

Machine-learning closure for Vlasov–Poisson dynamics in Fourier–Hermite space

Nathaniel Barbour^{1,2} , William Dorland^{1,2} , Ian G. Abel² and Matt Landreman² 

¹Department of Physics, University of Maryland, College Park, MD 20742, USA

²Institute for Research in Electronics and Applied Physics, University of Maryland, College Park, MD 20742, USA

Corresponding author: Nathaniel Barbour, barbour@umd.edu

(Received 17 April 2025; revision received 17 August 2025; accepted 21 August 2025)

Accurate reduced models of turbulence are desirable to facilitate the optimisation of magnetic-confinement fusion reactor designs. As a first step towards higher-dimensional turbulence applications, we use reservoir computing, a machine-learning (ML) architecture, to develop a closure model for a limiting case of electrostatic gyrokinetics. We implement a pseudo-spectral Eulerian code to solve the one-dimensional Vlasov–Poisson system on a basis of Fourier modes in configuration space and Hermite polynomials in velocity space. When cast onto the Hermite basis, the Vlasov equation becomes an infinitely coupled hierarchy of fluid moments, presenting a closure problem. We exploit the locality of interactions in the Hermite representation to introduce an ML closure model of the small-scale dynamics in velocity space. In the linear limit, when the kinetic Fourier–Hermite solver is augmented with the reservoir closure, the closure permits a reduction of the velocity resolution, with a relative error within 2 % for the Hermite moment where the reservoir closes the hierarchy. In the strongly nonlinear regime, the ML closure model more accurately resolves the low-order Fourier and Hermite spectra when compared with a naive closure by truncation and reduces the required velocity resolution by a factor of 16.

Key words: plasma simulation, plasma instabilities, fusion plasma

1. Introduction

One of the most important ongoing efforts in the pursuit of magnetic fusion energy is the study of microturbulence (Yoshida *et al.* 2025). While magnetic-confinement fusion plasmas are designed to be stable to macroscopic magnetohydrodynamic perturbations, the steep gradients in temperature and density that are present in all current designs also drive small-scale instabilities. These instabilities drive sustained turbulence, which then dominates the transport of particles, energy and momentum in the plasma.

As the fusion industry turns its focus toward designing first-of-a-kind pilot plants, optimising these designs to mitigate turbulent heat losses is an urgent objective.

Significant progress in turbulence optimisation for tokamaks (Highcock *et al.* 2018) and stellarators (Roberg-Clark *et al.* 2023; Kim *et al.* 2024) has been achieved. However, the space of candidate magnetic equilibria remains vast, and high-resolution turbulence simulations are computationally expensive. A unifying feature of fluid (Kraichnan 1959; Kolmogorov 1991), magnetohydrodynamic (Goldreich & Sridhar 1995) and gyrokinetic (Bañón Navarro *et al.* 2011) turbulence theory is that free energy tends to cascade from large scales to small scales through local interactions in phase space, and these cascades are well described by power laws.¹ Therefore, models that can accurately calculate large-scale quantities, including spectra and heat fluxes, without needing to resolve small scales, are both desirable and plausible.

In the field of fluid turbulence modelling, two successful approaches to constructing reduced, lower-resolution models that capture large-scale behaviour are Reynolds-averaged Navier–Stokes (Reynolds 1895) and large-eddy simulation (Deardorff 1970). Large-eddy simulation also has been used to build reduced models of gyrokinetic turbulence (Bañón Navarro *et al.* 2014). These formalisms accelerate turbulence simulations by averaging over time and filtering out the small-scale dynamics. Closure models provide another pathway to filter out the small scales while retaining accurate large-scale statistics. In kinetic plasma physics, Hammett & Perkins (1990) developed a landmark analytic linear closure for the low-frequency response to small Langmuir oscillations.

An alternative method for building reduced models is to use data-driven, statistical techniques like machine learning (ML) and Bayesian optimisation. In recent years, the proliferation of ML models has led to advancements in pattern identification and model construction for magnetic fusion. Some example applications include improved prediction and avoidance of tokamak disruptions using the DECAF framework (Piccione *et al.* 2020) and on the DIII-D tokamak (Rea *et al.* 2019; Fu *et al.* 2020), improved active-feedback plasma control systems with reinforcement learning (Degraeve *et al.* 2022; Seo *et al.* 2024; Kerboua-Benlarbi *et al.* 2024), optimised execution of gyrokinetic simulations with the PORTALS framework (Rodríguez-Fernandez, Howard & Candy 2022) and accelerated, low-resolution turbulence simulations (Greif, Jenko & Thuerey 2023; Castagna *et al.* 2024; Clavier *et al.* 2025). Data-driven, configuration space (Ma *et al.* 2020; Qin *et al.* 2023) and heat flux (Ingelsten *et al.* 2024; Huang, Dong & Wang 2025) closure models of Landau damping have also been developed.

In this paper, we choose to use the one-dimensional, collisionless Vlasov–Poisson system as a test problem for implementing an ML closure in velocity space. This system is the one-dimensional, electrostatic limit of the full five-dimensional gyrokinetic system (Parker & Dellar 2015). It models dynamics parallel to the magnetic field, and it contains a quadratic nonlinearity present in the full gyrokinetic system. These features make the Vlasov–Poisson system a suitable foundation to assess the future potential for a closure model to be developed for gyrokinetic turbulence.

In parallel velocity space, we use properties of the Hermite basis to cast the objective of reducing resolution requirements as a closure problem. The Hermite basis

¹Power laws typically arise when a scale-local transfer of energy occurs, and a separation between energy injection and dissipation scales exists. The relevance of scale locality to energy cascades in gyrokinetics has been studied in detail by Barnes, Parra & Schekochihin (2011) and Teaca, Jenko & Told (2017). In some systems of interest, inverse cascades occur, driving invariants from small scales to large scales. The theory of inverse cascades has been developed for several two-dimensional systems (Hasegawa & Mima 1978; Plunk *et al.* 2010), yet an open question remains about their applicability to the full five-dimensional gyrokinetic system.

has been used extensively to solve problems in kinetic plasma dynamics (Armstrong 1967; Grant & Feix 1967; Joyce, Knorr & Meier 1971; Gibelli & Shizgal 2006; Zocco & Schekochihin 2011; Loureiro, Schekochihin & Zocco 2013; Kanekar *et al.* 2015; Parker & Dellar 2015; Jorge, Ricci & Loureiro 2017; Adkins & Schekochihin 2018; Mandell, Dorland & Landreman 2018; Frei *et al.* 2023). In this work, we opt to focus on velocity space because the Hermite basis naturally expresses the interaction between scales in velocity space as local interactions in a coupled hierarchy of moments. This property of the moment formalism directly presents a single-term target for a closure. Specifically, the interaction between resolved scales and unresolved scales in velocity space appears as a single unknown term in an equation in this representation. In contrast, the spectral representation of configuration space introduces some non-local interactions across scales, complicating the development of a closure model. Additionally, the Hermite basis explicitly expresses conservation laws for particle number, momentum and energy as evolution equations for the lowest-order moments. These properties allow us to develop a closure that preserves the low-order conservation laws. More details on the closure problem are presented in § 3.5.

We use reservoir computing, an ML architecture, to implement a spectral, proof-of-concept, velocity-space closure for the moment hierarchy that maintains accurate spectra and reduces resolution requirements. The reservoir computing model maintains an internal representation of the system state, which it evolves through a directed graph with fixed weights. It then predicts the system state at time $t + 1$ from the state at time t using a set of output weights trained by linear regression. Since its concurrent development by Jaeger (2001) and Maass, Natschläger & Makram (2002), reservoir computing has successfully been applied to the task of forecasting the dynamics of low-dimensional chaotic systems (Lu *et al.* 2017; Pathak *et al.* 2018). This ML paradigm is well suited to the task of modelling time-series data of physical processes. In climate physics, Arcomano *et al.* (2020) built a surrogate global atmospheric forecast model with reservoir computing, and in plasma physics, Jalalvand *et al.* (2022) used reservoir computing to analyse and classify Alfvén eigenmodes using DIII-D diagnostics data. In comparison with several other recurrent neural network architectures, reservoir computing has shown comparable ability to capture the statistics of chaotic systems, while requiring significantly less training time (Vlachas *et al.* 2020). These results motivate our decision to use reservoir computing for the closure model.

Some ML methods, including reservoir computing, are more challenging to interpret than traditional physics models. Artificial neural network architectures are often ‘black boxes’ that generate layered, analytically intractable networks of nonlinear activation functions. Gaining insightful intuition from these networks can be daunting. We mitigate this problem by constraining the ML portion of our simulations to represent only the unresolved small scales, preserving analytic evolution equations for large scales. Our methods are analogous to a large-eddy simulation approach for phase space.

The paper has the following structure. In § 2, we present our test problem, the normalised one-dimensional Vlasov–Poisson system. We then derive the projection of the system onto the pseudo-spectral Fourier–Hermite basis in § 3. We present the closure problem in § 3.5 and introduce our ML closure model in § 4 as a solution to that problem. In § 5, we present our ML model’s predictions, and we conclude with a summary and discussion in § 6.

2. Vlasov–Poisson system

As our test problem, we study the dynamics of a one-dimensional, collisionless, electrostatic hydrogen plasma. We focus on the electron dynamics, treating the ions as a cold, neutralising background. The equations that describe this plasma are the Vlasov–Poisson system (Vlasov 1968). The calculations in this paper use Gaussian units. The Vlasov equation for electrons in one dimension is

$$\frac{\partial f}{\partial t} + v \frac{\partial f}{\partial z} - \frac{eE}{m_e} \frac{\partial f}{\partial v} = 0, \quad (2.1)$$

where $f(z, v, t)$ is the electron distribution function, e is the elementary charge and m_e is the mass of an electron. Term $E(z, t)$ is the electric field, which we calculate using Poisson's equation:

$$-\frac{\partial^2 \Phi}{\partial z^2} = 4\pi\rho, \quad E = -\frac{\partial \Phi}{\partial z}, \quad (2.2)$$

where $\rho(z, t)$ is the total electric charge density, including both ions and electrons, and $\Phi(z, t)$ is the electric potential. For this quasi-neutral plasma with cold ions, we can evaluate the charge density as

$$\rho(z, t) = e \left(n_0 - \int dv f \right), \quad (2.3)$$

where n_0 is the mean number density of both species.

Plasmas with near-Maxwellian velocity distributions are of particular interest for magnetic fusion applications. Therefore, we separate the distribution function into a Maxwellian mean and time-dependent perturbations:

$$f(z, v, t) = F_0(v) + g(z, v, t), \quad (2.4)$$

where

$$F_0(v) = \left(\frac{n_0}{v_{te}} \right) \frac{1}{\sqrt{2\pi}} e^{-v^2/2v_{te}^2} \quad (2.5)$$

is the one-dimensional Maxwellian velocity distribution for the electrons, and $g(z, v, t)$ is the fluctuating part of the distribution function. We have defined the electron thermal speed as $v_{te} = \sqrt{T_e/m_e}$, where T_e is the temperature of the electrons in units of energy. The Vlasov equation supports longitudinal Langmuir waves, whose dynamics occurs on time scales of the order of the plasma frequency:

$$\omega_{pe} \equiv \sqrt{\frac{4\pi n_0 e^2}{m_e}}. \quad (2.6)$$

The characteristic length scale for those oscillations is the Debye length:

$$\lambda_D \equiv \sqrt{\frac{T_e}{4\pi n_0 e^2}}. \quad (2.7)$$

To support numerical integration of this system of equations, we normalise the variables into dimensionless forms, denoted with primes:

$$\begin{aligned}
t &\equiv \frac{1}{\omega_{pe}} t', & v &\equiv v_{te} v', & z &\equiv \lambda_D z', & E &\equiv \frac{T_e}{e\lambda_D} E', & f &\equiv F_0 + g \equiv \frac{n_0}{v_t} (F'_M + g'), \\
\rho &\equiv n_0 e \rho', & \Phi &\equiv \frac{T_e}{e} \Phi', & F'_M &\equiv \frac{1}{\sqrt{2\pi}} e^{-v'^2/2}, & \int dv' F'_M &= 1.
\end{aligned}
\tag{2.8}$$

After dropping the primes for legibility, we can now write the model equations in dimensionless form:

$$\frac{\partial f}{\partial t} + v \frac{\partial f}{\partial z} - E \frac{\partial f}{\partial v} = 0, \tag{2.9}$$

$$\frac{\partial E}{\partial z} = 1 - \int dv f. \tag{2.10}$$

Finally, after splitting the distribution function into its mean and fluctuating components, we arrive at the normalised evolution equation for the fluctuating part of the electron distribution function:

$$\frac{\partial g}{\partial t} + v \frac{\partial g}{\partial z} + v E F_M - E \frac{\partial g}{\partial v} = 0, \tag{2.11}$$

$$\frac{\partial E}{\partial z} = - \int dv g. \tag{2.12}$$

3. Pseudo-spectral methods

We solve the Vlasov–Poisson system (2.11)–(2.12) using an orthonormal spectral basis by projecting the velocity-space dependence onto a Hermite polynomial basis and the configuration-space dependence onto Fourier harmonics. We choose this approach because pseudo-spectral methods are often more efficient than pure finite-difference algorithms (Boyd 2000), and as mentioned in the introduction, there is an extensive history of the application of the Fourier–Hermite basis to solve problems in kinetic plasma physics.

3.1. Hermite moment formalism

To discretise velocity space, we impose a Hermite basis representation, using notation from Mandell *et al.* (2018). Including both the Maxwellian weight and the normalisation factors, the Hermite basis expansion ($\varphi^m(v)$) and projection ($\varphi_m(v)$) functions are

$$\varphi^m(v) = \frac{\text{He}_m(v)}{\sqrt{2\pi m!}} e^{-v^2/2}, \quad \varphi_m(v) = \frac{\text{He}_m(v)}{\sqrt{m!}}, \tag{3.1}$$

where

$$\text{He}_m(v) = (-1)^m e^{v^2/2} \frac{d^m}{dv^m} e^{-v^2/2} \tag{3.2}$$

are the probabilists' Hermite polynomials (Abramowitz & Stegun 1972). Note that these functions are orthonormal,

$$\int_{-\infty}^{\infty} dv \varphi^m(v) \varphi_{m'}(v) = \delta_{m,m'}, \tag{3.3}$$

and that the background Maxwellian is identical to the $m = 0$ basis element:

$$\varphi^0 = \frac{1}{\sqrt{2\pi}} e^{-v^2/2} = F_M. \quad (3.4)$$

This property makes the Hermite moment representation a natural choice for near-Maxwellian velocity distributions. We project the fluctuating portion of the distribution function onto the Hermite basis using

$$g(z, v, t) = \sum_{m=0}^{\infty} \varphi^m G_m(z, t), \quad G_m(z, t) \equiv \int_{-\infty}^{\infty} dv \varphi_m g(z, v, t). \quad (3.5)$$

3.2. Vlasov equation in the Hermite basis

We seek an evolution equation for the amplitudes, G_m , of the Hermite expansion functions. To derive this moment hierarchy, we project (2.11) onto the Hermite basis term by term. The first term is the most straightforward:

$$\int_{-\infty}^{\infty} dv \varphi_m \frac{\partial g}{\partial t} = \frac{\partial G_m}{\partial t}. \quad (3.6)$$

We then project the linear ballistic term, $v(\partial g / \partial z)$, onto the basis:

$$v \frac{\partial g}{\partial z} = \frac{\partial}{\partial z} \sum_{m=0}^{\infty} v \varphi^m G_m = \frac{\partial}{\partial z} \sum_{m=0}^{\infty} \left(\sqrt{m+1} \varphi^{m+1} + \sqrt{m} \varphi^{m-1} \right) G_m, \quad (3.7)$$

where we used a recurrence relation for the Hermite basis functions to remove the explicit dependence on v (Abramowitz & Stegun 1972):

$$\sqrt{m+1} \varphi^{m+1} = v \varphi^m - \sqrt{m} \varphi^{m-1}. \quad (3.8)$$

Applying the projection function to (3.7) and using (3.3) yields

$$\int_{-\infty}^{\infty} dv \varphi_m v \frac{\partial g}{\partial z} = \frac{\partial}{\partial z} \left(\sqrt{m} G_{m-1} + \sqrt{m+1} G_{m+1} \right). \quad (3.9)$$

The linear portion of the electric field drive term contains the $m = 1$ expansion function, so applying the projection function to this term yields

$$\int_{-\infty}^{\infty} dv \varphi_m v E F_M = E \int_{-\infty}^{\infty} dv \varphi_m \varphi^1 = E \delta_{m,1}. \quad (3.10)$$

The final term to consider is the nonlinear term, $-E(\partial g / \partial v)$. Combining a second recurrence relation (Abramowitz & Stegun 1972),

$$\frac{d}{dv} \text{He}_m(v) = v \text{He}_m(v) - \text{He}_{m+1}(v), \quad (3.11)$$

with the definition of the Hermite basis functions φ^m (3.1), presents a useful new identity:

$$\frac{\partial \varphi^m}{\partial v} = -\sqrt{m+1} \varphi^{m+1}. \quad (3.12)$$

The expansion of the nonlinear term in the Hermite basis is then

$$-E \frac{\partial g}{\partial v} = -E \sum_m \frac{\partial \varphi^m}{\partial v} G_m = E \sum_m \sqrt{m+1} \varphi^{m+1} G_m. \quad (3.13)$$

After projecting this term onto the basis,

$$\begin{aligned} \int_{-\infty}^{\infty} dv \varphi_m \left(E \sum_{m'} \sqrt{m'+1} \varphi^{m'+1} G_{m'} \right) &= \sum_{m'} E \sqrt{m'+1} \delta_{m,m'+1} G_{m'} \\ &= E \sqrt{m} G_{m-1}, \end{aligned} \quad (3.14)$$

we can collect all of the terms to recover the general form of the equation for G_m :

$$\frac{\partial G_m}{\partial t} + \frac{\partial}{\partial z} \left(\sqrt{m} G_{m-1} + \sqrt{m+1} G_{m+1} \right) + E \sqrt{m} G_{m-1} + E \delta_{m,1} = 0. \quad (3.15)$$

3.3. Pseudo-spectral method in Fourier space

Next, we discretise space with a Fourier basis. The Fourier transform and inverse Fourier transform operators for a spatial grid of N points are

$$\mathcal{F}[g] = \frac{1}{N} \sum_z g e^{-ikz} \quad \mathcal{F}^{-1}[G_k] = \sum_k G_k e^{ikz}, \quad (3.16)$$

where the Fourier wavenumbers are given by $k = 2\pi j/N$, with $-N/2 \leq j < N/2$ and $j \in \mathbb{Z}$. The Fourier convolution theorem,

$$\mathcal{F}[fg] = \sum_{k'} F_k G_{k-k'}, \quad (3.17)$$

transforms the nonlinear term into a coupled sum over all wavenumbers. This coupling introduces additional computational overhead. We mitigate that problem by using a pseudo-spectral method (Boyd 2000), evaluating the nonlinear term on the original configuration space and projecting the result back onto the space of Fourier wavenumbers. Applying the Fourier transform, (3.16), to (3.15) and then applying the Fourier convolution theorem yields

$$\begin{aligned} \frac{\partial G_{m,k}}{\partial t} + ik \left(\sqrt{m} G_{m-1,k} + \sqrt{m+1} G_{m+1,k} \right) + \sqrt{m} \mathcal{F} \left[\mathcal{F}^{-1}[E_k] \mathcal{F}^{-1}[G_{m-1,k}] \right] \\ = -E_k \delta_{m,1}. \end{aligned} \quad (3.18)$$

Simulating this collisionless system with finite resolution without using any regularisation is numerically infeasible, as the dynamics can generate structures in phase space at arbitrarily small scales (Parker & Dellar 2015; Loureiro *et al.* 2016; Mandell *et al.* 2024). To mitigate that effect, we introduce numerical regularisation through hyperdiffusion ($-\nu_k k^4$) and hypercollision ($-\nu_m m^4$) operators:

$$H_{mk} = \begin{cases} 0 & \text{if } m \leq 2 \\ -\nu_k k^4 - \nu_m m^4 & \text{if } m > 2, \end{cases} \quad (3.19)$$

where ν_k and ν_m are parameters which control the strengths of the hyperdiffusion and hypercollisions, respectively. Note that we do not apply this regularisation to

the $m = 0, 1$ and 2 equations, as they encode conservation of particle number, momentum and energy, respectively.

We solve the system for a finite number of moments M . Defining compact notation for the nonlinear operator,

$$\mathcal{N}[E_k, G_{m-1,k}] = \sqrt{m} \mathcal{F}[\mathcal{F}^{-1}[E_k] \cdot \mathcal{F}^{-1}[G_{m-1,k}]], \quad (3.20)$$

and accounting for the special cases of the $m = 0, m = 1$ and $m = M - 1$ equations, we have derived the projection of the Vlasov equation onto the Fourier–Hermite basis:

$$\frac{\partial G_{0,k}}{\partial t} + ikG_{1,k} = 0, \quad (3.21)$$

$$\frac{\partial G_{1,k}}{\partial t} + ik \left(G_{0,k} + \sqrt{2}G_{2,k} \right) + \mathcal{N}[E_k, G_{0,k}] + E_k = 0, \quad (3.22)$$

$$\begin{aligned} \frac{\partial G_{m,k}}{\partial t} + ik \left(\sqrt{m}G_{m-1,k} + \sqrt{m+1}G_{m+1,k} \right) + \mathcal{N}[E_k, G_{m-1,k}] \\ = H_{m,k}G_{m,k}, \quad 2 \leq m < M-1, \end{aligned} \quad (3.23)$$

$$\frac{\partial G_{M-1,k}}{\partial t} + ik\sqrt{M-1}G_{M-2,k} + \mathcal{N}[E_k, G_{M-2,k}] = H_{M-1,k}G_{M-1,k}. \quad (3.24)$$

3.4. Poisson's equation in the Fourier–Hermite basis

To complete the system, we project Poisson's equation onto the Fourier–Hermite basis and solve for the electric field, E_k . First, we solve for the charge density in the Hermite basis:

$$\rho = - \int_v dv g = - \int_v dv \varphi_0 \sum_m \varphi^m G_m = -G_0. \quad (3.25)$$

Next, we insert this expression for the normalised charge density into Poisson's equation (2.12) and apply the Fourier transform (3.16) to calculate the electric field as a function of the $m = 0$ density moment in the Fourier–Hermite basis:

$$E_k = \frac{iG_{0,k}}{k}. \quad (3.26)$$

We require that the $k = 0$ mode is zero and stationary, corresponding to zero mean electric field.

3.5. The closure problem

A continuous representation formally requires an infinite number of Hermite moments, but we must discretise the system with a finite number of moments, M . Observe that in the Fourier–Hermite basis, the parallel streaming term in the convective derivative, $v(\partial g / \partial z)$, becomes $ik(\sqrt{m}G_{m-1,k} + \sqrt{m+1}G_{m+1,k})$ for general m . This term couples nearest neighbours in Hermite space and forms a hierarchy of equations that must be closed. This closure problem is analogous to the classic closure problem encountered when taking fluid moments of the kinetic equation (2.1). The resulting fluid equations for density, momentum, energy and higher-order moments exhibit coupling between moment m and moments $m + 1$ and $m - 1$. This same feature occurs in the Hermite representation of velocity space, and from (3.5)

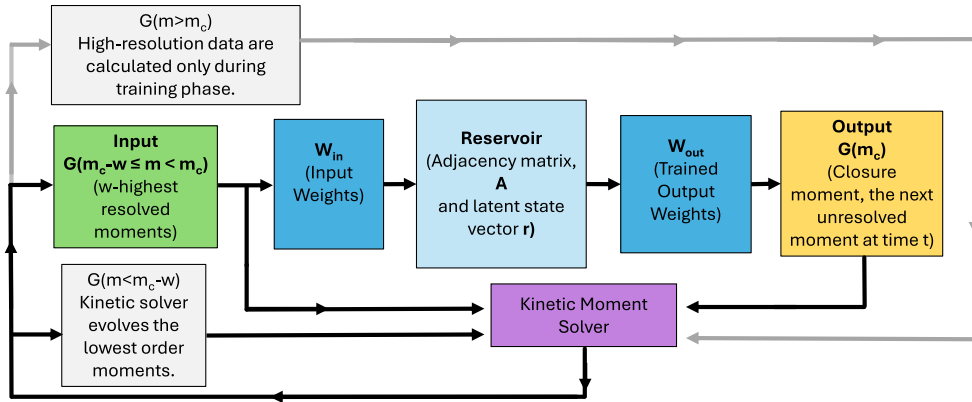


FIGURE 1. Diagram of the reservoir computing closure model, integrated with the kinetic moment solver.

and (3.1), the Hermite moment basis is obtained through a similar process of multiplying the distribution function by increasing powers of v via increasing orders of the Hermite polynomials and then integrating over velocity space.

The simplest model of closure is by truncation, which we use in the direct numerical solution (DNS) by omitting the $G_{m+1,k}$ term in the evolution equation for the final moment, $G_{M-1,k}$. Our objective is to introduce a dynamical closure model to close the hierarchy at a moment number $m = m_c$, $m_c < M - 1$, while still achieving accurate predictions of the Fourier and Hermite spectra. We use a reservoir recurrent neural network as the closure model, which we describe in more detail in § 4.

4. Machine-learning closure: reservoir computing

Inspecting the evolution equation for $G_{m,k}$ (3.15) reveals that the coupling between moment m and moment $m + 1$ only explicitly appears as a linear term in the equation, acting on each wavenumber k independently. As a result, we introduce a unique and independent Hermite closure model for each k in the system, for a total of N_k reservoirs. We use methods similar to those of Pathak *et al.* (2018) to design the reservoirs.

Each reservoir is a network of D_r nodes defined by a latent state vector $\mathbf{r}_k(t) \in \mathbb{R}^{D_r}$, where each element of $\mathbf{r}_k(t)$ represents the state of one node at time t . The structure of the reservoir computing model is presented in figure 1. The connections between nodes are defined by a $D_r \times D_r$ sparse adjacency matrix, $\mathbf{A}$, where exactly κ non-zero elements per row are initialised to random values drawn from the uniform distribution on $[0, 1]$ and scaled by the appropriate factors to set the largest eigenvalue of $\mathbf{A}$ equal to a specified value, ρ_{sp} . The magnitude of the largest eigenvalue of a reservoir's adjacency matrix has been demonstrated to significantly impact the accuracy of the reservoir's predictions (Jiang & Lai 2019). The runtime is separated into two phases, training and prediction. In the training phase, the inputs are derived from a high-resolution DNS. In the subsequent prediction phase, the inputs are derived from a low-resolution simulation with the output of our reservoir used to compute the closure. The input to a given reservoir at each time step is a vector $\mathbf{u}_k(t)$ of dimension $2w$, where the elements of $\mathbf{u}$ are the real and imaginary

parts of the final w Hermite moments before m_c :

$$\mathbf{u}(t) = [G_{m_c-w,k}(t-w\Delta t), G_{m_c-(w-1),k}(t-(w-1)\Delta t), \dots, G_{m_c-1,k}(t-\Delta t)]. \quad (4.1)$$

In both phases, the inputs form a diagonal in m and t of Hermite moments, and this structure is inspired by the phenomenon of linear Landau damping in the linearised Vlasov–Poisson system, where energy cascades from low m to high m in time. The input layer is a $D_r \times 2w$ matrix $\mathbf{W}_{\text{in}}$, with its elements drawn from the uniform distribution on $[-\sigma, \sigma]$, where σ is a parameter. Critically, $\mathbf{A}$ and $\mathbf{W}_{\text{in}}$ remain fixed after initialisation. Only the output layer, $\mathbf{W}_{\text{out}}$, is trained. The reservoir state vector evolves with the equation

$$\mathbf{r}_k(t + \Delta t) = \tanh[\mathbf{A}\mathbf{r}_k(t) + \mathbf{W}_{\text{in}}\mathbf{u}(t)], \quad (4.2)$$

where the hyperbolic tangent activation function operates element-wise on the vector argument.

The elements of the reservoir-to-output coupling, $\mathbf{W}_{\text{out}}$, are parameters that must be tuned to train the reservoir. We introduce a hyperparameter b to exclude data from initial transients from the reservoir training process. This hyperparameter serves as a buffer of time steps between the initialisation time and the data assigned for training $\mathbf{W}_{\text{out}}$. During the training phase, defined by the times t where $b\Delta t \leq t \leq T$, the reservoir output is trained to approximate the closure moment, $G_{m_c,k}$.

The most straightforward choice for the closure evaluation would be the linear readout operation, $\mathbf{W}_{\text{out}}\mathbf{r}_k$. Empirically, that operation does not permit reservoirs to predict the dynamics of several other nonlinear systems, including the Lorenz system (Lu *et al.* 2017; Chattopadhyay, Hassanzadeh & Subramanian 2020; Pyle *et al.* 2021), Kuramoto–Sivashinsky equation (Pathak *et al.* 2018) and the global atmospheric system (Arcomano *et al.* 2020). As Lu *et al.* (2017) and Pathak *et al.* (2018) describe, the hyperbolic tangent operation in the evolution equation for the reservoir state (4.2) introduces an odd-parity symmetry to the reservoir state vector $\mathbf{r}$. As in the Kuramoto–Sivashinsky equation and the Lorenz systems studied by those authors, the quadratic nonlinear term in the Vlasov equation breaks odd-parity symmetry. In other words, the general evolution equation for the Hermite moments (3.18) is not invariant under the transformation $G_{m,k} \rightarrow -G_{m,k}$, while the evolution equation for the reservoir state (4.2) is invariant under the transformations $\mathbf{u} \rightarrow -\mathbf{u}$ and $\mathbf{r} \rightarrow -\mathbf{r}$. Following the methods of Pathak *et al.* (2018) and explored in more detail by Chattopadhyay *et al.* (2020) and Pyle *et al.* (2021), we resolve this conflict by applying a symmetry-breaking transformation ($\mathbf{r}_k \rightarrow \mathbf{r}_k^*$) to the reservoir latent state before readout:

$$r_{i,k}^*(t) = \begin{cases} r_{i,k}(t), & \text{even } i \\ r_{i,k}^2(t), & \text{odd } i. \end{cases} \quad (4.3)$$

Because the reservoir output, $\mathbf{W}_{\text{out}}\mathbf{r}_k^*(t)$, is a linear operation on $\mathbf{r}^*$, we can train the weights of $\mathbf{W}_{\text{out}}$ using linear regression and avoid the computational expense of backpropagation. To mitigate potential overfitting, we introduce a regularisation parameter, β , in the method of Tikhonov-regularised linear regression. The training problem is then to find the values of $\mathbf{W}_{\text{out}}$ which minimise

$$\sum_{j=b}^{T/\Delta t} \|G_{m_c,k}(j\Delta t) - \mathbf{W}_{\text{out}}\mathbf{r}_k^*(j\Delta t)\|^2 + \beta \|\mathbf{W}_{\text{out}}\|^2. \quad (4.4)$$

Explicitly, the closed-form solution that solves this linear regression problem is

$$\mathbf{W}_{\text{out}} = \mathbf{G}_{m_c, k} \mathbf{R}_k^{*T} (\mathbf{R}_k^* \mathbf{R}_k^{*T} + \beta \mathbf{I})^{-1}, \quad (4.5)$$

where $\mathbf{G}_{m_c, k}$ is a $2 \times (T/\Delta t - b)$ matrix consisting of the time series of the real and imaginary parts of the closure moment, $\mathbf{G}_{m_c, k}$, $\mathbf{R}_k^*$ is a $D_r \times (T/\Delta t - b)$ matrix consisting of the time series of the modified state vector $\mathbf{r}_k^*(t)$, $\mathbf{R}_k^{*T}$ is the transpose of $\mathbf{R}_k^*$ and $\mathbf{I}$ is the identity matrix.

4.1. Closure integration

Once the weights of $\mathbf{W}_{\text{out}}$ are trained, the algorithm switches to the prediction phase, and the system does not evolve moments beyond $m = m_c$. Instead, the output of the reservoir is fed back into the system of equations as a closure for the moment hierarchy. To make the discussion more concrete, we introduce a time integration operator $\mathcal{D}$, such that $\mathcal{D}[\mathbf{G}(t)] = \mathbf{G}(t + \Delta t)$, where $\mathbf{G}(t)$ is the $M \times N_k$ matrix of Fourier–Hermite amplitudes representing the state of the system at time t . Operator $\mathcal{D}$ is a generic operator, and its implementation details are flexible. For the results presented in this paper, we implement $\mathcal{D}$ using the Python libraries NumPy (Harris *et al.* 2020) and SciPy (Virtanen *et al.* 2020) with the classic fourth-order explicit Runge–Kutta method (RK4) and the third-order strong stability-preserving Runge–Kutta method (SSPRK3) as detailed by Durran (2010).

When we introduce the closure models, the time integration becomes a two-step process. For each wavenumber k , we use the time integration operator $\mathcal{D}$ to calculate the resolved moments $\mathbf{G}_{m < m_c, k}$:

$$\mathbf{G}_{m < m_c, k}(t + \Delta t) = \mathcal{D}[\mathbf{G}_{m \leq m_c}(t)]_k. \quad (4.6)$$

We then use a reservoir to predict the closure moment $\mathbf{G}_{m_c, k}$:

$$\mathbf{G}_{m_c, k}(t + \Delta t) = \mathbf{W}_{\text{out}} \mathbf{r}_k^*(t + \Delta t). \quad (4.7)$$

4.2. Computational complexity

Though performance profiling metrics for numerical solutions of partial differential equations are highly dependent on computational hardware architectures and algorithm implementation choices, we can calculate the computational complexity of the DNS and the ML closure for the cases in this paper. Both methods have an identical cost to evolve the lowest-order moments. Thus, we only report the cost of evolving moments $m_c \leq m < M$ for the DNS and the closure moment m_c for the ML method. While the time-integration operator $\mathcal{D}$ is generic, we will remove a layer of abstraction by calculating the result for the SSPRK3 algorithm. For simplicity, we will neglect the additional cost of operations on complex numbers relative to those same operations on real numbers, with the understanding that this will underestimate the cost of the DNS in comparison to the ML closure. The cost of transcendental functions including tanh depends on instruction sets. This estimate treats them as equivalent in cost to multiplication, with the understanding that this underestimates the cost of updating the reservoir state. We also neglect the cost of memory access and copy operations for both methods.

4.2.1. Complexity of DNS

First, let us introduce a new parameter $M' = M - m_c$ that represents the additional number of moments that the DNS must evolve (3.23) in comparison with the ML

closure model. Our choice of time-integration operator, SSPRK3, is a three-stage algorithm. Each stage includes an evaluation of the electric field, nonlinear term, linear streaming term and hypercollisions. The electric field is calculated by (3.26). The vector (i/k) can be stored in memory during initialisation, so the electric field calculation requires N_k multiplication operations. The nonlinear term is evaluated with a pseudo-spectral method using (3.20). Using the $\mathcal{O}(N \log N)$ scaling of the fast Fourier transform algorithm (Cooley & Tukey 1965), the total computational complexity of the nonlinear term is $3M'\mathcal{O}(N_k \log N_k)$. If the vector (ik) is also stored during initialisation, then the linear streaming term requires $6M'N_k$ operations. The dissipation matrix in (3.19) can be stored, and the cost of the regularisation term is $2M'N_k$. Finally, we include the operations needed to combine the results of each stage into the state of the system at the next time step. For SSPRK3, this requires $12M'N_k$ operations. The total cost of DNS for each time step is then

$$\text{Cost}_{\text{DNS}} = 3M'\mathcal{O}(N_k \log N_k) + 20M'N_k + N_k. \quad (4.8)$$

4.2.2. Complexity of ML closure

The ML closure evaluation has two stages: updating the latent state of each reservoir and evaluating the closure moment. There are a total of N_k reservoirs. The state of each reservoir, $\mathbf{r}_k$, updates at every time step using (4.2). This requires the sum of two matrix-vector products. The product $\mathbf{W}_{\text{in}}\mathbf{u}$ has a cost of $4D_r w$ operations. The other product $\mathbf{A}\mathbf{r}_k$ involves a sparse matrix. Though the cost of this product is implementation-specific, we will count operations on non-zero elements, resulting in a cost of $2D_r \kappa$ operations, where $\kappa < D_r$. Then, the tanh operation is applied to the sum of these products. The total cost of the reservoir update is then $2D_r[\kappa + 2w + 1]$. The symmetry-breaking transformation (4.3) is applied to the state vector, which requires $D_r/2$ multiplication operations. Finally, each reservoir predicts a Hermite closure using (4.7), which uses $4D_r$ operations. The ML closure is linear in k . In total, the cost of calculating the ML closure for each time step is then

$$\text{Cost}_{\text{ML}} = N_k D_r [8w + 4\kappa + 13/2]. \quad (4.9)$$

After eliminating the common factor of N_k from the cost of the DNS and the ML closure, we find that as N_k increases, the dominant scaling for the DNS is $\mathcal{O}(M' \log N_k)$. In contrast, the cost of the ML closure scales linearly with the size of each reservoir D_r . For sufficiently large $(M' \log N_k)/(D_r w)$, the reservoir closure will be faster than the DNS.

5. Closure results

5.1. Linearised Vlasov–Poisson

For our first test of the ML closure, we confirm that the model accurately solves the linear limit of the Vlasov–Poisson system. First, we linearise the Vlasov equation (2.11), dropping the nonlinear term:

$$\frac{\partial g}{\partial t} + v \frac{\partial g}{\partial z} + v E F_M = 0. \quad (5.1)$$

In Fourier space, this allows us to restrict our focus to a single wavenumber, as the coupling between wavenumbers occurs only in the nonlinear term. We simulate an approximately steady-state solution by driving the system, as explored by Kanekar *et al.* (2015). We inject energy into the $m = 0$ moment, and it cascades down to finer

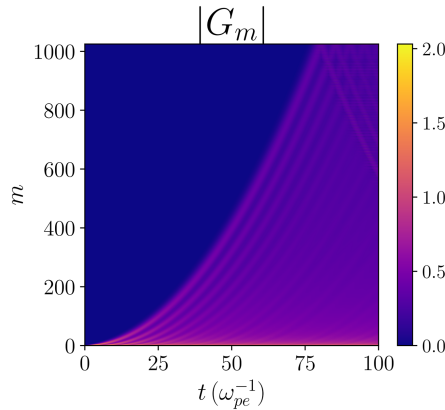


FIGURE 2. High-velocity-resolution ($M = 1025$) baseline time series of Hermite amplitudes for the driven, linearised Vlasov–Poisson system. Energy is injected as a density perturbation in the $m = 0$ Hermite moment and advects to higher moments through linear Landau damping. No hypercollisional regularisation is applied to this example, and hundreds of Hermite moments are required to prevent numerical reflection at the high- m boundary from impacting the low- m spectrum.

scales through Landau damping. We incorporate these features into the projection of this equation onto the Fourier–Hermite basis using the results of § 3:

$$\frac{\partial G_{m,k}}{\partial t} + ik \left(\sqrt{m} G_{m-1,k} + \sqrt{m+1} G_{m+1,k} \right) = \chi(t) \delta_{m,2} - E_k \delta_{m,1}, \quad (5.2)$$

where $\chi(t)$ is a forcing coefficient independently drawn at each time step from the uniform distribution on $[0, 1)$. The electric field E_k is given by (3.26). The purpose of this mechanism for energy injection is to achieve a statistical steady-state solution. Our primary systems of interest exhibit nonlinear chaos and typically settle into time-dependent steady states. While constant forcing would also achieve a steady-state solution, a time-dependent forcing mechanism more closely resembles typical problems encountered in plasmas. Other injection mechanisms, including driving the $m = 2$ moment or applying a stochastic external electric field, yield similar performance for the closure model. Figure 2 depicts a high-resolution ($M = 1025$) baseline simulation of (5.2) with $k = 0.4$.

When no hypercollisional regularisation is applied, the cascade of free energy numerically reflects at the high- m boundary. In the simulation in figure 2, this reflection occurs around $t = 80$. This phenomenon always occurs at a finite time for a given Hermite resolution (M), meaning that extending the duration of a simulation where the low-order moments are valid requires increasing the Hermite resolution. Additionally, the rate at which energy advects from low m to high m increases with Fourier wavenumber k (Parker & Dellar 2015). Therefore, though this linearised simulation with $k = 0.4$ is feasible with high velocity resolution, more Hermite resolution would be required for a nonlinear simulation with moderate spatial resolution. Some form of artificial dissipation is necessary for baseline simulations of higher-dimensional systems to be computationally feasible. We therefore apply hypercollisional regularisation to the nonlinear form of the Vlasov–Poisson system in §§ 5.2 and 5.3.

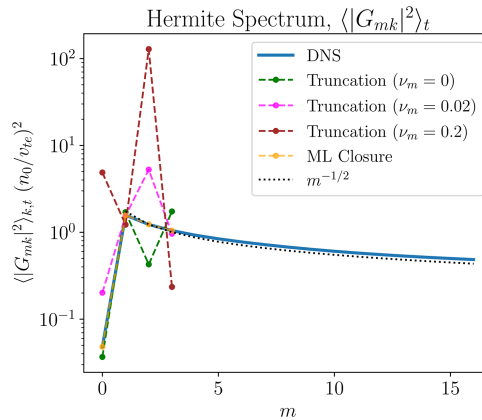


FIGURE 3. Time-averaged Hermite spectra for the ML closure model compared with the high-velocity-resolution ($M = 1025$) baseline, closure-by-truncation with three different coefficients (ν_m) for hypercollisional regularisation and the theoretical $m^{-1/2}$ scaling. The ML closure model shows strong agreement with the baseline simulation, while no value of ν_m produces an accurate spectrum.

We then train a reservoir to predict the Hermite closure moment, $G_{m_c,k}$, where we evaluate the closure at $m_c = 3$. The choice of $m_c = 3$ is motivated by the tradition of heat flux closures in fluid theory and our desire for a closure that allows the conservation laws in the $m \in [0, 1, 2]$ equations to remain intact. One motivation for a closure is to reduce the resolution required to achieve accurate simulations, so we choose the lowest valid value for m_c . We use the time series of the density, momentum and temperature moments from the time window $15 \leq t < 25$ as training data. We find that a requirement is that enough time has passed to allow energy to cascade into $m = m_c$. We report results training on data from the time window $15 < t \leq 25$, but the reservoir supports accurate spectra for other training windows as well. Figure 3 depicts the excellent agreement between the low-order spectra calculated by the high-resolution baseline and the ML closure model. The mean-squared error in the ML closure spectrum is 2.25×10^{-5} . The ML closure shows favourable performance in comparison with a naive closure by truncation, where no hypercollisional regularisation is applied. It also outperforms a closure by truncation where the hypercollisional term, $-\nu_m m^4 G_{m,k}$, is introduced to the right-hand side of (5.2) to mitigate numerical reflection at the high- m boundary. As in (3.19), this term is only applied for $m > 2$ to preserve the low-order conservation laws. For an initial comparison, ν_m is set to 2×10^{-2} to maintain finite dissipation at $m_c = 3$, where $\nu_m m_c^4$ is of order one. For reference, we also compare with the theoretical scaling $G_m \sim m^{-1/2}$ (Zocco & Schekochihin 2011; Kanekar *et al.* 2015).

The reservoir hyperparameters used to construct the model are spectral radius $\rho_{sp} = 0.6$, adjacency matrix degree of three, Tikhonov regularisation parameter $\beta = 10^{-9}$, input scaling parameter $\sigma = 0.5$ and six nodes in the reservoir. We found these hyperparameters by comparing mean-squared errors in spectra. An unguided scan led to a reduction in error from 1.75×10^{-7} to 2.26×10^{-8} . No systematic optimisation was required, though some choices of parameters led to numerically unstable time integration of (5.2). We used the same procedure to select the hyperparameters in both the weakly and strongly nonlinear cases presented in §§ 5.2 and 5.3, respectively. The spectral radius ρ_{sp} of the adjacency matrix $\mathbf{A}$ must be set to

$\rho_{sp} < 1$ for stability. The block-diagonal structure for $\mathbf{W}_{in}$ that Vlachas *et al.* (2020) used in their reservoir computing implementation yielded unstable predictions. We speculate that this may be related to differences in requirements for complex-valued data in comparison with real-valued data, but more analysis would be required to confirm. For the strongly nonlinear case in § 5.3, using a reservoir smaller than 60 nodes or including the initial transient in the training set also led to unstable solutions. The computational complexity for evaluating the ML closure is the same as in (4.9), with $N_k = 1$. The DNS cost for this case is $18M' + 1$, after removing the contributions of the nonlinear term and hypercollisions from (4.8).

5.2. Weakly nonlinear Vlasov–Poisson

Next, we reintroduce the nonlinear term, $-E(\partial g/\partial v)$, which permits coupling between wavenumbers. This coupling in Fourier space allows energy to flow across spatial scales, in addition to the already-present energy cascade in velocity space. This effect is subdominant at low amplitudes, but it becomes more significant as amplitudes increase. We first examine the low-amplitude limit in this section before proceeding to a high-amplitude case in § 5.3. In the low-amplitude limit, Landau damping continues to be the dominant effect. Here, we solve an initial-value problem, where the initial condition is a cosine density perturbation with a Maxwellian velocity component:

$$g(z, v, t = 0) = \epsilon \cos(k_0 z) F_M(v), \quad (5.3)$$

where ϵ is the initial amplitude of the perturbation normalised by the background density and Maxwellian weight and k_0 is the wavenumber of the initial perturbation. In the spectral domain, this initial condition has the convenient form

$$G_{m,k,t=0} = \frac{\epsilon}{2} \delta_{m,0} (\delta_{k,k_0} + \delta_{k,-k_0}), \quad (5.4)$$

collapsing the initial system state into a single non-zero value after applying the reality condition. To explore the linear limit of the system, we choose $\epsilon = 0.001$, a 0.1 % density perturbation. We examine the $k_0 = 0.4$ case that Brunetti, Califano & Pegoraro (2000) explored in their analysis, accounting for a factor of 10 difference in normalisations. With this low initial amplitude, we choose $N_k = 17$ Fourier wavenumbers, including the $k = 0$ mode, anticipating that the dominant energy cascade will be linear Landau damping. For the high-resolution simulations, we continue to use $M = 17$ Hermite moments, including the $m = 0$ moment. The hypercollisional regularisation coefficient is $\nu_m = 5 \times 10^{-5}$, and we set $\nu_k = 0$. This value of ν_m is chosen to maintain finite dissipation at the finest velocity scale in the simulation, such that the magnitude of the coefficient of the hypercollisional term at that scale, $\nu_m(M-1)^4$, is of order one. As the Fourier spectrum in figure 6 shows, the flux of energy to high k in this regime is small, so ν_k is not necessary. We set the size of the spatial simulation domain to $L = 5\pi$ so that $1/k_0$ is the largest wavelength resolved in the system. Figure 4 presents a comparison between the baseline, high-Hermite-resolution simulation, the ML closure model and the theoretical damping rate. The time trace for the ML closure method begins at the end of the training phase, $t = 25$. To calculate the damping rate, we use the complex root-finding algorithm of Carpentier & Dos Santos (1982), implemented in the generalised dispersion relation solver developed by Ivanov & Adkins (2023). These results demonstrate that with a low initial amplitude, our baseline simulation converges to the expected linear Landau damping rate. Additionally, figure 4 demonstrates that the ML closure

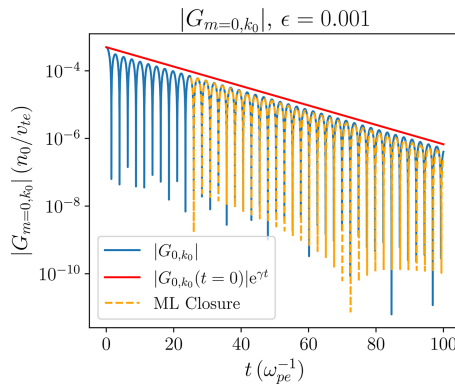


FIGURE 4. Comparison between baseline numerical solution, ML closure and theoretical damping rate of the Fourier–Hermite amplitude for an initial cosine density perturbation. The low-amplitude perturbation shows strong agreement with the theoretical damping rate. When augmented with the ML closure, the moment solver continues to capture the behaviour well at a lower Hermite resolution of $M = 4$, as opposed to the $M = 17$ baseline.

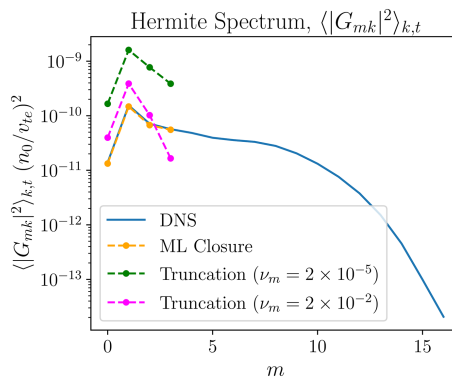


FIGURE 5. Hermite spectra of the high-velocity-resolution ($M = 17$) and truncated ($M = 4$) simulations and ML closure for the initial-value problem in figure 4. The spectra are averaged over time and Fourier wavenumber. The ML closure model permits a low-resolution simulation to accurately resolve the Hermite spectrum.

model preserves the frequency of the wave. Time traces of the $m = 1$ and $m = 2$ moments show similar performance and are presented in Appendix B.

In figure 5, we compare the ML closure model, with $m_c = 3$, to both the high-resolution ($M = 17$) baseline and two low-resolution simulations without the ML closure, i.e. closure by truncation. We also compare the Fourier spectra of these simulations in figure 6. For this problem, we place an independent reservoir at each wavenumber k and train it to learn a Hermite moment closure for that wavenumber. The ML closure shows strong agreement with the high-resolution baseline, with mean-squared errors of 1.26×10^{-42} for the Hermite spectrum and 8.01×10^{-38} for the Fourier spectrum. It successfully captures the Hermite and Fourier spectra of the system and confirms that its predictive capability continues to hold when a small, but non-zero, flux of energy flows between wavenumbers. To create a low-resolution

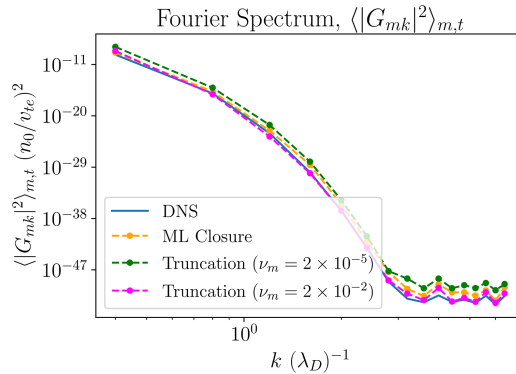


FIGURE 6. Fourier spectra of the simulations in figure 5. In this weakly nonlinear regime, the majority of the energy remains in the boxscale mode. Each reduced Hermite resolution model properly captures the rapid energy decay across k .

simulation without the ML closure, we truncate the moment hierarchy at $m_c = 3$ and set $G_{m_c+1,k} = 0$ in the evolution equation for $G_{m_c,k}$. As in § 5.1, we also compare to a low-resolution simulation with increased hypercollisional regularisation, with ν_m set to 2×10^{-2} to maintain finite dissipation at the finest resolved scales. An intuitive explanation for the poor performance of the truncated simulations in figure 5 can be described in relation to the general form of the evolution equation for this problem, (3.18). The truncation process removes the $G_{m+1,k}$ term from the final moment equation in the truncated system, eliminating an effective energy sink from that moment. Without access to the proper dissipation channel of smaller scales in velocity space, the truncated system experiences some energy reflection and pile-up in its smallest resolved scales in velocity. In contrast, the ML closure learns the pattern of dissipation through Landau damping and serves as a better boundary condition than truncation.

We find that the reservoir hyperparameters that result in the best closure performance for this initial condition are spectral radius $\rho_{sp} = 0.6$, Tikhonov regularisation parameter $\beta = 10^{-7}$ and $T = 25$ normalised time units for training. For each wavenumber, we set the reservoir input scaling parameter σ to normalise the reservoir inputs by the time average of the Hermite amplitude at the moment before the closure:

$$\sigma_k = \frac{1}{\langle |G_{m_c-1,k,t}| \rangle_t}. \quad (5.5)$$

This preserves the dynamic range of the hyperbolic tangent nonlinear activation function by mitigating the possibility of saturation to -1 or 1 . As in § 5.1, we set the degree of the adjacency matrix to three. Finally, we choose a reservoir size of two nodes per input value, totalling 12 nodes.

5.3. Strongly nonlinear Vlasov–Poisson

Finally, we create an ML closure model for the strongly nonlinear regime of the Vlasov–Poisson system. When the amplitude of the initial density perturbation becomes significant relative to the background, the nonlinear term dominates over the energy cascade in linear Landau damping. Figure 7 presents the numerical solution to the initial-value problem from § 5.2, with ϵ increased to 0.18. When compared

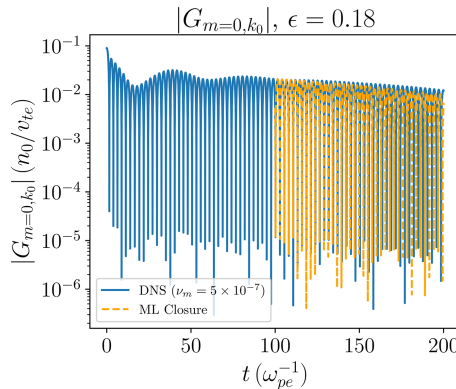


FIGURE 7. Simulated Fourier-Hermite amplitude for a high-initial-amplitude (18 % of background) cosine density perturbation. The dynamics exhibits strongly nonlinear behaviour, attenuating the damping of the mode. As in the low-amplitude case, the ML closure model captures well the frequency and amplitude of the wave.

with [figure 4](#), the behaviour at high amplitude deviates from the linear theory. Additionally, a convergence study indicated that more Hermite moments ($M = 65$) are necessary to accurately resolve the low-order spectrum than in the weakly nonlinear case. (For more detail on this, see [Appendix A](#).) We set ν_m to 5×10^{-7} , following the procedure we used to determine the baseline hypercollisional regularisation parameter for the linearised and weakly nonlinear cases. From the start of the simulation until approximately $t = 25$, the mode damps at a faster rate than in the linear theory. After the initial transient, the mode saturates to a steady state. When the ML closure model is introduced, the lower-Hermite-resolution ($M = 4$) system maintains an accurate frequency and slightly overdamps the amplitude. The training phase for the ML closure model was set to $50 \leq t < 100$ to avoid the most severe initial transient behaviour, and the time trace for the ML closure begins at $t = 100$.

As in the previous section, [figure 8](#) displays a comparison between the Hermite spectra of the high-resolution baseline, the ML closure model and two truncated low-resolution simulations, where one has an increased amount of hypercollisional regularisation to account for the lower resolution. At this higher initial amplitude, the Hermite spectrum of the baseline does not decrease monotonically, creating a more challenging scenario for the reservoirs to model. At the same value of ν_m , both the truncated low-resolution simulation and the ML closure model accurately capture the Hermite spectrum of the system at low m , but as [figure 9](#) reveals, only the ML closure model also agrees with the Fourier spectrum. Mean-squared errors in the ML closure spectra are 3.66×10^{-18} and 1.19×10^{-14} for the Hermite and Fourier spectra, respectively. The ML closure successfully captures both spectra of the system, reducing the required velocity-space resolution by a factor of 16. We observe some oscillatory behaviour in the Fourier spectrum of the ML closure, but our tests do not identify a mechanism that causes this behaviour. We obtain the reservoir hyperparameters by the same method as in the previous section, adjusting them to $\beta = 10^{-6}$, $T = 50$ and 120 nodes in each reservoir. Other parameters remain the same.

A potentially surprising result is that the Hermite spectrum of the naively truncated simulation is more accurate than its Fourier spectrum, despite the fact that the truncation occurs in Hermite space. We intuit that this result is due to a pile-up of

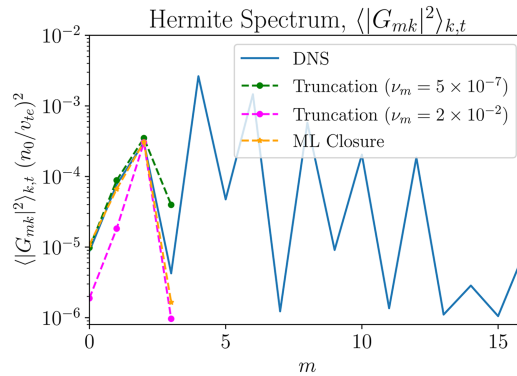


FIGURE 8. Hermite spectra of the high-velocity-resolution ($M = 65$) and truncated ($M = 4$) simulations and ML closure for the initial-value problem in figure 7 averaged over k and t . While both the ML closure model and truncated simulation agree with the high-resolution DNS, the ML closure model shows closer agreement.

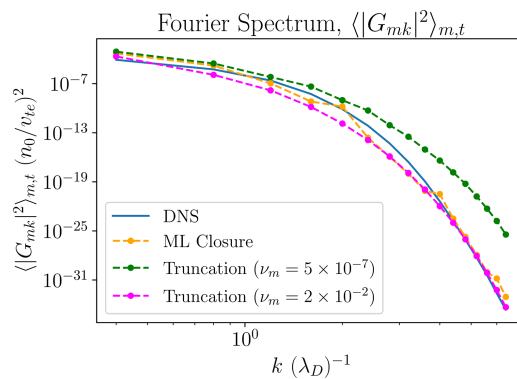


FIGURE 9. Fourier spectra of the simulations in figure 8. The ML closure resolves the wavenumber spectrum more accurately than the truncated simulations, improving simulation accuracy at low resolution in velocity space.

excess free energy at the closure moment, which then nonlinearly advects in Fourier space. Adjusting the strength of the hypercollisions slightly improves the accuracy of the Fourier spectrum, at the cost of overdamping the density and momentum moments. An analysis of the flux of free energy in this system similar to the work of Meyrand *et al.* (2019) may be used to investigate this result further, but that is beyond the scope of this work.

6. Summary and conclusion

In this paper, we have used reservoir computing to present an ML, velocity-space closure model for the hierarchy of Hermite moments in the one-dimensional Vlasov–Poisson system. The closure model serves as a proof-of-principle that reservoir computing can be used to reduce the simulation domain requirements for studying plasma dynamics.

A logical objective to pursue is to extend the closure to reduce the required resolution in the Fourier representation of position space. In practice, gyrokinetic turbulence simulations commonly use significantly more resolution for configuration space than they do for velocity space, in part because of the near-Maxwellian nature of the velocity distributions. Therefore, a Fourier-space closure model would potentially be more desirable than a velocity-space closure. However, the structure of the nonlinear term poses a challenge to achieving that goal. In the Vlasov–Poisson system, the coupling between resolved and unresolved scales in velocity space occurs as a sum of nearest neighbours in Hermite space. This locality in the spectral domain defines a clear, single closure term. In contrast, the convolution in the nonlinear term establishes non-local interactions across scales in Fourier space. An ML closure for Fourier space would require a different structure from the one developed for this paper.

The eventual goal is to build closure models that can reduce the domain requirements for higher-dimensional turbulence simulations, including the full gyrokinetic equation. A next step towards that goal would be to test this velocity-space closure in that system. There has been recent interest in applying spectral formulations of velocity space to gyrokinetic simulations, particularly with a Hermite basis used for parallel velocity and a Laguerre polynomial basis for perpendicular velocity (Jorge *et al.* 2017; Mandell *et al.* 2018; Hoffmann *et al.*, 2023a,b; Frei *et al.* 2023, 2024). A successful implementation of an ML closure for Fourier–Hermite–Laguerre codes, including Gyacomio (Hoffmann *et al.* 2023b) and GX (Mandell *et al.* 2024), may improve their capabilities to resolve turbulence statistics with lower resolution requirements. Empirically, both codes calculate turbulent heat fluxes with high accuracy using eight or fewer Laguerre modes, suggesting that future work on ML closures should prioritise reductions in spatial and parallel velocity resolution requirements.

An ideal closure model would have a compact, symbolic form from which physical intuition can be derived. Significant recent progress towards interpretable, symbolic closures has been achieved using sparse regression techniques (Cheng *et al.* 2023; Donaghy & Germaschewski 2023; Ingelsten *et al.* 2024). One challenge that these algorithms face is the difficulty of finding symbolic closures that are not local in space. The analytic closure derived by Hammett & Perkins (1990) for the linear limit of the system in this paper is local in Fourier space, but it includes a non-local Hilbert transform when expressed in configuration space. Closures for the nonlinear form of this system or other systems of interest may also require information that is non-local in configuration space, but local in a spectral representation. The ML closure developed in this paper and by Huang *et al.* (2025) apply a Fourier representation to the data, providing the neural networks direct access to locality in spatial scales, and the velocity-space closure in this paper is also local in Hermite space. Though interpreting moment closures that use neural networks is more challenging than closures derived by sparse regression techniques, future work may incorporate feature importance identification methods like those formalised by Lundberg & Lee (2017).

Acknowledgements

We express our gratitude to R. Gaur, N.R. Mandell and M. Nastac for helpful discussions about turbulence, M. Almanza, E.P. Alves, D. Carvalho, J. Chou, F. Fiuza, G. Guttormsen, B. Jang, N.F. Loureiro, M. McGrae-Menge, J. Pierce and

A. Velberg for their perspectives on subgrid modelling and E. Ott, D. Patel and A. Wikner for sharing their insights on reservoir computing.

Editor Nuno Loureiro thanks the referees for their advice in evaluating this article.

Funding

This work was made possible through the support of the United States Department of Energy under contract numbers DE-FG02-93ER54197 and DE-SC0022039.

Declaration of interest

M.L. is a consultant for Type One Energy.

Appendix A. Hermite resolution convergence

In § 5.3, we demonstrated that reservoir computing can be used to construct an ML closure that reduces the number of moments required to capture the low-order spectrum of the strongly nonlinear regime of the Vlasov–Poisson system by a factor of 16. In figure 10, we present the convergence study that we used to determine the high-resolution ($M = 65$) baseline for that case. The low-order moments represent important physical quantities, including density, momentum and energy. Therefore, accurately resolving those moments is a higher priority than resolving the fine-scale structure in velocity space that the high-order moments capture. In figure 10, the lowest Hermite resolution that captures the $m = 0$ and $m = 1$ density and momentum moments within a factor of 2 is $M = 65$, so that resolution was selected as the baseline case. Figure 11 shows a root-mean-square error (RMSE) metric for the low-order Hermite spectra at different Hermite resolutions, where we take the $M = 513$ spectrum as ground truth. Inaccuracy in the density and momentum moments leads to increased RMSE for the $M = 17$ and $M = 33$ cases, and subsequent errors converge exponentially, beginning with $M = 65$.

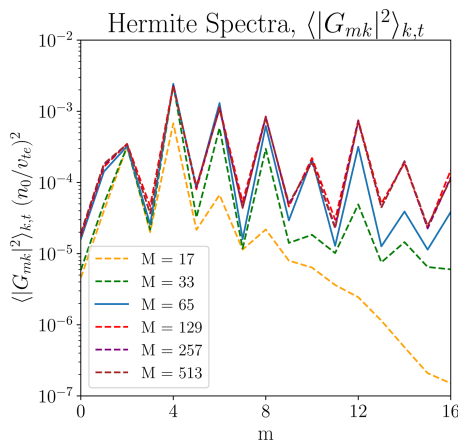


FIGURE 10. Convergence study of the Hermite spectra for the strongly nonlinear initial-value problem from § 5.3. We solve (3.21)–(3.24) without the ML closure at increasing levels of resolution in velocity space. We selected $M = 65$ moments for the high-resolution baseline case in § 5.3, as that case is the first to show convergence in the density and momentum moments.

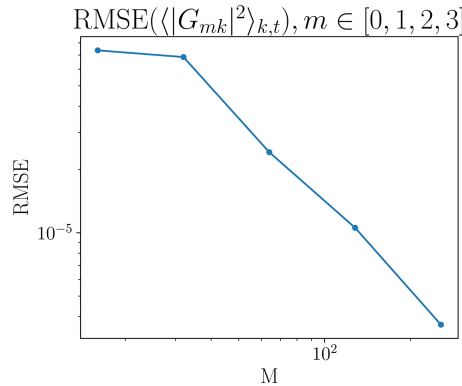


FIGURE 11. Root-mean-square error in the low-order ($m \in [0, 1, 2, 3]$) Hermite spectra at the resolutions plotted in figure 10, as compared with the $M = 513$ case. The $M = 17$ and $M = 33$ cases have similar errors, and exponential convergence occurs afterward.

Appendix B. Time evolution of low-order moments

Sections 5.2 and 5.3 demonstrate that the ML closure can capture spectra of the Vlasov–Poisson system and that the time evolution of the density moment is accurate. Figures 12–15 depict time traces of the $m = 1$ and $m = 2$ moments for the cases in those sections. The ML closure resolves the frequencies of the oscillations well. It also successfully captures the amplitude peaks soon after training ends, but it fails to match many of the minima. At long times after training, the amplitudes resulting from the ML closure begin to deviate slightly from the DNS amplitudes. In the weakly nonlinear regime presented in figures 12 and 13, the ML closure leads to a slightly lower amplitude at the end of the simulation for both the $m = 1$ and $m = 2$ moments. In the strongly nonlinear regime in figure 14, the ML closure trends towards a lower amplitude for $m = 1$ than in the DNS baseline. However, for $m = 2$ in that regime, figure 15 shows that the ML closure trends towards a slightly higher amplitude than the DNS. To quantitatively analyse this behaviour, we report damping rates for the DNS and ML closure in table 1. Damping rates were calculated by first extracting the local maxima from each time trace and then solving

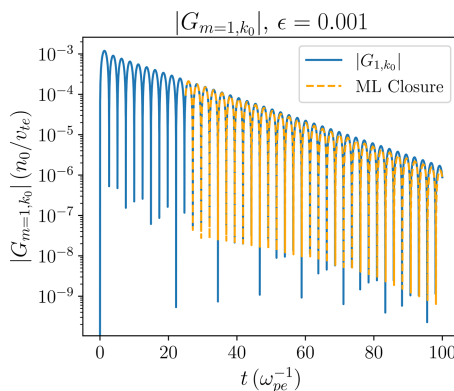


FIGURE 12. Time trace of the $m = 1$ moment for the low-amplitude initial-value case in § 5.2.

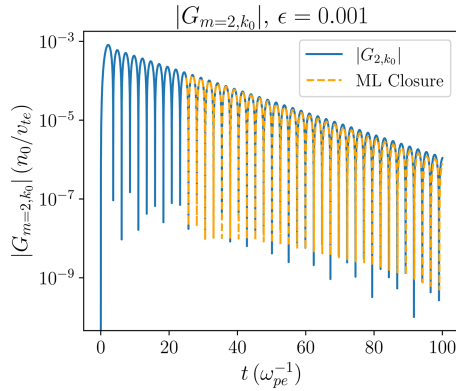


FIGURE 13. Time trace of the $m = 2$ moment for the low-amplitude initial-value case in § 5.2.

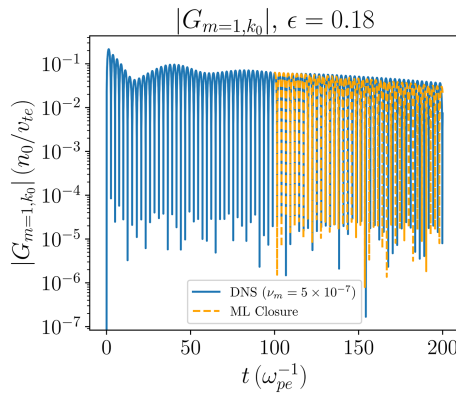


FIGURE 14. Time trace of the $m = 1$ moment for the high-amplitude initial-value case in § 5.3.

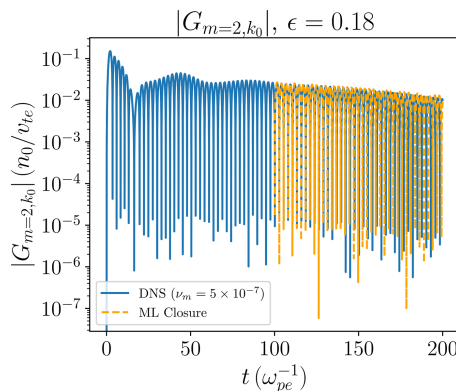


FIGURE 15. Time trace of the $m = 2$ moment for the high-amplitude initial-value case in § 5.3.

ϵ	m	DNS damping rate (ω_{pe})	ML closure damping rate (ω_{pe})
0.001	1	-6.581×10^{-2}	-6.911×10^{-2}
0.001	2	-6.581×10^{-2}	-6.911×10^{-2}
0.18	1	-5.476×10^{-3}	-7.572×10^{-3}
0.18	2	-8.732×10^{-3}	-7.518×10^{-3}

TABLE 1. Damping rates calculated from the initial-value simulations in figures 12–15. Here, ϵ is the amplitude of the initial cosine density perturbation and m is the Hermite moment number. Damping rates were calculated by solving a linear regression problem for the natural logarithm of the local maxima in each time series.

a linear regression problem for the natural logarithm of those maxima. An interesting observation is that the $m = 1$ and $m = 2$ results are identical for the low-amplitude case. In contrast, in the high-amplitude case, the DNS has a faster damping rate for $m = 2$ than for $m = 1$. In the strongly nonlinear regime, the ML closure yields similar damping rates for $m = 1$ and $m = 2$, but they diverge from the DNS damping rates.

REFERENCES

- ABRAMOWITZ, M. & STEGUN, I.A. 1972 *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. 10th edn., pp. 773–782. U.S. Dept. of Commerce, National Bureau of Standards, Dover Publications.
- ADKINS, T. & SCHEKOCHIHIN, A.A. 2018 A solvable model of Vlasov-kinetic plasma turbulence in Fourier–Hermite phase space. *J. Plasma Phys.* **84**, 905840107.
- ARCOMANO, T., SZUNYOGH, I., PATHAK, J., WIKNER, A., HUNT, B.R. & OTT, E. 2020 A machine learning-based global atmospheric forecast model. *Geophys. Res. Lett.* **47**, e2020GL087776.
- ARMSTRONG, T.P. 1967 Numerical studies of the nonlinear Vlasov equation. *Phys. Fluids* **10**, 1269–1280.
- BAÑÓN NAVARRO, A., TEACA, B., JENKO, F., HAMMETT, G.W., HAPPEL, T. & ASDEX Upgrade Team 2014 Applications of large eddy simulation methods to gyrokinetic turbulence. *Phys. Plasmas* **21**, 032304.
- BAÑÓN NAVARRO, A., MOREL, P., ALBRECHT-MARC, M., CARATI, D., MERZ, F., GÖRLER, T. & JENKO, F. 2011 Free energy cascade in gyrokinetic turbulence. *Phys. Rev. Lett.* **106**, 055001.
- BARNES, M., PARRA, F.I. & SCHEKOCHIHIN, A.A. 2011 Critically balanced ion temperature gradient turbulence in fusion plasmas. *Phys. Rev. Lett.* **107**, 115003.
- BOYD, J.P. 2000 *Chebyshev and Fourier Spectral Methods*. 2nd edn., pp. 1–17. Dover Publications.
- BRUNETTI, M., CALIFANO, F. & PEGORARO, F. 2000 Asymptotic evolution of nonlinear Landau damping. *Phys. Rev. E* **62**, 4109.
- CARPENTIER, M.P. & DOS SANTOS, A.F. 1982 Solution of equations involving analytic functions. *J. Comput. Phys.* **45**, 210.
- CASTAGNA, J., SCHIAVELLO, F., ZANISI, L. & WILLIAMS, J. 2024 StyleGAN as an AI deconvolution operator for large eddy simulations of turbulent plasma equations in BOUT++. *Phys. Plasmas* **31**, 033902.
- CHATTOPADHYAY, A., HASSANZADEH, P. & SUBRAMANIAN, D. 2020 Data-driven predictions of a multiscale Lorenz 96 chaotic system using machine-learning methods: reservoir computing, artificial neural network, and long short-term memory network. *Nonlinear Process. Geophys.* **27**, 373–389.
- CHENG, W., FU, H., WANG, L., DONG, C., JIN, Y., JIANG, M., MA, J., QIN, Y. & LIU, K. 2023 Data-driven, multi-moment fluid modeling of Landau damping. *Comput. Phys. Commun.* **282**, 108538.
- CLAVIER, B., ZARZOSO, D., DEL-CASTILLO-NEGRETTE, D. & FRÉNOT, E. 2025 Generative-machine-learning surrogate of plasma turbulence. *Phys. Rev. E* **111**, L013202.

- COOLEY, J.W. & TUKEY, J.W. 1965 An algorithm for the machine calculation of complex Fourier series. *Math. Comput.* **19**, 297–301.
- DEARDORFF, J.W. 1970 A numerical study of three-dimensional turbulent channel flow at large Reynolds numbers. *J. Fluid Mech.* **41**, 453–480.
- DEGRAVE, J., *et al.* 2022 Magnetic control of tokamak plasmas through deep reinforcement learning. *Nature* **602**, 414–419.
- DONAGHY, J. & GERMASCHESKI, K. 2023 In search of a data-driven symbolic multi-fluid ten-moment model closure. *J. Plasma Phys.* **89**, 895890105.
- DURRAN, D.R. 2010 *Numerical Methods for Fluid Dynamics*. **Chapter 2**, 2nd edn. Springer.
- FREI, B.J., HOFFMANN, A.C.D., RICCI, P., BRUNNER, S. & TECCHIOLO, Z. 2023 Moment-based approach to the flux-tube linear gyrokinetic model. *J. Plasma Phys.* **89**, 905890414.
- FREI, B.J., MENCKE, J. & RICCI, P. 2024 Full-F turbulent simulation in a linear plasma device using a gyro-moment approach. *Phys. Plasmas* **31**, 012301.
- FU, Y., ELTON, D., ERICKSON, K., KLEIWEGT, K., LUPIN-JIMENEZ, L., BOYER, M.D., EIDIETIS, N., BARBOUR, N., IZACARD, O. & KOLEMEN, E. 2020 Machine learning control for disruption and tearing mode avoidance. *Phys. Plasmas* **27**, 022501.
- GIBELLI, L. & SHIZGAL, B.D. 2006 Spectral convergence of the Hermite basis function solution of the Vlasov equation: the free-streaming term. *J. Comput. Phys.* **219**, 477–488.
- GOLDREICH, P. & SRIDHAR, S. 1995 Toward a theory of interstellar turbulence. II. Strong Alfvénic turbulence. *Astrophys. J.* **438**, 763–775.
- GRANT, F.C. & FEIX, M.R. 1967 Fourier–Hermite solutions of the vlasov equations in the linearized limit. *Phys. Fluids* **10**, 696–702.
- GREIF, R., JENKO, F. & THUERLEY, N. 2023 Physics-preserving AI-accelerated simulations of plasma turbulence. arXiv: [2309.16400](https://arxiv.org/abs/2309.16400).
- HAMMETT, G.W. & PERKINS, F.W. 1990 Fluid moment models for Landau damping with application to the ion-temperature-gradient instability. *Phys. Rev. Lett.* **64**, 3019.
- HARRIS, C.R., *et al.* 2020 Array programming with NumPy. *Nature* **585**, 357–362.
- HASEGAWA, A. & MIMA, K. 1978 Pseudo-three-dimensional turbulence in magnetized nonuniform plasma. *Phys. Fluids* **21**, 87–92.
- HIGHCOCK, E.G., MANDELL, N.R., BARNES, M. & DORLAND, W. 2018 Optimisation of confinement in a fusion reactor using a nonlinear turbulence model. *J. Plasma Phys.* **84**, 905840208.
- HOFFMANN, A.C.D., FREI, B.J. & RICCI, P. 2023a Gyrokinetic moment-based simulations of the Dimits shift. *J. Plasma Phys.* **89**, 905890611.
- HOFFMANN, A.C.D., FREI, B.J. & RICCI, P. 2023b Gyrokinetic simulations of plasma turbulence in a Z-pinch using a moment-based approach and advanced collision operators. *J. Plasma Phys.* **89**, 905890214.
- HUANG, Z., DONG, C. & WANG, L. 2025 Machine-learning heat flux closure for multi-moment fluid modeling of nonlinear landau damping. *Proc. Natl. Acad. Sci. USA* **122**, e2419073122.
- INGELSTEN, E.R., MCGRAE-MENGE, M.C., ALVES, E.P. & PUSZTAI, I. 2024 Data-driven discovery of a heat flux closure for electrostatic plasma phenomena. arXiv: [2411.18358](https://arxiv.org/abs/2411.18358).
- IVANOV, P.G. & ADKINS, T. 2023 An analytical form of the dispersion function for local linear gyrokinetics in a curved magnetic field. *J. Plasma Phys.* **89**, 905890213.
- JAEGER, H. 2001 The “echo state” approach to analysing and training recurrent neural networks. GMD Report 148. German National Research Center for Information Technology.
- JALALVAND, A., KAPTANOGLU, A.A., GARCIA, A.V., NELSON, A.O., ABBATE, J., AUSTIN, M.E., VERDOOLAEGE, G., BRUNTON, S.L., HEIDBRINK, W.W. & KOLEMEN, E. 2022 Alfvén eigenmode classification based on ECE diagnostics at DIII-D using deep recurrent neural networks. *Nucl. Fusion* **62**, 026007.
- JIANG, J. & LAI, Y.-C. 2019 Model-free prediction of spatiotemporal dynamical systems with recurrent neural networks: role of network spectral radius. *Phys. Rev. Res.* **1**, 033056.
- JORGE, R., RICCI, P. & LOUREIRO, N.F. 2017 A drift-kinetic analytical model for scrape-off layer plasma dynamics at arbitrary collisionality. *J. Plasma Phys.* **86**, 905830606.
- JOYCE, G., KNORR, G. & MEIER, H.K. 1971 Numerical integration methods of the Vlasov equation. *J. Comput. Phys.* **8**, 53–63.

- KANEKAR, A., SCHEKOCHIHIN, A.A., DORLAND, W. & LOUREIRO, N.F. 2015 Fluctuation-dissipation relations for a plasma-kinetic Langevin equation. *J. Plasma Phys.* **81**, 305810104.
- KERBOUA-BENLARBI, S., NOUAILLETAS, R., FAUGERAS, B., NARDON, E. & MOREAU, P. 2024 Magnetic control of WEST plasmas through deep reinforcement learning. *IEEE Trans. Plasma Sci.* **52**, 3698.
- KIM, P., *et al.* 2024 Optimization of nonlinear turbulence in stellarators. *J. Plasma Phys.* **90**, 905900402.
- KOLMOGOROV, A.N. 1991 The local structure of turbulence in incompressible viscous fluid for very large Reynolds numbers. *Proc. R. Soc. Lond. Ser. A* **434**, 9–13.
- KRAICHNAN, R.H. 1959 The structure of isotropic turbulence at very high Reynolds numbers. *J. Fluid Mech.* **5**, 497–543.
- LOUREIRO, N.F., DORLAND, W., FAZENDEIRO, L., KANEKAR, A., MALLET, A., VILELAS, M.S. & ZOCCO, A. 2016 Viriato: a Fourier–Hermite spectral code for strongly magnetized fluid-kinetic plasma dynamics. *Comput. Phys. Commun.* **206**, 45–63.
- LOUREIRO, N.F., SCHEKOCHIHIN, A.A. & ZOCCO, A. 2013 Fast collisionless reconnection and electron heating in strongly magnetized plasmas. *Phys. Rev. Lett.* **111**, 025002.
- LU, Z., PATHAK, J., HUNT, B., GIRVAN, M., BROCKETT, R. & OTT, E. 2017 Reservoir observers: model-free inference of unmeasured variables in chaotic systems. *Chaos* **27**, 041102.
- LUNDBERG, S.M. & LEE, S.-I. 2017 A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems* (ed. Guyon, I., Von Luxburg, U., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S. & Garnett, R.), vol. **30**, pp. 4765–4774. Curran Associates, Inc.
- MA, C., ZHU, B., XU, X.-Q. & WANG, W. 2020 Machine learning surrogate models for Landau fluid closure. *Phys. Plasmas* **27**, 042502.
- MAASS, W., NATSCHLÄGER, T. & MAKRAM, H. 2002 Real-time computing without stable states: a new framework for neural computation based on perturbations. *Neural Comput.* **14**, 2531–2560.
- MANDELL, N.R., DORLAND, W., ABEL, I., GAUR, R., KIM, P., MARTIN, M. & QIAN, T. 2024 GX: a GPU-native gyrokinetic turbulence code for tokamak and stellarator design. *J. Plasma Phys.* **90**, 905900402.
- MANDELL, N.R., DORLAND, W. & LANDREMAN, M. 2018 Laguerre-Hermite pseudo-spectral velocity formulation of gyrokinetics. *J. Plasma Phys.* **84**, 905840108.
- MEYRAND, R., KANEKAR, A., DORLAND, W. & SCHEKOCHIHIN, A.A. 2019 Fluidization of collisionless plasma turbulence. *Proc. Natl. Acad. Sci. USA* **116**, 1185–1194.
- PARKER, J.T. & DELLAR, P.J. 2015 Fourier–Hermite spectral representation for the Vlasov–Poisson system in the weakly collisional limit. *J. Plasma Phys.* **81**, 305810203.
- PATHAK, J., HUNT, B., GIRVAN, M., LU, Z. & OTT, E. 2018 Model-free prediction of large spatiotemporally chaotic systems from data: a reservoir computing approach. *Phys. Rev. Lett.* **120**, 024102.
- PICCIONE, A., BERKERY, J.W., SABBAGH, S.A. & ANDREOPOULOS, Y. 2020 Physics-guided machine learning approaches to predict the ideal stability properties of fusion plasmas. *Nucl. Fusion* **60**, 046033.
- PLUNK, G.G., COWLEY, S.C., SCHEKOCHIHIN, A.A. & TATSUNO, T. 2010 Two-dimensional gyrokinetic turbulence. *J. Fluid Mech.* **664**, 407–435.
- PYLE, R., JOVANOVIĆ, N., SUBRAMANIAN, D., PALEM, K.V. & PATEL, A.B. 2021 Domain-driven models yield better predictions at lower cost than reservoir computers in Lorenz systems. *Philos. Trans. R. Soc. A* **379**, 20200246.
- QIN, Y., MA, J., JIANG, M., DONG, C., FU, H., CHENG, W. & JIN, Y. 2023 Data-driven modeling of Landau damping by physics-informed neural networks. *Phys. Rev. Res.* **5**, 033079.
- REA, C., MONTES, K.J., ERICKSON, K.G., GRANETZ, R.S. & TINGUELY, R.A. 2019 A real-time machine learning-based disruption predictor on DIII-D. *Nucl. Fusion* **59**, 096016.
- REYNOLDS, O. 1895 On the dynamical theory of incompressible viscous fluids and the determination of the criterion. *Philos. Trans. R. Soc. Lond. (A)* **186**, 123–164.
- ROBERG-CLARK, G.T., PLUNK, G.G., XANTHOPOULOS, P., NÜHRENBURG, C., HENNEBERG, S.A. & SMITH, H.M. 2023 Critical gradient turbulence optimization toward a compact stellarator reactor concept. *Phys. Rev. Res.* **5**, L032030.

- RODRIGUEZ-FERNANDEZ, P., HOWARD, N.T. & CANDY, J. 2022 Nonlinear gyrokinetic predictions of SPARC burning plasma profiles enabled by surrogate modeling. *Nucl. Fusion* **62**, 076036.
- SEO, J., KIM, S., JALALVAND, A., CONLIN, R., ROTHSTEIN, A., ABBATE, J., ERICKSON, K., WAI, J., SHOUSA, R. & KOLEMEN, E. 2024 Avoiding fusion plasma tearing instability with deep reinforcement learning. *Nature* **626**, 746–751.
- TEACA, B., JENKO, F. & TOLD, D. 2017 Gyrokinetic turbulence: between idealized estimates and a detailed analysis of nonlinear energy transfers. *New J. Phys.* **19**, 045001.
- VIRTANEN, P., *et al.* 2020 SciPy 1.0: fundamental algorithms for scientific computing in python. *Nat. Methods* **17**, 261–272.
- VLACHAS, P.R., PATHAK, J., HUNT, B.R., SAPSIS, T.P., GIRVAN, M., OTT, E. & KOUMOUTSAKOS, P. 2020 Backpropagation algorithms and reservoir computing in recurrent neural networks for the forecasting of complex spatiotemporal dynamics. *Neural Netw.* **126**, 191–217.
- VLASOV, A.A. 1968 The vibrational properties of an electron gas. *Sov. Phys. Usp.* **10**, 721.
- YOSHIDA, M., *et al.* 2025 Transport and confinement physics: Chapter 2 of the special issue: on the path to tokamak burning plasma operation. *Nucl. Fusion* **65**, 033001.
- ZOCCO, A. & SCHEKOCHIHIN, A.A. 2011 Reduced fluid-kinetic equations for low-frequency dynamics, magnetic reconnection, and electron heating in low-beta plasmas. *Phys. Plasmas* **18**, 102309.