Accelerating Kinetic Simulations of Electrostatic Plasmas with Reduced-Order Modeling

Ping-Hsuan Tsai
Department of Computer Science
University of Illinois Urbana–Champaign
Urbana, IL 61801
pht2@illinois.edu

Seung Whan Chung
Lawrence Livermore National Laboratory
Livermore, CA 94550
chung28@llnl.gov

Debojyoti Ghosh
Lawrence Livermore National Laboratory
Livermore, CA 94550
ghosh5@llnl.gov

John Loffeld
Lawrence Livermore National Laboratory
Livermore, CA 94550
loffeld1@llnl.gov

Youngsoo Choi
Lawrence Livermore National Laboratory
Livermore, CA 94550
choi15@llnl.gov

Jonathan L. Belof
Lawrence Livermore National Laboratory
Livermore, CA 94550
belof1@llnl.gov

Abstract

Despite the advancements in high-performance computing and modern numerical algorithms, the cost remains prohibitive for multi-query kinetic plasma simulations. In this work, we develop data-driven reduced-order models (ROM) for collisionless electrostatic plasma dynamics, based on the kinetic Vlasov-Poisson equation. Our ROM approach projects the equation onto a linear subspace defined by principal proper orthogonal decomposition (POD) modes. We introduce an efficient tensorial method to update the nonlinear term using a precomputed third-order tensor. We capture multiscale behavior with a minimal number of POD modes by decomposing the solution into multiple time windows using a physical-time indicator and creating a temporally-local ROM. Applied to 1D–1V simulations, specifically the benchmark two-stream instability case, our time-windowed reduced-order model (TW–ROM) with the tensorial approach solves the equation approximately 280 times faster than Eulerian simulations while maintaining a maximum relative error of 4% for the training data and 13% for the testing data.

1 Introduction

Kinetic modeling of collisionless electrostatic plasma relies on the Vlasov–Poisson equations that describe charged particle distribution under self-consistent electrostatic fields. Due to high dimensionality, scale disparities, and nonlinearity, solving these equations is computationally challenging. Lagrangian particle-in-cell (PIC) methods [1] are often used, where particles are advanced along the characteristic curves of the Vlasov equation [2]. An alternative is the grid–based Eulerian methods that allow the use of advanced numerical algorithms for partial differential equations (PDEs) [3–5]. In this paper, we consider the latter as full-order models (FOMs). While high-performance computing and advanced algorithms enable high-fidelity kinetic plasma simulations, they remain intractable for parametric studies. In this work, we propose to address this by data-driven projection-based reduced-order models, which have been successfully applied to many physical simulations, e.g., fluid

Machine Learning and the Physical Sciences Workshop, NeurIPS 2023.

dynamics [6–13], nonlinear diffusion [14], Boltzmann transport [15], design optimization [16–19]. Starting with high-resolution Eulerian simulations, we derive a low-dimensional surrogate model that is solved at a reduced computational cost while providing approximate solutions with acceptable accuracy.

We consider the 1D–1V parametric Vlasov–Poisson equation discretized with a high-order conservative finite difference method. Traditional reduced-order modeling faces challenges of stability, efficiency, and accuracy due to the problem's multi-scale nature, limiting its benefits. We overcome this by partitioning the solution into multiple time windows and creating temporally-local ROMs [20–24]. The primary computational expense arises from the nonlinear hyperbolic term evaluation in the high-dimensional space. To improve efficiency, we introduce a tensorial approach for updating this term using a precomputed third-order tensor. Our numerical experiments demonstrate the efficacy of the proposed time-windowed ROM (TW–ROM) with the tensorial approach, and we achieve significant speed improvements.

2 Reduced-order modeling of the 1D–1V Vlasov–Poisson equations

In Section (2.1), we introduce the parametric 1D–1V Vlasov–Poisson equation and the initial and boundary conditions that give rise to the two-stream instability. In Section (2.2), we briefly introduce the full-order model. In Section (2.3), we present the framework for the time-windowing reduced-order model and the tensorial approach. The reduced order model is constructed in the offline phase and deployed in the online phase.

2.1 Parametric 1D–1V Vlasov–Poisson equations

We consider the parametric 1D–1V Vlasov–Poisson equations,

$$\begin{cases} \partial_t f(x,v,t) + v\partial_x f(x,v,t) + E(x,t)\,\partial_v f(x,v,t) = 0, \\ \partial_x^2 \phi(x,t) = \int f(x,v,t)\ dv \quad \int\int f(x,v,t)\ dv\,dx, \\ f(x,v,0) = \rho\frac{8}{2\pi T}\left(1+\alpha\cos\left(2k\pi\frac{x}{L}\right)\right)\left[\exp\left(\frac{(v\quad v_0)^2}{2T}\right)+\exp\left(\frac{(v+v_0)^2}{2T}\right)\right]. \end{cases} \tag{1}$$

Here $f(x,v,t)$ is the plasma distribution function, $\phi(x,t)$ is the electrostatic potential, and $E(x,t)\quad\partial_x\phi(x,t)$ is the electric field. The spatial and velocity coordinates are x and v, respectively. The simulation time interval is $[0,\ t_f]$ where $t_f\ 2\ \mathsf{R}_+$ is the final time, and the phase-space domain is $(x,v)\ 2\ \Omega := [0,2\pi]\quad[\ \ 3.5v_0, 3.5v_0]$ with periodic boundaries in x and homogeneous Dirichlet boundaries in v. The initial distribution $f(x,v,0)$ is characterized by three parameters: the temperature T, the perturbation amplitude α, and the initial stream velocity v_0 with $k=1$ and $L=2\pi$. We focus on studying the effect of parameterized initial distributions on the plasma dynamics.

2.2 Full-order model (FOM)

The semi-discretization of Eq. 1 in space can be written as:

$$\frac{d\mathbf{f}(t;\ \ )}{dt} = \mathbf{G}(\mathbf{f},t;\ \ ),\quad t\ 2\ [0,\ t_f],\quad \text{with}\quad \mathbf{f}(0;\ \ ) = \mathbf{f}_0(\ \ ), \tag{2}$$

where $\quad = (T,\alpha,v_0)$ is the parameter in a parameter domain D; $\mathbf{f}\ 2\ \mathsf{R}^{N_f}$ denotes the parameterized time-dependent solution to the dynamical system with an initial state $\mathbf{f}_0$ and N_f is the FOM degrees of freedoms; $\mathbf{G}$ represents the nonlinear matrix function coming from the hyperbolic term in (1).

We discretize Eq. 1 in space with a high-order conservative finite-difference code on a Cartesian grid. The domain Ω is discretized on a $256\quad 256$ grid leads to $N_f = 65,536$. The spatial derivatives are computed using the fifth–order weighted essentially nonoscillatory (WENO) scheme [25]. The electrostatic potential is computed by solving the periodic Poisson equation with the fast Fourier transformation (FFT). The classical four-stage fourth-order explicit Runge–Kutta time integrator is used to evolve the resulting ordinary differential equation (ODE), given by Eq. 2, with a uniform time step of $\Delta t = 0.0025$.

2.3 Time-windowing reduced-order model (TW-ROM)

The formation of the two-stream instability occurs due to the nonlinear evolution of (1). This instability is a well-known phenomenon in plasma physics, which is generated by two counter-streaming beams. In this process, the kinetic energy of particles excites a plasma wave, which then transfers to electrostatic potential energy [26]. The main interest of numerical studies in the two-stream instability is to investigate how the parameter affects the solutions and the growth rate of instability. The primary difficulty for the ROM are the three stages of the solutions, i.e., short transient, growth, and statistically stationary stages, which require many reduced bases for the ROM to accurately capture the behavior in each stage [27].

We propose a framework to overcome these difficulties by employing multiple reduced models in time. The idea of the methodology is to construct local ROMs in the parameter-time domain using physical time t as the indicator for clustering and classification [20, 23]. Notice that the methodology is not limited to the 1D-1V Vlasov-Poisson equations and has been applied to Euler equations [20] and Navier-Stokes equations [28]. For a generic problem parameter $\mu \in \mathcal{D}$, let t_f be the final time in the ROM simulation. The computation in the online phase is performed using different reduced bases in N_w subintervals of the temporal domain $[0,\ t_f]$, i.e., $0 = t_0 < t_1 < \quad < t_{N_w - 1} < t_{N_w} = t_f$. With the partition of the indicator range, instead of directly assembling all the snapshot samples into a single huge snapshot matrix, the FOM states are first classified into groups. Let $m \in \mathsf{N}(N_w)$ be a group of index. We denote the subset of the index of time whose corresponding snapshot belongs to the m-th group as

$$\mathcal{G}_m = \{n \in \mathcal{Z} : 0 \quad n \quad N_t \text{ and } t_n \in [t_{m-1},\ t_m)\}, \tag{3}$$

where N_t is the number of time steps and $t_{N_t} = t_f$. Then the snapshot matrix of the distribution $\mathbf{U}_{f;m}$ in the m-th group is formed by assembling the corresponding snapshots, i.e., $\mathbf{U}_{f;m} \quad [\mathbf{f}_n(\quad_k)]_{n \in \mathcal{G}_m}$. Therefore, for $t \in \mathcal{T}_j \quad [t_{j-1},\ t_j]$, we employ the reduced bases constructed from the snapshot group $\mathcal{G}_m$. More precisely, we use the solution representation

$$\tilde{\mathbf{f}}(t;\ \ ) = \mathbf{\Phi}_f^m \hat{\mathbf{f}}^j(t;\ \ ), \quad \text{where } \mathbf{\Phi}_f^m \quad \left[\begin{matrix} \phi^m_{1;f} & & \phi^m_{n_f;f} \end{matrix} \right] \in \mathsf{R}^{N_f \quad n_f}. \tag{4}$$

Here $\mathbf{\Phi}_f^m$ is the distribution solution basis matrix constructed using Proper orthogonal decomposition (POD) [29] from $\mathbf{U}_{f;m}$ with $\phi^m_{i;f}$ being the i-th reduced basis vector and $\hat{\mathbf{f}}^j : \mathcal{T}_j \quad \mathcal{D} \ ! \ \mathsf{R}^{n_f}$ are the time-dependent generalized coordinates for distribution field in the time interval $\mathcal{T}_j$ with n_f being number of distribution reduced basis vectors. The reduced order model for each time interval $\mathcal{T}_j$ is derived by replacing $\mathbf{f}$ with $\tilde{\mathbf{f}}$ in 2, and employ Galerkin projection to close the system:

$$\frac{d\hat{\mathbf{f}}^j}{dt} = (\mathbf{\Phi}_f^m)^T \mathbf{G}(\mathbf{\Phi}_f^m \hat{\mathbf{f}}^j, t;\ \ ). \tag{5}$$

Here, we use the assumption of basis orthonormality. The initial condition is given by projecting onto the ROM spaces, i.e., $\hat{\mathbf{f}}^j(t_{j-1};\ \ ) = (\mathbf{\Phi}_f^m)^T \tilde{\mathbf{f}}(t_{j-1};\ \ )$.

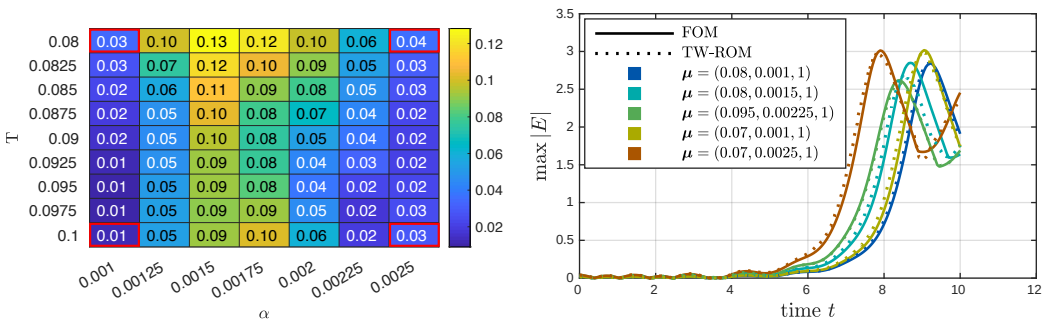
2.3.1 Tensorial approach

There is one major issue with (5). The nonlinear matrix function, $\mathbf{G}$, changes every time the state variables evolve. Additionally, it needs to be multiplied by the basis matrix whenever the updates in the nonlinear term occur, which scales with the FOM size N_f. Therefore, we cannot expect any speed-up without special treatment of the nonlinear term. To overcome this issue, we introduce a tensorial approach to efficiently update the nonlinear term using a precomputed third-order tensor. This requires solving an additional reduced system for the Poisson equation in time interval $\mathcal{T}_j$:

$$(\mathbf{\Phi}^m)^T \nabla^2(\mathbf{\Phi}^m \hat{\ }^j) = (\mathbf{\Phi}^m)^T \int (\mathbf{\Phi}_f^m \hat{\mathbf{f}}^j) dv \quad (\mathbf{\Phi}^m)^T \int \int (\mathbf{\Phi}_f^m \hat{\mathbf{f}}^j)\, dv dx, . \tag{6}$$

where $\mathbf{\Phi}^m$ is the potential solution basis matrix constructed using POD from $\mathbf{U}_{\ ;m} \quad [\ \ _n(\mu_k)]_{n \in \mathcal{G}_m}$ and $\hat{\ }^j : \mathcal{T}_j \quad \mathcal{D} \ ! \ \mathsf{R}^{n_\phi}$ are the time-dependent generalized coordinates for potential field with n being the number of potential reduced basis vectors. With the relation $E = \quad \nabla \phi$, an approximated electric field $\tilde{\mathbf{E}}^j$ can approximated:

$$\tilde{\mathbf{E}}^j(x,t) = \quad \nabla(\mathbf{\Phi}^m \hat{\ }^j) = \sum_i^{n_f} \hat{\ }^j_i \nabla \quad ^m_{;i} = \sum_i^{n_f} \hat{\ }^j_i \quad ^m_{E;i}. \tag{7}$$



(a) ϵ_{f,t_f} distribution

(b) max E history

Figure 1: Left: The $\epsilon_{f;t_f}$ distribution for the discrete parameter space D^h with TW-ROM. The error at training data is enclosed with a red box. Right: Comparison of the maxjEj history between FOM and TW-ROM for training data ($= (0.08, 0.001, 1)$) and testing data. $= (0.08, 0.0015, 1)$ and $(0.095, 0.00225, 1)$ for interpolation and $= (0.07, 0.001, 1)$ and $(0.07, 0.0025, 1)$ for extrapolation.

The evaluation of the right-hand side term in (5) can be approximated as tensor-vector-vector and matrix-vector multiplications:

$$
\begin{aligned}
(\boldsymbol{\Phi}_f^m)^T \mathbf{G}(\boldsymbol{\Phi}_f^m \hat{\mathbf{f}}^j, t; \quad) &= (\boldsymbol{\Phi}_f^m)^T ([\boldsymbol{D}_{\boldsymbol{x}} \quad diag(\boldsymbol{v})] + [diag(\mathbf{E}) \quad \boldsymbol{D}_{\boldsymbol{v}}]) \boldsymbol{\Phi}_f^m \hat{\mathbf{f}}^j \\
&(\boldsymbol{\Phi}_f^m)^T \left([\boldsymbol{D}_x \quad diag(\boldsymbol{v})] + \left[diag(\sum_i \hat{}_i^j \ {}_{E;i}^m) \quad \boldsymbol{D}_v \right] \right) \boldsymbol{\Phi}_f^m \hat{\mathbf{f}}^j \\
&= \boldsymbol{G}_1 \hat{\mathbf{f}}^j + \boldsymbol{\mathcal{G}}_2(\hat{}^j) \hat{\mathbf{f}}^j . \qquad (8)
\end{aligned}
$$

where $\boldsymbol{G}_{1;ij}$ $({}_{f;i}^m)^T [\boldsymbol{D}_x \quad diag(\boldsymbol{v})] \ {}_{f;j}^m$ and $\boldsymbol{\mathcal{G}}_{2;ijk}$ $({}_{f;i}^m)^T [diag({}_{E;j}^m) \quad \boldsymbol{D}_v] \ {}_{f;k}^m$. $\boldsymbol{D}_{\boldsymbol{x}}$ and $\boldsymbol{D}_{\boldsymbol{v}}$ are the 1D first-order derivative matrices in x and v, respectively. $diag(\boldsymbol{v})$ and $diag(\boldsymbol{E})$ are the diagonal matrices with $\boldsymbol{v}$ and $\boldsymbol{E}$ values in the diagonal. The matrix $\boldsymbol{G}_1$ 2 $\mathsf{R}^{n_f \ n_f}$ and the 3rd-order tensor $\boldsymbol{\mathcal{G}}_2$ 2 $\mathsf{R}^{n_f \ n_\phi \ n_f}$ can be precomputed in the offline phase. The memory requirement for the tensor $\boldsymbol{\mathcal{G}}_2$ does not scale with the size of the FOM but with the number of distribution POD modes n_f and potential POD modes n . In the online phase, the evaluation of (8) scales as $O(n_f^2 n \ )$ instead of $O(N_f)$.

3 Results and discussion

We consider $t_f = 10$ with $N_t = 4000$ for (1). The initial velocity deplacement v_0 is set to 1 and the parameter $= (T, \alpha, v_0)$ varies in the domain $D = [0.08, \ 0.1] \quad [0.001, \ 0.0025] \quad f1g$. The TW-ROM (5–6) is constructed with four FOM solutions data (training data) at $_1 = (0.08, 0.001, 1)$, $_2 = (0.08, 0.0025, 1)$, $_3 = (0.1, 0.001, 1)$, and $_4 = (0.1, 0.0025, 1)$ in the time interval $[0, \ t_f]$ with $N_w = 100$ subintervals and a total of 160 snapshots per subinterval. We test the TW-ROM on a discrete parameter space D^h D, including T 2 $[0.08, \ 0.1]$ and α 2 $[0.001, \ 0.0025]$, with 9 and 7 evenly distributed discrete points in the respective parameter range, resulting a total of 63 parameters. To evaluate the TW-ROM performance, the relative error of the approximated solution is measured against the corresponding FOM solution at the final time t_f, which is defined as $\epsilon_{f;t_f}$ $kf \quad \tilde{f}k_2/kfk_2$. Fig. (1a) displays the $\epsilon_{f;t_f}$ of 63 parameters 2 D^h with TW-ROM. The relative errors at the four training data (displayed on four corners) are less than 5%. The maximum relative error for testing data is 13%. The behavior of maxjEj, an important quantity to determine the growth rate, is also reported for both the FOM and the TW-ROM in Fig. (1b) for the training and testing data. The TW-ROM is able to capture the growth rate of the interpolation cases ($= (0.08, 0.0015, 1)$ and $(0.095, 0.00225, 1)$) and the extrapolation cases ($= (0.07, 0.001, 1)$ and $(0.07, 0.0025, 1)$) as well.

Fig. (2) further displays the predicted distribution $\mathbf{f}$ of the TW-ROM at time instance $t = 8$ and $t = 10$ for three values: $= (0.095, 0.00225, 1)$ and $= (0.08, 0.0015, 1)$ as interpolation cases, and $= (0.07, 0.0025, 1)$ as an extrapolation case. Despite some discrepancies in the predicted $\mathbf{f}$

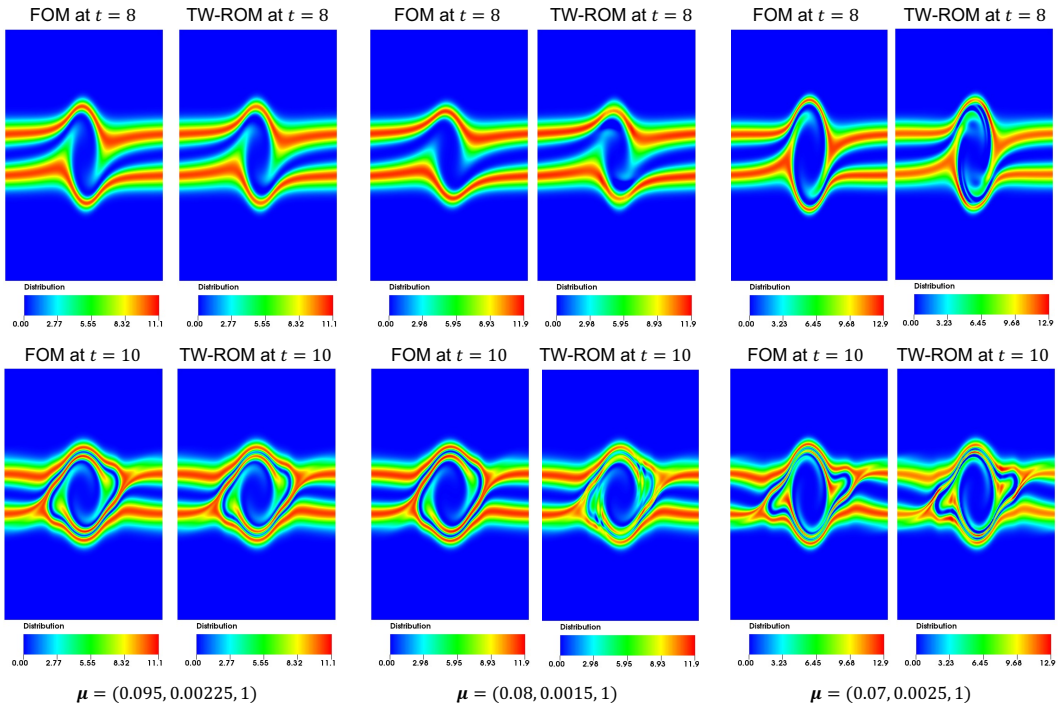


Figure 2: The predicted distribution $\mathbf{f}$ of the TW-ROM at time instance $t = 8$ and $t = 10$ for three values: $= (0.095, 0.00225, 1)$ and $= (0.08, 0.0015, 1)$ as interpolation cases, and $= (0.07, 0.0025, 1)$ as an extrapolation case.

values for $= (0.08, 0.0015, 1)$ and $= (0.07, 0.0025, 1)$, the TW-ROM accurately capture the occurrence of instability.

For a given parameter , the wall-clock time for the FOM is about 56 (seconds), whereas the wall-clock time for the TW-ROM is about 0.2 (seconds), achieving a 280 times speed-up and a $4,480$ times speed-up in CPU hours. The construction of the tensor G_2 for 100 time windows takes about 87.2 (seconds) which is longer than one FOM simulation. However, the speed-up gained in the online stage compensates the overhead. Furthermore, the current implementation of the tensor construction is serial, suggesting the potential for further reduction in construction time through parallel implementation.

The testing and training data (FOM solutions) and the TW-ROM offline and online phases are performed on Livermore Computing Quartz [30] with Intel Xeon CPUs, 128 GB memory, peak TFLOPS of 3251.4, and peak single CPU memory bandwidth of 77 GB/s. We use the functionalities in libROM [31], an existing asset with Apache license, to construct the TW-ROM.

A potential constraint lies in the choice of the indicator for partitioning the temporal domain. The existing TW-ROM is established using a physical-time indicator, resulting in a temporal partition that is independent of parameters. Figs. (1a-1b) demonstrate that the TW-ROM can effectively approximate solutions and capture growth rates within a reasonable range of T and α. Moreover, it can readily adapt to other parameterization types. However, to extend the application of TW-ROM to a broader or higher-dimensional parameter space, a more intelligent indicator based on physics is essential.

In the context of 2D-2V and 3D-3V problems, the solution dynamics are anticipated to become even more intricate. This complexity requires standard ROM to use large number of modes to accurately represent the dynamic behavior. Consequently, the development of a time-windowing ROM becomes essential. This approach is crucial not only for capturing the multi-scale behavior but also for maintaining computational efficiency.

4 Broader impact

This paper presents a novel data-driven reduced-order modeling approach employing time-windowing and a tensorial strategy to expedite kinetic simulations of electrostatic plasmas. The envisioned framework is anticipated to exert a significant influence on the computational science community and holds the potential for diverse applications across various engineering and scientific domains. It is important to note that this work has been conducted with careful consideration, and no adverse ethical or societal consequences are associated with its findings.

Acknowledgments and Disclosure of Funding

This work was performed under the auspices of the U.S. Department of Energy (DOE), by Lawrence Livermore National Laboratory (LLNL) under Contract No. DE-AC52–07NA27344 and was supported by Laboratory Directed Research and Development funding under projects 21-SI-006 and 22-SI-006. Y. Choi was also supported for this work by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, as part of the CHaRMNET Mathematical Multifaceted Integrated Capability Center (MMICC) program, under Award Number DE-SC0023164. IM release: LLNL-CONF-855151.

References

[1] Charles K Birdsall and A Bruce Langdon. *Plasma physics via computer simulation*. CRC press, 2018.

[2] John P Verboncoeur. Particle simulation of plasmas: review and advances. *Plasma Physics and Controlled Fusion*, 47(5A):A231, 2005.

[3] Jeffrey AF Hittinger and Jeffrey W Banks. Block-structured adaptive mesh refinement algorithms for Vlasov simulation. *Journal of Computational Physics*, 241:118–140, 2013.

[4] Eric Sonnendrücker, Jean Roche, Pierre Bertrand, and Alain Ghizzo. The semi-Lagrangian method for the numerical resolution of the Vlasov equation. *Journal of Computational Physics*, 149(2):201–220, 1999.

[5] Lukas Einkemmer. High performance computing aspects of a dimension independent semi-Lagrangian discontinuous Galerkin code. *Computer Physics Communications*, 202:326–336, 2016.

[6] Youngsoo Choi and Kevin Carlberg. Space–time least-squares Petrov–Galerkin projection for nonlinear model reduction. *SIAM Journal on Scientific Computing*, 41(1):A26–A58, 2019.

[7] Kevin Carlberg, Youngsoo Choi, and Syuzanna Sargsyan. Conservative model reduction for finite-volume models. *Journal of Computational Physics*, 371:280–314, 2018.

[8] Youngkyu Kim, Youngsoo Choi, David Widemann, and Tarek Zohdi. A fast and accurate physics-informed neural network reduced order model with shallow masked autoencoder. *Journal of Computational Physics*, 451:110841, 2022.

[9] Jessica T Lauzon, Siu Wun Cheung, Yeonjong Shin, Youngsoo Choi, Dylan Matthew Copeland, and Kevin Huynh. S-OPT: A points selection algorithm for hyper-reduction in reduced order models. *arXiv preprint arXiv:2203.16494*, 2022.

[10] Alejandro N Diaz, Youngsoo Choi, and Matthias Heinkenschloss. A fast and accurate domain-decomposition nonlinear manifold reduced order model. *arXiv preprint arXiv:2305.15163*, 2023.

[11] Kento Kaneko, Ping-Hsuan Tsai, and Paul Fischer. Towards model order reduction for fluid-thermal analysis. *Nuclear Engineering and Design*, 370:110866, 2020.

[12] Ping-Hsuan Tsai and Paul Fischer. Parametric model-order-reduction development for unsteady convection. *Frontiers in Physics*, 10:903169, 2022.

[13] Ping-Hsuan Tsai, Paul Fischer, and Edgar Solomonik. Accelerating the Galerkin reduced-order model with the tensor decomposition for turbulent flows. *arXiv preprint arXiv:2311.03694*, 2023.

[14] Chi Hoang, Youngsoo Choi, and Kevin Carlberg. Domain-decomposition least-squares Petrov–Galerkin (DD-LSPG) nonlinear model reduction. *Computer methods in applied mechanics and engineering*, 384:113997, 2021.

[15] Youngsoo Choi, Peter Brown, William Arrighi, Robert Anderson, and Kevin Huynh. Space–time reduced order model for large-scale linear dynamical systems with application to Boltzmann transport problems. *Journal of Computational Physics*, 424:109845, 2021.

[16] Youngsoo Choi, Gabriele Boncoraglio, Spenser Anderson, David Amsallem, and Charbel Farhat. Gradient-based constrained optimization using a database of linear reduced-order models. *Journal of Computational Physics*, 423:109787, 2020.

[17] Sean McBane and Youngsoo Choi. Component-wise reduced order model lattice-type structure design. *Computer methods in applied mechanics and engineering*, 381:113813, 2021.

[18] Sean McBane, Youngsoo Choi, and Karen Willcox. Stress-constrained topology optimization of lattice-like structures using component-wise reduced order models. *Computer Methods in Applied Mechanics and Engineering*, 400:115525, 2022.

[19] Youngsoo Choi, Geoffrey Oxberry, Daniel White, and Trenton Kirchdoerfer. Accelerating design optimization using reduced order models. *arXiv preprint arXiv:1909.11320*, 2019.

[20] Siu Wun Cheung, Youngsoo Choi, Dylan Matthew Copeland, and Kevin Huynh. Local Lagrangian reduced-order modeling for the Rayleigh-Taylor instability by solution manifold decomposition. *Journal of Computational Physics*, 472:111655, 2023.

[21] Kyle Washabaugh, David Amsallem, Matthew Zahr, and Charbel Farhat. Nonlinear model reduction for CFD problems using local reduced-order bases. In *42nd AIAA Fluid Dynamics Conference and Exhibit*, page 2686.

[22] David Amsallem, Matthew J Zahr, and Charbel Farhat. Nonlinear model order reduction based on local reduced-order bases. *International Journal for Numerical Methods in Engineering*, 92 (10):891–916, 2012.

[23] Dylan Matthew Copeland, Siu Wun Cheung, Kevin Huynh, and Youngsoo Choi. Reduced order models for Lagrangian hydrodynamics. *Computer Methods in Applied Mechanics and Engineering*, 388:114259, 2022.

[24] Siu Wun Cheung, Youngsoo Choi, H Keo Springer, and Teeratorn Kadeethum. Data-scarce surrogate modeling of shock-induced pore collapse process. *arXiv preprint arXiv:2306.00184*, 2023.

[25] G.-S. Jiang and C.-W. Shu. Efficient implementation of weighted ENO schemes. *Journal of Computational Physics*, 126(1):202–228, 1996. ISSN 0021-9991. doi: 10.1006/jcph.1996.0130.

[26] D Anderson, Renato Fedele, and M Lisak. A tutorial presentation of the two stream instability and Landau damping. *American Journal of Physics*, 69(12):1262–1266, 2001.

[27] Jan Hesthaven, Cecilia Pagliantini, and Nicolò Ripamonti. Adaptive symplectic model order reduction of parametric particle-based Vlasov–Poisson equation. *Mathematics of Computation*, 2023.

[28] Eric J Parish and Kevin T Carlberg. Windowed least-squares model reduction for dynamical systems. *Journal of Computational Physics*, 426:109939, 2021.

[29] Gal Berkooz, Philip Holmes, and John L Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual review of fluid mechanics*, 25(1):539–575, 1993.

[30] Livermore Computing – Quartz. `https://hpc.llnl.gov/hardware/compute-platforms/quartz`.

[31] Youngsoo Choi, William J. Arrighi, Dylan M. Copeland, Robert W. Anderson, and Geoffrey M. Oxberry. librom. [Computer Software] `https://doi.org/10.11578/dc.20190408.3`, oct 2019. URL `https://doi.org/10.11578/dc.20190408.3`.