J. Plasma Phys. (2026), *vol*. 92, E149 © The Author(s), 2026. Published by Cambridge University Press.
This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted re-use, distribution and reproduction, provided the original article is properly cited. doi:10.1017/S0022377826102517

A neural operator closure for Landau damping in electrostatic plasma

Samuel Burles[1], Enrico Camporeale[1,2] and Oreste Pezzi[3]

[1]School of Physical and Chemical Sciences, Queen Mary University of London, London E1 4NS, UK
[2]Space Weather TREC, University of Colorado, Boulder, CO 80303, USA
[3]Institute for Plasma Science and Technology, Consiglio Nazionale delle Ricerche Via G. Amendola 122/D, 70126 Bari, Italy

Corresponding author: Samuel Burles, s.burles@qmul.ac.uk

(Received 31 July 2026 UTC; revision received 8 September 2026 UTC; accepted 10 September 2026 UTC)

We present a data-driven plasma fluid closure for both linear and nonlinear electrostatic Landau damping in one dimension. A Fourier neural operator (FNO) is trained online within a differentiable fluid solver, with the loss computed on trajectories produced by the closed fluid simulation rather than on individual kinetic snapshots. The closure is non-Markovian, acting on a trailing window of the resolved-moment history so as to represent the memory of the unresolved dynamics. We demonstrate that a single FNO trained in this way reproduces both linear and nonlinear Landau damping, generalises to unseen initial perturbation amplitudes lying between those used in training, and remains numerically stable when deployed in independent fluid simulations. In the nonlinear regime the learned heat flux reproduces the resolved-moment dynamics without matching the kinetic heat flux pointwise, behaving as an effective closure that carries the influence of the unresolved dynamics through the resolved-field history, though the learned specific flux is expected to depend on the numerical scheme and training data. A sensitivity analysis of the trained model shows that it computes a genuine moment-to-flux relation whose reliance on the memory window is structured rather than uniform, concentrating at lags spaced by the oscillation period.

Key words: plasma simulation, plasma nonlinear phenomena, space plasma physics

1. Introduction

Collisionless plasma dynamics is multi-scale. Kinetic effects influence large-scale structures and accurate global simulation of a large-scale system therefore requires resolving kinetic interactions across all scales (Lautenbach & Grauer 2018; Lapenta *et al.* 2022). A quote from Krall and Trivelpiece summarises this well: 'The microscopic quantities are more difficult to measure directly, but frequently they play a dominant role in determining the macroscopic properties of plasma' (Krall & Trivelpiece 1973).

Historically, efforts to model collisionless plasma have been split between two approaches: kinetic models and fluid models. Kinetic simulations resolve particle-level interactions, but become computationally prohibitive at large scales due to the high dimensionality of the phase space. Fluid models of plasma, obtained through reductions of the kinetic description, are much more computationally viable by virtue of their low dimensionality, but are ill-suited to resolving kinetic effects because of the simplifying approximations required to construct them.

Indeed, fluid models can be obtained by taking velocity moments of the Vlasov equation. The evolution of each order of velocity moment depends on the next-higher moment, producing an infinite hierarchy of equations that must be truncated at some chosen order. Truncation requires a closure relation that expresses the chosen moment in terms of lower-order moments only. Closures are therefore approximations, and the quality of this approximation determines how much of the underlying kinetic physics the resulting fluid model can recover.

The development of collisionless plasma closures began with the seminal closure of Chew, Goldberger and Low (CGL closure) (Chew, Goldberger & Low 1956), which introduced pressure anisotropy to collisionless plasma fluid models and gave rise to the adiabatic plasma fluid models (Le *et al.* 2009, 2016; Ohia *et al.* 2012). Substantial subsequent effort has focused on the phenomenon of Landau damping, beginning with the Hammett–Perkins closure (Hammett & Perkins 1990), which describes the linear phase of Landau damping through a closure formulated in Fourier space and derived from linear kinetic theory. The wider family of fluid models incorporating Landau damping, known as Landau fluids, has developed from this initial work by Hammett and Perkins (Snyder, Hammett & Dorland 1997; Goswami, Passot & Sulem 2005; Passot & Sulem 2006; Sulem & Passot 2015; Wang *et al.* 2015; Ng *et al.* 2020), including systematic higher-order closures constructed from Padé approximants of the plasma dispersion function (Hunana *et al.* 2018, 2019), which improve the fidelity of the linear kinetic response across a wider range of phase velocities. Closures derived from linear theory, such as Hammett–Perkins and its higher-order Padé descendants, are by construction limited to the regimes in which that theory applies, motivating data-driven approaches that can extend beyond them.

Recent advances in parallel computing and graphics processing unit (GPU) hardware have driven rapid progress in machine learning (ML), resulting in new techniques applied across the physical sciences. In neutral-fluid dynamics, ML approaches have been adopted with particular enthusiasm for turbulent simulation and the closure problem more broadly (Brunton, Noack & Koumoutsakos 2020; Gupta & Lermusiaux 2023; Li *et al.* 2024; Sanderse *et al.* 2024).

In plasma physics, early efforts to apply ML to the closure problem focused on reproducing known analytic closures, such as the aforementioned Hammett–Perkins closure, with neural-network surrogate models, establishing that closure relations can in principle be represented by learned models (Ma *et al.* 2020; Maulik *et al.* 2020). Subsequent work moved beyond analytic targets and trained closures directly on data from fully kinetic simulations (Liu *et al.* 2022; Joglekar & Thomas 2023; Qin *et al.* 2023; Huang, Dong & Wang 2025; McGrae-Menge *et al.* 2026) to capture physics that lies outside the range of validity of any single analytic closure theory. More recent studies have extended this approach to higher-dimensional settings and to richer phenomena, including magnetic reconnection and plasma turbulence (Laperre *et al.* 2022; Miloshevich *et al.* 2025; McGrae-Menge *et al.* 2026). We present a much more detailed review of the field in Burles & Camporeale (2026).

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

Two recent studies are of particular relevance to the present work. Wei *et al.* (2023) presented the first attempt at using the Fourier neural operator (FNO) architecture to reproduce the heat-flux divergence from a Vlasov–Ampére simulation, showing that an FNO more accurately reproduces the kinetic heat flux when compared with a more typical multilayer perceptron architecture. Huang *et al.* (2025) later demonstrated that an FNO can serve as a closure for one-dimensional Landau damping, reproducing both the linear and nonlinear regimes when deployed within a fluid solver. Their closure is trained offline (fitted as a regression problem against moments precomputed from the kinetic simulation data) and subsequently tested online (the trained model is coupled with a fluid solver and performance is judged at inference). The closure from Huang *et al.* (2025) demonstrated the viability of this method for including nonlinear Landau damping in a fluid system; however, their results were limited to only the initial damping and growth phase of the nonlinear simulation.

This offline construction is an instance of what the neutral-fluid and climate closure-modelling communities call *a-priori* training, in which the model is fitted to reproduce the true closure term on precomputed data, as opposed to *a-posteriori* training, in which the model is embedded in a differentiable solver and optimised against the resulting trajectories (MacArt, Sirignano & Freund 2021; Frezat *et al.* 2022; Sanderse *et al.* 2024; Shankar *et al.* 2024). A recurring finding of that work is that *a-priori* optimal closures can be unstable once coupled to the solver, whereas *a-posteriori* training tends to yield closures that are more stable and accurate in deployment (Frezat *et al.* 2022); the online training we employ here is the plasma closure counterpart of this *a-posteriori* strategy. This leaves two questions unresolved. First, a low pointwise error on kinetic snapshots does not guarantee numerical stability once the closure is embedded in a fluid solver, where small per-step errors can compound over thousands of time steps and drive the simulation away from the physical solution. Concrete demonstrations of learned closures in genuine online tests remain scarce (Shukla *et al.* 2022; Joglekar & Thomas 2023; Huang *et al.* 2025). Of these, Joglekar & Thomas (2023) is closest in spirit to the present work: they embed a learnable closure term (a symbolic model augmented by a neural network) in a differentiable solver and recover both linear and nonlinear Landau damping across a range of initial conditions. Our approach is complementary, using a FNO acting on an explicit trailing memory window, which lets us probe the non-Markovian structure of the learned closure directly (§ 4.5) and characterise generalisation across a dense amplitude scan.

Second, for a closure to be useful in large-scale fluid modelling it must generalise across initial conditions, ideally capturing the linear and nonlinear regimes of a given process within a single fluid model. Existing studies typically demonstrate accuracy at one or a few initial conditions rather than across a span of initial conditions. The question of how reliably a learned closure transfers to unseen initial states has not been systematically addressed.

In this work we address both of these gaps. We embed an FNO closure directly within a differentiable fluid solver and train it online, computing the loss on trajectories produced by the closed fluid simulations rather than on individual kinetic snapshots (by trajectory we mean the time series of the resolved fluid moments n, u, p, E over the rollout, not any single field). By training the closure in the same setting in which it is subsequently deployed, online training yields a model that is, by construction, suited to stable use in independent fluid simulation. We then demonstrate that a single FNO closure trained in this manner reproduces both linear and nonlinear electrostatic Landau damping in one dimension, and that it

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

generalises to initial perturbation amplitudes unseen during training, interpolating between amplitudes included in the training set. This represents a concrete step towards a broadly applicable machine-learned plasma closure, capable of acting as a component within larger fluid simulations across the range of initial conditions such simulations encounter.

A distinguishing feature of our closure is that it is non-Markovian: rather than mapping the instantaneous resolved moments to the heat flux, it acts on a trailing window of the recent moment history. This choice is motivated by the observation that the exact closure obtained by projecting out the unresolved dynamics carries memory of the resolved fields, as made precise by the Mori–Zwanzig formalism, in which the unresolved contribution appears as a memory integral over the history of the resolved variables (Gouasmi, Parish & Duraisamy 2017; Parish & Duraisamy 2017). Finite-memory neural closures of this kind have been explored for reduced-order models of neutral fluids (Wang, Ripamonti & Hesthaven 2020); here the memory encodes the phase-mixing history that underlies Landau damping. Having trained the closure, we further probe what it has learned through a sensitivity analysis of its inputs, providing evidence that it computes a genuine moment-to-flux relation rather than extrapolating its own output, and that its reliance on the memory window is structured rather than uniform, concentrating at lags spaced by the oscillation period.

We choose Landau damping of a Langmuir wave as a test case for this method as it is well established in the existing machine-learned closure literature, and provides a simple case of kinetic dynamics with well-known behaviour in both the linear and nonlinear regimes. The two regimes are separated by the trapping (bounce) time $\tau_{\mathrm{tr}} = (m/eE_0k)^{1/2}$, after which the linear theory of Landau damping breaks down; here k is the wavenumber and E_0 is the amplitude of the electric field (O'Neil 1965). For a fixed wavenumber, this time scale is set by the perturbation amplitude, so scanning the initial amplitude, as we do here, moves the system continuously from the linear ($t \ll \tau_{\mathrm{tr}}$) to the trapping-dominated nonlinear ($t \gtrsim \tau_{\mathrm{tr}}$) regime.

The remainder of the paper is organised as follows. Section 2 reviews the general plasma closure problem and sets out the one-dimensional electrostatic fluid models, framing the closure problem in the form used here. Section 3 describes the FNO architecture and the online training procedure within the differentiable fluid solver. Section 4 presents the results for an FNO trained on single initial perturbation amplitudes representing a linear case and a nonlinear case of Landau damping, the generalisation of a single FNO trained across multiple initial perturbation amplitudes, a comparison between the learned effective heat flux and the kinetic heat flux, and a sensitivity analysis probing what the closure has learned. Section 5 discusses the implications of our findings and outlines directions for future work.

2. Plasma closure background

Following the procedure set out in Swanson (2003), we write the collisionless Vlasov equation as

$$\frac{\partial f_\alpha}{\partial t} + \boldsymbol{v} \cdot \nabla f_\alpha + \frac{q_\alpha}{m_\alpha} (\boldsymbol{E} + \boldsymbol{v} \times \boldsymbol{B}) \cdot \nabla_{\boldsymbol{v}} f_\alpha = 0. \tag{2.1}$$

Here $f_\alpha(\boldsymbol{x}, \boldsymbol{v}, t)$ is the phase-space distribution function of species α, $\boldsymbol{v}$ is the velocity, q_α and m_α are the charge and mass of the species, $\boldsymbol{E}$ and $\boldsymbol{B}$ are the electric and magnetic fields, and ∇ and ∇_v denote gradients in configuration and velocity space. The subscripts α denote the particle species, which we will henceforth drop as we

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

consider a one-fluid model of electrons with a static ion background distribution. We introduce the velocity monomial $\phi_j(v) = v^j$ of moment order j (the one-dimensional form used throughout; in higher dimensions the order-j moment is the rank-j velocity tensor $\langle v^{\otimes j} \rangle$, to which this reduces in the one-dimensional electrostatic setting), and define the moment-taking process as averaging over the velocity:

$$\langle \phi_j(\boldsymbol{v}) \rangle = \int \phi_j f \, \mathrm{d}\boldsymbol{v}. \tag{2.2}$$

We then proceed to take a general velocity moment of the Vlasov equation in order to obtain the evolution equation for a general-order moment, i.e.

$$\frac{\partial}{\partial t}\langle \phi_j \rangle + \boldsymbol{\nabla} \cdot \langle \phi_j \boldsymbol{v} \rangle - \langle \boldsymbol{A} \cdot \boldsymbol{\nabla}_{\boldsymbol{v}} \phi_j \rangle = 0, \tag{2.3}$$

where $\boldsymbol{A} = (q_\mathrm{e}/m)(\boldsymbol{E} + \boldsymbol{v} \times \boldsymbol{B})$, $q_\mathrm{e} = -e$ is the electron charge (with e the elementary charge) and m is the electron mass. It is clear from the second term in this equation that the evolution of each moment is dependent on the next highest-order moment, resulting in an infinite moment hierarchy of evolution equations that must be truncated in order to obtain a fluid model. A closure relation is an equation that approximates the highest-order moment retained in a fluid model using only lower-order moments.

Henceforth, we restrict ourselves to the case of a one-dimensional, electrostatic electron plasma with a static ion background, as this will be our set-up for investigating a closure relation for Landau damping. Fluid models of plasma are obtained by taking velocity moments of the Vlasov equation. The fluid variables are defined as the velocity moments of the one-dimensional electron distribution function $f(x, v, t)$:

$$n(x, t) = \int f(x, v, t) \, \mathrm{d}v, \tag{2.4}$$

$$u(x, t) = \frac{1}{n} \int v f(x, v, t) \, \mathrm{d}v, \tag{2.5}$$

$$p(x, t) = m \int (v - u)^2 f(x, v, t) \, \mathrm{d}v, \tag{2.6}$$

$$q(x, t) = m \int (v - u)^3 f(x, v, t) \, \mathrm{d}v. \tag{2.7}$$

Here n is the number density, u is the bulk (mean) velocity, p is the scalar pressure, q is the heat flux and m is the electron mass as above.

Applying the moment-taking procedure, up to second order, to the Vlasov equation in our one-dimensional electrostatic setting yields the fluid evolution equations

$$\frac{\partial n}{\partial t} = -\frac{\partial}{\partial x}(nu), \tag{2.8}$$

$$\frac{\partial u}{\partial t} = -u\frac{\partial u}{\partial x} - \frac{1}{mn}\frac{\partial p}{\partial x} + \frac{q_\mathrm{e}}{m}E, \tag{2.9}$$

$$\frac{\partial p}{\partial t} = -u\frac{\partial p}{\partial x} - 3p\frac{\partial u}{\partial x} - \frac{\partial q}{\partial x}. \tag{2.10}$$

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

Equation (2.8) is the continuity equation, (2.9) is the momentum equation closed by the electric field, and (2.10) is the pressure evolution equation. The system is closed electrostatically through Poisson's equation, which relates the electric field to the departure of the electron density ($n(x,t)$) from the neutralising ion background (n_0),

$$\frac{\partial E}{\partial x} = \frac{q_\mathrm{e}}{\epsilon_0}(n(x,t) - n_0). \tag{2.11}$$

The remaining closure problem is to express the third-order moment q, or equivalently its divergence $\partial q/\partial x$, as a function of the lower-order fluid quantities n, u, p and the electric field E.

The best-known analytic solution to this closure problem is the Hammett–Perkins closure (Hammett & Perkins 1990), obtained by matching the linear fluid response to the kinetic plasma dispersion function. In Fourier space it expresses the heat flux as a spatially non-local operator on the temperature perturbation,

$$\tilde{q}_k = -n_0\sqrt{\frac{8}{\pi}}v_\mathrm{th}\frac{\mathrm{i}k}{|k|}\tilde{T}_k, \tag{2.12}$$

with $v_\mathrm{th} = \sqrt{T_0/m}$ and $\tilde{T}_k$ the Fourier transform of the temperature perturbation $\tilde{T} = \tilde{p} - T_0\tilde{n}$. The factor $\mathrm{i}k/|k| = \mathrm{i}\,\mathrm{sgn}(k)$ is a Hilbert transform in configuration space, making the closure non-local in space but instantaneous in time and linear in the moments. Higher-order Landau-fluid closures generalise (2.12) by approximating the plasma dispersion function with higher-order Padé approximants (Hunana *et al.* 2018, 2019), improving the linear response over a broader range of phase velocities while remaining, like Hammett–Perkins, closures derived from linear kinetic theory. We take the Hammett–Perkins closure as an analytic benchmark for the learned closure (§§ 4.1, 4.2).

2.1. *Memory in the exact closure: the Mori–Zwanzig formalism*

Before turning to the learned closure, it is worth making precise why the exact closure is expected to carry memory. Truncating the moment hierarchy at (n, u, p) projects the Vlasov–Poisson dynamics onto a finite set of resolved variables; the discarded fine velocity-space structure of f re-enters only through the heat flux q. The Mori–Zwanzig formalism (Zwanzig 1961; Mori 1965) reformulates the projected dynamics exactly, and is the standard framework for reasoning about reduced-order models of this kind (Chorin, Hald & Kupferman 2000; Gouasmi *et al.* 2017; Parish & Duraisamy 2017). For a resolved observable g evolving under $\dot{g} = \mathcal{L}g$, where $\mathcal{L}$ is the Liouville operator, let P project onto the resolved variables and $Q = 1 - P$ onto the rest. Mori–Zwanzig then rewrites the projected dynamics, without approximation, as the generalised Langevin equation

$$\frac{\partial g}{\partial t} = \underbrace{P\mathcal{L}g}_{\text{Markovian}} + \underbrace{\int_0^t \mathcal{K}(t-s)\,g(s)\,\mathrm{d}s}_{\text{memory}} + \underbrace{F(t)}_{\text{noise}}, \qquad F(t) = \mathrm{e}^{tQ\mathcal{L}}Q\mathcal{L}g(0). \tag{2.13}$$

The three terms are the instantaneous response computable from the resolved state alone (Markovian term), a convolution over the resolved history (memory term) and the influence of the unresolved initial content (noise term). The memory kernel is the autocorrelation of the projected-out dynamics, $\mathcal{K}(\tau) \sim P\mathcal{L}\,e^{\tau Q\mathcal{L}}\,Q\mathcal{L}$: the

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

resolved fields excite the unresolved ones ($Q\mathcal{L}$), the excitation propagates under the orthogonal dynamics $e^{\tau Q\mathcal{L}}$ confined to the eliminated subspace, and is only later fed back onto the resolved dynamics ($P\mathcal{L}$). Memory therefore persists for as long as the eliminated degrees of freedom retain a coherent imprint of the state that excited them, and $\mathcal{K}$ decays on their correlation time. This is the general reason a reduced description carries memory.

In the present system, continuity and momentum ((2.8), (2.9)) involve only resolved quantities, the electric field being slaved to the density through Poisson's equation (2.11), so the entire unresolved influence is concentrated in the pressure equation, the instance of (2.13) applied to our system, i.e.

$$\frac{\partial p}{\partial t} = -u\frac{\partial p}{\partial x} - 3p\frac{\partial u}{\partial x} - \frac{\partial \bar{q}}{\partial x} + \int_0^t \mathcal{K}[t-s;\ n, u, p]\,\mathrm{d}s + F(x,t), \tag{2.14}$$

with $\bar{q} = Pq$ the conditional-mean (Markovian) heat flux, the integral the memory term and F the noise. Here the orthogonal dynamics is explicit. Expanding the perturbed f in Hermite polynomials makes the linearised kinetics a semi-infinite chain in the Hermite index, ℓ, coupled neighbour-to-neighbour by free streaming at a rate $\sim k v_{\mathrm{th}}$; the resolved block $\ell = 0, 1, 2$ connects to the unresolved tail $\ell \geqslant 3$ through the single amplitude q, and that tail evolves by pure streaming. Eliminating it returns the heat flux as a convolution over the past temperature gradient rather than an instantaneous function of it, with $\mathcal{K}$ decaying on the phase-mixing time $\sim(k v_{\mathrm{th}})^{-1}$ as streaming carries the excitation to ever finer velocity scales. The Hammett–Perkins closure (2.12) is the Markovian limit: collapsing $\mathcal{K}$ to a delta function that retains its zero-frequency weight, discards the time structure and leaves the $ik/|k|$ operator. Because our initial condition is a Maxwellian consistent with its own moments (3.1), the unresolved amplitudes vanish at $t = 0$, so $F \equiv 0$ and the exact closure is a deterministic functional of the resolved-field history.

Neither the convolution nor a closed form survives into the nonlinear regime, particle trapping disrupts phase mixing on the bounce time scale, precisely where the linear kernel fails, and $\mathcal{K}$ becomes a state-dependent functional of the resolved trajectory. A closure spanning both regimes must therefore retain a dependence on the resolved history, and the trailing window of K resolved states supplied to the FNO (§ 3.2) is a discrete, finite-horizon approximation of the memory integral in (2.14).

3. Methods

Our approach proceeds in three stages: generation of kinetic simulation data, online training of the FNO within a differentiable fluid solver and online testing of the trained model in an independent fluid simulation.

Kinetic simulations are run using a Vlasov–Poisson solver to produce noise-free distribution function outputs, from which we compute velocity moments. During online training, the FNO output replaces the heat-flux divergence in (2.10). Equations (2.8)–(2.10) are solved via a differentiable fluid solver, the electric field E being obtained from the Poisson equation (2.11) at each stage, and a loss function is defined as the difference between fluid and kinetic moments computed over a simulation window (so-called rollout window). This procedure trains the FNO to produce a heat-flux divergence that yields fluid trajectories matching the kinetic ones, without ever explicitly fitting the FNO to the kinetic heat flux itself. Finally, the trained FNO is tested online by deploying it within a fluid solver initialised from the same initial conditions as the corresponding kinetic simulation, and the resulting trajectories are compared with the kinetic ground truth over the full simulation duration.

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

3.1. *Kinetic simulation data*

We employ an Eulerian Vlasov–Poisson algorithm to generate the kinetic simulation data (Pezzi *et al.* 2013; Pezzi, Camporeale & Valentini 2016; Celebre, Servidio & Valentini 2023). Eulerian Vlasov solvers are particularly suitable here and, hence, preferred over particle-in-cell approaches, because they discretise the plasma distribution function directly on a phase-space grid. This ensures an extremely clean description of the distribution function – compared with the inherent statistical noise of particle-in-cell methods – and hence, of the velocity moments computed from these outputs. The algorithm numerically integrates the Vlasov equation for electrons – assuming that ions are a static background – coupled to the Poisson equation. The Vlasov equation is solved by implementing a splitting scheme (Cheng & Knorr 1976; Filbet & Pareschi 2002) that splits the multi-dimensional advection in phase space into several separate one-dimension advections in physical and velocity space. Each one-dimensional advection is then performed through the fourth-order volume-difference Van Leer scheme (Mangeney *et al.* 2002). It assumes periodic boundary conditions in physical space, while in velocity space $f(|v| > v_{\max}) = 0$, where $v_{\max}$ is a large multiple of the electron thermal speed to ensure a neat description of the distribution function. The code also implements collisional effects by introducing opportune Fokker–Planck operators at the right-hand side of the Vlasov equation. However, for this study, we focus on collisionless simulations.

Kinetic simulations are run at a time step of $\Delta t = 0.01\,\omega_p^{-1}$ and an output cadence of $0.1\,\omega_p^{-1}$. The $0.1\,\omega_p^{-1}$ output cadence samples the electric-field energy oscillation (period $\approx 2.5\,\omega_p^{-1}$ at this wavenumber) roughly 25 times per period. It is also the cadence at which the $K = 50$ step memory window is populated, giving the $5\,\omega_p^{-1}$ memory horizon of § 3.2, and is chosen fine enough to resolve the oscillation phase the closure reads from its memory (§ 4.5) while keeping the stored history manageable.

The simulation domain is one-dimensional with periodic boundary conditions, with system length $L = 2\pi/k$, where $k = 0.35$ is the wavenumber of the initial perturbation, such that the simulation length is normalised to a single period of the initial perturbation. Lengths are normalised by the Debye length, so $k = 0.35$ denotes $k\lambda_D = 0.35$. The treatment is non-relativistic, so the speed of light does not enter. The kinetic simulation uses 256 grid points in configuration space and 401 grid points in velocity space, spanning $v \in [-6, 6]$, normalised by the electron thermal velocity. Configuration-space data are downsampled to 128 grid points before being supplied to the fluid solver during training; the 128-point grid resolves the low-wavenumber structure the closure acts on (32 retained Fourier modes) while reducing the per-step memory and compute of the differentiable rollout, which dominates the training cost. In both the kinetic and fluid simulations, the simulation time is normalised by the plasma frequency, ω_{p}. The simulation is initialised with a static ion background distribution and a perturbed electron distribution; the perturbation is sinusoidal in space and is varied in amplitude across simulation runs in order to probe both the linear and nonlinear regimes of Landau damping. The initial perturbation amplitude A enters as a perturbation to the electron density,

$$n_0(x) = 1 + A\cos(kx). \tag{3.1}$$

The initial condition (3.1) excites a standing Langmuir wave; a propagating wave carries different parity in the heat flux, so confirming that this methodology extends to travelling-wave initial conditions is a useful check left to future work.

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

Parameter	Value
Spatial grid points, N_x	256
Velocity grid points, N_v	401
Velocity domain	$v \in [-6, 6]\, v_{\mathrm{th}}$
Perturbation wavenumber, k	0.35
Domain length, L	$2\pi/k$
Boundary conditions	Periodic
Ion background	Static, uniform
Initial density	$n_0(x) = 1 + A\cos(kx)$
Time step, Δt_{kin}	$10^{-2}\,\omega_{\mathrm{p}}^{-1}$
Output cadence	$0.1\,\omega_{\mathrm{p}}^{-1}$ (every 10 steps)
Run duration (linear)	$150\,\omega_{\mathrm{p}}^{-1}$
Run duration (nonlinear)	$500\,\omega_{\mathrm{p}}^{-1}$

TABLE 1. Parameters of the kinetic (Vlasov) simulations used to generate the training and ground-truth data. Velocities are normalised by the electron thermal speed v_{th}, lengths by the Debye length and time by the inverse plasma frequency ω_{p}^{-1}. Configuration-space fields are downsampled from $N_x = 256$ to the $N_G = 128$ fluid grid before being supplied to the closure.

From each kinetic snapshot, we compute the velocity moments up to and including the heat flux using Simpson's rule applied to the velocity-space integrals defined in § 2. These moment snapshots serve both as the input initial conditions, evolved during training in the fluid rollouts, and as the ground-truth targets against which the closed fluid trajectories are compared. The parameters of the kinetic simulations are collected in table 1.

3.2. *Fluid model*

In our strategy, following prior works by Wei *et al.* (2023) and Huang *et al.* (2025), the FNO is incorporated directly into the fluid equations, replacing the heat-flux divergence term in the pressure equation (2.10). Rather than treating the closure as an instantaneous algebraic map from the current lower-order moments to the heat-flux divergence, we allow the closure to depend on a short history of the resolved fields. The closure takes as input the trailing window of the K most recent states, together with an amplitude conditioning channel $\log A_{\mathrm{scale}}$ (defined in § 3.3), and returns the heat-flux divergence at the next step. The pressure equation becomes

$$\frac{\partial p}{\partial t} = -u\frac{\partial p}{\partial x} - 3p\frac{\partial u}{\partial x} - \mathrm{FNO}(\boldsymbol{M};\ \log A_{\mathrm{scale}})\,, \tag{3.2}$$

where $\boldsymbol{M}$ is defined below. The use of a finite-memory window is physically motivated. The exact closure obtained by projecting out the unresolved (higher-moment, fine velocity-space) dynamics is in general non-Markovian: the truncated hierarchy carries a memory of the resolved fields through a convolution kernel (Gupta & Lermusiaux 2023). In the specific case of Landau damping, the heat flux encodes the phase-mixing history of the distribution function. A trailing window of resolved-moment fields provides a simple, discrete approximation to this memory. In practice, we use $K = 50$ steps at the training output cadence of $\Delta t = 0.1\,\omega_{\mathrm{p}}^{-1}$, i.e. a memory horizon of $5\,\omega_{\mathrm{p}}^{-1}$; at early times, before K states are available, the

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

window is left-padded with the corresponding equilibrium values ($n = 1,\ p = 1,\ u = E = \partial_x q = 0$).

We choose to have the network output the heat-flux divergence directly, rather than the heat flux itself, for two reasons. First, as the heat flux appears in the pressure equation only as a divergence, learning the divergence directly is a simpler training target and avoids the additional error introduced by approximating a further derivative through, e.g. finite differences. Second, and more subtly, outputting $\partial_x q$ directly rather than q gives the closure a direct channel for the net heating associated with Landau damping. Landau damping transfers energy from the wave to resonant particles, so the spatially averaged pressure $\langle p\rangle = (1/L)\int p\,\mathrm{d}x$ rises as the perturbation damps. Averaging the pressure equation (2.10) over the periodic domain gives $\mathrm{d}\langle p\rangle/\mathrm{d}t = -2\langle p\,\partial_x u\rangle - \langle \partial_x q\rangle$, so the mean pressure can in principle change through the compressional term even when the heat-flux contribution is divergence free; allowing $\langle \partial_x q\rangle \neq 0$ nonetheless supplies a dedicated, well-conditioned degree of freedom for the net thermodynamic effect that a strictly periodic, single-valued q could not, since $\int \partial_x q\,\mathrm{d}x = 0$ identically for such a q. The network's output is therefore best read as an effective-heat-flux-divergence source term in the resolved pressure equation, rather than the exact spatial derivative of a physical periodic heat flux. This is a deliberate modelling choice, consistent with the effective-closure interpretation of § 4.4, and not a violation of the periodic boundary conditions imposed on the resolved fields n, u, p and E, which remain periodic throughout.

Time integration of the fluid equations ((2.8), (2.9), (2.10)) set out in § 2 is performed using a third-order strong stability preserving Runge–Kutta method (SSPRK3). For a state x evolving according to $\partial x/\partial t = f(x, t)$ with initial value x_0 at time $t = t_0$, a single SSPRK3 step of size Δt produces

$$x^{(1)} = x_0 + \Delta t\ f(x_0,\ t_0), \tag{3.3}$$

$$x^{(2)} = \frac{3}{4}x_0 + \frac{1}{4}\left(x^{(1)} + \Delta t\ f\left(x^{(1)},\ t_0 + \Delta t\right)\right), \tag{3.4}$$

$$x^{(3)} = \frac{1}{3}x_0 + \frac{2}{3}\left(x^{(2)} + \Delta t\ f\left(x^{(2)},\ t_0 + \frac{\Delta t}{2}\right)\right), \tag{3.5}$$

where superscripts label the SSPRK3 substages and $x^{(3)}$ is the accepted update at $t_0 + \Delta t$.

The presence of the FNO closure in the time-derivative function f requires care at each substage, because the closure is not a pointwise function of the instantaneous state but is conditioned on a trailing window of the moment history. Writing the per-slot channel bundle as

$$\boldsymbol{m}_t \equiv (n_t,\ u_t,\ p_t,\ E_t,\ (\partial_x q)_t), \tag{3.6}$$

the FNO takes as input the last K slots of the history window,

$$\boldsymbol{M}_t \equiv (\boldsymbol{m}_{t-(K-1)\Delta t},\ \ldots,\ \boldsymbol{m}_t), \qquad (\partial_x q) = \mathrm{FNO}\left(\boldsymbol{M}_t;\ \log A_{\mathrm{scale}}\right), \tag{3.7}$$

together with the log-amplitude conditioning scalar $\log A_{\mathrm{scale}}$. Here K is the memory length. Denote by $\mathcal{F}_n$, $\mathcal{F}_u$, $\mathcal{F}_p$ the right-hand sides of ((2.8), (2.9), (2.10)) and by $\mathcal{P}$ the spectral Poisson solve returning E from the density.

The subtlety introduced by the trailing window is that each provisional substage state must be inserted into the buffer before the FNO can be evaluated on it. We

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

handle this with a single rolling slot: stage 1 appends its predictor to the buffer, while stages 2 and 3 overwrite that trailing slot, so at most one provisional entry is present at any time and, after stage 3, the trailing slot holds the accepted state at t_{i+1}. The electric field is recomputed from Poisson at every substage, so the state fed to the FNO always satisfies the electrostatic constraint. Note also that the pressure update at each substage uses the closure produced at the previous substage: stage 1 uses the closure carried from t_i, and the FNO evaluated at the end of a substage supplies the closure consumed by the next. The explicit scheme advancing the accepted state $\{\boldsymbol{m}_{t_{i-K+1}}, \ldots, \boldsymbol{m}_{t_i}\}$ by one step is as follows.

Stage 1 (forward-Euler predictor):

$$n^{(1)} = n_{t_i} + \Delta t\, \mathcal{F}_n(n_{t_i}, u_{t_i}), \tag{3.8}$$

$$u^{(1)} = u_{t_i} + \Delta t\, \mathcal{F}_u(n_{t_i}, u_{t_i}, p_{t_i}, E_{t_i}), \tag{3.9}$$

$$p^{(1)} = p_{t_i} + \Delta t\, \mathcal{F}_p(u_{t_i}, p_{t_i}, (\partial_x q)_{t_i}), \tag{3.10}$$

$$E^{(1)} = \mathcal{P}(n^{(1)}), \tag{3.11}$$

$$(\partial_x q)^{(1)} = \mathrm{FNO}(\boldsymbol{M}_{t_i};\ \log A_{\mathrm{scale}}), \quad \boldsymbol{M}_{t_i} = \left(\boldsymbol{m}_{t_{i-K+1}}, \ldots, \boldsymbol{m}_{t_i}\right). \tag{3.12}$$

Append $\boldsymbol{m}^{(1)} = (n^{(1)}, u^{(1)}, p^{(1)}, E^{(1)}, (\partial_x q)^{(1)})$ to the trailing memory window.

Stage 2:

$$n^{(2)} = \frac{3}{4} n_{t_i} + \frac{1}{4}\left(n^{(1)} + \Delta t\, \mathcal{F}_n\left(n^{(1)}, u^{(1)}\right)\right), \tag{3.13}$$

$$u^{(2)} = \frac{3}{4} u_{t_i} + \frac{1}{4}\left(u^{(1)} + \Delta t\, \mathcal{F}_u\left(n^{(1)}, u^{(1)}, p^{(1)}, E^{(1)}\right)\right), \tag{3.14}$$

$$p^{(2)} = \frac{3}{4} p_{t_i} + \frac{1}{4}\left(p^{(1)} + \Delta t\, \mathcal{F}_p\left(u^{(1)}, p^{(1)}, (\partial_x q)^{(1)}\right)\right), \tag{3.15}$$

$$E^{(2)} = \mathcal{P}(n^{(2)}), \tag{3.16}$$

$$(\partial_x q)^{(2)} = \mathrm{FNO}\left(\boldsymbol{M}^{(1)};\ \log A_{\mathrm{scale}}\right), \quad \boldsymbol{M}^{(1)} = \left(\boldsymbol{m}_{t_{i-K+2}}, \ldots, \boldsymbol{m}_{t_i}, \boldsymbol{m}^{(1)}\right). \tag{3.17}$$

Overwrite the trailing slot with $\boldsymbol{m}^{(2)}$.

Stage 3:

$$n^{(3)} = \frac{1}{3} n_{t_i} + \frac{2}{3}\left(n^{(2)} + \Delta t\, \mathcal{F}_n\left(n^{(2)}, u^{(2)}\right)\right), \tag{3.18}$$

$$u^{(3)} = \frac{1}{3} u_{t_i} + \frac{2}{3}\left(u^{(2)} + \Delta t\, \mathcal{F}_u\left(n^{(2)}, u^{(2)}, p^{(2)}, E^{(2)}\right)\right), \tag{3.19}$$

$$p^{(3)} = \frac{1}{3} p_{t_i} + \frac{2}{3}\left(p^{(2)} + \Delta t\, \mathcal{F}_p\left(u^{(2)}, p^{(2)}, (\partial_x q)^{(2)}\right)\right), \tag{3.20}$$

$$E^{(3)} = \mathcal{P}(n^{(3)}), \tag{3.21}$$

$$(\partial_x q)^{(3)} = \mathrm{FNO}\left(\boldsymbol{M}^{(2)};\ \log A_{\mathrm{scale}}\right), \quad \boldsymbol{M}^{(2)} = \left(\boldsymbol{m}_{t_{i-K+2}}, \ldots, \boldsymbol{m}_{t_i}, \boldsymbol{m}^{(2)}\right). \tag{3.22}$$

Overwrite the trailing slot with $\boldsymbol{m}^{(3)}$ and accept $\boldsymbol{m}_{t_{i+1}} = \boldsymbol{m}^{(3)}$.

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

This scheme was chosen as the most straightforward implementation of the online closure. Its main drawback is that the trained FNO is tightly coupled to this particular numerical method, which may limit its portability to other fluid codes; a more solver-independent implementation is left to future work.

A related subtlety is that the trailing slot fed to the closure is not the same kind of state at every substage: at stage 1 it holds the accepted state at t_i, whereas at stages 2 and 3 it holds the provisional predictors $m^{(1)}$ and $m^{(2)}$, which are not themselves accepted fluid states. The closure is trained end-to-end through the full three-stage update, so it encounters this substage structure identically in training and deployment and can in principle absorb it; we expect any residual effect to be small, since the provisional states differ from the accepted state only at $O(\Delta t)$ over a single step and occupy only the most recent of $K = 50$ slots. We have not isolated this effect, and a substage-consistent buffering scheme would be one way to remove it; we leave this to the solver-independent reimplementation noted above.

Spatial derivatives in the fluid equations are computed via second-order central finite differences on the 128-point grid, utilising the periodic boundary conditions. The fluid solver shares the same spatial domain, $L = 2\pi/k$, as the kinetic simulation. The entire differentiable fluid solver, including the SSPRK3 time integration, the spectral Poisson solve and the finite-difference spatial operators, is implemented in PyTorch (Paszke *et al.* 2019), so that the full fluid rollout supports reverse-mode automatic differentiation and gradients can be propagated back through it to the FNO parameters during online training.

3.3. *Fourier neural operator*

We use a FNO as the architecture for our closure function (Li *et al.* 2021), implemented using the `FNO` class of the `neuraloperator` Python package (Kossaifi *et al.* 2024). The closure network maps the trailing window of resolved-moment fields to the heat-flux divergence one output step ahead. Each of the five resolved fields $\{n, u, p, E, \partial_x q\}$ is supplied over the $K = 50$ most recent time steps on the 128-point spatial grid, with the density and pressure entered as perturbations about their unit equilibrium ($n - 1$ and $p - 1$), and a single constant channel carries the amplitude conditioning variable $\log A_{\mathrm{scale}}$, giving $5K + 1 = 251$ input channels of length 128. The network outputs a single channel, the predicted $\partial_x q$ at the next step, also of length 128. The FNO comprises eight Fourier layers, each retaining the lowest 32 Fourier modes and using 64 hidden channels. Because Landau damping is periodic and dominated by a low-wavenumber structure, this truncated-mode architecture is well matched to the problem (Li *et al.* 2021); the periodic geometry likewise makes the Fourier layers a natural choice. The online training method is in principle independent of the closure architecture, and a systematic comparison of the FNO against alternative architectures, such as other operator-learning approaches (Lu *et al.* 2021), trained under the same online procedure is a worthwhile avenue for further study, which we leave to future work. These architectural choices are common to every closure trained in this work, the single-amplitude linear and nonlinear closures and the multi-amplitude closure, and are summarised in table 2.

The closure is a standard FNO (Li *et al.* 2021); we summarise its forward map so the architecture is self-contained. Write the channel-stacked input described above as $\mathbf{a} \in \mathbb{R}^{d_a \times N}$, with $d_a = 5K + 1 = 251$ input channels each of length $N = 128$ (the spatial grid). The operator lifts the input to a hidden width $d_v = 64$, applies $L = 8$

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

Hyperparameter	Value
Fourier layers	8
Retained Fourier modes (per layer)	32
Hidden channels	64
Input channels ($5K + 1$)	251
Output channels	1
Activation function	GELU
Block precision	full

TABLE 2. The FNO closure architecture, common to all trained models (the single-amplitude linear and nonlinear closures and the multi-amplitude closure). The number of input channels follows from the five resolved fields $\{n, u, p, E, \partial_x q\}$ supplied over the $K = 50$ step memory window, together with the single spatially constant $\log A_{\text{scale}}$ conditioning channel, giving $5K + 1 = 251$. The architecture is implemented using the `FNO` class of the `neural operator` library (Kossaifi *et al.* 2024).

Fourier layers, and projects to the single output channel $d_u = 1$:

$$v^{(0)} = P\,\mathbf{a}, \tag{3.23}$$

$$v^{(l+1)}(x) = \sigma\big(W_l\, v^{(l)}(x) + b_l + (\mathcal{K}_l v^{(l)})(x)\big), \qquad l = 0, \ldots, L-1, \tag{3.24}$$

$$\partial_x q = Q\, v^{(L)}. \tag{3.25}$$

Here P and Q are pointwise (channel-wise) affine lift and projection maps, W_l is a pointwise linear operator (1×1 convolution mixing channels at each grid point) with bias b_l and σ is the Gaussian error linear unit (GELU) activation. The spectral convolution $\mathcal{K}_l$ is the defining operation of the architecture, i.e.

$$(\mathcal{K}_l v) = \mathcal{F}^{-1}(R_l \cdot \mathcal{F} v), \qquad (R_l \cdot \widehat{v})(\kappa) = \begin{cases} R_l(\kappa)\,\widehat{v}(\kappa), & \kappa < k_{\max}, \\ 0, & \kappa \geqslant k_{\max}, \end{cases} \tag{3.26}$$

where $\mathcal{F}$ is the discrete Fourier transform along the single spatial dimension, $\widehat{v} = \mathcal{F} v$, and at each retained mode κ the learned weight $R_l(\kappa) \in \mathbb{C}^{d_v \times d_v}$ mixes the hidden channels; modes $\kappa \geqslant k_{\max}$ are truncated to zero before the inverse transform. Learning the response mode-by-mode in Fourier space, rather than through fixed-width spatial stencils, is what lets a single layer act non-locally across the periodic domain in one step. Retaining only $k_{\max} = 32$ modes matches the low-wavenumber structure of Landau damping noted above and makes the weights $\{R_l\}$ the bulk of the model's parameters. The values $(L, k_{\max}, d_v) = (8, 32, 64)$ are those of table 2.

Supplying a history of the fields, rather than only the instantaneous state, allows the closure to represent the non-Markovian dependence discussed in § 3.2. During a fluid rollout the memory window is populated from the fluid trajectory itself: at each SSPRK3 substage the FNO is evaluated on the current and recent resolved states, so that the closure is consistent with the same finite-memory map it was trained under. The spectral transform in each Fourier layer acts along the single spatial dimension (the 128-point grid) only. The $K = 50$ memory steps and the five resolved fields enter as input channels, the $5K + 1 = 251$ channels of table 2, rather than as

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

additional transform axes, so the closure is a one-dimensional spatial FNO applied to a channel-stacked history, not a space–time transform.

A practical consideration that proved central to obtaining good performance is the scaling of the FNO inputs. The fluid moments span very different magnitudes: the density and the electric field are typically of order unity in our normalisation, while the heat flux can be many orders of magnitude smaller, particularly in the linear regime; moreover, the magnitude of each moment varies significantly over the course of a single simulation, and across simulations by the several orders of magnitude that separate the linear and nonlinear regimes. We address this through two scaling steps applied to the FNO inputs and outputs.

First, we normalise the perturbation fields by a characteristic amplitude computed from the most recent density field in the input window,

$$A_{\mathrm{scale}} = \left(\frac{1}{N} \sum_{x} \big(n(x) - 1\big)^2 \right)^{1/2}, \tag{3.27}$$

where N is the number of grid points in x and $n(x)$ is the density at the most recent time step in the memory window; A_{scale} is therefore the root-mean-square amplitude of the density perturbation, and is floored at a small value (10^{-6}) to remain well defined at exact equilibrium. Because A_{scale} is computed from the resolved density within the closure itself, it requires no external input at inference and is recomputed at every step, so it tracks the instantaneous perturbation level rather than a fixed initial amplitude. Here the perturbation fields are the deviations of each moment from its spatially uniform equilibrium: the density and pressure are supplied to the network as $n - 1$ and $p - 1$, their normalised background values both being fixed at unity, whereas the velocity u, electric field E and heat-flux divergence $\partial_x q$ vanish at equilibrium and are supplied directly. Each perturbation field is divided by A_{scale} before being supplied to the network, and the output $\partial_x q$ is multiplied by A_{scale} to restore physical units. Normalising in this way lets the network operate on order-unity fields irrespective of the underlying amplitude regime, so that it learns the shape of the closure relation rather than its overall scale; without this step, training would be biased towards the larger-amplitude cases at the expense of the linear regime, where the perturbations are vanishingly small. Because normalisation removes the amplitude, we reintroduce it explicitly as a conditioning input by appending $\log A_{\mathrm{scale}}$ as an additional, spatially constant channel. This tells the network which amplitude regime it is operating in and allows a single model to modulate its behaviour continuously between the linear (small A_{scale}) and nonlinear (large A_{scale}) limits, and to interpolate between the amplitude scales seen in training.

Second, we apply an inverse hyperbolic sine (arcsinh) transformation, which acts approximately linearly on small inputs and approximately logarithmically on large inputs. Here $\mathrm{arcsinh}(x) = \ln(x + \sqrt{x^2 + 1})$, applied element-wise to each A_{scale}-normalised input field. This compresses the dynamic range of the inputs while preserving the structure of the small scales, which is crucial in the linear regime where perturbation amplitudes become very small. Although input scaling is not generally given prominent treatment in the existing ML plasma closure literature, in our experience it is one of the most consequential design choices for producing closures that perform well across a wide range of perturbation amplitudes.

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

3.4. *Online training method*

The online training procedure operates as follows. At each training iteration, a batch of kinetic snapshots is sampled from the training set; each snapshot provides the initial conditions for a short rollout of the closed fluid simulation, in which the FNO replaces the heat-flux divergence term as described in § 3.2. The fluid rollout is integrated for a chosen number of time steps, with the FNO evaluated at each substage of the SSPRK3 update. The resulting fluid trajectory is compared with the corresponding kinetic trajectory at the same time points, and the loss is computed as the relative L^2 norm between the two over all rolled-out time steps and all resolved moments, i.e. the fields $\{n, u, p, E\}$ evolved by the fluid solver (with E slaved to the density through Poisson's equation). The heat-flux divergence $\partial_x q$ is the output of the closure, not an evolved field, and never enters the loss; it is reported as a diagnostic only (table 5, figure 9). The loss is evaluated on the unscaled (physical) fluid moments: it is a per-field relative L^2 error in which each evolved field's discrepancy is normalised by the norm of the corresponding kinetic field. The arcsinh and A_{scale} scalings of § 3.3 apply only to the FNO inputs and output, not to the loss, and because the relative norm is formed field by field no single moment or amplitude regime dominates the objective.

Because the entire fluid solver is implemented in PyTorch (Paszke *et al.* 2019), a differentiable framework, gradients of the loss with respect to the FNO parameters can be computed through the rollout via automatic differentiation, and standard stochastic gradient methods can therefore be used to update the parameters.

This procedure differs fundamentally from offline training in that the loss never references the kinetic heat flux directly. The FNO is trained only against the trajectories of the lower-order moments and is free to converge to whichever heat-flux profile produces the best agreement with those trajectories under the specific fluid solver in use. This effective-heat-flux divergence is, in general, not expected to coincide with the one computed directly from the kinetic simulation. It is worth being explicit about why, since the fluid moment equations ((2.8)–(2.10)) are themselves exact consequences of the Vlasov equation: were we to supply the exact kinetic heat-flux divergence $\partial_x q_{kin}$ at every point and time step, the retained moments n, u and p would reproduce the kinetic evolution up to numerical error. The learned effective flux departs from $\partial_x q_{kin}$ for two reasons. First, the online loss is evaluated on the resolved-moment trajectory and never references $\partial_x q_{kin}$; the closure is therefore free to converge to whichever $\partial_x q$ best steers the discretised fluid solver over a rollout, and instantaneous departures from $\partial_x q_{kin}$ are tolerated provided they do not accumulate into trajectory drift. Second, the closure acts on a trailing history window rather than the instantaneous state, so the influence of the truncated higher moments is carried through the memory of the resolved fields rather than through an instantaneous pointwise correction. The residual dependence on the particular numerical scheme, discretisation and time stepping then renders the learned flux solver-specific. We develop this reading, and its relation to the Mori–Zwanzig picture, in § 4.4.

For the dataset used in single-amplitude experiments, the train/test split is taken chronologically; the earlier portion of each simulation provides the training snapshots and the later portion provides the test snapshots. For the generalisation experiments described in § 4.3, the split is instead taken across initial perturbation amplitudes, as described in that section.

Practical implementation requires care with memory and compute requirements. Each fluid rollout requires storing intermediate states for backpropagation, and the

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

Hyperparameter	Value
Spatial grid points, N_G	128
Fluid time step, Δt	$0.1\,\omega_{\mathrm{p}}^{-1}$
Rollout length	100 steps ($10\,\omega_{\mathrm{p}}^{-1}$)
Memory window, K	50 steps ($5\,\omega_{\mathrm{p}}^{-1}$)
Loss	Relative L^2 (all moments except $\partial_x q$, all rollout steps)
Optimiser	AdamW
Learning rate (initial)	10^{-3}
Learning-rate schedule	Cosine annealing ($T_{\max} = 1000$ epochs, $\eta_{\min} = 0$)
Training epochs	1000
Batch size	50
Gradient clipping	None

TABLE 3. Training hyperparameters, common to all trained models. The batch size is selected dynamically to make use of the available GPU memory; the value of 50 corresponds to the 40 GB A100 used here. In each epoch the entire set of training snapshots is sampled once, each snapshot providing the initial condition for a 100-step fluid rollout. The models differ only in their training data: the single-amplitude closures are trained on one initial amplitude with a chronological train/test split, whereas the multi-amplitude closure is trained on the amplitudes defined in § 4.3 with the split taken across amplitudes.

memory footprint scales with the number of rollout steps; longer rollouts therefore force smaller batch sizes and require additional accelerator memory. Tuning these hyperparameters involves a trade-off: shorter rollouts permit faster training and larger batches but provide a weaker training signal for the long-time behaviour of the closure, while longer rollouts improve the final performance and stability at substantial computational cost. All training conducted for this investigation was performed on a single NVIDIA A100 GPU with 40 GB of VRAM. In each epoch the full set of training snapshots is sampled once, each snapshot seeding a separate 100-step fluid rollout, with the batch size chosen dynamically to fill the available GPU memory; on this hardware that permitted a batch size of 50. For each model discussed here, training was performed with a rollout length of 100 fluid time steps at a time-step length of $\Delta t = 0.1\omega_{\mathrm{p}}^{-1}$, which we found to balance well between ensuring the stability of the trained closure, while remaining computationally viable. We did not employ a rollout-length curriculum: the 100-step rollout was used from the first epoch. We found this unnecessary for convergence here; the per-field relative loss and the input normalisation keep the early epoch gradients well conditioned even when the untrained closure is inaccurate, though a curriculum may prove helpful for longer rollouts. The models trained over single-amplitude simulation runs converged in approximately 10–15 hours of training time on a single A100 GPU, and the general model, which had a larger training set, converged in approximately 50 hours of training time on a single A100 GPU. In future work, it will be informative to conduct online training with different lengths of Δt, specifically investigating if ML based closures can reproduce kinetic effects even when trained and/or tested with fluid-like time scales. The training hyperparameters, common to all models, are summarised in table 3.

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

3.5. *Offline training for comparison*

To isolate the effect of online training, we compare against an otherwise identical closure trained offline (*a priori*). The offline model is deliberately matched to the online model in every respect except the training objective: it uses the same FNO architecture and hyperparameters (§ 3.3), the same input construction, including the $K = 50$ step trailing memory window and the $\log A_{\mathrm{scale}}$ conditioning channel, the same input scaling and the same relative L^2 loss and optimiser settings. The single difference is the origin of the data on which the loss is evaluated. In the offline case the memory windows are sampled directly from the kinetic simulation trajectories, and the network is trained to regress the kinetic heat-flux divergence, $\partial_x q_{\mathrm{kin}}$, computed from those same kinetic snapshots; the loss never involves a fluid rollout. This is in contrast to the online case, in which each memory window is generated by the closed fluid solver during the rollout and the loss is evaluated on the resulting fluid trajectory (§ 3.4).

Both models are then evaluated identically. Testing is performed online in every case: the trained closure is embedded in the fluid solver, initialised from the same conditions as the corresponding kinetic run and integrated over the full simulation duration, with the resulting trajectory compared against the kinetic ground truth. All comparisons between the online- and offline-trained closures reported in § 4 therefore differ only in how the closure was trained, not in how it is deployed or assessed.

We stress that this matched comparison is intended to isolate the effect of the training method, and not to argue that offline training cannot produce competent closures for this problem. Offline-trained closures have been shown to work well in this setting: Huang *et al.* (2025), for example, obtained an offline-trained FNO closure that reproduced both the linear and nonlinear regimes of one-dimensional Landau damping. The offline results reported here should accordingly be read as specific to our matched configuration rather than as a general limitation of offline training, since design choices left fixed for the comparison can materially affect offline performance. The temporal sampling of the training data is one such choice: Huang *et al.* (2025) sampled their training snapshots at a considerably finer time step than the output cadence used here, which provides the offline regression with a denser and more informative target and may partly explain the stronger offline performance they report. A systematic study of how data sampling and related choices affect offline relative to online training is beyond our present scope; our narrower claim is that, under identical hyperparameters and training data, we find online training to be the more robust of the two.

3.6. *Hammett–Perkins closure benchmark*

As a second, non-learned reference, we deploy the analytic Hammett–Perkins closure (2.12) in the same fluid solver. Unlike the FNO, the Hammett–Perkins closure is a memoryless algebraic map from the instantaneous resolved state to the heat-flux divergence; we therefore evaluate it inline at every SSPRK3 stage, on the stage's own (n, p), rather than under the trailing-window substep convention used for the FNO (§ 3.2). The heat flux is formed from the $\mathrm{i}\,\mathrm{sgn}(k)$ multiplier by fast Fourier transform and differentiated with the same finite-difference operator as the rest of the solver, so the discretisation is consistent across the two.

The coefficient in (2.12) is fixed by the linear kinetic response, but the three-pole Padé approximation underlying it reproduces the exact damping rate only

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

approximately at any single wavenumber. At our $k = 0.35$ the canonical coefficient $\sqrt{8/\pi} \approx 1.60$ over-damps the mode; we therefore calibrate the coefficient to match the kinetic linear damping rate, obtaining $\chi = \sqrt{4.8/\pi} \approx 1.24$, 22 % below the canonical value. This offset is consistent with the known mode-dependent accuracy of the three-pole closure, which fits the response function across all phase velocities rather than reproducing the exact rate at one. Calibrating in this way gives the Hammett–Perkins closure its best-case linear performance, so that any departure in the nonlinear regime reflects the structural limitations of the closure rather than a sub-optimal coefficient.

4. Results

In this section we present the performance of the online-trained FNO closure across three experimental settings. Section 4.1 examines the case of linear Landau damping, § 4.2 focuses on the nonlinear case and § 4.3 examines a single FNO trained across multiple initial perturbation amplitudes and tested on amplitudes both within and outside the training set. In each setting we benchmark the online-trained FNO against the same FNO architecture trained offline against the kinetic heat flux, and where appropriate against the analytic Hammett–Perkins closure (§ 3.6).

Throughout, the time-averaged relative L^2 error of an evolved field g is

$$\mathcal{E}(g) = \frac{1}{N_t} \sum_t \frac{\|g_{\text{fluid}}(\cdot, t) - g_{\text{kin}}(\cdot, t)\|_2}{\|g_{\text{kin}}(\cdot, t)\|_2}, \tag{4.1}$$

the spatial L^2 norm of the fluid-kinetic discrepancy at each output time, normalised by the kinetic norm at that time and averaged over the N_t output times in the window. The norm is taken over the full moment (mean plus fluctuation) on the spatial grid for u, E, $\partial_x q$ and over the fluctuation only for n, p; the per-time kinetic norms are non-zero for every reported field, so no flooring is required.

4.1. *Linear landau damping*

For the linear case, we use an initial perturbation amplitude of $A = 10^{-3}$, which produces electric-field perturbations small enough that nonlinear coupling between modes remains negligible over the full simulation duration. Figure 1 shows the electric-field energy $W_E = (1/2) \int E^2 \, \mathrm{d}x$ as a function of time for the kinetic simulation, the fluid simulation closed with the online-trained FNO (figure 1*a*) and the fluid simulation closed with the offline-trained FNO (figure 1*b*). In both cases the fluid simulation is integrated over the full simulation time, $0 < t < 150\, \omega_{\text{p}}^{-1}$, and uses the same initial conditions as the kinetic 'ground truth'. The shaded regions denoted with 'train' and 'test' mark the portions of the kinetic trajectory that were and were not sampled during training, the split falling at $t = 75\, \omega_{\text{p}}^{-1}$.

The online-trained closure reproduces the linear damping in detail (figure 1*a*). Throughout the training window the fluid $W_E(t)$ tracks the kinetic curve peak for peak, matching both the exponential decay envelope and the phase of the oscillation. We extract the damping rate by fitting a straight line to the peaks of $\log W_E$ over the training and test windows separately; the resulting rates are summarised in table 4. In the training window the fitted rate agrees with the kinetic value to within 0.2 % ($\gamma_{\text{FNO}} = -3.43 \times 10^{-2}$ against $\gamma_{\text{kin}} = -3.43 \times 10^{-2}$), confirming that the online closure recovers the correct Landau rate rather than merely a stable decay. This fidelity is reflected across all moments in table 5: within the training window every resolved moment (E, n, u, p) is reproduced to a time-averaged relative L^2 error at or below

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

	Linear ($A = 10^{-3}$)	
	Train ($t < 75$)	Test ($t \geqslant 75$)
γ_{kin}	-3.43×10^{-2}	-3.30×10^{-2}
$\gamma_{\text{FNO(online)}}$	-3.43×10^{-2}	-3.91×10^{-2}
γ_{HP}	-3.53×10^{-2}	-3.54×10^{-2}
Rel. err. (FNO (online))	1.56×10^{-3}	1.82×10^{-1}
Rel. err. (HP)	2.69×10^{-2}	6.94×10^{-2}

TABLE 4. Damping rates fitted to the peaks of $\log W_E(t)$ over the in-training (Train) and out-of-training (Test) windows, for the linear single-simulation case, for both the online-trained FNO and the Hammett–Perkins closures. The relative error is $|\gamma_{\text{i}} - \gamma_{\text{kin}}|/|\gamma_{\text{kin}}|$.

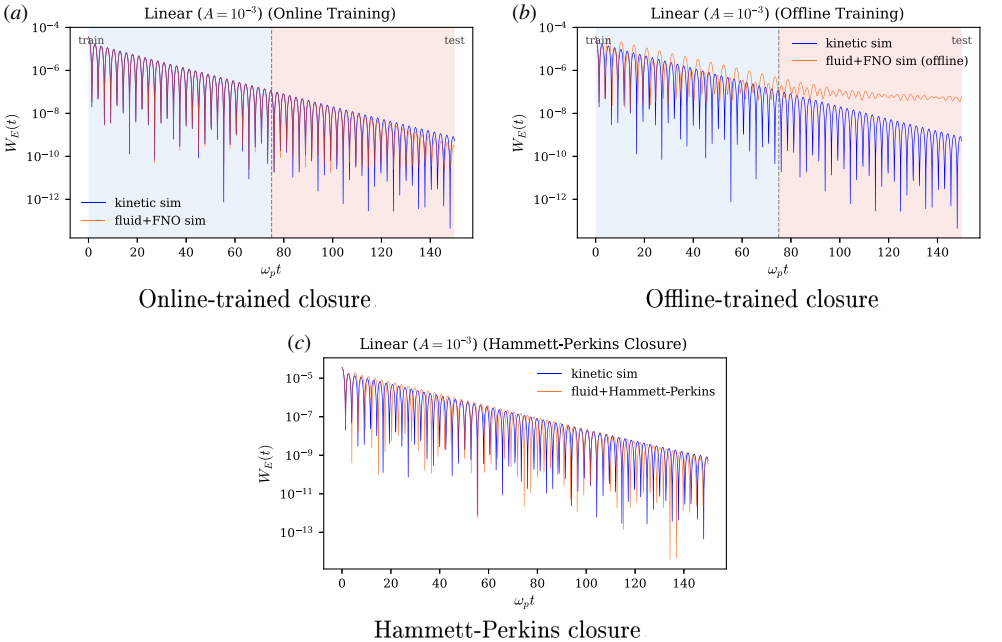


FIGURE 1. Electric-field energy $W_E(t)$ for the linear case ($A = 10^{-3}$): kinetic ground truth against the fluid simulation closed with (*a*) the online-trained FNO, (*b*) the offline-trained FNO, and (*c*) the analytic Hammett–Perkins closure ($\chi = \sqrt{4.8/\pi}$). All panels share the same axes and initial conditions; shaded regions in (*a*) and (*b*) mark the FNO training window and do not apply to the untrained Hammett–Perkins closure in (*c*).

the 10^{-2} level, with the density and pressure errors as low as $\sim 10^{-6}$; the diagnostic $\partial_x q$ error sits slightly higher, at $\approx 6 \times 10^{-2}$ (table 5), consistent with the effective-flux discussion of § 4.4.

Beyond the training window the two trajectories begin to separate slowly. The fitted test-window damping rate departs from the kinetic value by ≈18 % (table 4), and the moment errors grow accordingly, with the electric-field and velocity errors rising from the 10^{-3} level in the training window to $\sim 10^{-1}$ in the test window while

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

	Linear ($A = 10^{-3}$)		Nonlinear ($A = 10^{-1}$)	
Field	Train	Test	Train	Test
E	6.81×10^{-3}	8.78×10^{-2}	5.30×10^{-3}	1.42×10^{-1}
n	1.76×10^{-6}	1.96×10^{-6}	1.44×10^{-4}	2.86×10^{-3}
u	5.69×10^{-3}	1.04×10^{-1}	8.64×10^{-3}	1.45×10^{-1}
p	8.33×10^{-6}	6.53×10^{-6}	7.63×10^{-4}	1.25×10^{-2}
$\partial_x q$	6.05×10^{-2}	9.72×10^{-2}	1.55×10^{-1}	1.01

TABLE 5. Time-averaged relative L^2 error of each evolved fluid moment between the FNO-closed fluid simulation and the kinetic ground truth, evaluated over the in-training (Train) and out-of-training (Test) windows. Window splits are $t_{\text{split}} = 75$ for the linear case and $t_{\text{split}} = 250$ for the nonlinear case.

Threshold	Linear ($A = 10^{-3}$)	Nonlinear ($A = 10^{-1}$)
10 %	1.40	1.40
50 %	45.1	207
100 %	55.4	Not reached

TABLE 6. Elapsed time t_{div} (in ω_{p}^{-1}), measured from the end of the training window, at which the time-averaged relative error in the electric field E first exceeds the indicated threshold, for the linear and nonlinear single-simulation cases (training windows ending at $t = 75$ and $t = 250\,\omega_{\text{p}}^{-1}$, respectively). The 10 % threshold is crossed almost immediately in both cases, but the linear case reaches 50 % and 100 % error far sooner than the nonlinear case, which does not reach 100 % within the simulated window.

the density and pressure errors remain at the 10^{-6} level (table 5). Table 6 quantifies the onset of this drift: the relative electric-field error crosses 10 % almost immediately on leaving the training window ($t_{\text{div}} \approx 1.4\,\omega_{\text{p}}^{-1}$ after the split), but takes considerably longer to reach 50 % and 100 % (45 and 55 ω_{p}^{-1}, respectively), so the closure degrades gracefully rather than destabilising. The divergence is confined to the field variables that carry the oscillation phase: the deep nulls of W_E fall out of phase and the FNO peaks drop slightly below the kinetic peaks at late times, while the smoothly varying density and pressure remain accurate.

This late-time drift is expected. By the end of the run the linear signal has damped by many orders of magnitude below its initial value (figure 1*a*), into a regime the closure never saw during training: the chronological split means the smallest-amplitude, most deeply damped states lie entirely in the test window, and small phase errors accumulate over a rollout far longer than the 100-step ($10\,\omega_{\text{p}}^{-1}$) window used in training. Figure 2 shows the relative error in each evolved fluid moment between the online-tested fluid simulation and the kinetic ground truth as a function of space and time, and confirms this picture: the error is small and spatially uniform through the training window, and grows after the split, reaching order unity in the field variables (E, n, u) by the end of the run.

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

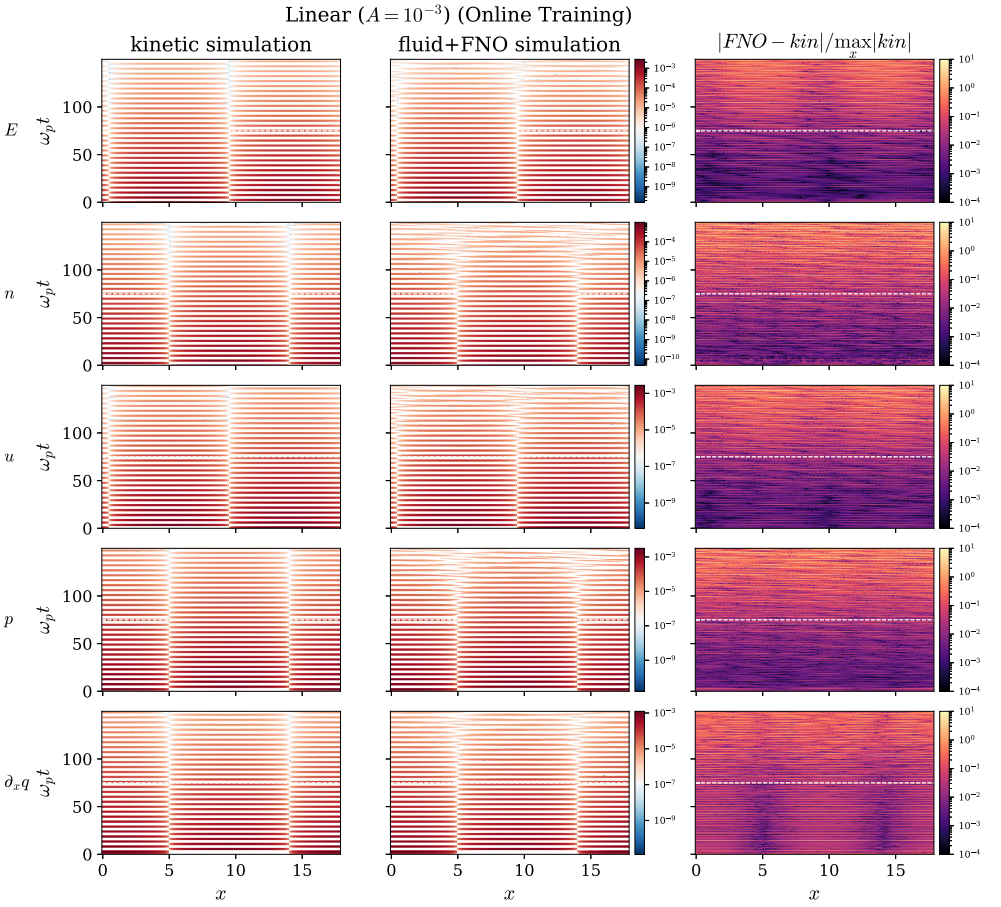


FIGURE 2. Space–time evolution of the evolved moments for the linear single-amplitude case ($A = 10^{-3}$, online-trained closure). Each row corresponds to one field (E, n, u, p, $\partial_x q$); the first column shows the kinetic ground truth, the second the fluid+FNO simulation and the third the pointwise relative error $|\mathrm{FNO} - \mathrm{kin}|/\max_x |\mathrm{kin}|$. Time runs vertically (0–150 ω_p^{-1}) and space horizontally over one perturbation wavelength $L = 2\pi/k$; colour scales are logarithmic and the horizontal dashed line marks the $t = 75\,\omega_\mathrm{p}^{-1}$ train/test split. Within the training window the kinetic and fluid columns are visually indistinguishable and the pointwise error is small and spatially uniform. Beyond the $t = 75\omega_p^{-1}$ split the fluid solution departs visibly from the kinetic reference: the coherent vertical phase structure breaks up and the relative error grows, reaching order unity (locally $1 - 10x$ the spatial maximum of the kinetic field) in E, n and u by the end of the run, as the deeply damped signal approaches the numerical noise floor of the solver. Density and pressure are shown as fluctuations about their equilibria ($n - 1$, $p - 1$).

The offline-trained closure, matched to the online model in architecture, hyperparameters and training data (§ 3.5), does not reproduce the linear dynamics under the same matched configuration (figure 1*b*). Rather than following the oscillatory decay, it produces a smooth $W_E(t)$ that damps far too slowly and misses the oscillation structure entirely, sitting orders of magnitude above the kinetic minima by the end of the run. As discussed in § 3.5, this behaviour is specific to the matched comparison and is sensitive to the temporal sampling of the offline training data; it is not a claim that offline training cannot succeed on this problem.

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

The analytic Hammett–Perkins closure reproduces the linear damping well (figure 1*c*), as expected of a closure constructed to capture precisely this regime. With the calibrated coefficient the Hammett–Perkins fluid $W_E(t)$ tracks the kinetic decay envelope and rate across the full run, with a fitted damping rate $\gamma_{\mathrm{HP}} = -3.53 \times 10^{-2}$ (table 4), matching the kinetic rate by construction. A closer look shows, however, that the calibrated closure reproduces the damping rate but not the oscillation frequency, with the peaks of W_E drifting progressively out of alignment with the kinetic trace, a $\approx$5 % error in the real oscillation frequency ($\omega_{\mathrm{r}} = 1.16$ against the kinetic 1.22) even as the decay envelope is matched. This is expected of a single-coefficient closure. Calibrating χ fixes the imaginary part of the complex frequency, leaving the real part set by the remaining fluid dispersion, which is not independently matched to the kinetic value. Because the online FNO and the Hammett–Perkins closure are deployed in the same fluid solver, the discrepancy cannot be attributed to the fluid discretisation, the online closure reproduces the kinetic W_E peak for peak (prior to reaching times unseen during training), matching both the damping envelope and the oscillation phase, whereas Hammett–Perkins matches only the envelope. Even in the linear regime, then, the learned closure captures the full linear dispersion that the analytic closure only partially reproduces.

4.2. *Nonlinear landau damping*

For the nonlinear case, we increase the initial perturbation amplitude to $A = 10^{-1}$, large enough that the system enters the trapping regime: the electric field undergoes initial damping followed by saturation and a subsequent slow modulation at the bounce frequency. Figure 3 shows $W_E(t)$ for the kinetic simulation, the fluid simulation closed with the online-trained FNO (figure 3*a*) and the fluid simulation closed with the offline-trained FNO (figure 3*b*).

Within the training window ($t < 250\,\omega_p^{-1}$) the online-trained closure reproduces the principal features of the nonlinear evolution (figure 3*a*): the initial damping phase, the level at which W_E saturates, and the period and amplitude of the bounce oscillation that modulates the saturated envelope. Beyond the training window the closure remains stable but not quantitatively accurate: it holds the saturation level and continues to track the bounce modulation qualitatively over the remaining $250\,\omega_p^{-1}$, but a phase offset develops at the training boundary and grows with time, and the resolved-moment errors are larger in the test window than in training (table 5), with the electric-field and velocity errors reaching $\sim 10^{-1}$ and the density and pressure errors at the 10^{-3}–10^{-2} level.

Table 6 reports the elapsed time from the end of each training window at which the relative electric-field error first crosses a given threshold. We caution against reading it as evidence that the closure is more accurate in the nonlinear regime. The 10 % threshold is in fact crossed almost immediately past the boundary in both cases ($\approx 1.4\,\omega_p^{-1}$); the nonlinear case then reaches 50 % only $207\,\omega_p^{-1}$ later and does not reach 100 % within the run, against 45 and $55\,\omega_p^{-1}$ in the linear case, but this slower growth is largely a property of the signal rather than of the closure. The relative error is measured against a kinetic W_E that remains $O(1)$ throughout the nonlinear run yet damps many orders of magnitude into the numerical noise floor in the linear run, so a comparable absolute discrepancy registers as a much larger relative error in the linear case. Consistent with this, the time-averaged relative errors of table 5 are larger in the nonlinear test window than in the linear one (density and pressure at 10^{-3}–10^{-2} against 10^{-6}), reflecting the richer trapping-generated structure the truncated moments must represent. The claim we make is therefore only that the

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

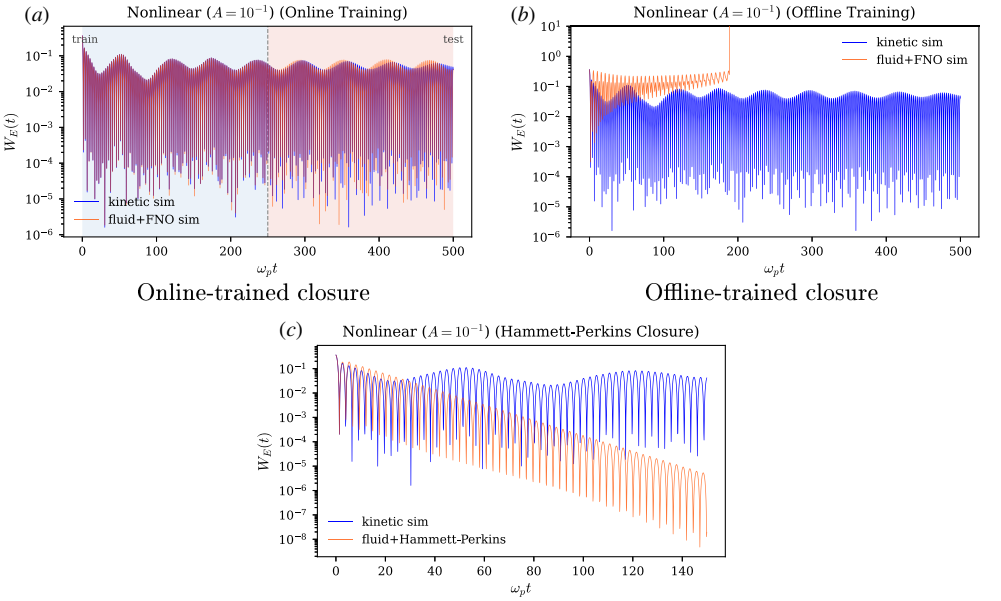


FIGURE 3. Electric-field energy $W_E(t)$ for the nonlinear case ($A = 10^{-1}$): kinetic ground truth against the fluid simulation closed with (*a*) the online-trained FNO, (*b*) the offline-trained FNO, and (*c*) the analytic Hammett–Perkins closure ($\chi = \sqrt{4.8/\pi}$). All panels share the initial conditions; shaded regions in (*a*) and (*b*) mark the FNO training window and do not apply to the untrained Hammett–Perkins closure in (*c*). Panel (*c*) is restricted to $t < 150\,\omega_{\mathrm{p}}^{-1}$ for demonstrative purposes.

nonlinear closure degrades gracefully, it drifts in phase and accumulates error but does not destabilise over the full $500\,\omega_p^{-1}$ run, not that it is more accurate beyond the training window than in the linear case.

The one field that departs substantially from the kinetic reference is the heat-flux divergence itself. The time-averaged $\partial_x q$ error grows from $\sim 10^{-1}$ in training to order unity in the test window (table 5), even as the observable fluid dynamics remain well reproduced. Figure 4 shows the relative error in each fluid moment over space and time: the errors in E, n, u and p remain bounded and moderate over the full duration, with their spatial structure reflecting the phase-space structures generated by particle trapping that the truncated moments cannot fully resolve, whereas the $\partial_x q$ error is large and broadly distributed across the domain. That the closure reproduces the resolved-moment dynamics while producing a heat flux that diverges pointwise from the kinetic one is the central observation we return to in § 4.4.

The offline-trained closure again fails under the matched configuration (figure 3*b*). It does not settle to the correct saturation level, instead sitting well above the kinetic envelope and drifting slowly upward rather than saturating, and it does not reproduce the oscillatory structure of the field energy. As in the linear case, and as discussed in § 3.5, this reflects the matched offline configuration rather than a general limitation of offline training.

The Hammett–Perkins closure cannot follow the nonlinear evolution (figure 3*c*). After the initial damping it fails to reproduce the saturation and the bounce modulation of the trapped state, since trapping is a nonlinear kinetic effect lying outside the

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

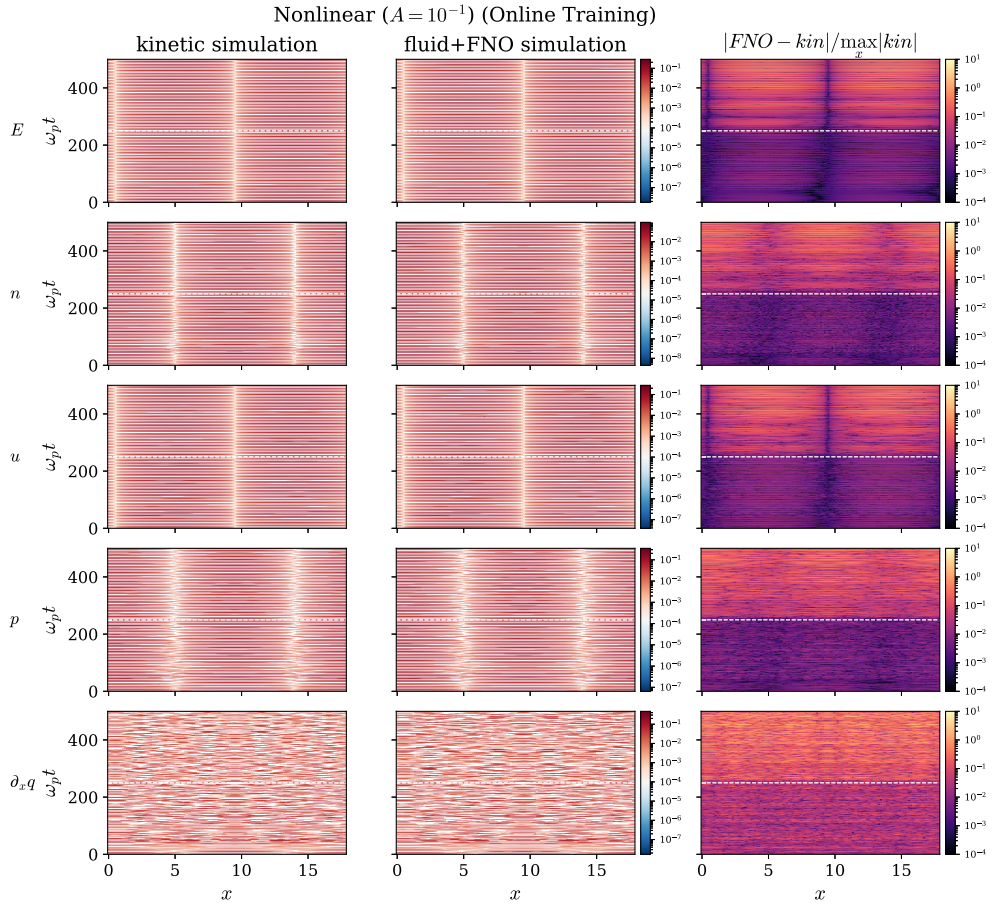


FIGURE 4. Space–time evolution of the evolved moments for the nonlinear single-amplitude case ($A = 10^{-1}$, online-trained closure), in the same layout as figure 2. Rows are the fields E, n, u, p, $\partial_x q$; columns are the kinetic ground truth, the fluid+FNO simulation and the pointwise relative error $|\mathrm{FNO} - \mathrm{kin}|/\max_x |\mathrm{kin}|$. Time runs vertically (0–500 ω_p^{-1}) and space over one wavelength $L = 2\pi/k$; colour scales are logarithmic and the dashed line marks the $t = 250\,\omega_\mathrm{p}^{-1}$ train/test split. The errors in E, n, u and p remain bounded over the full run, their spatial structure tracking the trapping-generated phase-space structure that the truncated moments cannot resolve, while the $\partial_x q$ error is large and broadly distributed, reflecting the departure of the learned effective heat flux from the kinetic one (§ 4.4).

linear theory from which the closure is derived, and no choice of coefficient recovers it. The same analytic closure that succeeds in the linear regime therefore fails here, whereas the single online-trained FNO reproduces both regimes within one model, albeit after re-training the FNO; however, in the following section we will present a single FNO model that can reproduce both the linear and nonlinear cases without re-training.

4.3. *Generalising across different initial amplitudes*

To test whether a single trained closure can capture both linear and nonlinear regimes within one model, we train an FNO on data drawn from a range of initial

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

perturbation amplitudes spanning the linear-to-nonlinear transition. The training amplitudes are sampled with four points per decade in $\log A$, giving

$$A_{\text{train}} = 10^{(j/4-3)}, \qquad j \in \{0, 1, 2, \ldots, 8\}, \tag{4.2}$$

ranging from $A = 10^{-3}$ (strongly linear) to $A = 10^{-1}$ (strongly nonlinear). The choice of four points per decade is intended to provide reasonable coverage of the linear, transition and nonlinear regimes, with sufficient density in the transition region where the qualitative dynamics change rapidly with amplitude.

We test the trained FNO on a denser grid of amplitudes spanning, and extending just beyond, the two decades covered by the training set. The test set therefore comprises the initial perturbation amplitudes (7×10^{-4}, 8×10^{-4}, 9×10^{-4}, 1×10^{-3}, $2 \times 10^{-3}, \ldots, 9 \times 10^{-2}$, 1×10^{-1}, 2×10^{-1}, 3×10^{-1}). This includes amplitudes coinciding with training values (permitting in-sample evaluation), amplitudes between training values (testing interpolation) and amplitudes below the training minimum and above the training maximum (testing extrapolation in both directions; § 4.3, figure 5*a*). Training data are mixed across amplitudes during training: at each iteration, snapshots are sampled uniformly across the training set, ensuring that no single amplitude dominates the loss signal.

In this amplitude scan, every run, linear and nonlinear alike, is integrated to $t = 150\,\omega_p^{-1}$, rather than the $500\,\omega_p^{-1}$ used for the single-amplitude nonlinear case of § 4.2; all diagnostics in this section (figures 5–9) are computed over this common window.

Figure 5 summarises the performance of the trained closure across all test amplitudes. Panel (*a*) shows the linear damping rate, obtained by a linear fit to the peaks of $\log W_E$ over the initial damping phase ($0 < t < 15\,\omega_p^{-1}$) of each run for both the kinetic simulation and the fluid+FNO simulation; the agreement is good across the majority of the amplitude range. The closure tracks the kinetic rate not only through the linear plateau but through the nonlinear enhancement of the damping as the amplitude increases, and interpolates accurately to test amplitudes that lie between training values. The agreement breaks down only for the two largest test amplitudes ($A = 0.2$ and 0.3), which lie beyond the maximum amplitude included in training: here the FNO rate departs sharply from the kinetic one, whereas test amplitudes that fall below the minimum training amplitude remain well captured. This asymmetry, a failure to extrapolate above the training envelope but reliable interpolation and extrapolation below it, demonstrates the need to span all dynamically relevant regimes when assembling the training set. Panels (*b–d*) show the time evolution of W_E for representative deep-linear, threshold and deep-nonlinear cases, confirming that a single trained model captures the qualitative evolution in all three regimes. The full set of per-amplitude $W_E(t)$ traces is collected in figure 6, which shows close kinetic–FNO agreement at every trained and interpolated amplitude and the same over-prediction of the envelope at the two extrapolated amplitudes.

Figure 7 reports the late-time electric-field energy $\langle W_E \rangle$ for each amplitude run, computed as the time average of W_E over the final 25 % of the run (i.e. $112.5{-}150\omega_p^{-1}$). We use 'late-time' rather than 'saturation' because the linear cases do not truly saturate within the window; the metric is well defined for the nonlinear cases, which have reached an approximate steady state by this point, and is retained for the linear cases only as a late-time average. The fluid–FNO simulations reproduce the saturation behaviour observed in the kinetic simulations across the nonlinear regime; the linear cases, in which no true saturation is reached within the

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

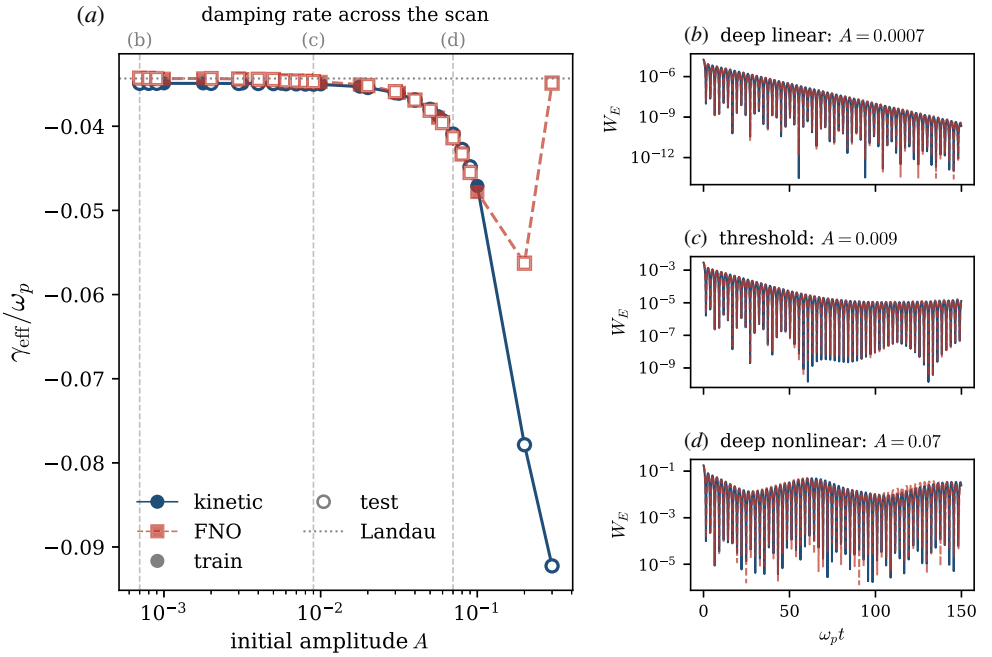


FIGURE 5. (*a*) Damping rate of the initial linear phase of each amplitude run, comparing the kinetic and fluid+FNO. Damping rate is calculated by a linear fit to the peaks of the W_E plots over 0–15 $\omega_{\rm p}t$ of each simulation run. Panels (*b*,*c*,*d*) are plots of W_E over simulation time for a linear, threshold and nonlinear case of Landau damping, respectively. The dotted line marked 'Landau' in panel (*a*) is the analytic linear Landau damping rate for the $k = 0.35$ mode, shown for reference.

simulation window, are still well captured by the FNO closure, despite the ill-defined nature of the late-time electric-field energy for these cases.

Figure 8 shows the increase in total energy, $E_{\rm tot}$, of both kinetic and fluid–FNO simulations over time. The total energy is a Vlasov invariant and when computed from the retained moments it is already complete without a closure: the kinetic energy $(1/2)\int(nu^2 + p)\,{\rm d}x$ involves no moment above the pressure, and the heat flux and higher moments contribute nothing to it. The residual drift in the kinetic curve is therefore not a truncation effect but reflects the finite numerical accuracy of the Vlasov solver (finite velocity grid and phase-space filamentation), growing with amplitude as the dynamics develop a finer velocity-space structure; it sets the baseline against which the fluid+FNO drift, which does evolve only the truncated moment set under the learned closure, should be read. The fluid–FNO simulations exhibit comparable conservation behaviour to the kinetic reference in the linear regimes, while remaining bounded in the deeply nonlinear regimes, despite growing to roughly one to two orders of magnitude larger than the kinetic results; indicating that the learned closure does not introduce significant spurious energy injection or dissipation over the simulation window. We note that the FNO closure is not constructed to enforce energy conservation explicitly, the level of conservation observed therefore reflects the consistency of the learned closure with the physical structure of the underlying equations rather than an imposed constraint.

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

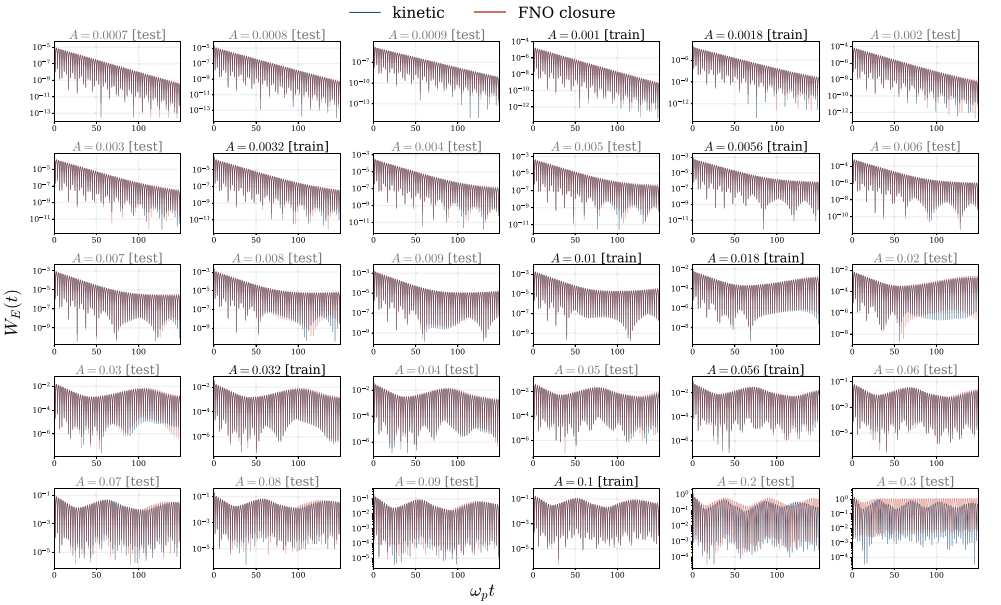


FIGURE 6. Electric-field energy $W_E(t)$ for every amplitude in the test grid, comparing the kinetic simulation (blue) with the online-tested generalised FNO closure (red); each panel is labelled by its initial amplitude and by whether that amplitude was included in training. Amplitude increases across rows. All FNO curves are produced by online testing of the single trained closure from the same initial conditions as the corresponding kinetic run. Agreement is close across the linear, threshold and nonlinear regimes and at both trained and interpolated amplitudes; the two largest amplitudes ($A = 0.2,\ 0.3$), which lie above the training range, show the FNO envelope over-predicting the kinetic one, consistent with the extrapolation failure seen in figure 5(*a*).

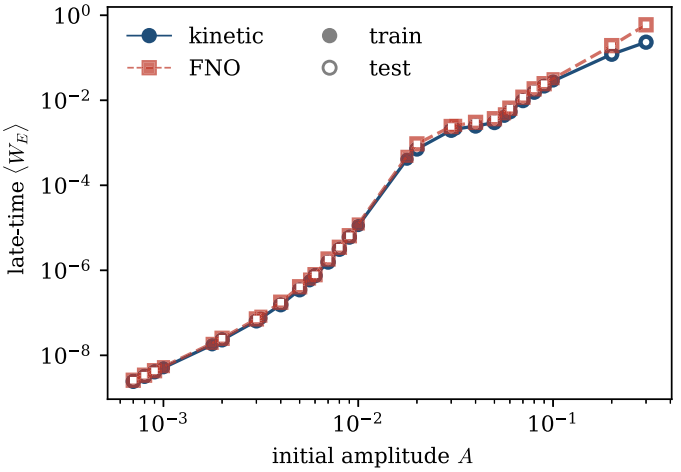


FIGURE 7. Late-time electric-field energy as a function of initial amplitude, comparing the kinetic and fluid+FNO simulations. The late-time value is computed as the time average of W_E over the final 25 % of each simulation run, at which point the nonlinear cases have reached an approximate steady state. The fluid+FNO closure reproduces the late-time energy across the nonlinear regime; the linear cases, for which no true saturation is reached within the simulation window and the metric is correspondingly ill-defined, are nonetheless captured well.

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

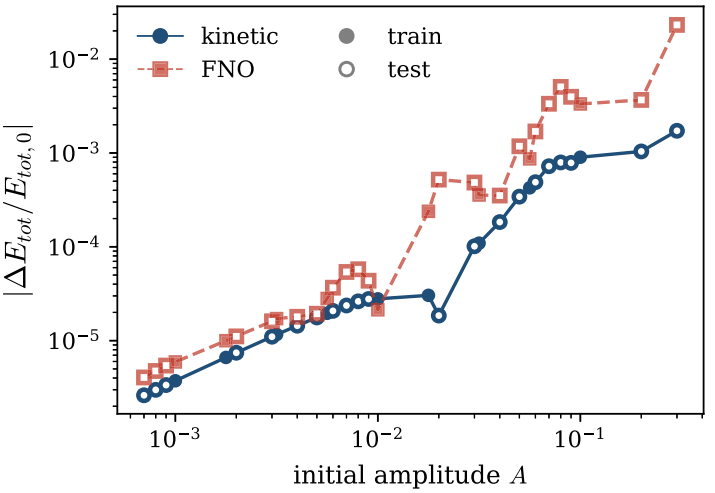


FIGURE 8. Increase of the total energy over time for the kinetic and fluid+FNO simulations across the amplitude scan, quantifying spurious energy injection or dissipation introduced by the closure. In the linear regime the fluid+FNO drift is comparable to the kinetic reference; in the deeply nonlinear regime it remains bounded, though roughly one to two orders of magnitude larger than the kinetic drift. Energy conservation is not imposed on the closure, so the bounded drift reflects consistency of the learned closure with the underlying equations rather than an enforced constraint.

Figure 9 resolves the per-moment relative L^2 error against the kinetic reference as a function of initial amplitude, separated into training and test amplitudes. For every field, the error rises smoothly with amplitude as the dynamics become more nonlinear, and the training and test markers lie essentially on top of one another across the whole scan, confirming that the closure interpolates between trained amplitudes without a generalisation gap. The errors in E, n, u and p climb from below 10^{-2} in the linear regime to order unity only at the largest amplitudes, with a characteristic step near $A \approx 2 \times 10^{-2}$ where the system crosses into the trapping regime. The heat-flux divergence error is the exception: it jumps to order unity as soon as the dynamics become nonlinear and remains there, mirroring the single-amplitude behaviour of § 4.2 and anticipating the effective-heat-flux discussion of § 4.4.

4.4. *The learned closure as an effective heat flux*

Because the online closure is never trained against the kinetic heat flux directly (§ 3.4), there is no guarantee, nor any requirement, that the heat-flux divergence it produces coincides with the one measured from the kinetic simulation. The closure is instead free to converge to whichever $\partial_x q$ best reproduces the evolution of the resolved moments under the specific fluid discretisation in use. We therefore distinguish the effective heat flux learned by the online closure from the kinetic heat flux computed as a velocity moment of the distribution function.

In the linear regime the two remain close. Table 5 shows that the relative L^2 error of $\partial_x q$ against the kinetic value stays at the $\approx 6 \times 10^{-2}$ level throughout the linear training window. In the nonlinear regime, by contrast, the effective and kinetic heat fluxes diverge markedly: the $\partial_x q$ error grows to order unity in the test window, an order of magnitude or more above the density and pressure errors, which remain at

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

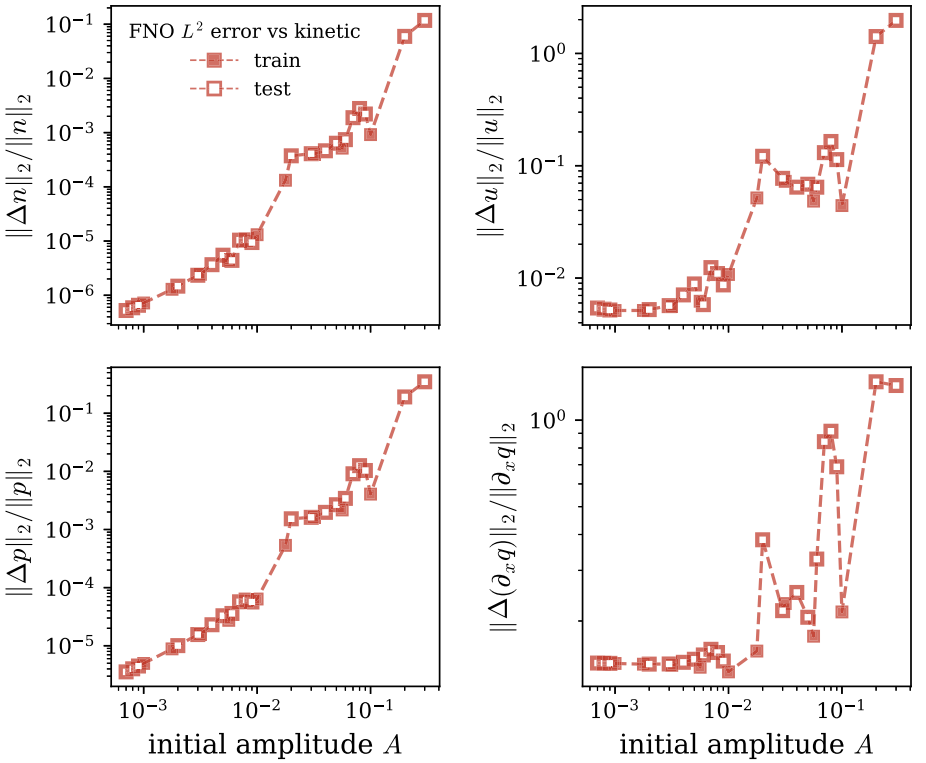


FIGURE 9. Relative L^2 error of each evolved moment's trajectory between the kinetic and fluid+FNO simulations, as a function of initial perturbation amplitude, for the single generalised FNO closure. Errors are the time-averaged relative L^2 error of (4.1). One panel is shown per field (E, n, u, p, $\partial_x q$); filled markers denote amplitudes included in the training set and open markers the held-out test amplitudes. Errors increase monotonically with amplitude as the dynamics become more nonlinear, with training and test markers coinciding across the scan. The heat-flux divergence error (bottom centre) reaches order unity as soon as the trapping regime is entered, whereas the resolved-moment errors rise more gradually.

the 10^{-3}–10^{-2} level (table 5), while the electric-field energy, including its saturation and bounce oscillations, is well reproduced (§ 4.2). The spatial structure of the discrepancy is visible in the $\partial_x q$ row of the contour plots (figure 4). The closure thus reproduces the observable fluid dynamics while producing a heat flux that departs substantially from the kinetic one.

This behaviour is consistent with the truncation of the moment hierarchy, an interpretation we introduced in § 3.4. The kinetic system carries the full velocity dependence of the distribution function, whereas the fluid model retains only moments up to the pressure; the influence of the higher moments, and of the fine velocity-space structure generated by phase mixing, has been projected out. The Mori–Zwanzig formalism shows that, in general, such projected influence reappears as a history-dependent term acting on the resolved variables (Gouasmi *et al.* 2017; Parish & Duraisamy 2017), and this provides a natural interpretation of what the online closure has learned: not the bare kinetic heat flux, but an effective flux that carries the influence of the truncated higher moments through the resolved-field history, so as to reproduce the evolution of the resolved fields. We emphasise, however,

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

that the specific effective flux obtained is not a unique or scheme-independent object. In this language the Hammett–Perkins closure (2.12) is the Markovian, linear limit of the effective flux: a memoryless, spatially non-local operator that captures the linear response but carries none of the history the trapping regime requires. The online closure can be read as a nonlinear, non-Markovian generalisation of the same object, its spectral layers acting as a learned counterpart of the $ik/|k|$ multiplier, consistent with the growth of its memory reliance into the nonlinear regime (§ 4.5). Since it is defined only through its effect on the discretised resolved-moment dynamics, it is expected to depend on the fluid discretisation, the time-integration scheme and the training distribution, and a different solver or training set would in general yield a quantitatively different effective closure. We therefore intend this as a qualitative interpretation rather than a recovery of the exact Mori–Zwanzig memory term. It is consistent with the memory-window design of the closure (§ 3.2) and with the sensitivity analysis of § 4.5, which shows the closure's dependence on the $\partial_x q$ history growing, and decoupling from the lower moments, as the dynamics become nonlinear.

The contrast with the offline closure is instructive. Because the offline model is trained to regress $\partial_x q_{\mathrm{kin}}$ directly (§ 3.5), it reproduces the kinetic heat flux more closely by construction; yet, in our matched comparison, this closer fidelity to the kinetic flux did not translate into better reproduction of the resolved-moment dynamics under online deployment (§§ 4.1 and 4.2). We do not read this as evidence that offline training is inherently inferior, offline closures can perform well under other configurations, including for this same problem (Huang *et al.* 2025). Rather, it suggests that, at least under identical hyperparameters and training data, reproducing the kinetic heat flux pointwise is not by itself sufficient for an accurate closed fluid model; what matters is the closure's effect on the retained moments, which online training optimises directly.

4.5. *What the closure has learned*

Having established that the closure reproduces the kinetic dynamics and generalises across amplitude (§§ 4.1–4.3), we now ask how it does so. The FNO closure is a black-box model, and such a closure could in principle reproduce a trajectory for the wrong reason, by extrapolating its own recent heat-flux output forward in time rather than computing the heat flux from the state of the plasma; such a model would not be expected to generalise across amplitudes as ours does. Two questions follow; first, does the closure compute a genuine relation from the resolved-moment fields (n, u, p, E) to the heat-flux divergence, or does it merely continue its own $\partial_x q$ history? Second, does the memory window introduced in § 3.2 carry information the closure actually uses?

We probe both by measuring how the predicted heat-flux divergence of the FNO responds to its inputs. The primary diagnostic is an ablation test, where we replace a single input field, or a group of them, with its average value, which removes the spatial and temporal variation, and record the resulting change in $\partial_x q$. A large change signals a strong dependence on that input. We complement this with the Hilbert–Schmidt independence criterion (HSIC) (Gretton *et al.* 2005), a measure of statistical dependence that, unlike a correlation coefficient, captures nonlinear as well as linear relationships. It generalises covariance by computing it in a transformed (kernel) feature space, so that it detects association of any functional form and vanishes only when the two quantities are statistically independent. Both diagnostics act on the deployed closure over representative data; since the ablation test displaces

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

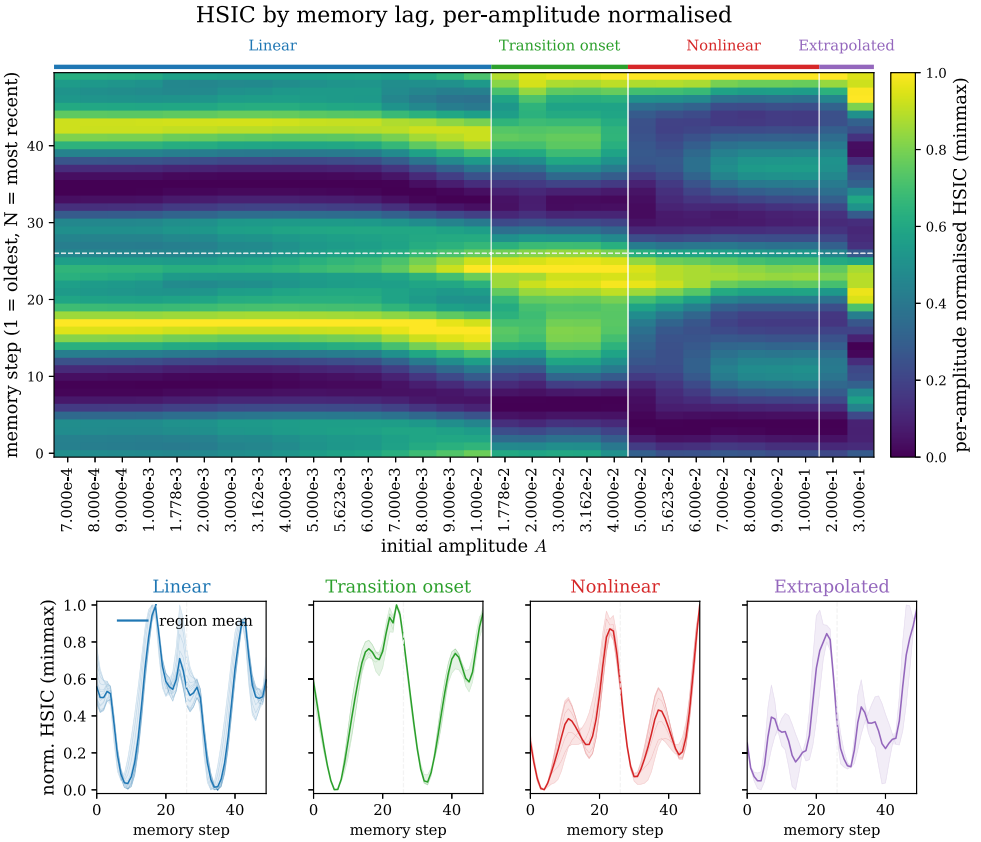


FIGURE 10. Strength of the closure's dependence on its memory inputs, as a function of how far back in time the input lies (vertical axis, the memory lag) and of the initial perturbation amplitude (horizontal axis). Dependence is measured by the HSIC (a statistical association measure; see text) between each past field and the predicted heat-flux divergence, and is rescaled to [0, 1] separately at each amplitude so the lag pattern can be compared across amplitudes. The dependence concentrates at time lags spaced by the oscillation period of the electric-field energy (dashed line), showing that the closure reads the phase of the oscillation from its memory rather than an undifferentiated recent history. The pattern also changes character across the linear, threshold and nonlinear regimes (left to right).

the inputs from the states the closure normally sees, we read the results as qualitative trends rather than precise values. Both ablation and HSIC diagnostics are computed over all available time steps of each amplitude run.

The memory window carries structured rather than uniform information (figure 10). The dependence on the past fields concentrates at particular lags, spaced by the oscillation period of the electric-field energy, so the closure reads the phase of the underlying oscillation from its memory rather than an undifferentiated recent history; the lag structure additionally differs between the linear, threshold and nonlinear regimes. This structure should be read with a caveat intrinsic to the measure. Because the resolved fields oscillate at close to the plasma frequency, each past field is autocorrelated with the present state, which is in turn related to the target, so a given lag may register a high dependence either because the closure draws on it or because the field there is a proxy for the present state. The measured pattern is thus

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

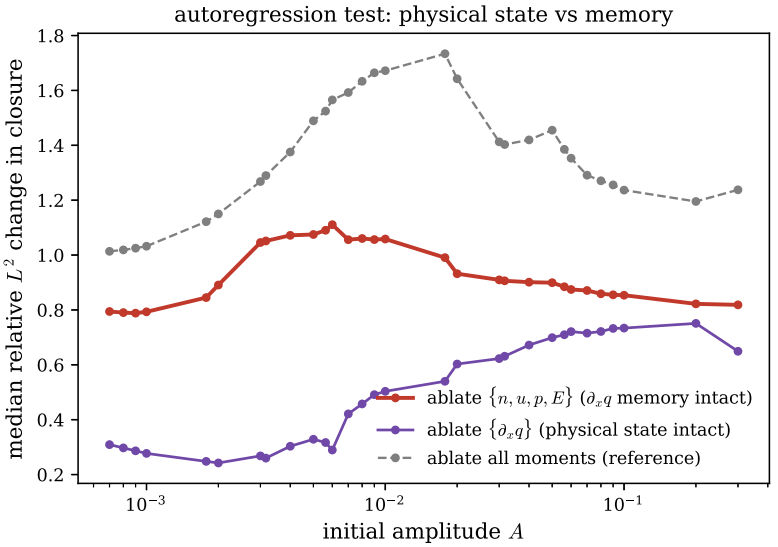


FIGURE 11. Test of whether the closure reads the plasma state or merely continues its own output. Each curve shows how much the predicted heat-flux divergence changes (median relative L^2 change) when a set of inputs is replaced by its average value, as a function of initial amplitude. Replacing the physical moment fields $\{n, u, p, E\}$ while keeping the heat-flux history intact (solid) changes the prediction strongly at every amplitude, and by more than removing the heat-flux history alone: the closure depends on the plasma state and is not simply extrapolating its past output. The overall dependence on the moment inputs (dashed) grows as the dynamics become nonlinear.

a combination of the true memory dependence and the autocorrelation of the fields, which HSIC alone cannot separate; the autocorrelation of an oscillation peaks at lags spaced by its period, the same spacing we attribute to the closure reading of the oscillation phase, so here the two reinforce rather than distinguish one another. What the autocorrelation cannot produce is the response to the ablation test below, in which the physical state is removed while the heat-flux history is retained.

The ablation test shows the closure to compute a genuine relation from the plasma state to the heat flux, rather than continuing its own output (figure 11). Replacing the physical moment fields $\{n, u, p, E\}$ with their averages while leaving the $\partial_x q$ history intact changes the prediction substantially at every amplitude, and by more than removing the $\partial_x q$ history alone; a closure extrapolating its past output would be almost unaffected by the loss of the physical state. This dependence on the moment fields remains strong (and does not weaken) as the dynamics become nonlinear, and taken with the generalisation of § 4.3 indicates a transferable physical relation rather than memorised trajectories.

Among the resolved fields the closure draws primarily on the pressure. Density and pressure are nearly proportional across the amplitude scan (figure 12, correlation $\approx$0.9) and so carry almost the same information, and the ablation tests identify pressure as the field that supplies it (figure 13): removing density and pressure together changes the prediction by little more than removing pressure alone, leaving density largely redundant once pressure is present. The electric field is redundant with the density in the same way, adding little to the $\{n, p\}$ group when included (figure 14), as the electrostatic constraint requires: Poisson's equation (2.11) ties

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

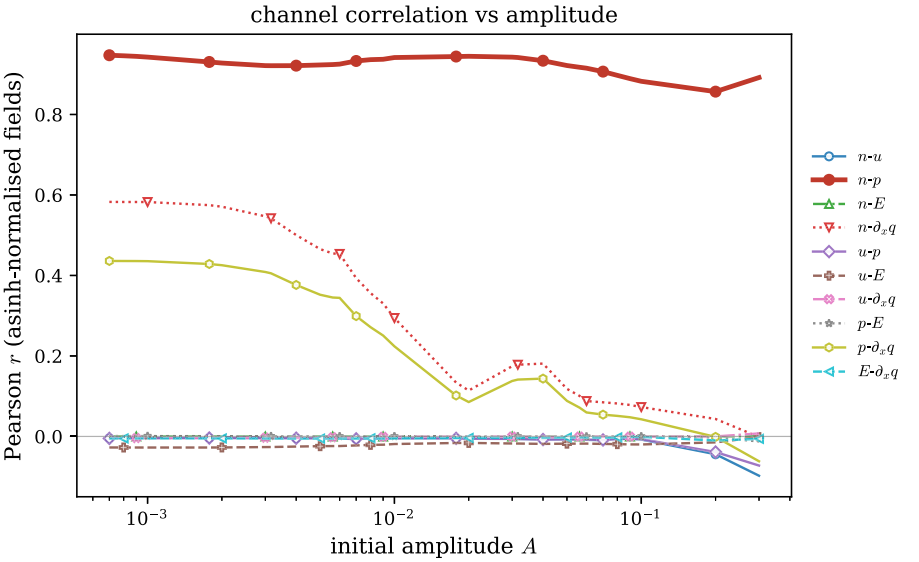


FIGURE 12. How closely the input fields track one another, as a function of initial amplitude (computed on the fields as supplied to the network). Correlation between pairs of input fields: values near ± 1 mean the two carry nearly the same information, values near 0 mean they are unrelated. The density–pressure pair (emphasised) stays near 0.9 at all amplitudes, the two are nearly interchangeable, while the heat-flux history becomes progressively unrelated to n and p as the dynamics become nonlinear.

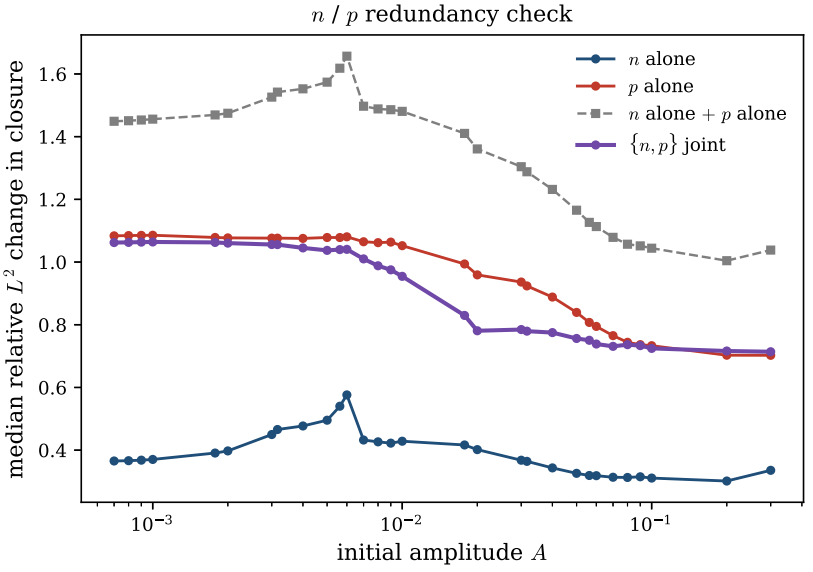


FIGURE 13. Which of density and pressure the closure actually uses. Curves show the change in the predicted heat-flux divergence when density alone, pressure alone or both together are replaced by their averages, versus initial amplitude, together with the sum of the two single-field values. Removing both together has an effect close to removing pressure alone, and smaller than the two single-field effects added together: pressure carries most of the shared density–pressure information, and density adds little once pressure is present.

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

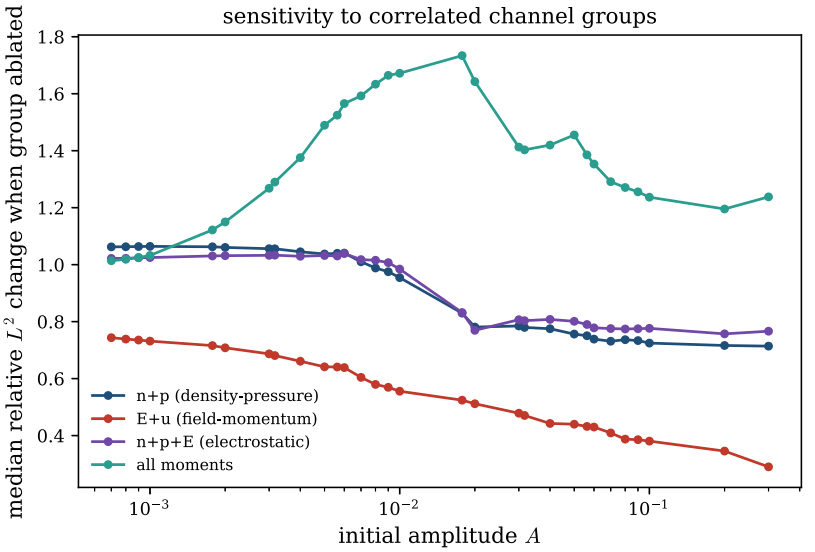


FIGURE 14. Dependence of the closure on groups of related inputs, measured as the median relative L^2 change in the predicted heat-flux divergence when each group is replaced by its average, versus initial amplitude. The $\{n, p\}$ and $\{n, p, E\}$ curves almost coincide, so adding the electric field to the density–pressure group makes little difference: E carries essentially no information the density does not (as expected from Poisson's equation). The curve for all moment fields together rises through the transition region before partially falling in the deeply nonlinear regime; the specifically growing reliance on the $\partial_x q$ history is shown in figure 11 rather than here.

E to the density through $\partial_x E = (q_e/\epsilon_0)(n - n_0)$, so E and the density perturbation carry the same information, up to the integration constant fixed by the periodic boundary condition.

Finally, the diagnostics indicate why the memory is needed, beyond confirming that it is used. As the amplitude increases into the nonlinear regime, the heat-flux history becomes at once more important to the closure, the effect of ablating the $\partial_x q$ history in figure 11 grows with amplitude, and less related to the instantaneous lower moments, its correlation with n and p falling towards zero (figure 12). The heat flux thus acquires dynamics that cannot be reconstructed from the present values of the other moments, the signature of phase mixing, precisely the regime in which a memory-dependent, non-Markovian closure is necessary. The measured behaviour of the closure therefore mirrors the physical reasoning that motivated its finite-memory design (§ 3.2). This growing reliance on the heat-flux history is the same trend invoked in § 4.4: the closure's departure from the kinetic heat flux in the nonlinear regime, and its increasing dependence on memory there, are two readings of one fact, that the effective flux becomes a genuinely history-dependent quantity as the truncated higher-moment dynamics grow in importance.

5. Discussion

The results of the previous section establish three claims. First, training a FNO closure online, within a differentiable fluid solver, produces a closure that is numerically stable when subsequently deployed in independent fluid simulations and that reproduces the kinetic dynamics of one-dimensional electrostatic Landau damping

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

in both linear and nonlinear regimes. Second, a single online-trained FNO can generalise across initial perturbation amplitudes that lie between those represented in the training set. Third, in the nonlinear regime the heat flux learned by the online-trained closure does not exactly reproduce the heat flux measured directly from the kinetic simulation.

Taken together, these results establish online training as a viable route to producing learned closures that operate stably and accurately within fluid simulations and that generalise across the range of initial conditions one would encounter in a typical large-scale modelling application. Our work builds on Joglekar & Thomas (2023), who first demonstrated that a learnable closure embedded in a differentiable solver can capture both the linear and nonlinear regimes of Landau damping across a range of initial conditions. We extend this line of work in three ways. First, rather than combining a symbolic model with a neural-network component, we use a single FNO as the entire closure – a pure operator-learning approach that assumes no predetermined functional form for the heat flux. Second, because the operator acts on an explicit trailing memory window, the resulting closure is non-Markovian in a form we can interrogate directly; the sensitivity analysis of § 4.5 shows what the learned memory encodes. Third, we characterise online generalisation systematically, across a dense amplitude scan with explicit interpolation and extrapolation tests, and we quantify the stability and accuracy of the online-trained closure over long rollouts and against matched offline-trained and analytic Hammett–Perkins baselines, over which it shows a clear advantage. We present these as complementary advances that build on the earlier demonstration.

Several limitations of the present study suggest directions for further work. The online training procedure is substantially more memory intensive than offline training, owing to the need to backpropagate through the fluid rollout. This high-memory requirement places constraints on the rollout length and batch size achievable on standard hardware and motivates an investigation of memory-efficient automatic differentiation strategies such as gradient checkpointing or implicit differentiation. The choice of the FNO architecture, while well motivated for the periodic, low-wavenumber dynamics of Landau damping, has not been compared against alternative architectures trained under the same online procedure; such a comparison would clarify the extent to which our results reflect properties of the FNO specifically as opposed to properties of online training more generally. The geometry of our problem, one-dimensional, electrostatic, with a fixed wavenumber, is also deliberately simple. Extending the online training method to fully electromagnetic, multi-dimensional settings, and to phenomena such as magnetic reconnection or plasma turbulence (Pezzi *et al.* 2019), will require addressing both the increased dimensionality of the inputs and outputs and the much greater computational cost of the underlying kinetic simulations.

A further question for production use is whether a trained closure transfers, without retraining, to a solver run at a different time step or spatial resolution. The FNO's spectral parameterisation is resolution flexible in space, so transfer across grid resolutions should be feasible; the time step is more delicate, because the memory window is defined in integration steps ($K = 50$ at $\Delta t = 0.1\,\omega_p^{-1}$, a fixed $5\,\omega_p^{-1}$ horizon), so changing Δt at inference rescales the physical memory horizon and the lag spacing the closure has learned to read (§ 4.5). Testing $\Delta t-$ and resolution transfer, and if necessary re-expressing the memory window in physical time rather than steps, is a next step we leave to future work.

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

The spectral truncation of the FNO, which retains only the lowest Fourier modes in each layer, is well matched to the single-mode, low-wavenumber dynamics considered here, but may become a limitation in richer settings where several wavemodes are dynamically active or where important structure resides at higher wavenumbers. The number of retained modes can of course be increased, but at the cost of additional parameters and training expense, and the appropriate mode content is ultimately problem dependent. Extending the method to multi-mode or broadband problems will therefore require matching the retained spectrum to the range of scales that carry the physics of interest, and assessing how the closure behaves for modes whose damping rates differ substantially across the retained band.

Two features of the closure bear on its use in domain-decomposed production codes: the temporal memory window ($K = 50$ stored resolved states per cell) and the spatial non-locality of the FNO. The storage overhead relative to a Markovian closure is a factor of order K per evolved field, which is modest in one dimension but grows with dimension. The spectral layers, as implemented here, act through global Fourier transforms over the periodic domain; in a distributed setting one would instead evaluate the closure on local patches with halo exchange, which raises the questions of how large a patch must be relative to the characteristic wavelength (here set by the retained 32 modes in the FNO at $k = 0.35$) and how to handle the loss of periodicity on an individual patch, for example, through windowed transforms or overlap regions. We regard establishing patch-local, distributed-memory FNO closures as a prerequisite for multi-dimensional deployment and leave it to future work.

Our generalisation tests vary the initial perturbation amplitude at fixed wavenumber k and temperature T. Generalisation across other parameters is not demonstrated here and is not automatic: the arcsinh and A_{scale} normalisation (§ 3.3) removes amplitude but not wavenumber or temperature, so a single model would not be expected to interpolate across k or T without either including them as conditioning inputs or retraining. Wavenumber generalisation is the more delicate of the two, since the retained Fourier-mode content and the physical memory horizon are tied to the dominant scale; temperature enters mainly through the normalisation of velocity and the linear damping rate, and is the more plausible candidate for conditioning-based generalisation. A systematic scan across k and T is left to future work.

Despite these limitations, the present results are a concrete step towards machine-learned plasma closures suitable for large-scale fluid simulation. Where kinetic simulations are too expensive to run repeatedly but the relevant physical regimes are spanned by a tractable subset of fully kinetic runs, an online-trained closure derived from that subset offers a route to fluid-cost simulations that retain the kinetic effects most relevant to the regime of interest.

Acknowledgements

Editor Luís O. Silva thanks the referees for their advice in evaluating this article.

Funding

This research utilised Queen Mary's Apocrita HPC facility, supported by QMUL Research-IT. (DOI: http://doi.org/10.5281/zenodo.438045) This research was supported by a Queen Mary University of London Principal's Award Studentship. OP acknowledges support by the FIS2 Starting Grant FIS-2023-00246 'PhAse-sPAce cOmplexity in turbulent nearly reversible plasmas (PAPAO)' (CUP B53C24009610001) funded by the Italian Ministry of University and Research.

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

Declaration of interests

The authors report no conflict of interest.

Data availability

Code and training data are openly available on Zenodo: Burles, S., Camporeale, E. & Pezzi, O. (2026). A Neural Operator Closure for Landau Damping in Electrostatic Plasma - Code + Data (Version 1.0). Zenodo. https://doi.org/10.5281/zenodo.22800112

REFERENCES

BRUNTON, S.L., NOACK, B.R. & KOUMOUTSAKOS, P. 2020 Machine learning for fluid mechanics. *Annu. Rev. Fluid Mech.* **52**, 477–508.

BURLES, S. & CAMPOREALE, E. 2026 The machine learning approach to moment closure relations for plasma: a review. *J. Plasma Phys.* **92**, E100.

CELEBRE, G., SERVIDIO, S. & VALENTINI, F. 2023 Phase space dynamics of unmagnetized plasmas: collisionless and collisional regimes. *Phys. Plasmas* **30**, 092304.

CHENG, C.Z. & KNORR, G. 1976 The integration of the Vlasov equation in configuration space. *J. Comput. Phys.* **22**, 330–351.

CHEW, G.F., GOLDBERGER, M.L. & LOW, F.E. 1956 The Boltzmann equation and the one-fluid hydromagnetic equations in the absence of particle collisions. *Proc. R. Soc. Lond. A* **236**, 112–118.

CHORIN, A.J., HALD, O.H. & KUPFERMAN, R. 2000 Optimal prediction and the Mori–Zwanzig representation of irreversible processes. *Proc. Natl Acad. Sci.* **97**, 2968–2973.

FILBET, F. & PARESCHI, L. 2002 A numerical method for the accurate solution of the Fokker–Planck–Landau equation in the nonhomogeneous case. *J. Comput. Phys.* **179**, 1–26.

FREZAT, H., LE SOMMER, J., FABLET, R., BALARAC, G. & LGUENSAT, R. 2022 A posteriori learning for quasi-geostrophic turbulence parametrization. *J. Adv. Model. Earth Syst.* **14**, e2022MS003124.

GOSWAMI, P., PASSOT, T. & SULEM, P.L. 2005 A Landau fluid model for warm collisionless plasmas. *Phys. Plasmas* **12**, 102109.

GOUASMI, A., PARISH, E.J. & DURAISAMY, K. 2017 *A priori* estimation of memory effects in reduced-order models of nonlinear systems using the Mori-Zwanzig formalism. *Proc. R. Soc. A* **473**, 20170385.

GRETTON, A., BOUSQUET, O., SMOLA, A. & SCHÖLKOPF, B. 2005 Measuring statistical dependence with Hilbert-Schmidt norms. In *Algorithmic Learning Theory (ALT 2005), Lecture Notes in Computer Science*, vol. 3734, pp. 63–77. Springer.

GUPTA, A. & LERMUSIAUX, P.F.J. 2023 Generalized neural closure models with interpretability. *Sci. Rep-UK.* **13**, 10634.

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**, 3019–3022.

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.* **122**, e2419073122.

HUNANA, P., TENERANI, A., ZANK, G.P., GOLDSTEIN, M.L., WEBB, G.M., KHOMENKO, E., COLLADOS, M., CALLY, P.S., ADHIKARI, L. & VELLI, M. 2019 An introductory guide to fluid models with anisotropic temperatures. Part 2. Kinetic theory, Padé approximants and Landau fluid closures. *J. Plasma Phys.* **85**, 205850603.

HUNANA, P., ZANK, G.P., LAURENZA, M., TENERANI, A., WEBB, G.M., GOLDSTEIN, M.L., VELLI, M. & ADHIKARI, L. 2018 New closures for more precise modeling of Landau damping in the fluid framework. *Phys. Rev. Lett.* **121**, 135101.

JOGLEKAR, A.S. & THOMAS, A.G.R. 2023 Machine learning of hidden variables in multiscale fluid simulation. *Mach. Learn.* **4**, 035049.

KOSSAIFI, J., KOVACHKI, N., LI, Z., PITT, D., LIU-SCHIAFFINI, M., GEORGE, R.J., BONEV, B., AZIZZADENESHELI, K., BERNER, J. & ANANDKUMAR, A. 2024 A library for learning neural operators. arXiv:2412.10354.

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

KRALL, N.A. & TRIVELPIECE, A.W. 1973 *Principles of Plasma Physics*. McGraw-Hills.

LAPENTA, G., SCHRIVER, D., WALKER, R.J., BERCHEM, J., ECHTERLING, N.F., ALAOUI, E., MOSTAFA, & TRAVNICEK, P. 2022 Do we need to consider electrons' kinetic effects to properly model a planetary magnetosphere: the case of Mercury. *J. Geophys. Res.: Space Phys.* **127**, e2021JA030241.

LAPERRE, B., AMAYA, J., JAMAL, S. & LAPENTA, G. 2022 Identification of high order closure terms from fully kinetic simulations using machine learning. arXiv:2110.09916.

LAUTENBACH, S. & GRAUER, R. 2018 Multiphysics simulations of collisionless plasmas. *Front. Phys.* **6**, 113.

LE, A., DAUGHTON, W., KARIMABADI, H. & EGEDAL, J. 2016 Hybrid simulations of magnetic reconnection with kinetic ions and fluid electron pressure anisotropy. *Phys. Plasmas* **23**, 032114.

LE, A., EGEDAL, J., DAUGHTON, W., FOX, W. & KATZ, N. 2009 Equations of state for collisionless guide-field reconnection. *Phys. Rev. Lett.* **102**, 085001.

LI, T., BIFERALE, L., BONACCORSO, F., SCARPOLINI, M.A. & BUZZICOTTI, M. 2024 Synthetic Lagrangian turbulence by generative diffusion models. *Nat. Mach. Intell.* **6**, 393–403.

LI, Z., KOVACHKI, N., AZIZZADENESHELI, K., LIU, B., BHATTACHARYA, K., STUART, A. & ANANDKUMAR, A. 2021 Fourier neural operator for parametric partial differential equations. arXiv:2010.08895.

LIU, K., CHENG, W., FU, H., JIANG, M., MA, J. & QIN, Y. 2022 Data-driven Modeling of Plasma Fluid Closure and Parameter Prediction. In *2022 International Applied Computational Electromagnetics Society Symposium (ACES-China)*, pp. 1–3.

LU, L., JIN, P., PANG, G., ZHANG, Z. & KARNIADAKIS, G.E. 2021 Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature Mach. Intell.* **3**, 218–229.

MA, C., ZHU, B., XU, X.-Q. & WANG, W. 2020 Machine learning surrogate models for Landau fluid closure. *Phys. Plasmas* **27**, 042502.

MACART, J.F., SIRIGNANO, J. & FREUND, J.B. 2021 Embedded training of neural-network subgrid-scale turbulence models. *Phys. Rev. Fluids* **6**, 050502.

MANGENEY, A., CALIFANO, F., CAVAZZONI, C. & TRAVNICEK, P. 2002 A numerical scheme for the integration of the Vlasov–Maxwell system of equations. *J. Comput. Phys.* **179**, 495–538.

MAULIK, R., GARLAND, N.A., BURBY, J.W., TANG, X.-Z. & BALAPRAKASH, P. 2020 Neural network representability of fully ionized plasma fluid model closures. *Phys. Plasmas* **27**, 072106.

MCGRAE-MENGE, M.C., PIERCE, J.R., FIUZA, F. & ALVES, E.P. 2026 Embedding physical symmetries into machine-learned reduced plasma physics models via data augmentation. *Phys. Rev. Res.* **8**, 013200.

MILOSHEVICH, G., VRANCKX, L., LOPES, F.N. DE O., DAZZI, P., ARRÒ, G. & LAPENTA, G. 2025 Electron neural closure for turbulent magnetosheath simulations: energy channels. arXiv:2510.00282 [physics].

MORI, H. 1965 Transport, collective motion, and Brownian motion. *Prog. Theor. Phys.* **33**, 423–455.

NG, J., HAKIM, A., WANG, L. & BHATTACHARJEE, A. 2020 An improved ten-moment closure for reconnection and instabilities. *Phys. Plasmas* **27**, 082106.

OHIA, O., EGEDAL, J., LUKIN, V.S., DAUGHTON, W. & LE, A. 2012 Demonstration of anisotropic fluid closure capturing the kinetic structure of magnetic reconnection. *Phys. Rev. Lett.* **109**, 115004.

O'NEIL, T. 1965 Collisionless damping of nonlinear plasma oscillations. *Phys. Fluids* **8**, 2255–2262.

PARISH, E.J. & DURAISAMY, K. 2017 Non-Markovian closure models for large eddy simulations using the Mori-Zwanzig formalism. *Phys. Rev. Fluids* **2**, 014604.

PASSOT, T. & SULEM, P.L. 2006 A fluid model with finite Larmor radius effects for mirror mode dynamics. *J. Geophys. Res.: Space Phys* **111**, A04203.

PASZKE, A. *et al.* 2019 PyTorch: an imperative style, high-performance deep learning library. *Adv. Neural Inform. Process. Syst.* **32**, 8024–8035.

PEZZI, O., CAMPOREALE, E. & VALENTINI, F. 2016 Collisional effects on the numerical recurrence in Vlasov-Poisson simulations. *Phys. Plasmas* **23**, 022103.

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press

PEZZI, O., COZZANI, G., CALIFANO, F., VALENTINI, F., GUARRASI, M., CAMPOREALE, E., BRUNETTI, G., RETINÒ, A. & VELTRI, P. 2019 ViDA: a Vlasov-DArwin solver for plasma physics at electron scales. *J. Plasma Phys.* **85**, 905850506.

PEZZI, O., VALENTINI, F., PERRONE, D. & VELTRI, P. 2013 Eulerian simulations of collisional effects on electrostatic plasma waves. *Phys. Plasmas* **20**, 092111.

QIN, Y., MA, J., JIANG, M., DONG, C., FU, H., WANG, L., CHENG, W. & JIN, Y. 2023 Data-driven modeling of Landau damping by physics-informed neural networks. *Phys. Rev. Res.* **5**, 033079.

SANDERSE, B., STINIS, P., MAULIK, R. & AHMED, S.E. 2024 Scientific machine learning for closure models in multiscale problems: a review. *Found. Data Sci.* **7**, 298–337.

SHANKAR, V., CHAKRABORTY, D., VISWANATHAN, V. & MAULIK, R. 2024 Differentiable turbulence: closure as a partial differential equation constrained optimization, arXiv:2307.03683.

SHUKLA, A., HATCH, D.R., DORLAND, W. & MICHOSKI, C. 2022 A learned closure method applied to phase mixing in a turbulent gradient-driven gyrokinetic system in simple geometry. *J. Plasma Phys.* **88**, 905880115.

SNYDER, P.B., HAMMETT, G.W. & DORLAND, W. 1997 Landau fluid models of collisionless magneto-hydrodynamics. *Phys. Plasmas* **4**, 3974–3985.

SULEM, P.L. & PASSOT, T. 2015 Landau fluid closures with nonlinear large-scale finite Larmor radius corrections for collisionless plasmas. *J. Plasma Phys.* **81**, 325810103.

SWANSON, D.G. 2003 Plasma waves (2nd edition). *Plasma Phys. Control. Fusion* **45**, 1069.

WANG, L., HAKIM, A.H., BHATTACHARJEE, A. & GERMASCHEWSKI, K. 2015 Comparison of multi-fluid moment models with particle-in-cell simulations of collisionless magnetic reconnection. *Phys. Plasmas* **22**, 012108.

WANG, Q., RIPAMONTI, N.Ò & HESTHAVEN, J.S. 2020 Recurrent neural network closure of parametric POD-Galerkin reduced-order models based on the Mori-Zwanzig formalism. *J. Comput. Phys.* **410**, 109402.

WEI, S., LIU, Y., FU, H., DONG, C. & WANG, L. 2023 Data-driven modeling of Landau damping by Fourier neural operator. In *2023 International Applied Computational Electromagnetics Society Symposium (ACES-China)*, pp. 01–03.

ZWANZIG, R. 1961 Memory effects in irreversible thermodynamics. *Phys. Rev.* **124**, 983–992.

https://doi.org/10.1017/S0022377826102517 Published online by Cambridge University Press