonal projection to (4.9), only the bases of node $(3, 4)$ are affected. In particular, let $\mathbf{P}^{(3,4)}$ be the projection operator at the node $(3, 4)$, then

$$\begin{aligned}
\mathbf{P}^{(3,4)}\tilde{\mathbf{f}} &= \mathbf{P}^{(3,4)} \sum_{l_{12}=1}^{r_{12}} \sum_{l_{34}=1}^{r_{34}} \mathbf{B}_{l_{12},l_{34}}^{(1,2,3,4)} \mathbf{U}_{l_{12}}^{(1,2)} \otimes \frac{1}{\mathbf{w}} \star \mathbf{U}_{l_{34}}^{(3,4)} \\
&= \sum_{l_{12}=1}^{r_{12}} \sum_{l_{34}=1}^{r_{34}} \mathbf{B}_{l_{12},l_{34}}^{(1,2,3,4)} \mathbf{U}_{l_{12}}^{(1,2)} \otimes \left(\mathbf{P}^{(3,4)} (\frac{1}{\mathbf{w}} \star \mathbf{U}_{l_{34}}^{(3,4)}) \right),
\end{aligned} \tag{4.28}$$

where

$$\mathbf{P}^{(3,4)} (\frac{1}{\mathbf{w}} \star \mathbf{U}_{l_{34}}^{(3,4)}) = \mathbf{P}^{(3,4)} \tilde{\mathbf{U}}_{l_{34}}^{(3,4)} = \sum_{k=1}^{4} \langle \tilde{\mathbf{U}}_{l_{34}}^{(3,4)}, \mathbf{V}_k \rangle_{\mathbf{w}} \mathbf{V}_k \doteq \sum_{k=1}^{4} \mathbf{S}_{k,l_{34}} \mathbf{V}_k, \tag{4.29}$$

with

$$\mathbf{S}_{k,l_{34}} = \langle \tilde{\mathbf{U}}_{l_{34}}^{(3,4)}, \mathbf{V}_k \rangle_{\mathbf{w}} = \langle \frac{1}{\mathbf{w}} \star \mathbf{U}_{l_{34}}^{(3,4)}, \mathbf{V}_k \rangle_{\mathbf{w}} = \langle \mathbf{U}_{l_{34}}^{(3,4)}, \mathbf{V}_k \rangle. \tag{4.30}$$

Plug (4.29) into (4.28),

$$\begin{aligned}
\mathbf{P}^{(3,4)}\tilde{\mathbf{f}} &= \sum_{l_{12}=1}^{r_{12}} \sum_{l_{34}=1}^{r_{34}} \mathbf{B}_{l_{12},l_{34}}^{(1,2,3,4)} \mathbf{U}_{l_{12}}^{(1,2)} \otimes (\sum_{k=1}^{4} \mathbf{S}_{k,l_{34}} \mathbf{V}_k), \\
&= \sum_{l_{12}=1}^{r_{12}} \sum_{l_{34}=1}^{r_{34}} \mathbf{B}_{l_{12},l_{34}}^{(1,2,3,4)} \left(\sum_{l_1=1}^{r_1} \sum_{l_2=1}^{r_2} (\mathbf{U}_{l_1}^{(1)} \otimes \mathbf{U}_{l_2}^{(2)}) \mathbf{B}_{l_1,l_2,l_{12}}^{(1,2)} \right) \otimes (\sum_{k=1}^{4} \mathbf{S}_{k,l_{34}} \mathbf{V}_k), \\
&= \sum_{k=1}^{4} \left(\sum_{l_1=1}^{r_1} \sum_{l_2=1}^{r_2} \underbrace{\left(\sum_{l_{12}=1}^{r_{12}} \sum_{l_{34}=1}^{r_{34}} \mathbf{B}_{l_{12},l_{34}}^{(1,2,3,4)} \mathbf{B}_{l_1,l_2,l_{12}}^{(1,2)} \mathbf{S}_{k,l_{34}} \right)}_{\doteq (\mathbf{B}_1^{(1,2)})_{l_1,l_2,k}} (\mathbf{U}_{l_1}^{(1)} \otimes \mathbf{U}_{l_2}^{(2)}) \right) \otimes \mathbf{V}_k, \qquad (4.31) \\
&= \sum_{k=1}^{4} \left(\sum_{l_1=1}^{r_1} \sum_{l_2=1}^{r_2} (\mathbf{B}_1^{(1,2)})_{l_1,l_2,k} (\mathbf{U}_{l_1}^{(1)} \otimes \mathbf{U}_{l_2}^{(2)}) \right) \otimes \mathbf{V}_k, \qquad (4.32)
\end{aligned}$$

where we let $(\mathbf{B}_1^{(1,2)})_{l_1,l_2,k}$ as specified in (4.19), and

$$(\mathbf{U}_1^{(1,2)})_k \doteq \left(\sum_{l_1=1}^{r_1} \sum_{l_2=1}^{r_2} (\mathbf{B}_1^{(1,2)})_{l_1,l_2,k} (\mathbf{U}_{l_1}^{(1)} \otimes \mathbf{U}_{l_2}^{(2)}) \right) \tag{4.33}$$

as in (4.18).

The macroscopic charge, momentum and energy density of $\mathbf{f}_1$, (4.21)-(4.24) can be derived, from the form of $\mathbf{f}_1$ in (4.12), the form of $\mathbf{U}_1^{(1,2)}$ in (4.33), and

$$\mathbf{1}_{v_1 \otimes v_2} = (c_1)^2 \mathbf{V}_1, \quad \mathbf{v}_1 \otimes \mathbf{1}_{v_2} = c_1 c_2 \mathbf{V}_2, \quad \mathbf{1}_{v_1} \otimes \mathbf{v}_2 = c_1 c_2 \mathbf{V}_3,$$

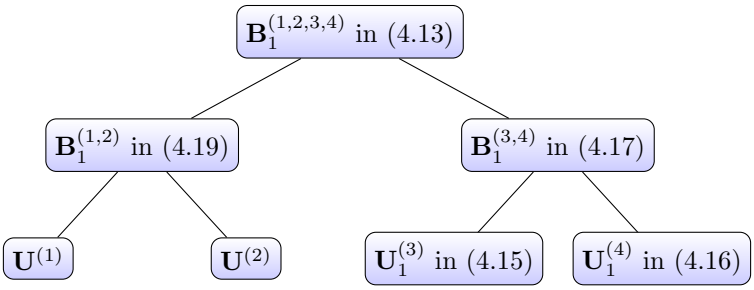


FIG. 4.2. *The data layout of* $\mathbf{f}_1$. *Here* $\mathbf{U}^{(1)}$ *and* $\mathbf{U}^{(2)}$ *are the same as those vector frames for* $\mathbf{f}$.

$$\frac{1}{2}\mathbf{v}_1^2 \otimes \mathbf{1}_{v_2} + \frac{1}{2}\mathbf{1}_{v_1} \otimes \mathbf{v}_2^2 = \frac{1}{\sqrt{2}} c_1 c_3 \mathbf{V}_4 + c(c_1)^2 \mathbf{V}_1,$$

which is due to (4.27). The agreement of macroscopic charge, momentum and energy density of $\mathbf{f}$ and $\mathbf{f}_1$ is a direct consequence of manipulation of equalities in (4.18), (4.19), (4.25) and orthonormal property of the basis in $\mathbf{U}_1^{(3,4)}$.

Once the orthogonal projection is performed, $\mathbf{f}_1$ stays untouched for the truncation. As with the 1D1V case, we will perform HOSVD to truncate the remainder $\mathbf{f}_2 \doteq \mathbf{f} - \mathbf{f}_1$, with a cost scales as $\mathcal{O}(dNr^2 + dr^4)$. If a standard HT truncation is directly applied to $\mathbf{f}_2$, unlike the 1D1V case the conservation property cannot be guaranteed. Below, we elaborate and investigate such an issue, and further develop a conservative projection procedure after the HT truncation to ensure charge, current and kinetic energy density conservation.

We start with a brief description of a naive weighted hierarchical HOSVD truncation procedure, as a direct analog of the 1D1V case. First, we scale $\mathbf{f}_2$ according to the weights and define $\tilde{\mathbf{f}}_2 = \frac{1}{\sqrt{\mathbf{w}}} \star \mathbf{f}_2$. Note that the rescaling is computed dimension-by-dimension. The standard hierarchical HOSVD root-to-leaf truncation with threshold ε is applied to $\tilde{\mathbf{f}}_2$ and $\mathcal{T}_\varepsilon(\tilde{\mathbf{f}}_2)$ is obtained. Finally, $\mathbf{f}_2$ is defined as

$$\mathbf{f}_2 = \sqrt{\mathbf{w}} \star \mathcal{T}_\varepsilon(\tilde{\mathbf{f}}_2). \tag{4.34}$$

The issue with such procedure is the loss of macroscopic conservation in the root-to-leaf truncation. In particular, the HT truncation $\mathcal{T}_\varepsilon(\tilde{\mathbf{f}}_2)$ can be represented as, see [14],

$$\mathcal{T}_\varepsilon(\tilde{\mathbf{f}}_2) = (\pi_1 \otimes \pi_2 \otimes \pi_3 \otimes \pi_4)(\pi_{12} \otimes \pi_{34})\, \mathbf{f}_2, \tag{4.35}$$

where π_α denotes the orthogonal projection on the subspace spanned by $\tilde{r}_\alpha$ leading left singular vectors $\{\tilde{\mathbf{U}}_i^{(\alpha)}\}_{i=1}^{\tilde{r}_\alpha}$ of the matricization $\mathcal{M}^{(\alpha)}(\mathbf{f}_2)$ with respect to node α. However, the bases at node $(3,4)$ are no longer orthogonal to $\{\mathbf{V}_1, \mathbf{V}_2, \mathbf{V}_3, \mathbf{V}_4\}$ in (4.27) due to the truncation π_3 and π_4 at leaf nodes. Hence $\mathbf{f}_2$ as in (4.34) may have nonzero charge, current and kinetic energy density, breaking the conservation property. To fix the issue, we propose to apply the operator $(\mathbf{I} - \tilde{P})$, to $\sqrt{\mathbf{w}} \star \mathcal{T}_\varepsilon(\tilde{\mathbf{f}}_2)$ to ensure zero charge, current and kinetic energy density of truncated $\mathbf{f}_2$. Here $\tilde{P}$ is the same as that specified in (4.11) in Proposition 4.1. We introduce the following notation,

$$\widetilde{\mathcal{T}}_\varepsilon(\mathbf{f}_2) \doteq (\mathbf{I} - \tilde{P}) \left(\sqrt{\mathbf{w}} \star \mathcal{T}_\varepsilon(\tilde{\mathbf{f}}_2) \right). \tag{4.36}$$

Finally, the conservative HT truncation of $\mathbf{f}$ is done as follows

$$T_c(\mathbf{f}) \doteq \mathbf{f}_1 + \widetilde{\mathcal{T}}_\varepsilon(\mathbf{f}_2). \tag{4.37}$$

The following proposition is a straightforward consequence from the orthogonal projection.

Proposition 4.2. $\widetilde{\mathcal{T}}_\varepsilon(\mathbf{f}_2)$ *has zero charge density, zero current density, and zero kinetic energy density. Hence,* $T_c(\mathbf{f})$ *preserves the charge, current, and kinetic energy densities* ($\boldsymbol{\rho}$, $\mathbf{J}_1$, $\mathbf{J}_2$, $\boldsymbol{\kappa}$) *of the original* $\mathbf{f}$.

Proof. The zero charge, current and kinetic energy density is a direct consequence of the $\mathbf{I} - \tilde{P}$ projection operator. From this fact, together with from Proposition 4.1, $T_c(\mathbf{f})$ preserves the charge, current, and kinetic energy densities ($\boldsymbol{\rho}$, $\mathbf{J}_1$, $\mathbf{J}_2$, $\boldsymbol{\kappa}$) of the original $\mathbf{f}$.

Proposition 4.3. *(Local mass and momentum conservation for the 2D2V VP system.) If the discrete differential operators D_x, D_v employed are conservative, i.e., can be written in a flux difference form, and linear, i.e., can preserve linear relations, then the proposed low rank method with an SSP multi-step time integrator preserves the mass and momentum locally; that is the schemes for $\boldsymbol{\rho}$, $\mathbf{J}_1$, $\mathbf{J}_2$, from integrating the scheme on $\mathbf{f}^n$, are consistent and conservative discretization of the macroscopic moment equations* (2.7)-(2.8).

Proof. The proof is based on a conservative and linear discretization of the discrete differential operators D_x, D_v employed and from the conservative truncation procedure as proposed above. The details are similar to that of Proposition 3.4 and hence omitted for brevity.

Remark 4.4. *To ensure the local conservation, $\mathbf{f}_1$ should not be further compressed. In the numerical simulation, we still truncate $\mathbf{f}_1$ at nodes (1, 2) with threshold 10^{-15} to remove the redundancy from the add basis procedure. Thus the local conservation property is preserved on the same scale of machine precision, i.e. 10^{-15}. In Section 5, we present the numerical result for a linear 3D3V Vlasov system with a known electric field. The solution exhibits a very low rank structure, and our method is effective in capturing this structure while also conserving the mass and momentum. However, it's important to note that the proposed method may be too expensive to apply in the fully nonlinear 3D3V setting, particularly when the solution does not exhibit a low rank structure, and hence it may be more efficient not to employ the low rank decomposition in the $\mathbf{x}$ direction.*

We summarized the conservative truncation procedure as Algorithm 2 below for the 2D2V VP solution.

Algorithm 2: The conservative truncation procedure for the 2D2V VP solution.

- Input: the pre-compressed low rank solution $\mathbf{f}$ in the HT format with dimension tree given in Figure 4.1 (a) and the associated data including the frame tensors $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$, $\mathbf{U}^{(3)}$, and $\mathbf{U}^{(4)}$ at nodes (1), (2), (3), and (4), respectively, and transfer tensors $\mathbf{B}^{(1,2)}$, $\mathbf{B}^{(3,4)}$, and $\mathbf{B}^{(1,2,3,4)}$ at nodes $(1,2)$, $(3,4)$, and $(1,2,3,4)$, respectively.
- Output: the compressed low rank solution $T_c(\mathbf{f})$ in the HT format with the same charge, current, and kinetic energy density functions as $\mathbf{f}$.

1. Compute the rescaled orthogonal projection to obtain $\mathbf{f}_1 = \tilde{P}(\mathbf{f})$, in the HT format with data layout in Figure 4.2. The transfer tensor $\mathbf{B}_1^{(1,2,3,4)}$ is the identity matrix of size 4×4, $(\mathbf{B}_1^{(1,2)})_{l_{12}}$ is a matrix of size $r_1 \times r_2 \times 4$ from (4.19), and $\mathbf{B}_1^{(3,4)}$ is a matrix of size $3 \times 3 \times 4$ from (4.17), with the frame tensors $\mathbf{U}_1^{(1)} = \mathbf{U}^{(1)}$, $\mathbf{U}_1^{(2)} = \mathbf{U}^{(2)}$, $\mathbf{U}_1^{(3)}$ and $\mathbf{U}_1^{(4)}$ from (4.15) and (4.16), respectively. At the end of this step, we truncate node (1, 2) of $\mathbf{f}_1$ with threshold 10^{-15} to remove redundant basis in $\mathbf{U}_1^{(1,2)}$.
2. Perform the HOSVD truncation, together with an orthogonal projection operator, to $\mathbf{f}_2 \doteq \mathbf{f} - \mathbf{f}_1$ to ensure zero charge, current and energy densities:
   (a) Compute $\mathbf{f}_2 = \mathbf{f} - \mathbf{f}_1$, and scale it to obtain $\tilde{\mathbf{f}}_2 = \frac{1}{\sqrt{\mathbf{w}}} \star \mathbf{f}_2$.
   (b) Apply the standard HOSVD truncation to $\tilde{\mathbf{f}}_2$, and apply rescaling to obtain $\sqrt{\mathbf{w}} \star \mathcal{T}_\varepsilon(\tilde{\mathbf{f}}_2)$.
   (c) Apply $\mathbf{I} - \tilde{P}$ to $\sqrt{\mathbf{w}} \star \mathcal{T}_\varepsilon(\tilde{\mathbf{f}}_2)$ to obtain $\widetilde{\mathcal{T}}_\varepsilon(\mathbf{f}_2)$, i.e. (4.36), with the same $\tilde{P}$ operator as in the previous step.
3. Update the compressed low rank solution $T_c(\mathbf{f}) = \mathbf{f}_1 + \widetilde{\mathcal{T}}_\varepsilon(\mathbf{f}_2)$ from (4.37).

---

5. Numerical results. In this section we present a collection of numerical examples to demonstrate the efficacy of the proposed conservative low rank tensor method for simulating the VP system. In the simulations, fifth order upwind finite difference methods are employed for spatial discretization, together with a second order SSP multi-step method denoted by SSPML2 for temporal discretization. The numerical solutions of high dimensions are represented in the HT format [18]. We also compare the proposed conservative low rank methods against the nonconservative version in terms of efficiency and ability to conserve the physical invariants. Unless otherwise noted, we let the weight function $w(\mathbf{v}) = \exp(-\frac{|\mathbf{v}|^2}{2})$.

5.1. 1D1V Vlasov-Poisson system.

Example 5.1. *(Weak Landau damping.) We consider the weak Landau damping test with initial condition*

$$f(x, v, t = 0) = \frac{1}{\sqrt{2\pi}} \left(1 + \alpha \cos(kx)\right) \exp\left(-\frac{v^2}{2}\right), \tag{5.1}$$

where $\alpha = 0.01$ *and* $k = 0.5$.

The computational domain is set to be $[0, L_x] \times [-L_v, L_v]$ with $L_x = 2\pi/k$ and $L_v = 6$. We set $\varepsilon = 10^{-5}$ for truncation. In Figure 5.1, we report the time histories of the electric energy and numerical ranks of the solutions computed by the conservative and non-conservative methods for comparison. It is observed that both methods are able to predict the correct damping rate of the electric energy. Meanwhile, the numerical ranks of the conservative method are bigger than those of the non-conservative counterpart for the same threshold. It is partially ascribed to the fact that the weighted SVD truncation is applied to $\mathbf{f}_2$ only and $\mathbf{f}_1$ is a rank three tensor that remains unchanged for the conservative method, while the standard non-weighted SVD truncation is applied to the pre-compressed solution for the non-conservative method. Due to the larger rank, the conservative method will take more CPU time than a non-conservative method. On the other hand, with a higher representation rank, the conservative method can track the damping phenomenon more accurately, see Figure 5.4 for the next example. In Figure 5.2, the time histories of the relative deviation of total mass, momentum and energy are plotted. The conservative method is found to be able to conserve the total mass and momentum up to machine precision regardless of the mesh size used. As mentioned above, the proposed conservative method cannot conserve the total energy, as the time integrator employed is not energy conserving. With the mesh refinement, the conservation error of total energy decreases. Meanwhile, the non-conservative method can conserve the total mass and energy up to the magnitude of truncation threshold ε, but the total momentum is conserved to the machine precision, which is attributed to the symmetry of the solution in the velocity direction. Noteworthy, though both methods cannot conserve the total energy, the conservative one does a better job in energy conservation compared to the non-conservative counterpart, and it is because that the kinetic energy is preserved in the truncation. It is also observed that the numerical ranks of the solution on the coarser mesh are higher than that on the finer mesh. This is because the numerical errors coming from discretization could contribute to extra rank increase, and hence the use of finer mesh is beneficial for the proposed low rank method. Furthermore, as mentioned in Remark 3.6, the choice of the weight function $w(v)$ may affect the performance of the conservative low rank method. In Figure 5.3, we report the numerical results for the conservative method with a different weight function $w(v) = \exp(-v^2/4)$. Note that the new weight function has a larger value at the boundary compared to $\exp(-v^2/2)$. It is observed that the method with the new weight function enjoys smaller numerical ranks of the solutions, thereby leading to improved computational efficiency; however, the conservation errors in mass and momentum become slightly larger.

Example 5.2. *(Strong Laudau damping.) We consider the strong Landau damping test with the initial condition* (5.1) *and a bigger perturbation parameter* $\alpha = 0.5$.

Two truncation thresholds $\varepsilon = 10^{-3}, 10^{-4}$ are used to compare the performance of the proposed conservative method with the non-conservative one. In Figure 5.4, we report the time evolution of the electric energy together with the ranks of the numerical solutions for $\varepsilon = 10^{-3}$. We observe that the conservative method is able to capture correctly the nonlinear dynamics of the strong Landau damping as opposed to the non-conservative method. This is because the truncation error due to the large threshold used greatly pollutes the accuracy for the non-conservative method, leading to severe rank increase. By design the conservative method exactly conserves the mass and momentum densities in the low rank setting. Such conservation helps resolve the nonlinear Vlasov dynamics with a relatively large truncation threshold. As observed in Figure 5.5, the conservative method can conserve the total mass and momentum up to the machine precision regardless of the mesh size used. Again, the energy conservation is not observed, but the conservation error decreases with mesh refinement, which is not the case for the non-conservative method. Then we consider a smaller truncation threshold $\varepsilon = 10^{-4}$, and the truncation error is reduced accordingly. Both methods generate consistent results as plotted in Figure 5.6. As with the weak case, the conservative method has slightly larger ranks than the non-conservative method. We have a similar observation of the methods in conserving the invariants in Figure 5.7, as that in Figure 5.5.

Example 5.3. *(Bump on tail.) In this example, we simulate the bump-on-tail test with the initial condition*

$$f(x, v, t = 0) = (1 + \alpha \cos(kx)) \left(n_p \exp\left(-\frac{v^2}{2}\right) + n_b \exp\left(-\frac{(v-u)^2}{2v_t}\right) \right), \tag{5.2}$$

where $\alpha = 0.04$, $k = 0.3$, $n_p = \frac{9}{10\sqrt{2\pi}}$, $n_b = \frac{2}{10\sqrt{2\pi}}$, $u = 4.5$, $v_t = 0.5$.

In the simulation, we set $\varepsilon = 10^{-4}$ for truncation. The weight function $w(v) = \exp(-\frac{v^2}{3})$ is chosen for the conservative method. Note that unlike the previous Landau damping examples, the solution of bump

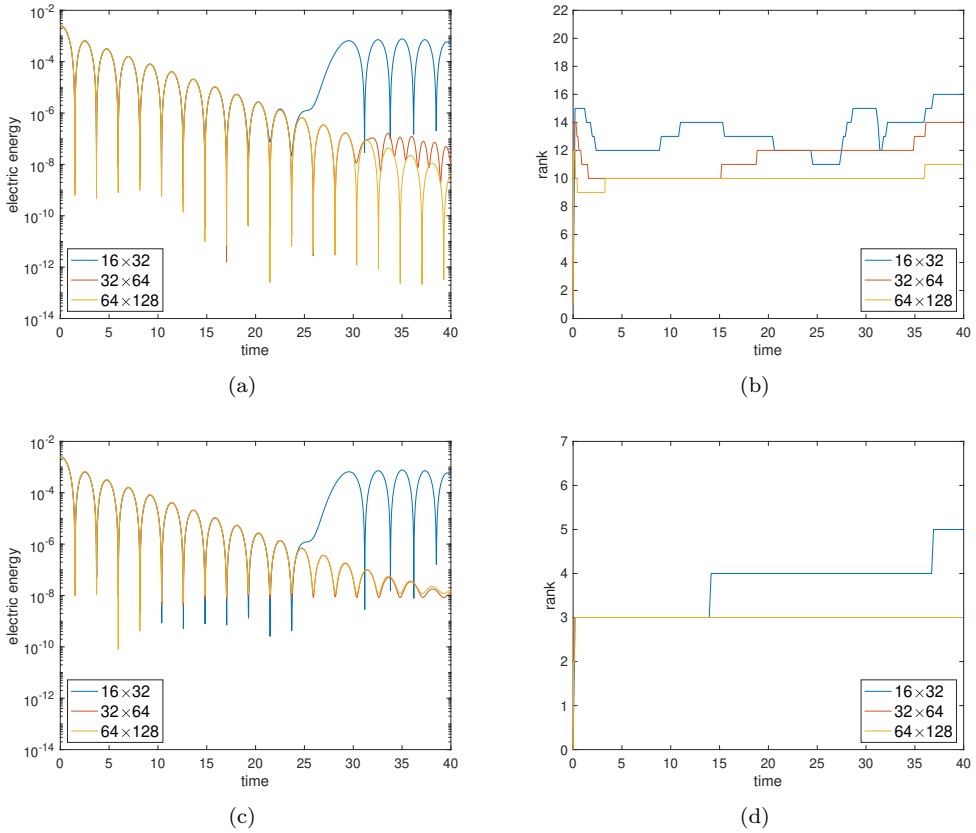


FIG. 5.1. *Example 5.1. The time evolution of the electric energy (a, c) and the rank of the numerical solutions (b, d). Conservative method (a, b) and non-conservative method (c, d).* $\varepsilon = 10^{-5}$.

on tail does not have the symmetry. In Figure 5.8, we report the time evolution of the electric energy as well as the ranks of the solutions. Consistent numerical results are observed. In Figure 5.9, we plot the time evolution of the relative deviation of the total mass, momentum and energy for both methods. The non-conservative method is able to conserve the invariants on the scale of the truncation threshold. Note that the exact momentum conservation is not observed as expected. Meanwhile, the conservative method is able to conserve the total mass and momentum up to the machine precision and has smaller conservation errors of total energy compared to the non-conservative method. Note that the conservation error of the total momentum increases after $t = 15$ for the conservative method with the coarse mesh size 32×64 which is ascribed to the boundary error. In Figure 5.10, we report the contour plots of the solutions by the two methods. The numerical solutions are observed to qualitatively match each other. Last, we test the performance of projector $P_{\mathcal{N}}$ with different subspaces $\mathcal{N}$ as discussed in Remark 3.6. In particular, we denote by P_1, P_2 and P_3 the orthogonal projectors with the weighted inner product (3.15) onto the subspaces $\text{span}\{1_v\}$, $\text{span}\{1_v, \mathbf{v}\}$, $\text{span}\{1_v, \mathbf{v}, \mathbf{v}^2\}$, respectively. In Figure 5.11, we plot the time evolution of the electric energy, ranks, total mass, total momentum, and total energy of the solutions from the conservative method with P_1, P_2 and P_3. The three methods generate consistent results for the time evolution of the electric energy, and the ranks are comparable. By construction all three methods can conserve the total mass, and the methods with P_2 and P_3 can conserve the total momentum. Further, the method with P_3 does the best job in conserving the total energy as P_3 preserves the kinetic energy for truncation.

5.2. 2D2V Vlasov-Poisson system.

Example 5.4. *(Weak Landau damping.) We consider the 2D2V weak Landau damping, the dynamics of*

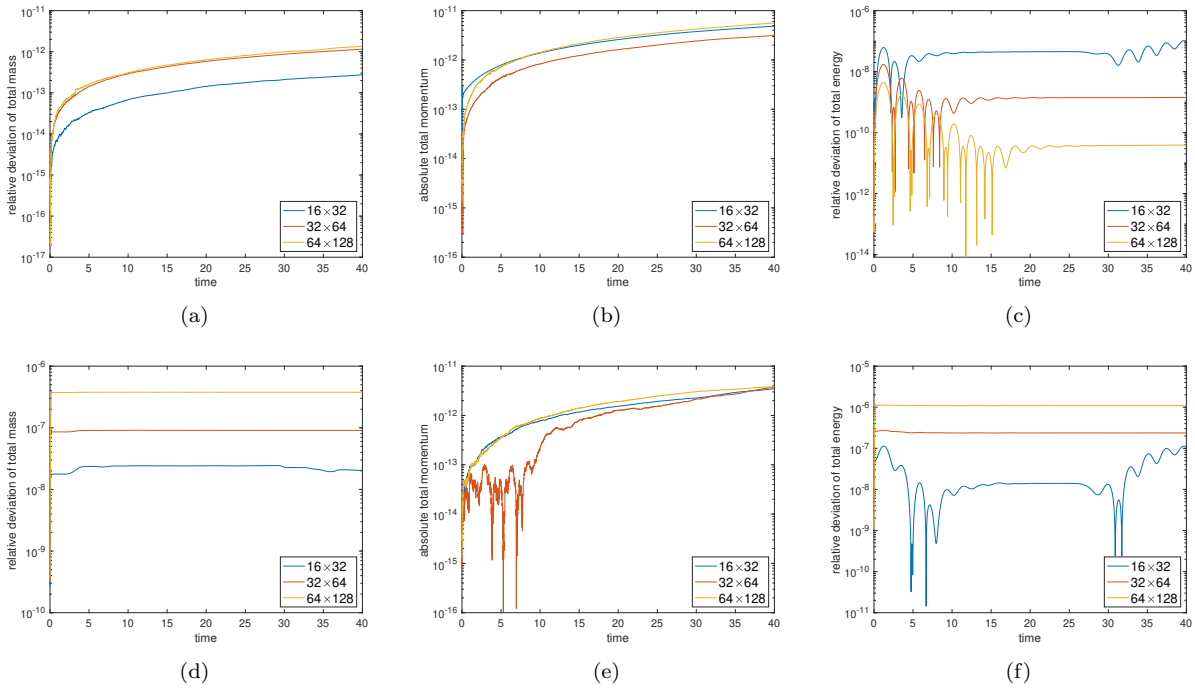


FIG. 5.2. *Example 5.1. The time evolution of relative deviation of total mass (a, d), absolute total momentum (b, d), and relative deviation of total energy (c, f). Conservative method (a, b, c) and non-conservative method (d, e, f).* $\varepsilon = 10^{-5}$.

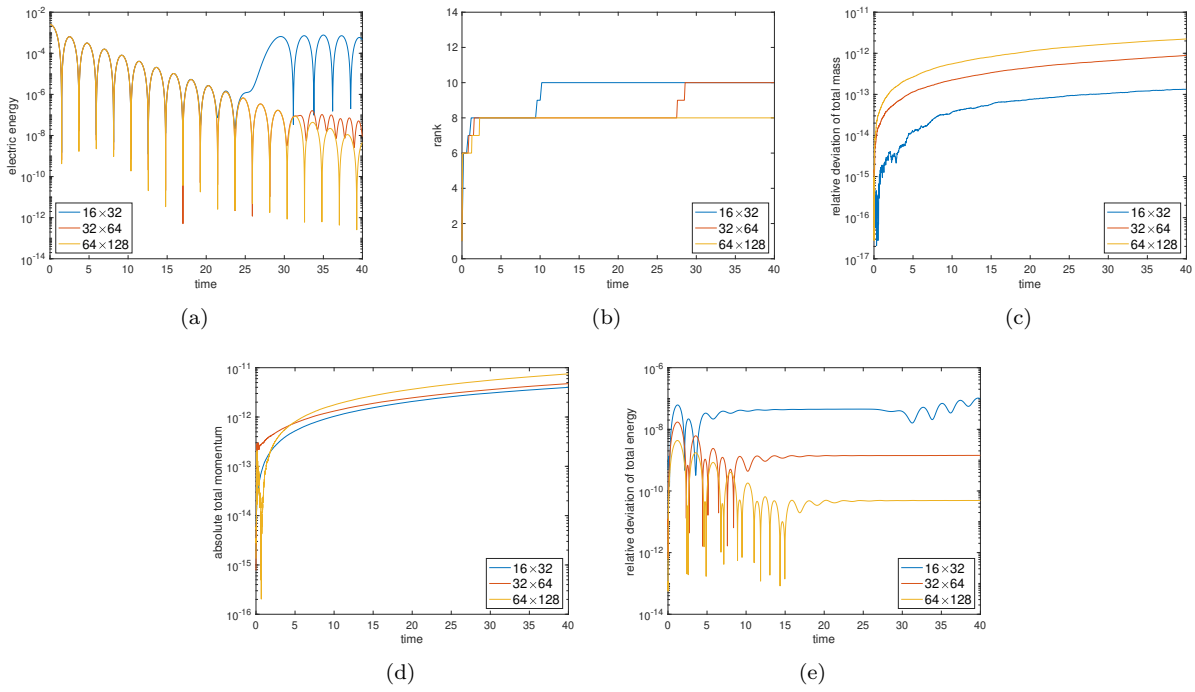


FIG. 5.3. *Example 5.1. Weight function* $w(v) = \exp(-v^2/4)$. *The time evolution of the electric energy (a), the rank of the numerical solutions (b), relative deviation of total mass (c), absolute total momentum (d), and relative deviation of total energy (f).* $\varepsilon = 10^{-5}$.

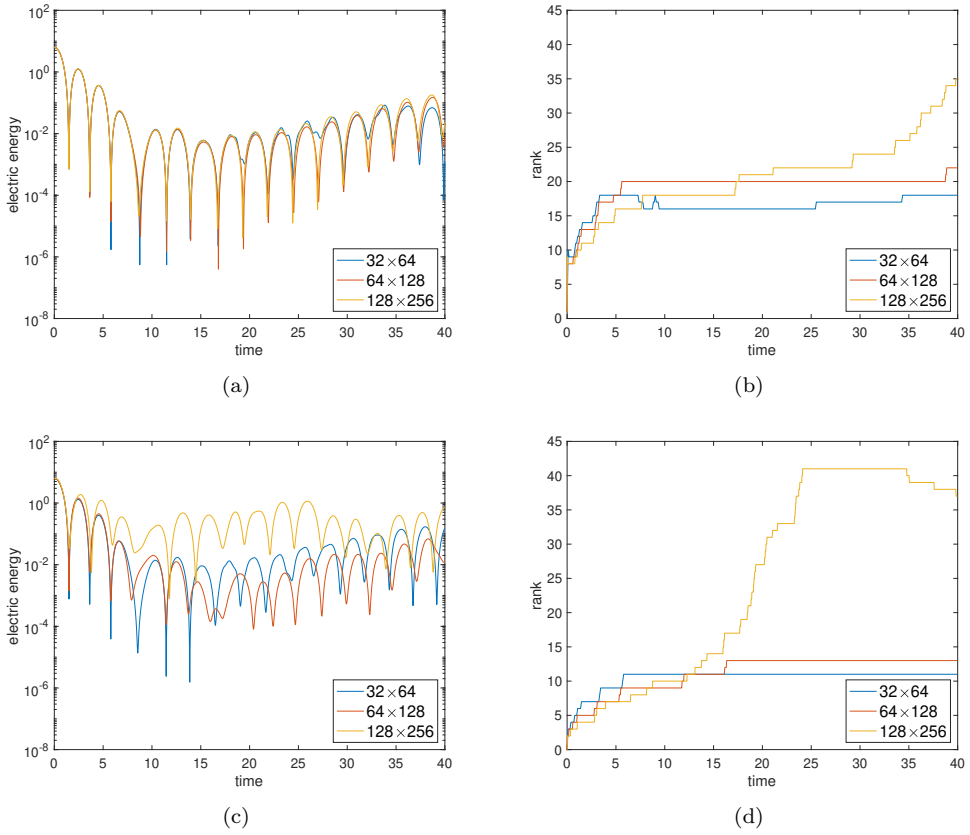


FIG. 5.4. *Example 5.2. The time evolution of the electric energy (a, c) and the rank of the numerical solutions (b, d). Conservative method (a, b) and non-conservative method (c, d).* $\varepsilon = 10^{-3}$.

which is similar to the 1D1V case. The initial condition is

$$f(\mathbf{x}, \mathbf{v}, t = 0) = \frac{1}{(2\pi)^{d/2}} \left(1 + \alpha \sum_{m=1}^{d} \cos(kx_m)\right) \exp\left(-\frac{|\mathbf{v}|^2}{2}\right), \tag{5.3}$$

where $d = 2$, $\alpha = 0.01$, *and* $k = 0.5$.

We set the computation domain as $[0, L_x]^2 \times [-L_v, L_v]^2$, where $L_x = \frac{2\pi}{k}$ and $L_v = 6$, and the truncation threshold $\varepsilon = 10^{-5}$. We simulate the problem with both conservative and non-conservative methods, and the solutions are represented in the fourth order HT format. In Figures 5.12-5.13, we report the time evolution of the electric energy, hierarchical ranks of the numerical solution, relative deviation of total mass and energy together with absolute total momentum J_1 and J_2. It is observed that both methods are able to predict the damping rate of the electric energy. Furthermore, the conservative method is able to conserve the total mass and momentum J_1 and J_2 up to the machine precision and enjoys better total energy conservation compared to the non-conservative counterpart. On the other hand, the hierarchical ranks of the solution from the conservative method, especially r_1 and r_2, are larger than that from the non-conservative method. This is because $\mathbf{f}_1$ is constructed without compression to guarantee the local conservation (in fact $\mathbf{f}_1$ is truncated with threshold 10^{-15} in the simulation), and then the solution tensor is not compressed in the $\mathbf{x}$ direction.

Example 5.5. *We consider the 2D2V two-stream instability with initial condition*

$$f(\mathbf{x}, \mathbf{v}, t = 0) = \frac{1}{2^d (2\pi)^{d/2}} \left(1 + \alpha \sum_{m=1}^{d} \cos(kx_m)\right) \prod_{m=1}^{d} \left(\exp\left(-\frac{(v_m - v_0)^2}{2}\right) + \exp\left(-\frac{(v_m + v_0)^2}{2}\right)\right), \tag{5.4}$$

where $d = 2$, $\alpha = 0.001$, $v_0 = 2.4$, *and* $k = 0.2$.

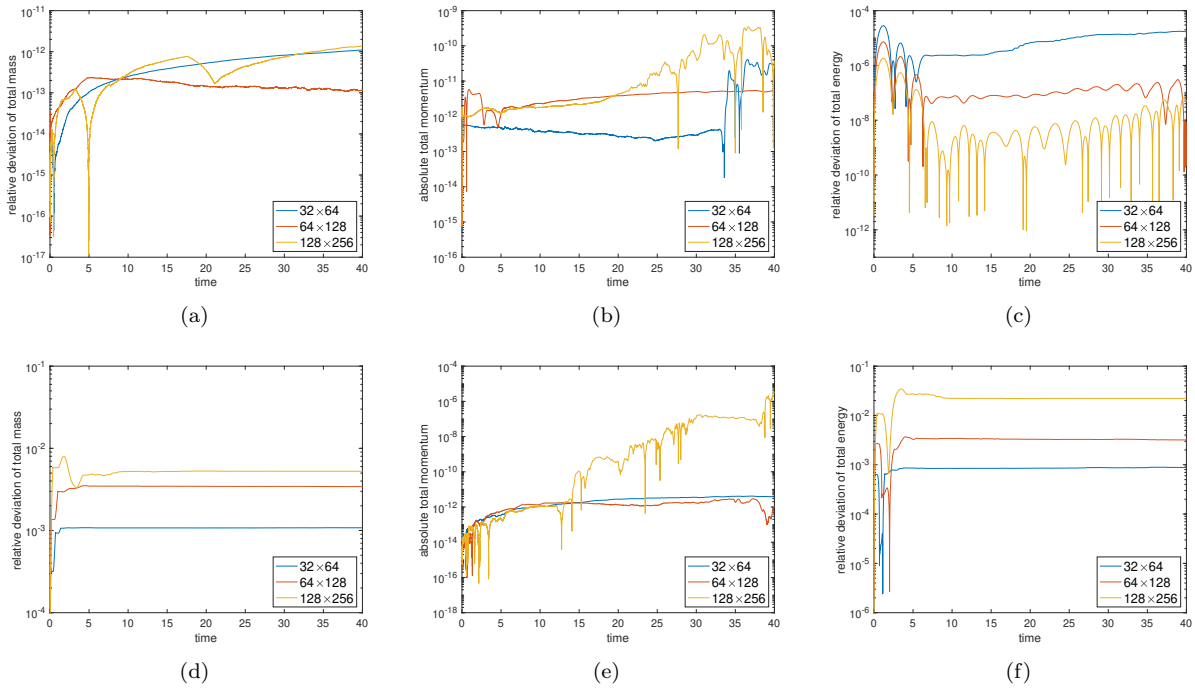


FIG. 5.5. *Example 5.2. The time evolution of relative deviation of total mass (a, d), absolute total momentum (b, e), and relative deviation of total energy (c, f). Conservative method (a, b, c) and non-conservative method (d, e, f).* $\varepsilon = 10^{-3}$.

The computation domain is set as $[0, L_x]^2 \times [-L_v, L_v]^2$, where $L_x = \frac{2\pi}{k}$ and $L_v = 8$, and the truncation threshold is set as $\varepsilon = 10^{-5}$. In Figures 5.14-5.15, respectively for conservative and non-conservative methods, we report the time evolution of the electric energy, hierarchical ranks of the numerical solution, relative deviation of total mass and energy together with absolute total momentum J_1 and J_2. The observation is similar to the previous example that the proposed conservative method is able to conserve the total mass and momentum, and meanwhile, the hierarchical ranks of the solution tensor from the conservative method are larger than that from the non-conservative counterpart.

5.3. 3D3V Linear Vlasov system.

Example 5.6. *In this example, we simulate a 3D3V forced linear Vlasov system*

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla_{\mathbf{x}} f + \mathbf{E}(\mathbf{x}, t) \cdot \nabla_{\mathbf{v}} f = \psi(\mathbf{x}, \mathbf{v}, t),$$

where the external electric field is $\mathbf{E} = (E_1, E_2, E_3)$*:*

$$\begin{cases} E_1 &= \sin(2x_1 - 2\pi t), \\ E_2 &= \sin(2x_2 - 2\pi t), \\ E_3 &= \sin(2x_3 - 2\pi t). \end{cases}$$

ψ is chosen such that the Vlasov equation has the exact solution

$$f(\mathbf{x}, \mathbf{v}, t) = \left(\prod_{i=1}^{3} \cos(2x_i - 2\pi t) \right) \exp\left(-\frac{1}{4} |\mathbf{v} - 1|^2 \right).$$

Note that the solution remains a rank one tensor over time and the system conserves the mass and momentum.

The computational domain is set as $[-\pi, \pi]^3 \times [-4, 4]^3$, and we let $\varepsilon = 10^{-4}$ for truncation. Additionally, we employ the dimension tree given in Figure 5.3 to represent the 3D3V Vlasov solution tensor in the HT

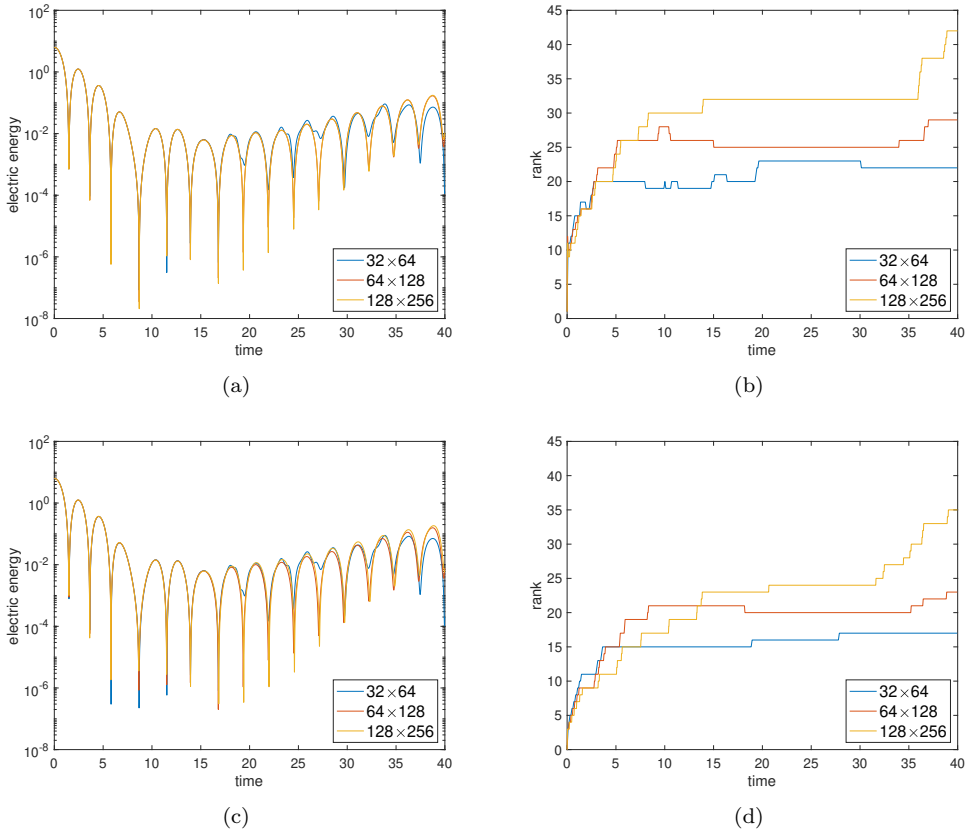


FIG. 5.6. *Example 5.2. The time evolution of the electric energy (a, c) and the ranks of the numerical solutions (b, d). Conservative method (a, b) and non-conservative method (c, d).* $\varepsilon = 10^{-4}$.

format. We apply the conservative and nonconservative low rank methods to simulate the problems up to $t = 0.5$ and summarize the convergence study in Table 5.1. Second order accuracy is observed for that both methods due to the use of SSPML2 for time discretization. In Figure 5.17, we plot the time evolution of hierarchical ranks of the numerical solution with mesh size $512^3 \times 512^3$, together with the time evolution of the absolute total mass and absolute total momentum J_1, J_2 and J_3 computed by the conservative low rank method. As with the previous 2D2V case, the proposed method can effectively capture the underlying low rank structure, observing that the hierarchical ranks of the solution remain low over the time evolution. Further, the method is able to conserve the mass and momentum up to 10^{-10}.

TABLE 5.1
Example 5.6. $t = 0.5$. *Accuracy test.*

	Conservative method		Nonconservative method	
$N_x^3 \times N_v^3$	L^2-error	order	L^2-error	order
$64^3 \times 64^3$	2.93E-03		2.89E-03	
$128^3 \times 128^3$	1.14E-03	1.37	1.36E-03	1.08
$256^3 \times 256^3$	2.74E-04	2.05	3.27E-04	2.06
$512^3 \times 512^3$	3.91E-05	2.81	4.53E-05	2.85

6. Conclusion. In this paper, we proposed a conservative truncation procedure for a low-rank tensor approach for performing a grid-based Vlasov simulations. The basic idea is initialized in the 1D1V setting, and is further developed to the 2D2V setting with the HT tensor decompositions. The newly developed conservative low rank tensor algorithm is theoretically proved to be a locally conservative scheme to the

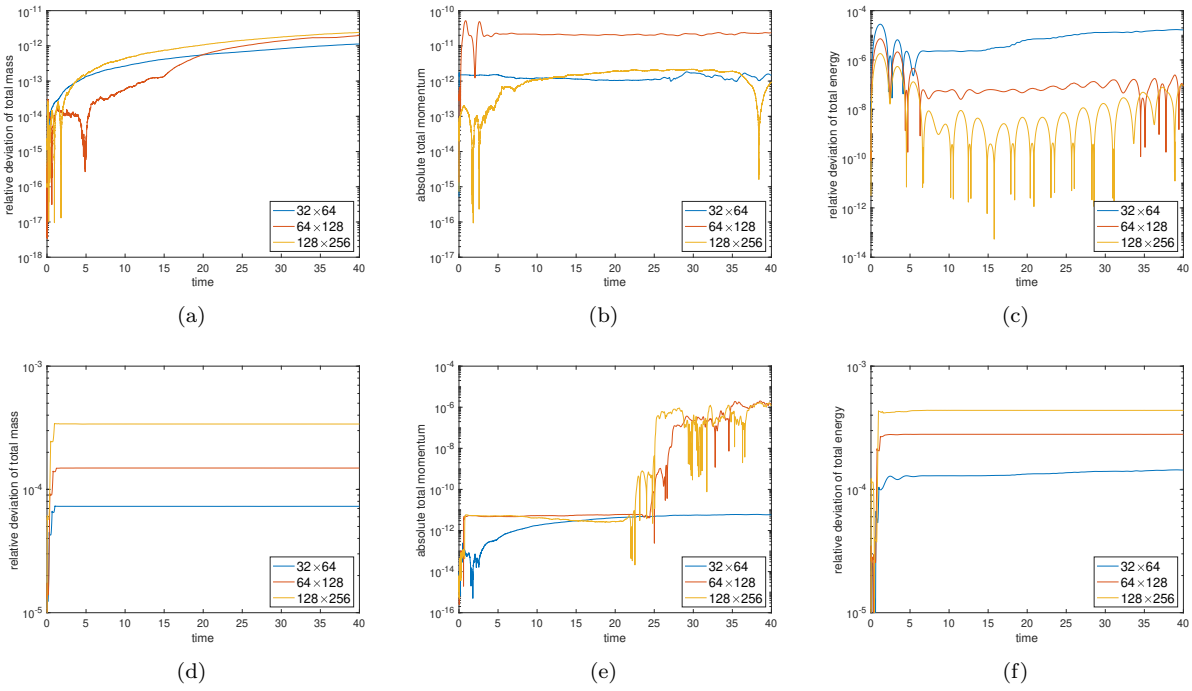


FIG. 5.7. *Example 5.2. The time evolution of relative deviation of total mass (a, d), absolute total momentum (b, e), and relative deviation of total energy (c, f). Conservative method (a, b, c) and non-conservative method (d, e, f).* $\varepsilon = 10^{-4}$.

macroscopic equations for charge and current densities, and is numerically verified to globally conserve the total charge and current. We plan to extend our algorithm in two future directions. Firstly, we aim to develop a low rank tensor algorithm that conserves local energy in addition to local charge and current conservation. Secondly, we plan to extend our approach to the fully nonlinear 3D3V setting. Due to the preservation of local conservation laws in physical spaces, the computational complexity is still high, especially in 3D physical spaces. To enable realistic 3D3V simulations, we will explore high-performance computing aspects of implementation and develop algorithms that further reduce computational complexity.

REFERENCES

[1] F. Allmann-Rahn, R. Grauer, and K. Kormann. A parallel low-rank solver for the six-dimensional vlasov-maxwell equations. *arXiv preprint arXiv:2201.03471*, 2022.

[2] C. K. Birdsall and A. B. Langdon. *Plasma physics via computer simulation*. CRC press, 2004.

[3] J. D. Carroll and J.-J. Chang. Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition. *Psychometrika*, 35(3):283–319, 1970.

[4] J. Dawson. Particle simulation of plasmas. *Rev. Mod. Phys.*, 55(2):403, 1983.

[5] L. De Lathauwer, B. De Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.*, 21(4):1253–1278, 2000.

[6] A. Dektor and D. Venturi. Dynamically orthogonal tensor methods for high-dimensional nonlinear pdes. *Journal of Computational Physics*, 404:109125, 2020.

[7] V. Ehrlacher and D. Lombardi. A dynamical adaptive tensor method for the Vlasov–Poisson system. *J. Comput. Phys.*, 339:285–306, 2017.

[8] L. Einkemmer and I. Joseph. A mass, momentum, and energy conservative dynamical low-rank scheme for the vlasov equation. *Journal of Computational Physics*, page 110495, 2021.

[9] L. Einkemmer and C. Lubich. A low-rank projector-splitting integrator for the Vlasov–Poisson equation. *SIAM J. Sci. Comput.*, 40(5):B1330–B1360, 2018.

[10] L. Einkemmer and C. Lubich. A quasi-conservative dynamical low-rank algorithm for the vlasov equation. *SIAM Journal on Scientific Computing*, 41(5):B1061–B1081, 2019.

[11] L. Einkemmer, A. Ostermann, and C. Piazzola. A low-rank projector-splitting integrator for the vlasov–maxwell equations with divergence correction. *J. Comput. Phys.*, 403:109063, 2020.

[12] F. Filbet and E. Sonnendrucker. Comparison of eulerian Vlasov solvers. *Computer Physics Communications*, 150(3):247–266, 2003.

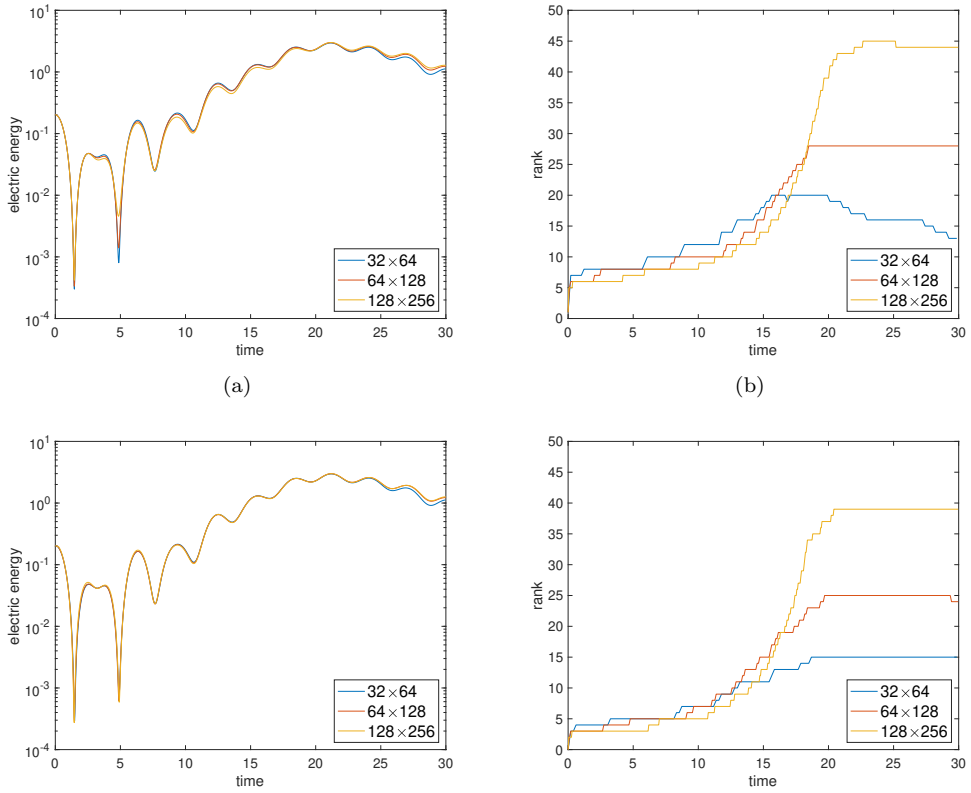


FIG. 5.8. *Example 5.3. The time evolution of the electric energy (a, c) and the ranks of the numerical solutions (b, d). Conservative method (a, b) and non-conservative method (c, d).* $\varepsilon = 10^{-4}$.

[13] S. Gottlieb, D. I. Ketcheson, and C.-W. Shu. *Strong stability preserving Runge-Kutta and multistep time discretizations*. World Scientific, 2011.

[14] L. Grasedyck. Hierarchical singular value decomposition of tensors. *SIAM J. Matrix Anal. Appl.*, 31(4):2029–2054, 2010.

[15] M. Griebel. A parallelizable and vectorizable multi-level algorithm on sparse grids. In W. Hackbusch, editor, *Parallel algorithms for partial differential equations*, volume 31 of *Notes on numerical fluid mechanics*, pages 94–100. 1991.

[16] W. Guo and Y. Cheng. A sparse grid discontinuous Galerkin method for high-dimensional transport equations and its application to kinetic simulations. *SIAM J. Sci. Comput.*, 38(6):A3381–A3409, 2016.

[17] W. Guo and J.-M. Qiu. A low rank tensor representation of linear transport and nonlinear vlasov solutions and their associated flow maps. *arXiv preprint arXiv:2106.08834*, 2021.

[18] W. Hackbusch. *Tensor spaces and numerical tensor calculus*, volume 42. Springer Science & Business Media, 2012.

[19] W. Hackbusch and S. Kühn. A new scheme for the tensor representation. *J. Fourier Anal. Appl.*, 15(5):706–722, 2009.

[20] R. A. Harshman et al. Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multimodal factor analysis. *UCLA Working Papers in Phonetics*, pages 1–84, 1970.

[21] F. L. Hitchcock. The expression of a tensor or a polyadic as a sum of products. *J. Math. Phys.*, 6(1-4):164–189, 1927.

[22] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Rev.*, 51(3):455–500, 2009.

[23] K. Kormann. A semi-Lagrangian Vlasov solver in tensor train format. *SIAM J. Sci. Comput.*, 37(4):B613–B632, 2015.

[24] K. Kormann and E. Sonnendrücker. Sparse grids for the Vlasov–Poisson equation. In *Sparse Grids and Applications-Stuttgart 2014*, pages 163–190. Springer, 2016.

[25] M. Kraus, K. Kormann, P. J. Morrison, and E. Sonnendrücker. GEMPIC: Geometric electromagnetic particle-in-cell methods. *J. Plasma Phys.*, 83(4), 2017.

[26] I. Oseledets. Tensor-train decomposition. *SIAM J. Sci. Comput.*, 33(5):2295–2317, 2011.

[27] I. V. Oseledets and S. V. Dolgov. Solution of linear systems and matrix inversion in the TT-format. *SIAM J. Sci. Comput.*, 34(5):A2718–A2739, 2012.

[28] I. V. Oseledets and E. E. Tyrtyshnikov. Breaking the curse of dimensionality, or how to use SVD in many dimensions. *SIAM J. Sci. Comput.*, 31(5):3744–3759, 2009.

[29] Z. Peng, R. G. McClarren, and M. Frank. A low-rank method for two-dimensional time-dependent radiation transport calculations. *Journal of Computational Physics*, 421:109735, 2020.

[30] S. Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. In *Dokl. Akad. Nauk SSSR*, volume 4, pages 240–243, 1963.

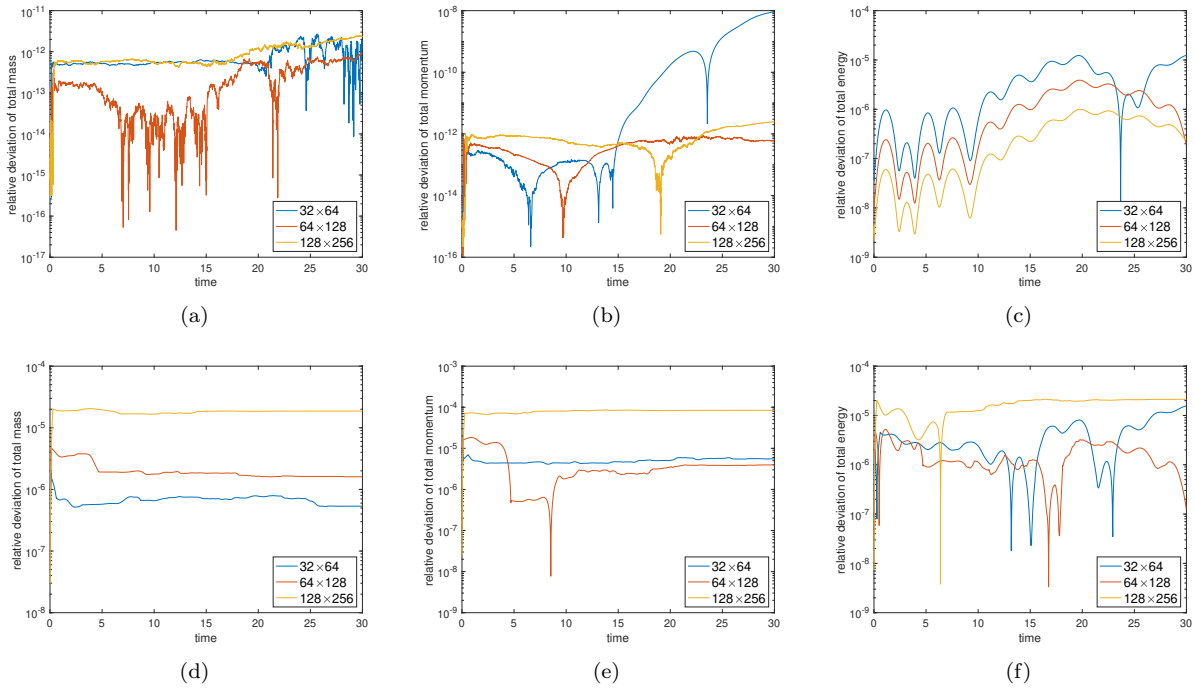
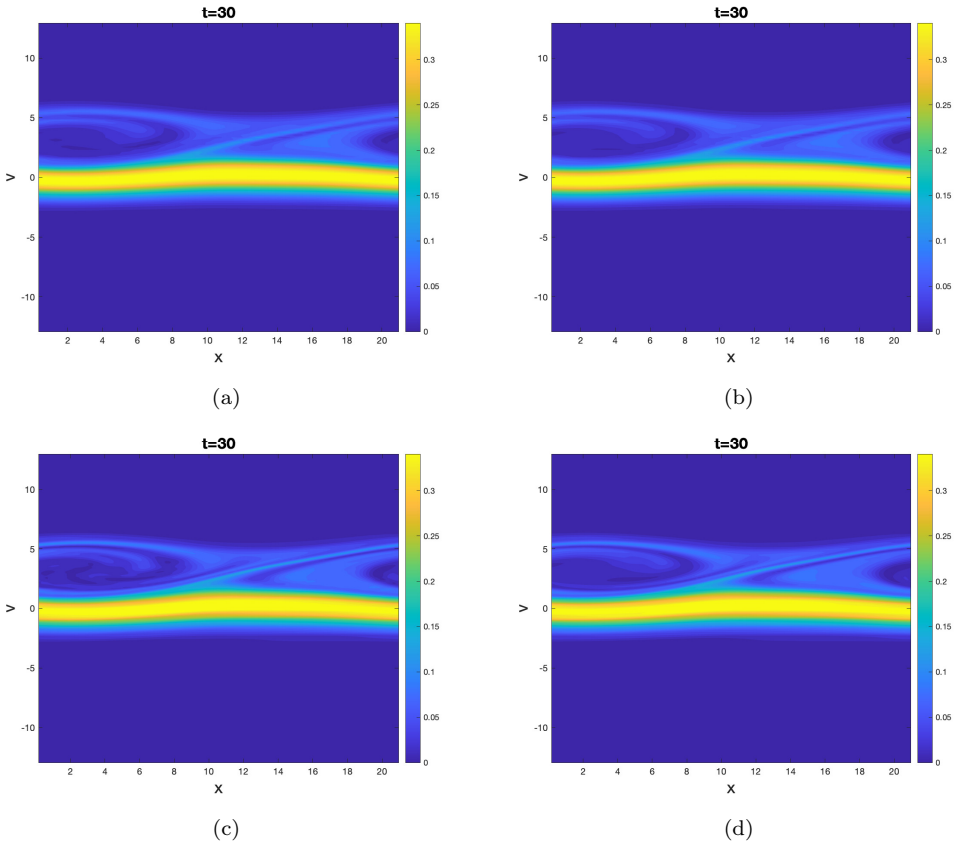


FIG. 5.9. *Example 5.3. The time evolution of relative deviation of total mass (a, d), relative deviation of total momentum (b, e), and relative deviation of total energy (c, f). Conservative method (a, b, c) and non-conservative method (d, e, f).* $\varepsilon = 10^{-4}$.

[31] Z. Tao, W. Guo, and Y. Cheng. Sparse grid discontinuous Galerkin methods for the Vlasov-Maxwell system. *J. Comput. Phys: X*, 3:100022, 2019.

[32] L. R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31(3):279–311, 1966.

[33] C. Zenger. Sparse grids. In *Parallel Algorithms for Partial Differential Equations, Proceedings of the Sixth GAMM-Seminar*, volume 31, 1990.



(a) (b) (c) (d)

FIG. 5.10. *Example 5.3. Contour plots of the solutions at* $t = 30$. *Conservative method with mesh* $N_x \times N_v = 64 \times 128$ *(a). Non-conservative method with mesh* $N_x \times N_v = 64 \times 128$ *(b). Conservative method with mesh* $N_x \times N_v = 128 \times 256$ *(c). Non-conservative method with mesh* $N_x \times N_v = 128 \times 256$ *(d).* $\varepsilon = 10^{-4}$.

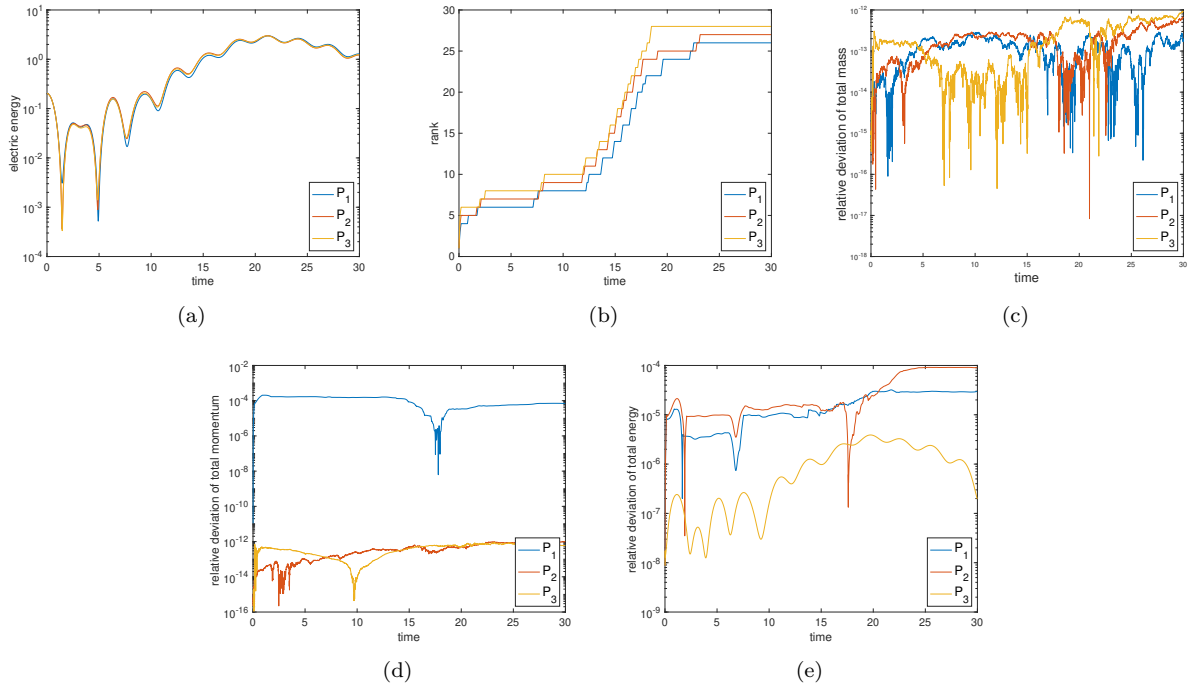


FIG. 5.11. *Example 5.3. Comparison of projections P_1, P_2, P_3 of the proposed conservative low rank method. The time evolution of electric energy (a), ranks of the numerical solutions (b), relative deviation of total mass (c), relative deviation of total momentum (e), and relative deviation of total energy (f). $N_x \times N_v = 64 \times 128$. $\varepsilon = 10^{-4}$.*

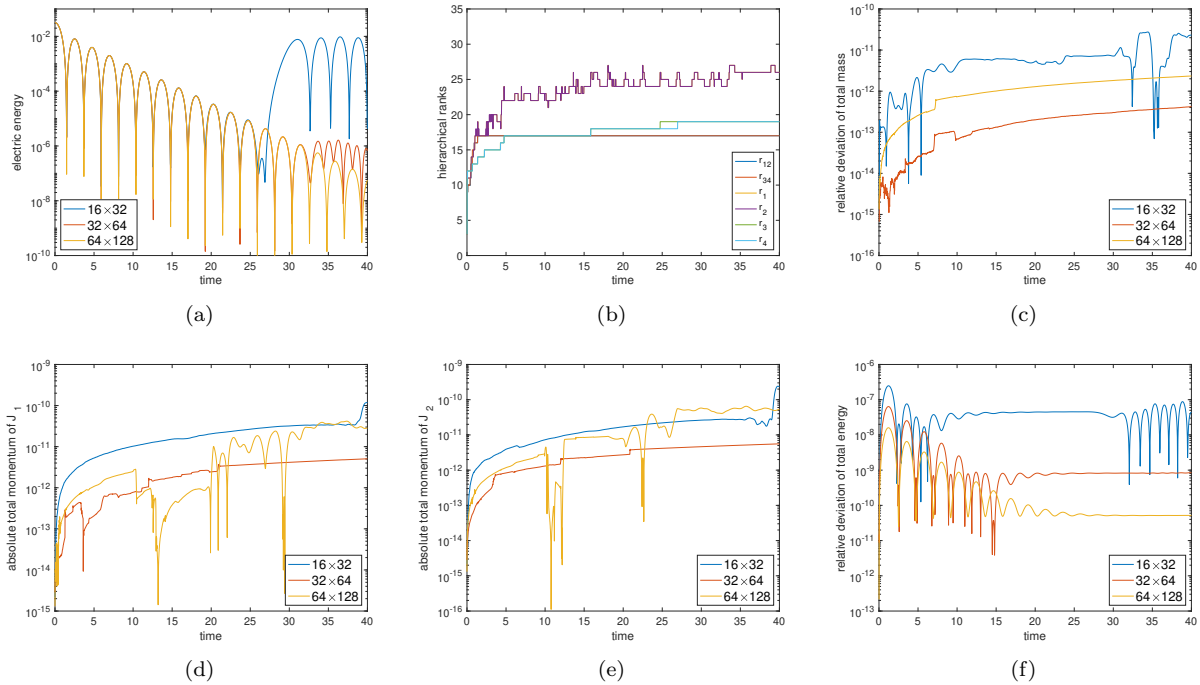


FIG. 5.12. *Example 5.4. Conservative low rank method. The time evolution of electric energy (a), hierarchical ranks of the xnumerical solution of mesh size $N_x \times N_v = 64 \times 128$ (b), relative deviation of total mass (c), absolute total momentum J_1 (d), absolute total momentum J_2 (e), and relative deviation of total energy (f). $\varepsilon = 10^{-5}$. In (b), r_{12} and r_{34} are close, r_1 and r_2 are close, and r_3 and r_4 are close.*

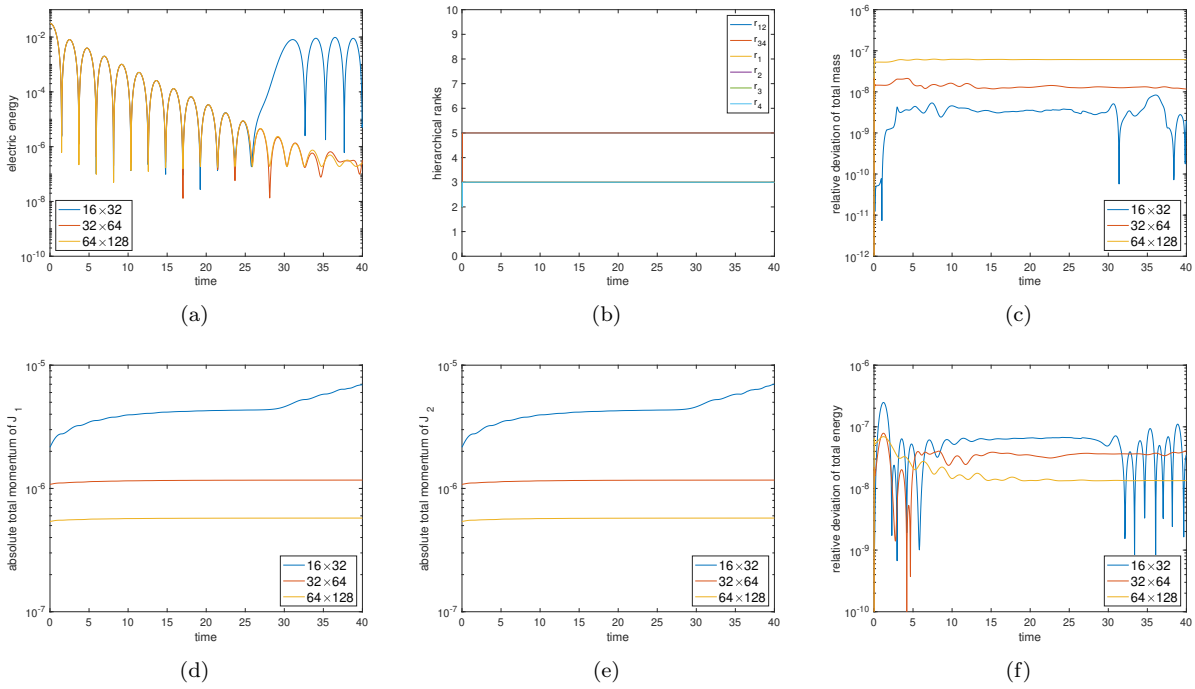


FIG. 5.13. *Example 5.4. Non-conservative low rank method. The time evolution of electric energy (a), hierarchical ranks of the numerical solution of mesh size $N_x \times N_v = 64 \times 128$ (b), relative deviation of total mass (c), absolute total momentum J_1 (d), absolute total momentum J_2 (e), and relative deviation of total energy (f). $\varepsilon = 10^{-5}$. In (b), r_{12} and r_{34} are close. r_1, r_2, r_3, and r_4 are close.*

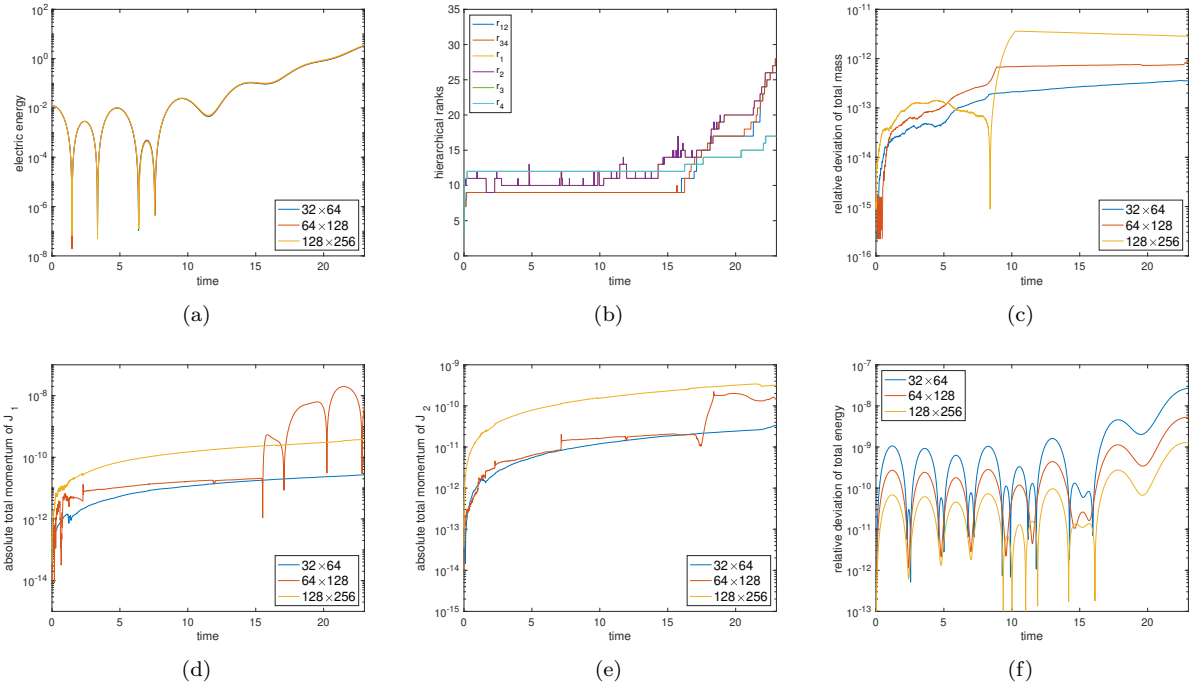


FIG. 5.14. *Example 5.5. Conservative low rank method. The time evolution of the electric energy (a), hierarchical ranks of the numerical solutions (b), relative deviation of total mass (c), absolute total momentum J_1 (d), absolute total momentum J_2 (e), and relative deviation of total energy (f). $\varepsilon = 10^{-5}$.*

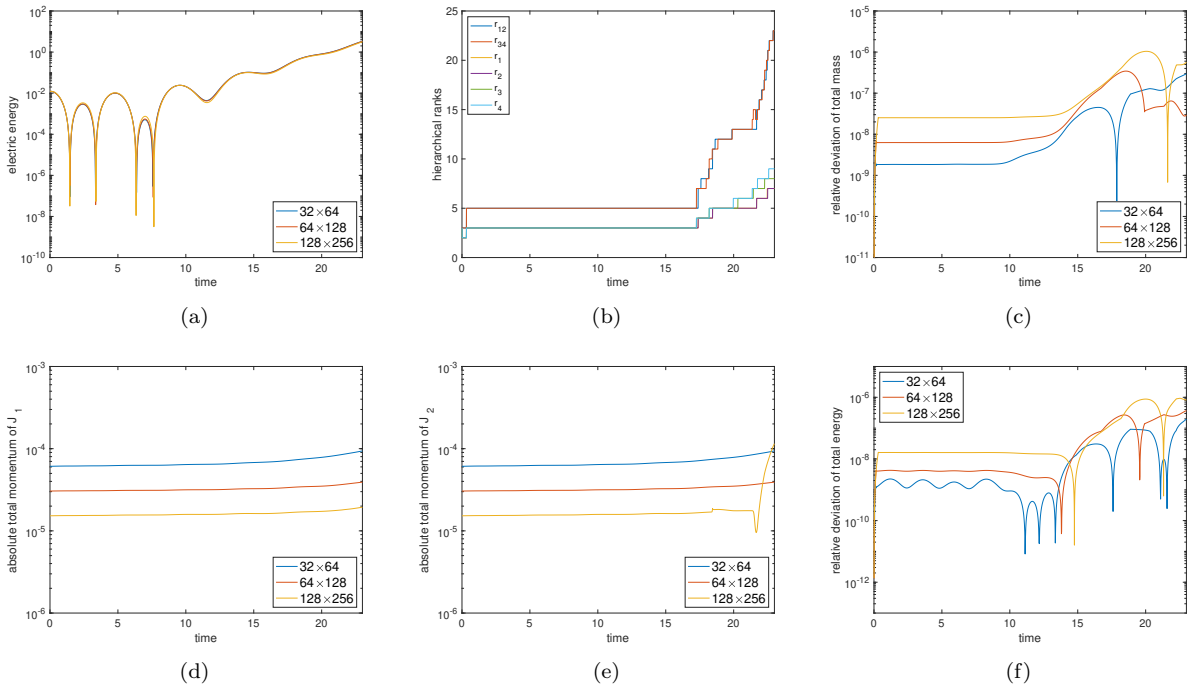


FIG. 5.15. *Example 5.5. Non-conservative low rank method. The time evolution of electric energy (a), hierarchical ranks of the numerical solutions (b), relative deviation of total mass (c), absolute total momentum J_1 (d), absolute total momentum J_2 (e), and relative deviation of total energy (f). $\varepsilon = 10^{-5}$.*

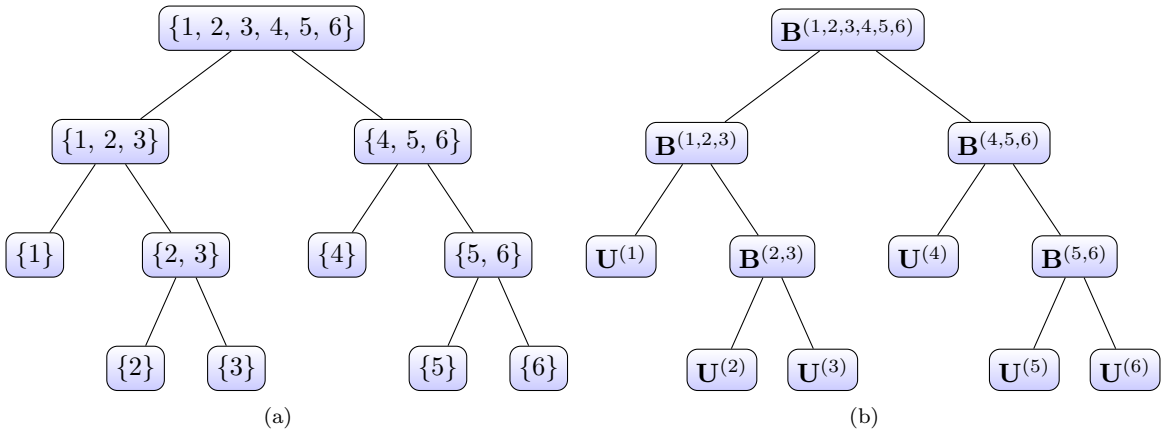


FIG. 5.16. *Dimension tree $\mathcal{T}$ to express 3D3V Vlasov solution tensors in the HT format.*

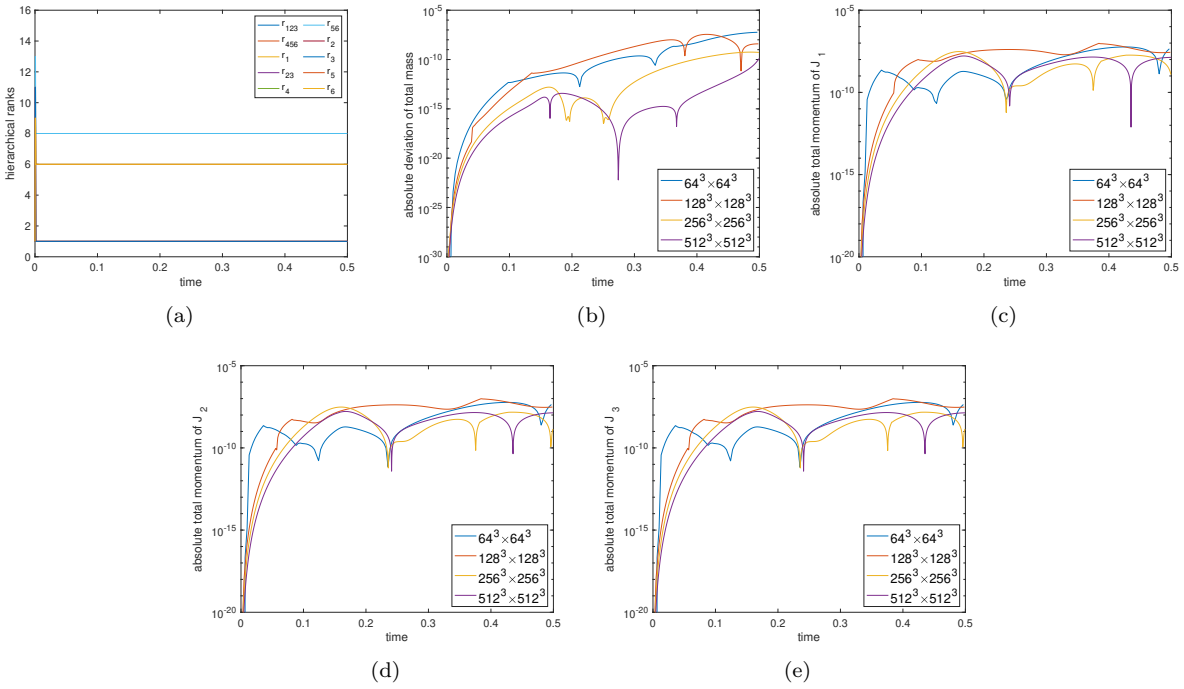


FIG. 5.17. *Example 5.6. Conservative low rank method. The time evolution of the hierarchical ranks of the numerical solution with mesh size* $512^3 \times 512^3$ *(a), absolute total mass (b), absolute total momentum* J_1 *(c),* J_2 *(d) and* J_3 *(e).* $\varepsilon = 10^{-4}$.