

Symplectic model reduction methods for the Vlasov equation

Tomasz M. Tyranowski^{1,2} | Michael Kraus^{1,2}

¹Max-Planck-Institut für Plasmaphysik, Garching, Germany

²Zentrum Mathematik, Technische Universität München, Garching, Germany

Correspondence

Tomasz M. Tyranowski,
Max-Planck-Institut für Plasmaphysik,
Boltzmannstraße 2, 85748 Garching,
Germany.
Email: tomasz.tyranowski@ipp.mpg.de

Funding information

Helmholtz-Gemeinschaft, Grant/Award
Number: ZT-I-0010

Abstract

Particle-based simulations of the Vlasov equation typically require a large number of particles, which leads to a high-dimensional system of ordinary differential equations. Solving such systems is computationally very expensive, especially when simulations for many different values of input parameters are desired. In this work, we compare several model reduction techniques and demonstrate their applicability to numerical simulations of the Vlasov equation. The necessity of symplectic model reduction algorithms is illustrated with a simple numerical experiment.

KEYWORDS

data-driven model reduction, geometric integration, particle-in-cell method, Vlasov equation

1 | INTRODUCTION

Kinetic models like the Vlasov equation provide the most accurate description for a plasma beyond the interaction of individual particles. While physically comprehensive, such models are often too expensive to be solved numerically under realistic conditions, especially in many-query contexts like uncertainty quantification, inverse problems or optimization. In such situations, reduced complexity models can provide a good compromise between computational cost and physical completeness of numerical models and facilitate the solution of problems that are otherwise unfeasible.

A common approach for the construction of reduced complexity models are modal decomposition techniques such as proper orthogonal decomposition (POD, see ref. [1]), also known as principal component analysis (PCA, see ref. [2]), and dynamic mode decomposition (DMD, see ref. [3]). Here, existing complex models are replaced by reduced models, which preserve the essential features of the original systems, but require less computational effort.

Algorithms obtained by such model order reduction techniques usually consist of two stages: an offline stage, where the reduced basis is constructed from empirical or simulation data of a physical system, referred to as snapshots, and an online stage, where the system is solved in the reduced basis. These techniques use singular-value decomposition (SVD) to identify dominant global modes in the snapshots, in such a way that the basis constructed from these modes spans the data optimally. Standard Galerkin projection methods are used to obtain approximate operators on such a basis.^[4] This approach is often amended by techniques such as the discrete empirical interpolation method to allow for efficient evaluation of non-linearities.^[5]

While model order reduction has been successfully applied to finite difference and finite volume discretizations of fluid equations (see ref. [6] for an overview), its applicability is not yet well studied for kinetic equations. Thus, the main goal of this work is to demonstrate the usefulness of model reduction techniques for numerical simulations of the Vlasov

equation using particle methods. Further, we will show that preserving the Hamiltonian structure of the system in the reduction procedure is key to obtaining stable and accurate reduced models.

1.1 | Particle methods for the Vlasov equation

In this work we consider the Vlasov equation

$$\frac{\partial f}{\partial t} + v \frac{\partial f}{\partial x} - E(x) \frac{\partial f}{\partial v} = 0, \quad (1)$$

for the particle density function $f = f(t, x, v)$ of plasma consisting of charged particles of unit mass and unit negative charge, where $E(x) = -\frac{\partial \varphi}{\partial x}$ is an external electrostatic field with the potential $\varphi = \varphi(x)$, and x and v are vectors in \mathbb{R}^d with $d = 1, 2, 3$. The standard approach to particle methods consists of the Ansatz

$$f(t, x, v) = \sum_{i=1}^n w_i \delta(x - X_i(t)) \delta(v - V_i(t)), \quad (2)$$

for the particle density function, where $X_i(t)$ and $V_i(t)$ represent the position and velocity of the i -th particle, respectively, and w_i its weight. Substituting Equation (2) in Equation (1), one obtains a system of ordinary differential equations (ODEs) for $X_i(t)$ and $V_i(t)$, namely

$$\dot{X}_i = V_i, \quad \dot{V}_i = \frac{\partial \varphi}{\partial x}(X_i), \quad i = 1, \dots, n. \quad (3)$$

It can be easily verified that Equation (3) has the form of a Hamiltonian system of equations

$$\dot{X}_i = \frac{\partial H}{\partial V_i}, \quad \dot{V}_i = -\frac{\partial H}{\partial X_i}, \quad i = 1, \dots, n, \quad (4)$$

with the Hamiltonian H given by

$$H(X, V) = \sum_{i=1}^n \left[\frac{1}{2} V_i^2 - \varphi(X_i) \right], \quad (5)$$

where $X = (X_1, \dots, X_n)$ and $V = (V_1, \dots, V_n)$ are vectors in \mathbb{R}^{nd} . Note that, for simplicity and brevity, we express the Hamiltonian in terms of the velocity vector V rather than the canonical conjugate momentum vector P , since for a fixed¹ external electric field and particles of unit mass we have $P_i = V_i$.

1.2 | Geometric integration

The Hamiltonian system (4) possesses several characteristic properties. Its flow $F_t : \mathbb{R}^{2nd} \rightarrow \mathbb{R}^{2nd}$ preserves the Hamiltonian, that is, the total energy, as well as the canonical symplectic form $\omega = \sum_{i=1}^n \sum_{j=1}^d dX_i^j \wedge dV_i^j$. The latter property expressed in terms of the standard basis for \mathbb{R}^{2nd} takes the form of the condition

$$(DF_t)^T \mathbb{J}_{2nd} DF_t = \mathbb{J}_{2nd}, \quad (6)$$

where DF_t denotes the Jacobi matrix of the flow map F_t , \mathbb{J}_{2nd} denotes the canonical symplectic matrix defined as

$$\mathbb{J}_{2nd} = \begin{pmatrix} 0 & \mathbb{I}_{nd} \\ -\mathbb{I}_{nd} & 0 \end{pmatrix}, \quad (7)$$

and \mathbb{I}_{nd} is the $nd \times nd$ identity matrix.^[8–10]

¹In the general case of self-consistent electromagnetic fields we have that $P_i = mV_i + qA(X_i, t)$, where A is the vector potential of the electromagnetic field, and m and q are the mass and charge of the particles.^[7,8]

In principle, general purpose numerical schemes for ODEs can be applied to Hamiltonian systems such as Equation (3). However, when simulating these systems numerically, it is advisable that the numerical scheme also preserves geometric features such as symplecticity (6). Geometric integration of Hamiltonian systems has been thoroughly studied (see ref. [9,11–23] and the references therein) and symplectic integrators have been shown to demonstrate superior performance in long-time simulations of such systems, compared with non-symplectic methods. Long-time accuracy and near preservation of the Hamiltonian by symplectic integrators have been rigorously studied using the so-called backward error analysis (see, e.g., ref. [9] and the references therein). Application of geometric integration to particle-in-cell (PIC) simulations of the Vlasov equation coupled to self-consistent electromagnetic fields satisfying the Maxwell equations (i.e., the Vlasov–Maxwell system) was proposed in ref. [24–28].

1.3 | Symplectic model reduction

For the aforementioned reasons it appears desirable to preserve the Hamiltonian structure also in model reduction. In fact, it has been found that preserving the Hamiltonian structure in the construction of the reduced spaces preserves stability,^[29] which is not guaranteed using non-structure-preserving model reduction techniques.^[30,31] To this end, standard model reduction techniques such as POD have been modified towards the so-called proper symplectic decomposition,^[32–34] which does indeed preserve the canonical symplectic structure of many Hamiltonian systems in the reduction procedure. Similarly, greedy algorithms^[29,35,36] can be used to construct the reduced basis in a Hamiltonian-structure preserving way, and recently also non-orthonormal bases have been considered,^[37] showing improved efficiency over orthonormal bases. See also ref. [38–41].

1.4 | Outline

The main content of the remainder of this paper is as follows. In Section 2, we review several model reduction techniques and set the appropriate notation. In Section 3, we present the results of our numerical experiment demonstrating the applicability of model reduction techniques to particle methods for the Vlasov equation. Section 4 contains the summary of our work.

2 | MODEL REDUCTION

In this section, we briefly review several model reduction techniques and set the notation appropriate for the problem defined in the introduction.

2.1 | Proper orthogonal decomposition

Proper orthogonal decomposition (POD) is one of the standard model reduction techniques.^[1,2] Consider a general ODE

$$\dot{u} = g(u), \quad \text{with } g : \mathbb{R}^N \rightarrow \mathbb{R}^N, \quad (8)$$

and with the initial condition $u(0) = u_0$. Equation (3) has this form with $N = 2nd$, $u = (X, V)$, and $g(u) = \left(V_1, \dots, V_n, \frac{\partial \varphi}{\partial x}(X_1), \dots, \frac{\partial \varphi}{\partial x}(X_n) \right)$. When N is a very high number, as is typical for particle methods, the system (8) becomes very expensive to solve numerically. The main idea of model reduction is to approximate such a high-dimensional dynamical system using a lower-dimensional one that can capture the dominant dynamic properties. Let Δ be an $N \times M$ matrix representing empirical data on the system (8). For instance, Δ can be a collection of snapshots of a solution of this system,

$$\Delta = [u(t_1) \ u(t_2) \ \dots \ u(t_M)], \quad (9)$$

at times t_1, \dots, t_M . These snapshots are calculated for some particular initial conditions or values of parameters that the system (8) depends on. A low-rank approximation of Δ can be done by performing a singular-value decomposition (SVD)

of Δ and truncating it after the first K largest singular values, that is,

$$\Delta = U\Sigma V^T \approx U_K \Sigma_K V_K^T, \quad (10)$$

where $\Sigma = \text{diag}(\sigma_1, \sigma_2, \dots)$ is the diagonal matrix of the singular values, U and V are orthogonal matrices, Σ_K is the diagonal matrix of the first K largest singular values, and U_K and V_K are orthogonal matrices constructed by taking the first K columns of U and V , respectively. Let ξ denote a vector in \mathbb{R}^K . Substituting $u = U_K \xi$ in (8) yields a reduced ODE for $\xi(t)$ as

$$\dot{\xi} = U_K^T g(U_K \xi), \quad (11)$$

with the projected initial condition $\xi(0) = \xi_0 = U_K^T u_0$. If the singular values of Δ decay sufficiently fast, then one can obtain a good approximation of Δ for K such that $K \ll N$ (see Section 2.3). Equation (11) is then a low-dimensional approximation of Equation (8) and can be solved more efficiently. For more details about the POD method, we refer the reader to ref. [1]. In the context of particle methods for the Vlasov equation, the reduced model (11) allows one to perform numerical computations with a much smaller number of degrees of freedom. Note, however, that while Equation (3) is a Hamiltonian system, there is no guarantee that the reduced model (11) will also have that property. In Section 3, we demonstrate that retaining the Hamiltonian structure in the reduced model greatly improves the quality of the numerical solution.

2.2 | Proper symplectic decomposition

Note that the Hamiltonian system (4) can be equivalently written as

$$\dot{u} = \mathbb{J}_{2N} \nabla_u H(u), \quad (12)$$

where $u = (X, V)$ and $N = nd$. A model reduction technique that retains the symplectic structure of Hamiltonian systems was introduced in ref. [34]. In analogy to POD, this method is called the proper symplectic decomposition (PSD). A $2N \times 2K$ matrix is called symplectic if it satisfies the condition

$$A^T \mathbb{J}_{2N} A = \mathbb{J}_{2K}. \quad (13)$$

For a symplectic matrix A , we can define its symplectic inverse $A^+ = \mathbb{J}_{2K}^T A^T \mathbb{J}_{2N}$. It is an inverse in the sense that $A^+ A = \mathbb{J}_{2K}$. Let ξ be a vector in \mathbb{R}^{2K} . Substituting $u = A\xi$ in (12) yields a reduced equation

$$\dot{\xi} = A^+ \mathbb{J}_{2N} \nabla_u H(u) = \mathbb{J}_{2K} \nabla_\xi H(A\xi), \quad (14)$$

which is a lower-dimensional Hamiltonian system with the Hamiltonian $\tilde{H}(\xi) = H(A\xi)$. Given a set of empirical data on a Hamiltonian system, the PSD method constructs a symplectic matrix A , which best approximates that data in a lower-dimensional subspace. We have tested two PSD algorithms, namely the cotangent lift and complex SVD algorithms.

2.2.1 | Cotangent lift algorithm

This algorithm constructs a symplectic matrix A , which has the special block diagonal structure

$$A = \begin{pmatrix} \Phi & 0 \\ 0 & \Phi \end{pmatrix}, \quad (15)$$

where Φ is an $N \times K$ matrix with orthogonal columns, that is, $\Phi^T \Phi = \mathbb{I}_K$. Suppose snapshots of a solution are given as an $N \times 2M$ matrix Δ of the form

$$\Delta = [X(t_1) \dots X(t_M) V(t_1) \dots V(t_M)]. \quad (16)$$

The SVD of Δ is truncated after the first K largest singular values, similar to Equation (10). The matrix Φ is then chosen as $\Phi = U_K$. More details can be found in ref. [34].

2.2.2 | Complex SVD algorithm

By allowing a broader class of symplectic matrices, we may get a better approximation. The complex SVD algorithm constructs a symplectic matrix of the form

$$A = \begin{pmatrix} \Phi & -\Psi \\ \Psi & \Phi \end{pmatrix}, \quad (17)$$

where Φ and Ψ are $N \times K$ matrices satisfying the conditions

$$\Phi^T \Phi + \Psi^T \Psi = \mathbb{I}_K, \quad \Phi^T \Psi = \Psi^T \Phi. \quad (18)$$

Suppose snapshots of a solution are given as an $N \times M$ complex matrix Δ of the form

$$\Delta = [X(t_1) + iV(t_1) \dots X(t_M) + iV(t_M)], \quad (19)$$

where i denotes the imaginary unit. The complex SVD of Δ is truncated after the first K largest singular values, that is,

$$\Delta = U \Sigma V^\dagger \approx U_K \Sigma_K V_K^\dagger. \quad (20)$$

The matrices Φ and Ψ are then chosen as the real and imaginary parts of U_K , respectively, that is, $U_K = \Phi + i\Psi$. More details can be found in ref. [34].

2.2.3 | Preservation of the Hamiltonian

Let $u(t)$ be a solution of Equation (12) with the initial condition u_0 , and let $\xi(t)$ be a solution of the reduced system (14) with the projected initial condition $\xi_0 = A^+ u_0$. Since the solutions of both the unreduced and reduced equations preserve their respective Hamiltonians, we have that the error of the Hamiltonian $\Delta H(t) = H(u(t)) - \tilde{H}(\xi(t))$ arising due to the symplectic projection is constant in time and equal to its initial value $\Delta H(0) = H(u_0) - \tilde{H}(\xi_0)$, whose magnitude can be controlled by choosing a sufficiently high K . If geometric integrators are used to solve the reduced model (14), then the numerical values of the reduced Hamiltonian \tilde{H} stay very close to the exact value $\tilde{H}(\xi_0)$, and therefore they also stay close to the exact energy $H(u_0)$ of the unreduced system (12).

2.3 | Approximation of data by linear subspaces

The POD and PSD methods described in Sections 2.1 and 2.2, respectively, rely on the assumption that the set of the empirical data $\Omega = \{u(t_1), \dots, u(t_M)\} \subset \mathbb{R}^N$ can be approximated well by a linear subspace of dimension $K < N$, otherwise these techniques do not bring computational savings. The so-called Kolmogorov n -width $d_K(\Omega)$ describes the error arising from the projection of Ω onto the best-possible subspace of \mathbb{R}^N of a given dimension K (see ref. [42]). It has been proven that for a class of problems written as parametrized PDEs, the Kolmogorov n -width decays exponentially fast, that is, $d_K(\Omega) = O(e^{-\gamma K})$ with some constant $\gamma > 0$.^[43,44] This extremely fast decay plays a critical role in any model reduction strategy based upon projecting to linear subspaces, since it allows one to select a low to moderate K to achieve small approximation errors. Such theoretical results have not yet been proven for particle-based simulations of the Vlasov equation. We will show by performing explicit numerical computations that for the example presented in Section 3 the evolution of the particles is indeed well approximated in a low-dimensional subspace. In case the empirical data do not appear to lie in a linear subspace, instead of the POD and PSD methods described above, one may apply online adaptive methods that update local reduced spaces depending on time,^[45,46] as well as a structure-preserving dynamic reduced

basis method for Hamiltonian systems.^[47] The application of the latter two techniques to the Vlasov equation will be investigated in a follow-up work.

3 | NUMERICAL EXPERIMENT

In this section, we present the results of a simple numerical experiment demonstrating the applicability of model reduction techniques to particle methods for the Vlasov equation.

3.1 | Initial and boundary conditions

We consider the Vlasov equation (1) on a one-dimensional ($d = 1$) spatial domain $-\infty \leq x \leq \infty$ with the initial condition

$$f(0, x, v) = f_0(x, v) = \frac{1}{\sqrt{2\pi\eta}} e^{-\frac{1}{2\eta^2}x^2} \left(\frac{1}{1+a} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}v^2} + \frac{a}{1+a} \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{1}{2\sigma^2}(v-v_0)^2} \right), \quad (21)$$

where the parameters are set as follows:

$$\eta = 10, \quad a = 0.3, \quad v_0 = 4, \quad \sigma = 0.5. \quad (22)$$

This is a ‘‘bump-on-tail’’ distribution in velocity space combined with a Gaussian distribution in position space. The initial conditions for the particle positions $X_i(0)$ and velocities $V_i(0)$ in Equation (3) are generated as random variables drawn from the probability distribution (21) using rejection sampling.

3.2 | Empirical data

Let the Vlasov equation (1) be parameter-dependent. For example, the external electric field may depend on some parameter β , that is, $E = E(x; \beta)$. Suppose we have the following computational problem: we would like to scan the domain of β , that is, compute the numerical solution of (1) for a large number of values of β . Given that in practical applications the system (3) is very high-dimensional, this task is computationally very intensive. Model reduction can alleviate this substantial computational cost. One can conduct full-scale computations using high-fidelity numerical methods only for a selected small number of values of β . These data can then be used to identify reduced models, as described in Section 2. The lower-dimensional equations (11) or (14) can then be solved more efficiently for other values of β , thus reducing the overall computational cost. For our simple experiment, we consider a linear external electric field, namely

$$E(x; \beta) = \beta^2 x, \quad (23)$$

where β is a real parameter. While this is a rather academic example, it allows an easy demonstration of how model reduction works in the case of particle methods for the Vlasov equation. With this electric field one can solve the Vlasov equation (1) using the method of characteristics to obtain the exact solution

$$f(t, x, v) = f_0 \left(x \cos \beta t - \frac{1}{\beta} v \sin \beta t, \beta x \sin \beta t + v \cos \beta t \right), \quad (24)$$

satisfying the initial condition (21). Moreover, the equation for the trajectories of the particles (3) is solved by

$$X_i(t) = \frac{1}{\beta} V_i(0) \sin \beta t + X_i(0) \cos \beta t, \quad V_i(t) = V_i(0) \cos \beta t - \beta X_i(0) \sin \beta t. \quad (25)$$

Instead of solving the full-scale system numerically, for convenience we used the exact solution (25) to generate the empirical data for the following six values of the parameter β :

$$\beta_1 = 5.95, \quad \beta_2 = 5.97, \quad \beta_3 = 5.99, \quad \beta_4 = 6.01, \quad \beta_5 = 6.03, \quad \beta_6 = 6.05. \quad (26)$$

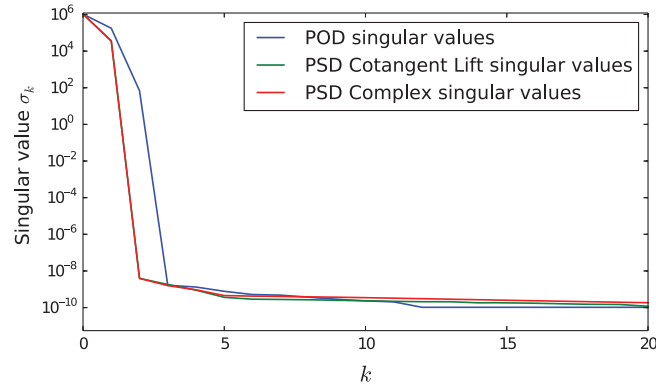


FIGURE 1 The decay of the singular values as calculated for the empirical data with each of the singular-value decomposition-based algorithms

The exact solution (25) was sampled for $n = 1000$ particles at times $t_k = k\Delta t$ for $\Delta t = 0.01$ and $k = 0, 1, \dots, 100,000$, that is, over the time interval $0 \leq t \leq 1000$. It should be noted that the same initial values for the positions and velocities of the particles were used for each value of β . The generated data for all values of β were put together and used to form the snapshot matrices (9), (16), and (19). For instance, for the POD snapshot matrix (9) we used

$$\Delta = [u(t_0; \beta_1) u(t_1; \beta_1) u(t_2; \beta_1) \dots u(t_0; \beta_2) u(t_1; \beta_2) u(t_2; \beta_2) \dots]. \quad (27)$$

Then, following the description of each algorithm in Section 2, reduced models were derived. The decay of the singular values for the POD, PSD cotangent lift and complex SVD methods is depicted in Figure 1. Since the sizes of the snapshot matrices Δ in our experiment were not exceedingly high (e.g., $2000 \times 600,006$ for (27)), we computed the full SVD decompositions using the standard SVD algorithm implemented in the Julia programming language. This algorithm requires memory and time that are superlinear in the dimensions of Δ ,^[48] which is prohibitive for very large data sets. However, the purpose of model reduction is to determine only a small number of the largest singular values and their corresponding singular vectors, therefore algorithms such as the truncated^[49] or the randomized^[50] SVD decompositions can be used instead. These algorithms require significantly less memory and time than the full SVD decomposition. Overall, this *offline* stage of model reduction is expensive, but it is performed only once. Then, in the *online* stage, the reduced systems can be solved cheaply for an arbitrary number of values of β .

3.3 | Reduced model simulations

To test the accuracy of the considered model reduction methods, we have compared the results of reduced model simulations to a full-scale reference solution. The reference solution for $\beta = 6.0$ was calculated on the time interval $0 \leq t \leq 1000$ in the same way as the empirical data in Section 3.2. Note that for this choice of β the period of the reference solution is $T \approx 1.05$ (see (25)), and the considered time interval encompasses roughly 955 periods. The reduced models were solved numerically on the same time interval using the second-order explicit and implicit midpoint methods. Note that when applied to a Hamiltonian system, the implicit midpoint method is a symplectic integrator, while the explicit midpoint method is not.^[9] All simulations were carried out with the time step $\Delta t = 0.0001$. The POD model (11) was solved for $K = 5$ and $K = 10$ (thus reducing the dimensionality of the problem from $2n = 2000$ to 5 and 10, respectively). Similarly, the PSD model (14) was solved for $K = 5$ and $K = 10$, both for the cotangent lift and complex SVD algorithms, in both cases reducing the dimensionality to 10 and 20, respectively. The choice of K is a compromise between the speed and the accuracy: the smaller K the faster the computation, but also the larger the projection error. In practice, one may choose K based on the initial value of the error (28), that is, the value of the projection error for the initial condition. For instance, in our experiment, the initial relative error for the POD simulations was equal to 1.09×10^{-14} for $K = 5$, and 7.81×10^{-15} for $K = 10$. All computations were performed in the Julia programming language with the help of the *GeometricIntegrators.jl* library.^[51] The three main conclusions from the numerical experiments are described below.

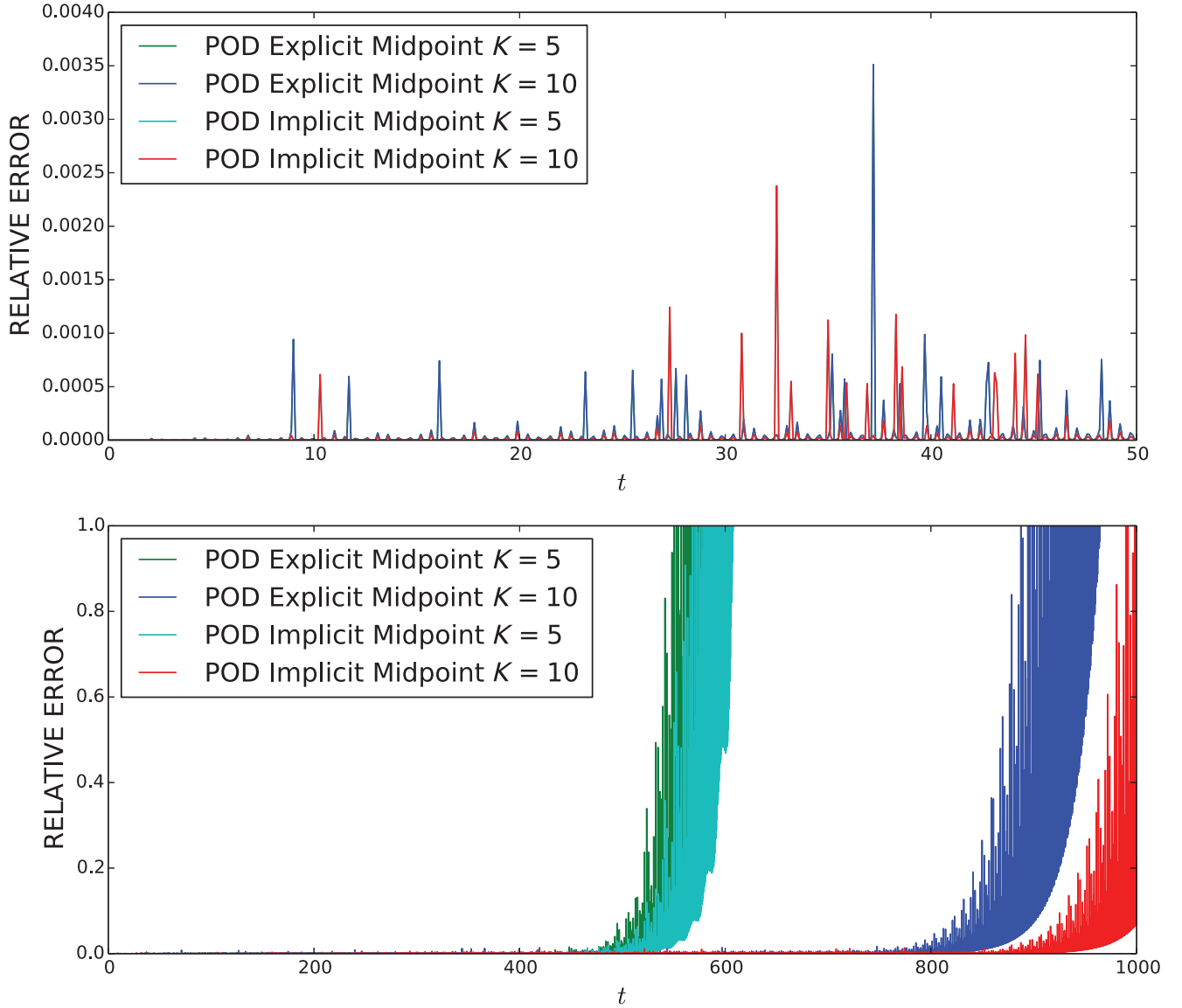


FIGURE 2 *Top:* The relative error of the proper orthogonal decomposition (POD) simulations on the time interval $0 \leq t \leq 50$. Both the explicit and implicit midpoint methods yield accurate approximations. Note that the plots for the explicit midpoint method with $K = 5$ and $K = 10$, as well as for the implicit midpoint method with $K = 5$ and $K = 10$, overlap very closely and are therefore indistinguishable. *Bottom:* The same plot over the whole simulation interval $0 \leq t \leq 1000$. It is evident that over a long integration time the errors blow up and both the explicit and implicit method simulations become unstable

3.4 | Long-time instability of the POD simulations

As a measure of accuracy of the reduced models we take the relative error

$$\frac{\|u(t) - u_{\text{ref}}(t)\|}{\|u_{\text{ref}}(t)\|}, \quad (28)$$

where u_{ref} is the reference solution of Equation (8), as described above, and $u = U_K \xi$ is the reconstructed solution, with ξ being the numerical solution of the reduced model (11). The relative error as a function of time is depicted in Figure 2. We see that the POD simulations give very accurate results on shorter time intervals, but the errors blow up over a long integration time, and both the explicit and implicit midpoint method simulations become unstable. This is a consequence

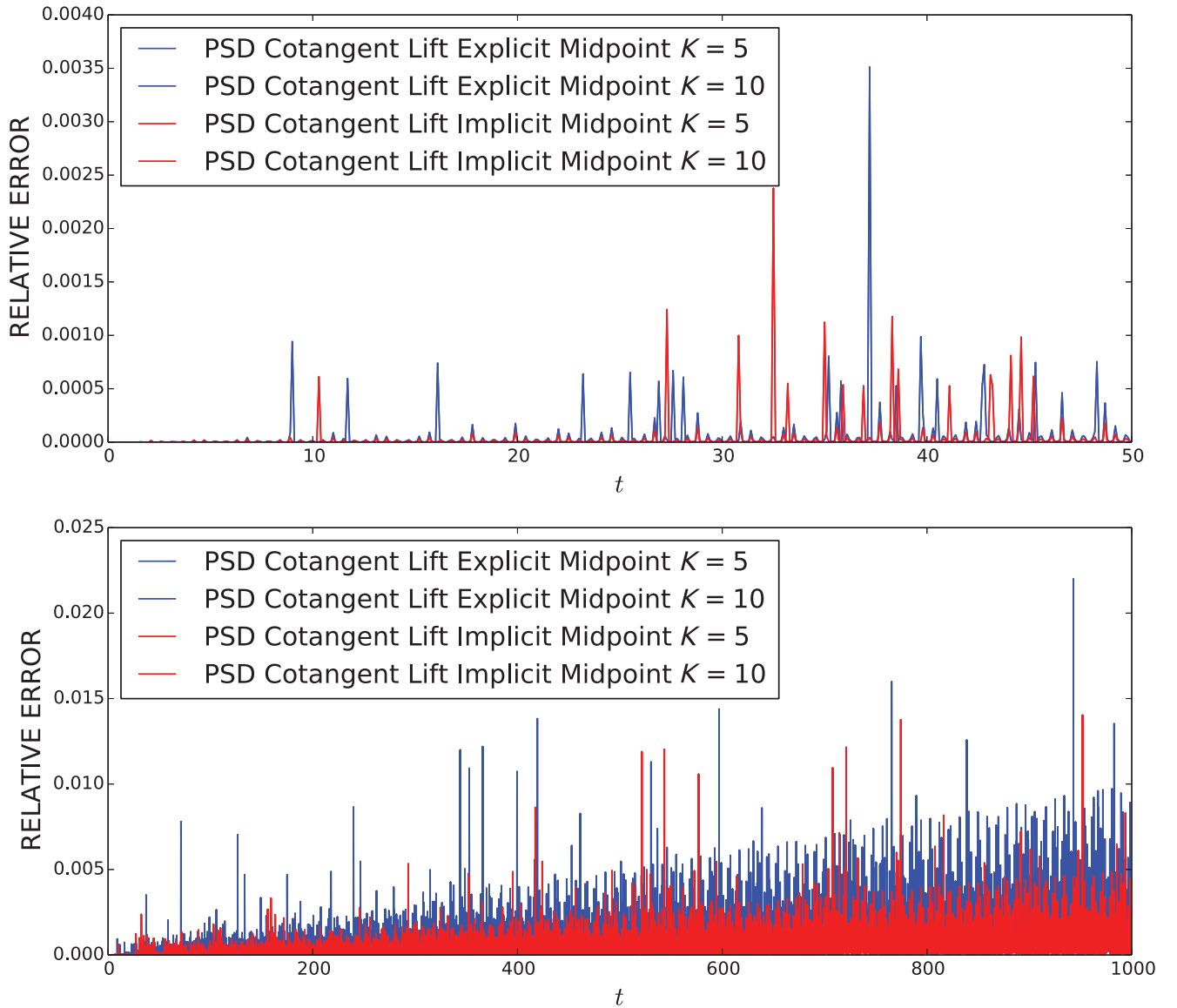


FIGURE 3 The relative error of the proper symplectic decomposition (PSD) simulations carried out with the cotangent lift algorithm on the time intervals $0 \leq t \leq 50$ (*Top*) and $0 \leq t \leq 1000$ (*Bottom*). The errors on the interval $0 \leq t \leq 50$ are essentially identical for the POD and PSD simulations, but the PSD simulations retain good accuracy and stability over the whole integration time. The results for the complex singular-value decomposition algorithm are nearly identical. Note that the same colour code is used for the plots that overlap very closely and are therefore indistinguishable

of the fact that there is no guarantee that the reduced system (11) retains any stability properties of the original system (8). In fact, the reduced equation for our example takes the form of the linear equation $\dot{\xi} = \Lambda_K \xi$, where the $K \times K$ matrix Λ_K is given by

$$\Lambda_K = U_K^T \begin{pmatrix} 0 & \mathbb{I}_n \\ -\beta^2 \mathbb{I}_n & 0 \end{pmatrix} U_K. \quad (29)$$

In our experiment, for $K = 5$ the matrix Λ_K has five eigenvalues with positive real parts, the largest one of which equals $\text{Re } \lambda \approx 0.0655$. Similarly, for $K = 10$, the eigenvalue with the largest real part is $\text{Re } \lambda \approx 0.0403$. The modes corresponding to these eigenvalues grow exponentially, and after a certain amount of time dominate the solution. This means that the reduced system is unstable; therefore, the errors arising from projecting the initial condition $\xi_0 = U_K^T u_0$ and from applying numerical integration schemes amplify over the simulation time, eventually leading to the observed loss of accuracy.^[30,31]

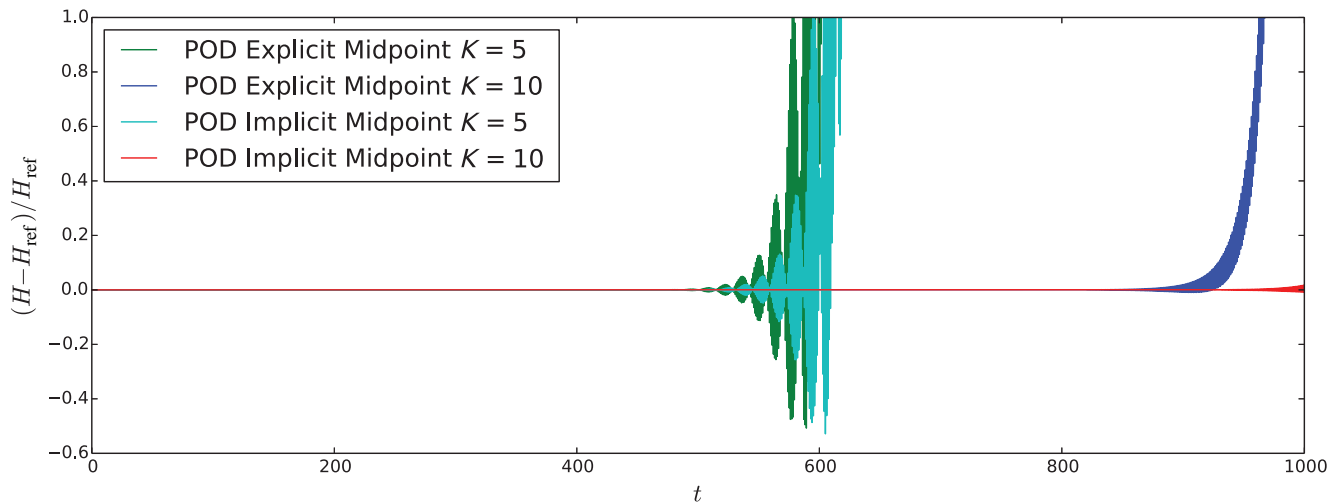


FIGURE 4 The relative error of the total energy of the particles as a function of time is depicted for the proper orthogonal decomposition simulations. The energy eventually blows up for both the explicit and implicit midpoint methods

3.5 | Long-time stability of the PSD simulations

In the case of the PSD models, the reconstructed solution $u = A\xi$ is obtained from the numerical solution ξ of the reduced Hamiltonian system (14). The relative error as a function of time for the cotangent lift algorithm is depicted in Figure 3. We see that both the explicit and implicit midpoint methods retain good accuracy and stability over the whole integration time. It is therefore evident that by preserving the Hamiltonian structure of the particle equations, symplectic model reduction significantly improves the stability of the numerical computations even if a non-symplectic integrator (here the explicit midpoint method) is used. The numerical results for the complex SVD algorithm are nearly identical, therefore for brevity and clarity we skip presenting a separate figure.

3.6 | Long-time energy behaviour

As the particle equations (4) are Hamiltonian, the total energy (5) of the particles should be preserved. In our numerical experiment, the Hamiltonian for the reference solution (25) with $\beta = 6.0$ was $H_{\text{ref}} \approx 1.803 \times 10^6$. The relative error of the total energy of the particles for each of the algorithms is depicted in Figures 4 and 5. One can clearly see that while the POD simulations initially retain the total energy relatively well, there is an evident linear growth trend for both the explicit and implicit midpoint methods, and the energy eventually blows up over a long integration time. On the other hand, the energy behaviour for the PSD simulations is more stable. The non-symplectic explicit midpoint method applied to the cotangent lift algorithm also shows the same linear growth trend, but does not blow up. Furthermore, the implicit midpoint method, which is symplectic in this case, demonstrates near preservation of the total energy, with only a minor linear growth throughout the whole simulation time. This demonstrates another important advantage of symplectic model reduction: since the reduced equations (14) are also Hamiltonian, one can employ symplectic time integrators to obtain a numerical solution that nearly conserves energy.

4 | SUMMARY AND FUTURE WORK

We have compared several model reduction techniques and demonstrated their usefulness for particle-based simulations of the Vlasov equation. We have pointed out the importance of retaining the Hamiltonian structure of the equations governing the evolution of particles. Our work can be extended in several directions. First, model reduction methods can be applied to the Vlasov equation coupled to a self-consistent electric field satisfying the Poisson equation, or electromagnetic

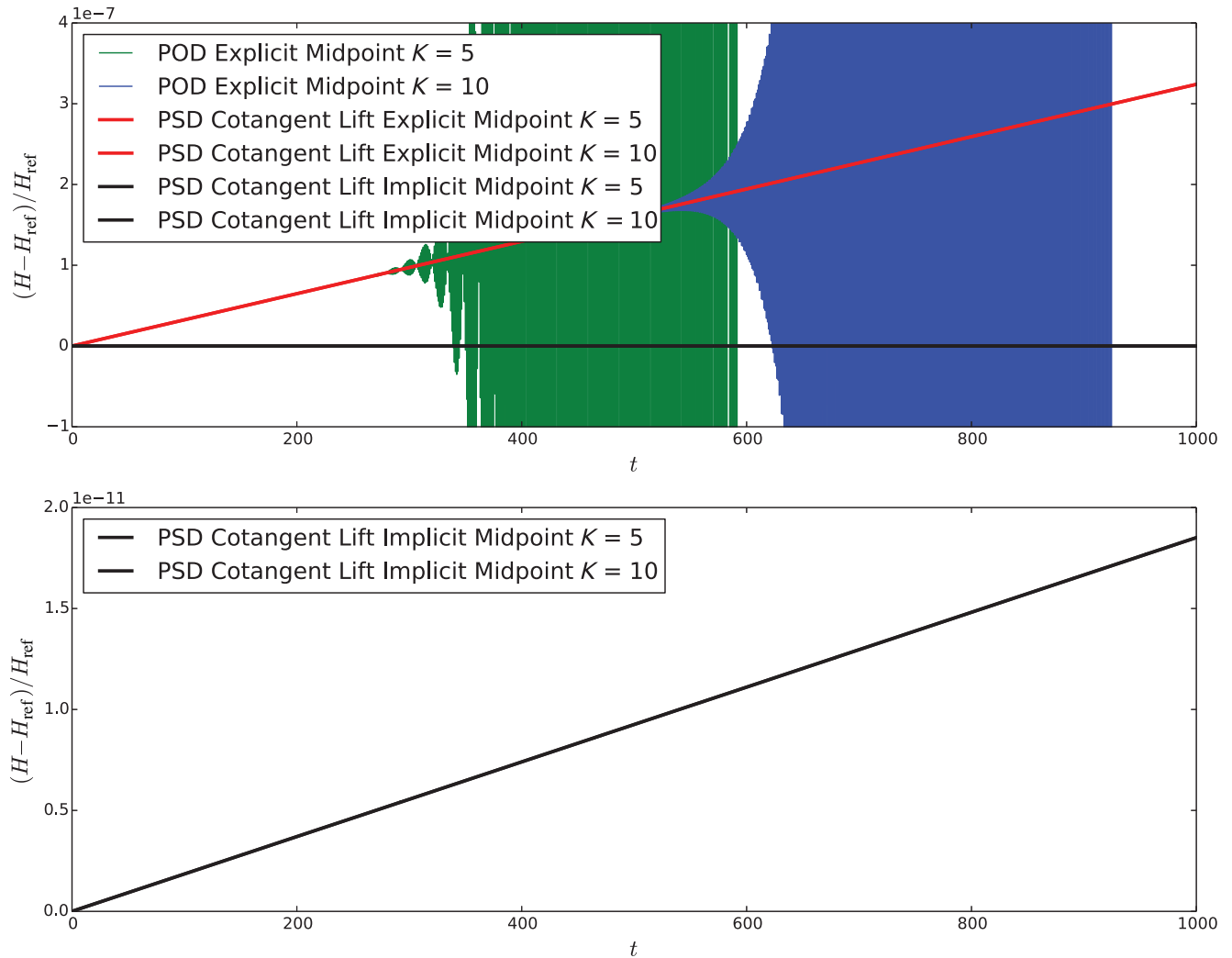


FIGURE 5 *Top:* The relative error of the total energy of the particles as a function of time is depicted for the proper symplectic decomposition simulations. The energy behaviour is more stable than for the proper orthogonal decomposition (POD) simulations. While the non-symplectic explicit midpoint method shows a linear growth trend, the symplectic implicit midpoint method appears to nearly preserve the total energy. For comparison, also two POD simulations are depicted. *Bottom:* Same plot, but only the simulations for the symplectic implicit midpoint method are depicted in order to show the scale of energy preservation. In fact these simulations also show a linear trend, which is nevertheless four orders of magnitude smaller than for the non-symplectic explicit midpoint method. Note that the same colour code is used for the plots that overlap very closely and are therefore indistinguishable. The results for the complex singular-value decomposition algorithm are nearly identical, and are therefore omitted for clarity

fields satisfying the Maxwell equations.^[24] Second, model reduction of collisional Vlasov equations stemming from metriplectic brackets^[52] or stochastic action principles^[53–55] would be an interesting and useful extension of the work presented here.

ACKNOWLEDGMENTS

We would like to thank Christopher Albert, Tobias Blickhan, Eric Sonnendrücker and Udo von Toussaint for useful comments and references. The study is a contribution to the Reduced Complexity Models grant number ZT-I-0010 funded by the Helmholtz Association of German Research Centers. Open access funding enabled and organized by Projekt DEAL.

CONFLICT OF INTEREST

The authors declare no potential conflict of interest.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

- [1] A. C. Antoulas, D. C. Sorensen, S. Gugercin, *Contemp. Math.* **2001**, 280, 193.
- [2] B. Moore, *IEEE Trans. Autom. Control* **1981**, 26(1), 17.
- [3] A. Alla, J. N. Kutz, *SIAM J. Sci. Comput.* **2017**, 39, B778.
- [4] K. Kunisch, S. Volkwein, *SIAM J. Numer. Anal.* **2002**, 40, 492.
- [5] S. Chaturantabut, D. C. Sorensen, *SIAM J. Sci. Comput.* **2010**, 32, 2737.
- [6] T. Lassila, A. Manzoni, A. Quarteroni, G. Rozza, In *Reduced Order Methods for Modeling and Computational Reduction* (Eds: A. Quarteroni, G. Rozza), Springer International Publishing, Cham **2014**, p. 235.
- [7] J. Jackson, *Classical Electrodynamics*, 3rd ed., Vol. 67, John Wiley & Sons, New York **1999**, p. 841.
- [8] J. Marsden, T. Ratiu, *Introduction to Mechanics and Symmetry*, volume 17 of *Texts in Applied Mathematics*, Springer Verlag, New York, NY **1994**.
- [9] E. Hairer, C. Lubich, G. Wanner, *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*. Springer Series in Computational Mathematics, Springer, New York **2002**.
- [10] D. D. Holm, T. Schmah, C. Stoica, *Geometric Mechanics and Symmetry: From Finite to Infinite Dimensions*. Oxford Texts in Applied and Engineering Mathematics, Oxford University Press, Oxford **2009**.
- [11] J. Hall, M. Leok, *Numer. Math.* **2015**, 130, 681.
- [12] C. Kane, J. E. Marsden, M. Ortiz, M. West, *Int. J. Numer. Methods Eng.* **2000**, 49(10), 1295.
- [13] M. Kraus, *Variational integrators in plasma physics*. PhD Thesis, Technische Universität München, **2013**.
- [14] M. Leok, J. Zhang, *IMA J. Numer. Anal.* **2011**, 31, 1497.
- [15] J. E. Marsden, G. W. Patrick, S. Shkoller, *Commun. Math. Phys.* **1998**, 199, 351.
- [16] J. E. Marsden, M. West, *Acta Numer.* **2001**, 10(1), 357.
- [17] R. I. McLachlan, G. R. W. Quispel, *J. Phys. A: Math. General* **2006**, 39(19), 5251.
- [18] S. Ober-Blöbaum, N. Saake, *Adv. Comput. Math.* **2015**, 41, 955.
- [19] C. W. Rowley, J. E. Marsden. presented at 2002, *Proc. 41st IEEE Conf. Decision Control*, Vol. 2 (pp. 1521–1527), **2002**.
- [20] J. M. Sanz-Serna, *Acta Numer.* **1992**, 1, 243.
- [21] T. M. Tyranowski. Geometric integration applied to moving mesh methods and degenerate Lagrangians. PhD thesis, California Institute of Technology, **2014**.
- [22] T. M. Tyranowski, M. Desbrun, *Mathematics* **2019**, 7(7), 642.
- [23] T. M. Tyranowski, M. Desbrun, *Mathematics* **2019**, 7(9), 861.
- [24] M. Kraus, K. Kormann, P. Morrison, E. Sonnendrücker, *J. Plasma Phys.* **2017**, 83, 905830401.
- [25] H. Qin, J. Liu, J. Xiao, R. Zhang, Y. He, Y. Wang, Y. Sun, J. W. Burby, L. Ellison, Y. Zhou, *Nucl. Fusion* **2016**, 56(1), 014001.
- [26] J. Xiao, J. Liu, H. Qin, Z. Yu, *Phys. Plasmas* **2013**, 20(10), 102517.
- [27] J. Xiao, H. Qin, J. Liu, *Plasma Sci. Technol.* **2018**, 20(11), 110501.
- [28] J. Xiao, H. Qin, J. Liu, Y. He, R. Zhang, Y. Sun, *Phys. Plasmas* **2015**, 22, 112504.
- [29] B. Afkham, J. Hesthaven, *SIAM J. Sci. Comput.* **2017**, 39, A2616.
- [30] S. Prajna, presented at *42nd IEEE Int. Conf. Decision Control (IEEE Cat. No.03CH37475)*, Vol. 5 (pp. 5254–5258), **2003**.
- [31] M. Rathinam, L. R. Petzold, *SIAM J. Numer. Anal.* **2003**, 41, 1893.
- [32] L. Peng, K. Mohseni, presented at *2016 IEEE 55th Conf. Decision Control (CDC)* (pp. 7465–7470), **2016**.
- [33] L. Peng, K. Mohseni. Unpublished, arXiv:1603.03514, **2016**.
- [34] L. Peng, K. Mohseni, *SIAM J. Sci. Comput.* **2016**, 38(1), A1.
- [35] B. M. Afkham, A. Bhatt, B. Haasdonk, J. S. Hesthaven, Unpublished, arXiv:1803.07799, **2018**.
- [36] B. M. Afkham, J. S. Hesthaven, *J. Sci. Comput.* **2019**, 81, 3.
- [37] P. Buchfink, A. Bhatt, B. Haasdonk, *Math. Comput. Appl.* **2019**, 24, 43.
- [38] K. Carlberg, R. Tuminaro, P. Boggs, *SIAM J. Sci. Comput.* **2015**, 37, B153.
- [39] S. Chaturantabut, C. Beattie, S. Gugercin, *SIAM J. Sci. Comput.* **2016**, 38, B837.
- [40] J. Hesthaven, C. Pagliantini, N. Ripamonti, Unpublished, arXiv:2109.12367, **2021**.
- [41] S. Lall, P. Krysl, J. E. Marsden, *Phys. D: Nonlinear Phenom.* **2003**, 184(1), 304.
- [42] A. Pinkus, Folge, Vol. 3, Springer-Verlag, Berlin, Germany **1985**.
- [43] A. Buffa, Y. Maday, A. T. Patera, C. Prud'homme, G. Turinici, *ESAIM: M2AN* **2012**, 46, 595.
- [44] M. Ohlberger, S. Rave, presented at *Proc. Conf. Algoritmy* (pp. 1–12), **2016**.
- [45] K. Carlberg, *Int. J. Numer. Methods Eng.* **2015**, 102, 1192.
- [46] B. Peherstorfer, K. Willcox, *SIAM J. Sci. Comput.* **2015**, 37, A2123.
- [47] C. Pagliantini, *Numer. Math.* **2021**, 148, 409.
- [48] P. Drineas, R. Kannan, M. W. Mahoney, *SIAM J. Comput.* **2006**, 36(1), 158.
- [49] T. F. Chan, P. C. Hansen, *SIAM J. Sci. Stat. Comput.* **1990**, 11, 519.
- [50] N. Halko, P. G. Martinsson, J. A. Tropp, *SIAM Rev.* **2011**, 53, 217.

- [51] M. Kraus, T. Tyranowski, C. Albert, C. Rackauckas. DDMGNI/GeometricIntegrators.jl: v0.2.0, **2020**. <https://doi.org/10.5281/zenodo.3648326>.
- [52] E. Hirvijoki, M. Kraus, J. W. Burby, Preprint arXiv:1802.05263, **2018**.
- [53] M. Kraus, T. M. Tyranowski, *IMA J. Numer. Anal.* **2020**, *41*, 1318.
- [54] T. M. Tyranowski, *Proc. R. Soc. A: Math. Phys. Eng. Sci.* **2021**, *477*, 20210167.
- [55] T. M. Tyranowski, Preprint arXiv:2201.13391, **2022**.

How to cite this article: T. M. Tyranowski, M. Kraus, *Contributions to Plasma Physics*. **2023**, *63*(5-6), e202200046.
<https://doi.org/10.1002/ctpp.202200046>