A Physics-Informed, Global-in-Time Neural Particle Method for the Spatially Homogeneous Landau Equation

Yeoneung Kim [1]

Incoming faculty, Department of Industrial Engineering,
Yonsei University (from September 2026)

Joint work with Minseok Kim[1] Sung-Jun Son[2] Donghyun Lee[2]

[1] Department of Applied Artificial Intelligence, SeoulTech
[2] Department of Mathematics, POSTECH

PARTIQAL 2026

Motivation: Collisional Plasmas and the Landau Operator

Where does this equation come from?

The **Landau (Fokker–Planck–Landau) operator** is the kinetic model of choice for *weakly-coupled, collisional plasmas*: magnetic-confinement fusion (tokamaks), inertial confinement, and astrophysical plasmas.

- **Physics:** Coulomb collisions are dominated by *grazing* (small-angle) encounters. Summing these many small deflections yields the Landau operator (Coulomb case $\gamma = -3$), not the Boltzmann operator.
- **Structure to preserve:** conservation of mass, momentum, energy; the H-theorem (entropy dissipation); relaxation to the Maxwellian.
- **Computational challenge:** the distribution lives in up to **6D phase space** ($3x + 3v$); the collision operator is *nonlinear and nonlocal* in velocity. Grid-based solvers scale poorly.

Goal: a mesh-free, dimension-robust solver that keeps the physics — motivating deterministic particle + machine-learning approaches.

The General Landau Equation

Full Phase-Space Dynamics

The general Landau equation describes the time evolution of the particle distribution function $f_t(x, v)$ in full phase space (position $x \in \mathbb{R}^d$, velocity $v \in \mathbb{R}^d$):

$$\partial_t f_t(x, v) + v \cdot \nabla_x f_t(x, v) = Q_L(f_t, f_t)(x, v)$$

- **Left-hand side (Free Transport):** Represents the advection of particles moving through physical space with velocity v.
- **Right-hand side (Collision Operator):** The Landau collision operator Q_L, which models grazing collisions (small-angle scattering) between particles. It acts exclusively on the velocity variable v.

Reduction to the Spatially Homogeneous Case

If we assume the particle distribution is uniform across physical space, the spatial gradient vanishes ($\nabla_x f_t = 0$). The density then depends solely on time and velocity: $f_t(v)$.

$$\partial_t f_t(v) = Q_L(f_t, f_t)(v)$$

This isolated velocity-space dynamics is the primary focus of our work.

Spatially Homogeneous Landau Equation

Transport Formulation

The spatially homogeneous Landau equation governs the time evolution of a distribution function $\tilde{f}_t(v)$ of charged particles:

$$\partial_t \tilde{f}_t(v) = -\nabla_v \cdot \left(\tilde{f}_t(v) U[\tilde{f}_t](v)\right)$$

where the mean-field drift $U[\tilde{f}_t](v)$ is driven by the **score** $\tilde{s}_t = \nabla_v \log \tilde{f}_t$:

$$U[\tilde{f}_t](v) = -\int_{\mathbb{R}^d} A(v - v^*)\left(\tilde{s}_t(v) - \tilde{s}_t(v^*)\right)\tilde{f}_t(v^*)\, dv^*$$

- $A(z) = C_\gamma |z|^\gamma \left(|z|^2 I - z \otimes z\right)$, for $\gamma \in [-3, 1]$.
- The equation describes nonlinear, nonlocal dynamics with a gradient-flow structure.
- *At the PDE level*, the exact solution conserves mass, momentum, and energy, with monotone entropy dissipation. (A numerical scheme preserves these only up to its discretization/approximation error.)

Understanding the Landau Collision Kernel

Mathematical Form & Orthogonality

The collision kernel $A(z)$ dictates how particles with relative velocity $z = v - v^*$ interact:

$$A(z) = C_\gamma |z|^\gamma \left(|z|^2 I - z \otimes z \right)$$

Crucially, the matrix $(|z|^2 I - z \otimes z)$ acts as a **projection operator** onto the orthogonal complement of z. It satisfies $A(z)z = 0$, meaning the interaction force is always orthogonal to the relative velocity.

- **Physical Intuition (Grazing Collisions):** Unlike the Boltzmann equation which models abrupt, large-angle scattering, the Landau kernel models continuous, *small-angle (grazing) collisions*. Particles interact by continuously "sliding" past one another, which is typical in dense plasmas.
- **Interaction Potentials (γ):**
  - $\gamma = 1$: **Hard potentials** (interaction force grows with relative speed).
  - $\gamma = 0$: **Maxwellian molecules** (constant collision rate, mathematically highly tractable).
  - $\gamma = -3$: **Coulomb interactions** (highly singular, long-range forces dominating plasma physics).
- **Geometric Role in Gradient Flows:** $A(z)$ acts as a state-dependent metric tensor. It shapes the precise geometric pathway the density f_t takes as it dissipates entropy towards the Maxwellian equilibrium.

Why Particle Methods for Kinetic Equations?

The Curse of Dimensionality

Classical Eulerian approaches (Finite Volume, Finite Difference, Spectral methods) require grid discretization. In kinetic theory, the phase space (position + velocity) is high-dimensional ($d \geq 3$ or up to 6), making grid-based methods computationally prohibitive.

The Particle (Lagrangian) Paradigm

Representing the density as an empirical measure of interacting particles: $f_t(v) \approx \frac{1}{N} \sum_{i=1}^{N} \delta_{v_t^{(i)}}$

- **Mesh-free:** Effort scales with N, bypassing the grid-based curse of dimensionality.
- **Adaptive Resolution:** Particles naturally cluster in regions of high probability.
- **Positivity & Mass:** Guaranteed by construction (empirical measure); momentum/energy conservation is scheme-dependent.

A Physicist's Picture of Optimal Transport

Wasserstein distance = "earth-mover" cost

$W_2(f, g)$ is the minimal kinetic cost to rearrange the mass of one distribution f into another g. It endows the space of probability distributions with a **geometry** (a way to measure distance and "steepest descent").

- **Gradient flow:** just as an overdamped particle slides down a potential $\dot{x} = -\nabla V(x)$, a *distribution* can slide downhill on a free-energy landscape $\mathcal{E}[f]$ — but "downhill" is measured in the Wasserstein geometry.
- **For Landau:** the free energy is the (negative) entropy $\mathcal{H}[f] = \int f \log f \, dv$. The equation is the steepest descent of $\mathcal{H}$ in a *collision-kernel-weighted* transport metric.
- **The payoff:** this is exactly the **H-theorem you already know** ($\frac{d}{dt}\mathcal{H} \leq 0$), now read as energy dissipation along a gradient flow. The driving force is the **score** $\nabla_v \log f$.

Optimal transport is not new physics — it is a geometric language for the relaxation you already understand.

The Optimal Transport and Gradient Flow Perspective

- Over the last two decades, interacting particle systems have been deeply unified with **Optimal Transport** theory.
- Many kinetic and aggregation-diffusion equations can be reinterpreted as **Gradient Flows** in the Wasserstein space of probability measures:

$$\partial_t f = \nabla_v \cdot \left(f \nabla_v \frac{\delta \mathcal{E}(f)}{\delta f} \right)$$

- **Stochastic vs. Deterministic:**
  - *Stochastic (DSMC):* Uses Brownian motion/random collisions. Suffers from slow convergence $\mathcal{O}(N^{-1/2})$.
  - *Deterministic:* Particles interact via deterministic velocity fields. Exactly preserves the gradient-flow structure and entropy dissipation without statistical noise.

The Landau Equation as a Gradient Flow

Entropy and the Score Function

Consider the physical entropy functional (up to a sign):

$$\mathcal{H}(f) = \int_{\mathbb{R}^d} f(v) \log f(v)\, dv$$

The gradient of its first variation with respect to f naturally recovers the **score function**:

$$\nabla_v \frac{\delta \mathcal{H}(f)}{\delta f} = \nabla_v \left(\log f(v) + 1\right) = \nabla_v \log f(v) = \tilde{s}_t(v)$$

Gradient-Flow Formulation

Using this, the Landau equation can be elegantly rewritten to reveal its underlying non-local gradient-flow structure:

$$\partial_t f_t(v) = \nabla_v \cdot \left(f_t(v) \int_{\mathbb{R}^d} A(v - v^*) \left(\nabla_v \frac{\delta \mathcal{H}(f_t)}{\delta f}(v) - \nabla_{v^*} \frac{\delta \mathcal{H}(f_t)}{\delta f}(v^*) \right) f_t(v^*)\, dv^* \right)$$

- **Geometric Interpretation:** The equation drives the density f_t to minimize the entropy $\mathcal{H}(f)$ with respect to a specific geometry dictated by the collision kernel $A(z)$.
- **H-Theorem (Entropy Dissipation):** This structure strictly guarantees that $\frac{d}{dt}\mathcal{H}(f_t) \leq 0$, ensuring relaxation to the Maxwellian equilibrium.

Key Literature and Foundational Contributions

The development of structure-preserving deterministic particle methods has been driven by profound advances in applied analysis:

- **J. A. Carrillo et al.:** Pioneered the numerical analysis of Wasserstein gradient flows. Demonstrated that deterministic interacting particles can strictly preserve free-energy dissipation and macroscopic invariants for aggregation-diffusion equations.
- **Carrillo, Craig & Patacchini:** Advanced the rigorous analysis of deterministic particle approximations. Co-developed the **Blob Method for diffusion**, providing convergence guarantees for regularized particle interactions.
- **Recent Era (Huang & Wang, 2025):** Scaled these deterministic paradigms to high dimensions using Neural Score Matching (SBP), replacing costly KDE/Blob interactions with learned score fields.

Our Work: *We build on this legacy by elevating the deterministic, score-driven particle method to a fully global-in-time continuous neural framework.*

Classical Route to the Score: KDE and the Blob Method

The obstacle

The dynamics need the score $\tilde{s}_t = \nabla_v \log f_t$, which requires the *density* f_t. But a particle solver only has locations $\{v_t^{(i)}\}$ — not a density.

- **Classical answer (deterministic / Blob method):** reconstruct a smoothed density by **kernel density estimation (KDE)**,

$$f_t^\varepsilon(v) = \frac{1}{N}\sum_{i=1}^{N} \psi_\varepsilon\left(v - v_t^{(i)}\right), \qquad s_t(v) \approx \nabla_v \log f_t^\varepsilon(v).$$

- The **Blob method** (Carrillo, Craig & Patacchini) regularizes the entropy through this mollified density, retaining *exact conservation* and entropy dissipation.
- **Bottleneck:** KDE accuracy degrades *exponentially with dimension* (curse of dimensionality), and pairwise evaluation costs $\mathcal{O}(N^2)$ per step.

This motivates estimating the score **directly** — *without ever forming a density.*

Recent progress using score matching

Let us forget about PDEs for the moment !

The Score Function

The **score** of a probability density $f(v)$ is defined as the gradient of its log-density:

$$\tilde{s}(v) := \nabla_v \log f(v)$$

- **Goal:** Train a neural network $s_\theta(v)$ to approximate the true score $\tilde{s}(v)$.
- **Ideal Loss (Explicit Score Matching):**

$$\mathcal{L}_{\mathrm{ESM}}(\theta) = \mathbb{E}_{v\sim f}\left[\|s_\theta(v) - \nabla_v \log f(v)\|^2\right]$$

- **The Bottleneck:** In particle methods, we only have particle positions $V_i \sim f$. We **do not know** the true density $f(v)$ or its gradient $\nabla_v f(v)$!

What is the Score, Physically?

The score is the entropic (osmotic) drift

From $\tilde{s}_t = \nabla_v \log \tilde{f}_t = \dfrac{\nabla_v \tilde{f}_t}{\tilde{f}_t}$, we have $\tilde{f}_t\, \tilde{s}_t = \nabla_v \tilde{f}_t$. So the score is the velocity-space force by which entropy/diffusion spreads mass from high- to low-density regions.

- **Near equilibrium it is simply linear.** For a Maxwellian $M \propto \exp\big(-|v-u|^2/2T\big)$,

$$\tilde{s}(v) = -\frac{v-u}{T},$$

  a linear restoring force toward the mean velocity u — a familiar object.
- **The collision dynamics only sees score differences:** the drift depends on the density through $\tilde{s}_t(v) - \tilde{s}_t(v^*)$.

Hence "learning the score" is learning the physical driving field — not fitting a black box. The remaining question: how to estimate $\tilde{s}_t$ from particles alone?

Background: Implicit Score Matching (Hyvärinen Identity)

Hyvärinen Identity (2005)

Using integration by parts, the explicit score matching loss can be rewritten without requiring the unknown true score:

$$\mathbb{E}_{v\sim f}\left[\|s_\theta - \nabla_v \log f\|^2\right] = \mathbb{E}_{v\sim f}\left[\|s_\theta(v)\|^2 + 2\nabla_v \cdot s_\theta(v)\right] + C$$

where $C = \mathbb{E}_{v\sim f}[\|\nabla_v \log f(v)\|^2]$ is a constant independent of θ.

Why is this powerful?

We can train the score network using **only empirical particle samples**, by minimizing the computable Implicit Score Matching (ISM) functional:

$$\mathcal{L}_{\mathrm{ISM}}(\theta) := \mathbb{E}_{v\sim f}\left[\|s_\theta(v)\|^2 + 2\nabla_v \cdot s_\theta(v)\right]$$

Minimizing the empirical ISM loss is equivalent to minimizing the true L^2_v score error.

Proof of the Hyvärinen Identity

Derivation via Integration by Parts

We start by expanding the Explicit Score Matching (ESM) loss:

$$\begin{aligned}\mathcal{L}_{\mathrm{ESM}}(\theta) &= \mathbb{E}_{v\sim f}\left[\|s_\theta(v) - \nabla_v \log f(v)\|^2\right] \\ &= \mathbb{E}_{v\sim f}\left[\|s_\theta(v)\|^2\right] - 2\mathbb{E}_{v\sim f}\left[s_\theta(v)\cdot \nabla_v \log f(v)\right] + \underbrace{\mathbb{E}_{v\sim f}\left[\|\nabla_v \log f(v)\|^2\right]}_{=:C \text{ (independent of } \theta)}\end{aligned}$$

The Cross-Term & Integration by Parts

Using the identity $\nabla_v \log f(v) = \frac{\nabla_v f(v)}{f(v)}$, we rewrite the cross-term:

$$-2\mathbb{E}_{v\sim f}\left[s_\theta(v)\cdot \frac{\nabla_v f(v)}{f(v)}\right] = -2\int_{\mathbb{R}^d} s_\theta(v)\cdot \nabla_v f(v)\, dv$$

Assuming $f(v)\to 0$ as $\|v\|\to\infty$, we apply **integration by parts**:

$$-2\int_{\mathbb{R}^d} s_\theta(v)\cdot \nabla_v f(v)\, dv = 2\int_{\mathbb{R}^d} \big(\nabla_v \cdot s_\theta(v)\big) f(v)\, dv = 2\mathbb{E}_{v\sim f}\left[\nabla_v \cdot s_\theta(v)\right]$$

$$\mathcal{L}_{\mathrm{ISM}}(\theta) = \mathbb{E}_{v\sim f}\left[\|s_\theta(v)\|^2 + 2\nabla_v \cdot s_\theta(v)\right]$$

Bridging Score Matching and Particle Dynamics

Back to Landau: we now feed the learned score into the collision dynamics.

From the mean-field drift to interacting particles

Approximate the density in the mean-field drift by the **empirical measure** $f_t = \frac{1}{N}\sum_j \delta_{v_t^{(j)}}$, turning the integral into a sum:

$$\underbrace{-\int A(v-v^*)\big(\tilde{s}_t(v)-\tilde{s}_t(v^*)\big)\tilde{f}_t(v^*)dv^*}_{\text{mean-field drift } U[\tilde{f}_t](v)} \approx -\frac{1}{N}\sum_{j=1}^{N} A(v-v_t^{(j)})\big(\tilde{s}_t(v)-\tilde{s}_t(v_t^{(j)})\big).$$

Following the characteristics $\dot{v}_t^{(i)} = U(v_t^{(i)})$ gives the interacting particle system (driven by the **true score**):

$$\frac{d}{dt}v_t^{(i)} = -\frac{1}{N}\sum_{j=1}^{N} A(v_t^{(i)}-v_t^{(j)})\big(\tilde{s}_t(v_t^{(i)})-\tilde{s}_t(v_t^{(j)})\big)$$

The Score-Driven Approximation

By replacing the unknown true score $\tilde{s}_t$ with our neural approximation s_θ (learned via Implicit Score Matching), we obtain a computable, deterministic dynamical system:

$$\frac{d}{dt}v_t^{(i)} = -\frac{1}{N}\sum_{j=1}^{N}$$

Recent Progress: Score-Based Particle (SBP) Method

- **Huang-Wang (2025)** proposed the Score-Based Particle (SBP) method, which employs an alternating sequential scheme at each time step t_n:

Sequential Two-Step Update

Step 1: Score Learning (Implicit Score Matching): Train s_θ using the current particle configuration $\{v_{t_n}^{(i)}\}$:

$$\min_\theta \frac{1}{N}\sum_{i=1}^{N}\Big(\|s_\theta(v_{t_n}^{(i)}, t_n)\|^2 + 2\nabla_v \cdot s_\theta(v_{t_n}^{(i)}, t_n)\Big)$$

Step 2: Particle Update (Explicit Time-Stepping): Advance particles to t_{n+1} using the learned score s_θ:

$$v_{t_{n+1}}^{(i)} = v_{t_n}^{(i)} - \Delta t \frac{1}{N}\sum_{j=1}^{N} A(v_{t_n}^{(i)} - v_{t_n}^{(j)})\big(s_\theta(v_{t_n}^{(i)}, t_n) - s_\theta(v_{t_n}^{(j)}, t_n)\big)$$

Limitations of the Sequential Scheme

- **High Computational Cost:** Requires training (or fine-tuning) the score network at *every* time step.
- **Discretization Error:** Accumulates $\mathcal{O}(\Delta t)$ temporal error.
- **Sequential Bottleneck:** Cannot query future states without full step-by-step integration.

Primer: Neural Networks as Flexible Ansätze

What a neural network actually is

A neural network $\mathcal{N}(x;\theta)$ is just a highly flexible parameterized function — a composition of linear maps and simple nonlinearities. With enough parameters it approximates any smooth function on a compact set (universal approximation).

- **"Training"** means choosing parameters θ to minimize a loss $\mathcal{J}(\theta)$ by gradient descent, $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{J}$.
- **Automatic differentiation:** derivatives needed in the loss — both $\nabla_\theta \mathcal{J}$ and *spatial/temporal* derivatives like $\nabla_v \mathcal{N}$, $\partial_t \mathcal{N}$ — are computed **exactly** by the chain rule, not by finite differences.
- **Analogy for physicists:** like a variational / Galerkin ansatz, but with an *adaptive nonlinear basis* instead of fixed modes — and no mesh, only sampled points.

Next: choose the loss to be the PDE residual itself ⇒ the fitted function solves the equation.

Physics-Informed Neural Networks

Question: The PDE is high-dimensional and mesh-based solvers scale poorly. How do we solve it effectively?

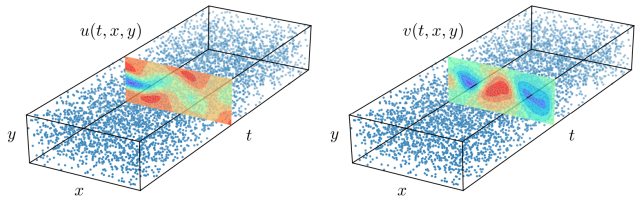
- PINNs do not require a mesh.
- Instead, we randomly sample **collocation points** in the domain.
- Represent solution by a neural network: $u(x) \approx \mathcal{N}(x;\theta)$
- Differential $\mathcal{F}$, $\mathcal{B}$, residuals:

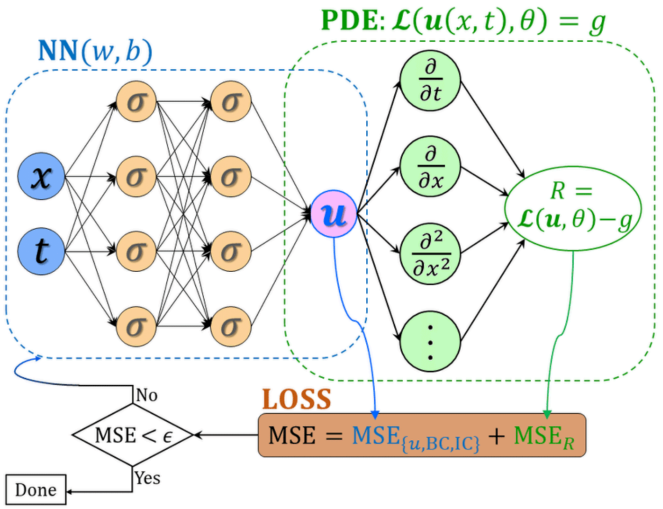
$$r(x;\theta) = \mathcal{F}(x, \mathcal{N}(x;\theta)),$$
$$b(x;\theta) = \mathcal{B}(x, \mathcal{N}(x;\theta)) - \alpha(x)$$

- Loss function

$$\mathcal{J}(\theta) = \|r(x;\theta)\|^2_{L^2(\Omega)} + \lambda \|b(x;\theta)\|^2_{L^2(\partial\Omega)}$$

- The neural network is trained to approximately satisfy the PDE at these scattered points in the domain.





Proposed Approach: Global-in-Time PINN-PM

Key Idea

Instead of explicit time-stepping, we learn the entire interacting particle dynamics as a **global-in-time neural flow**.

- **Trajectory Network (Flow Map):** $\Phi_\xi : \mathbb{R}^d \times [0, T] \to \mathbb{R}^d$

$$\hat{v}_t^{(i)} := \Phi_\xi(V_i, t)$$

- **Score Network:** $s_\theta : \mathbb{R}^d \times [0, T] \to \mathbb{R}^d$

$$s_\theta(v, t) \approx \nabla_v \log f_t(v)$$

Difference from SBP's flow map

SBP also uses a flow map, but it is *assembled by sequential Euler stepping*. Here, time t is a **direct input**: a single global network parameterizes the entire flow map, queryable at any t without integration.

Advantages:

- **Mesh-free in time** (No Δt discretization error).
- **Amortized inference:** query arbitrary times in a single forward pass.
- The Δt error of time-stepping is replaced by a **physics residual** that is *certifiable at deployment* and vanishes as training converges.

PINN-PM vs. SBP (Huang–Wang): Summary of Differences

	SBP (Huang–Wang, 2025)	**PINN-PM (ours)**
Time handling	Forward Euler, sequential	Global-in-time (t as input)
Flow map	Assembled by time-stepping	Directly parameterized by one network
Score learning	Retrained at every step	Trained once over $[0, T]$ (joint)
Inference	Sequential integration	Single forward pass (amortized)
Stability theory	**KL** divergence bound	W_1 / **trajectory** bound
Kernel assumption	**Coercivity** $\lambda_1 I \preceq A \preceq \lambda_2 I$	**Bounded + Lipschitz only** ($0 \preceq A \preceq \lambda_2 I$)
Error sources	Score error ($+ O(\Delta t)$)	Score + **physics residual** + MC
Conservation	Mass/momentum exact, energy $O(\Delta t)$	Approximate (tracked as diagnostic)
Particle count	$22{,}500 \sim 64{,}000$	1,000

- **Key theoretical gain:** our trajectory-based analysis drops the *coercivity* (lower-bound) requirement, covering degenerate/near-singular kernels that entropy-based SBP analysis does not.
- **Honest trade-off:** SBP conserves invariants *exactly by construction*; we trade this for global-in-time inference and end-to-end W_1/density certificates.

Physics-Informed Global Flow Learning

We train the networks jointly via two continuous-time losses evaluated at random collocation times $t_k \in [0, T]$:

1. Implicit Score Matching (ISM) Loss

Enforces statistical consistency (Hyvärinen identity):

$$\widehat{\mathcal{L}}_{\mathrm{ISM},t}(s_\theta) = \frac{1}{N_t N} \sum_{k=1}^{N_t} \sum_{i=1}^{N} \left(\|s_\theta(v_{t_k}^{(i)}, t_k)\|^2 + 2\nabla_v \cdot s_\theta(v_{t_k}^{(i)}, t_k) \right)$$

2. Physics Residual for Landau Dynamics

Penalizes deviation from the characteristic ODE:

$$\mathcal{L}_{\mathrm{phys}}(\xi, \theta) = \frac{1}{N_t N} \sum_{k=1}^{N_t} \sum_{i=1}^{N} \left\| \frac{\partial \Phi_\xi}{\partial t}(V_i, t_k) - U_{t_k}^{\delta}\left(\Phi_\xi(V_i, t_k)\right) \right\|^2$$

Total Loss: $\mathcal{L}(\xi, \theta) = \mathcal{L}_{\mathrm{phys}}(\xi, \theta) + \lambda_{\mathrm{score}} \widehat{\mathcal{L}}_{\mathrm{ISM},t}(s_\theta)$

PINN-PM Architecture

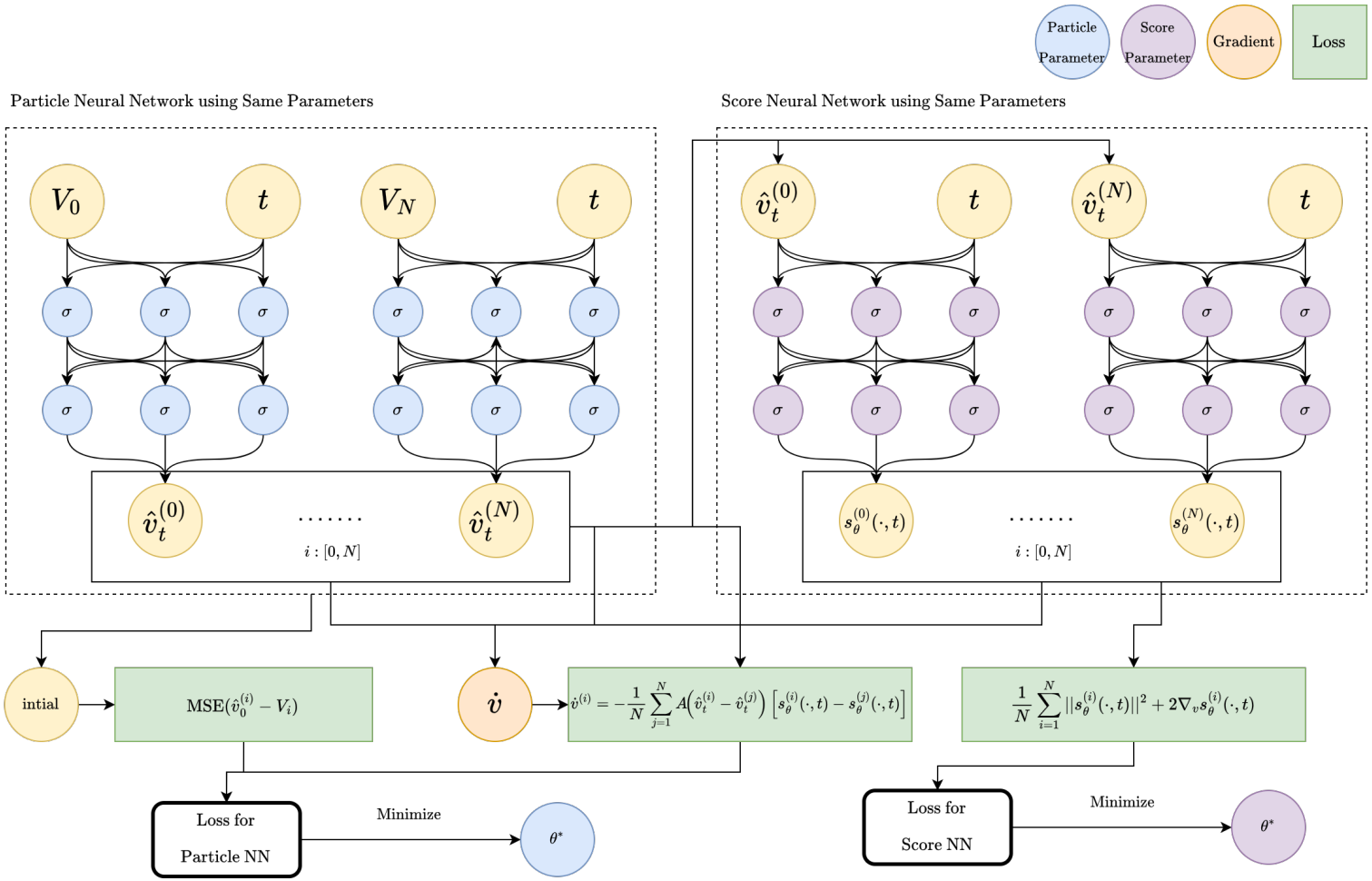


Figure: Conceptual architecture of PINN-PM with amortized inference.

Key Mathematical Notations

Densities and Measures

- $\tilde{f}_t$: Exact solution of the Landau equation (PDE level).
- $\mu_t^N = \frac{1}{N}\sum_{i=1}^N \delta_{v_t^{(i)}}$: Exact empirical measure (driven by true score).
- $f_t = \frac{1}{N}\sum_{i=1}^N \delta_{\hat{v}_t^{(i)}}$: Learned empirical measure (PINN-PM output).

Trajectories and Networks

- $v_t^{(i)}$: Exact interacting-particle trajectory.
- $\hat{v}_t^{(i)} = \Phi_\xi(V_i, t)$: Learned trajectory produced by the neural flow map.
- $\tilde{s}_t(v) = \nabla_v \log \tilde{f}_t(v)$: Exact score function.
- $s_\theta(v, t)$: Learned score network.

Drift and Metrics

- $U[\tilde{f}_t](v)$: Exact mean-field drift.
- $U_t^\delta(v)$: Approximate mean-field drift (using s_θ and f_t).
- $W_1(\mu, \nu)$: Wasserstein-1 distance between probability measures.

Hierarchy of Approximations: Three Levels of Dynamics

1. The PDE Level (Continuous: $\tilde{f}_t$)

Exact macroscopic solution driven by the true score $\tilde{s}_t = \nabla \log \tilde{f}_t$:

$$\partial_t \tilde{f}_t(v) = \nabla_v \cdot \left(\tilde{f}_t(v) \int A(v - v^*)\big(\tilde{s}_t(v) - \tilde{s}_t(v^*)\big) \tilde{f}_t(v^*)\, dv^* \right)$$

2. The Exact Particle Level (Discrete: μ_t^N)

Reference N-particle system. The integral is approximated by an empirical sum (Monte Carlo error), but it is still driven by the *true* score $\tilde{s}_t$:

$$\frac{d}{dt} v_t^{(i)} = -\frac{1}{N} \sum_{j=1}^{N} A(v_t^{(i)} - v_t^{(j)})\big(\tilde{s}_t(v_t^{(i)}) - \tilde{s}_t(v_t^{(j)})\big)$$

3. The Neural Particle Level (Learned: f_t)

Our computed system $\hat{v}_t^{(i)} = \Phi_\xi(V_i, t)$. Driven by the *learned* score s_θ, which introduces score error and a physics residual $\rho^{(i)}(t)$:

$$\frac{\partial}{\partial t} \hat{v}_t^{(i)} = -\frac{1}{N} \sum_{j=1}^{N} A(\hat{v}_t^{(i)} - \hat{v}_t^{(j)})\big(s_\theta(\hat{v}_t^{(i)}, t) - s_\theta(\hat{v}_t^{(j)}, t)\big) + \rho^{(i)}(t)$$

Level 1 & 2: PDE and Exact Particle Dynamics

Level 1: Exact PDE Solution ($\tilde{f}_t$)

The true solution to the spatially homogeneous Landau equation.

- **Density:** $\tilde{f}_t(v)$
- **Score:** $\tilde{s}_t(v) = \nabla_v \log \tilde{f}_t(v)$

Level 2: Exact Interacting Particles (μ_t^N)

A reference finite-particle system driven by the *true* score.

- **Trajectories ($v_t^{(i)}$):** Driven by the exact score differences $\tilde{s}_t(v_t^{(i)}) - \tilde{s}_t(v_t^{(j)})$.
- **Empirical Measure:** $\mu_t^N := \frac{1}{N} \sum_{i=1}^N \delta_{v_t^{(i)}}$
- **Role (Error Isolation):** Acts as a theoretical bridge to decouple the error sources via the triangle inequality:

$$W_1(f_t, \tilde{f}_t) \leq \underbrace{W_1(f_t, \mu_t^N)}_{\substack{\text{Neural Error} \\ \text{(Score error + Residual)}}} + \underbrace{W_1(\mu_t^N, \tilde{f}_t)}_{\substack{\text{Monte Carlo Error} \\ \text{Scaling as } \mathcal{O}(N^{-1/2})}}$$

Level 3: Learned Neural Particle Dynamics

Level 3: Neural Particles (f_t)

The actual computational output of our PINN-PM framework.

- **Trajectories ($\hat{v}_t^{(i)}$):** Generated directly by the neural flow map $\Phi_\xi(V_i, t)$.
- **Empirical Measure:** $f_t := \frac{1}{N}\sum_{i=1}^{N} \delta_{\hat{v}_t^{(i)}}$
- **Learned Score:** $s_\theta(v, t)$

Where does the error come from?

The neural trajectories $\hat{v}_t^{(i)}$ differ from the exact trajectories $v_t^{(i)}$ due to:

1. **Score Error:** $s_\theta \neq \tilde{s}_t$
2. **Physics Residual:** $\partial_t \Phi_\xi$ does not perfectly satisfy the ODE.

Connecting the Dots: The Error Decomposition Strategy

We quantify the total error using the Wasserstein-1 distance (W_1). By the triangle inequality, the total error splits naturally:

$$W_1(f_t, \tilde{f}_t) \leq \underbrace{W_1(f_t, \mu_t^N)}_{\text{Neural Error}} + \underbrace{W_1(\mu_t^N, \tilde{f}_t)}_{\text{Monte Carlo Error}}$$

- **Neural Error ($W_1(f_t, \mu_t^N)$):** Controlled by our deterministic trajectory stability analysis. It is driven by the ISM loss (score error) and the physics residual.
- **Monte Carlo Error ($W_1(\mu_t^N, \tilde{f}_t)$):** Controlled by classical propagation-of-chaos results. It scales as $\mathcal{O}(N^{-1/2})$.

This decoupling allows us to prove that driving the PINN losses to zero guarantees convergence to the true macroscopic physics.

Oracle-to-Dynamics Error Propagation

- We establish a deterministic, global-in-time stability analysis in an L^2_v framework.
- The total error is decoupled into three interpretable sources:
  1. **Score Error** (δ_2): Approximation quality of s_θ.
  2. **Measure Mismatch** (Δ_f): Distance between empirical and exact densities.
  3. **Physics Residual** (δ_{phys}): Deviation of Φ_ξ from exact transport.

Trajectory Error with PINN Residual (Theorem 1)

Let $e_t = \hat{v}_t - v_t$. The Grönwall stability bound yields:

$$\frac{d}{dt}\|e_t\|^2 \lesssim C(t)\|e_t\|^2 + \delta_2(t)^2 + \Delta_f(t)^2 + \delta_{\text{phys}}(t)^2$$

All errors accumulate additively inside the continuous integral.

Unlike SBP's entropy/KL analysis, this trajectory argument needs only **boundedness + Lipschitz** *of A — no coercivity (lower bound λ_1) on the collision kernel.*

Proof Sketch: Trajectory Stability (The Core Mechanism)

The Challenge: Non-local Drift Mismatch

To bound the trajectory error $e_t = \hat{v}_t - v_t$, we differentiate:

$$\frac{1}{2}\frac{d}{dt}\|e_t\|^2 = e_t \cdot \Big(\underbrace{U_t^\delta(\hat{v}_t) - U[\tilde{f}_t](v_t)}_{\text{Drift Mismatch}} + \rho(t) \Big)$$

The drift depends on the entire distribution. How do we disentangle the error?

Step 1: Drift Splitting

We split the drift mismatch into two parts:

$$U_t^\delta(\hat{v}_t) - U[\tilde{f}_t](v_t) = \underbrace{\Big(U_t^\delta(\hat{v}_t) - U[\tilde{f}_t](\hat{v}_t) \Big)}_{\text{(I) Model Error at fixed state}} + \underbrace{\Big(U[\tilde{f}_t](\hat{v}_t) - U[\tilde{f}_t](v_t) \Big)}_{\text{(II) Lipschitz Growth}}$$

- **Part (II)** is easily bounded by $L_U(t)\|e_t\|$ using the Lipschitz continuity of the exact drift.
- **Part (I)** contains both score error and measure mismatch. This is the hard part!

Proof Sketch: Bounding the Model Error via Optimal Transport

To bound Part (I), we introduce an **Intermediate Drift** $\bar{U}_t$ using the learned score s_θ but integrated against the *exact* measure $\tilde{f}_t$:

$$\bar{U}_t(v) := -\int A(v - v^*)\big(s_\theta(v,t) - s_\theta(v^*,t)\big)\,\tilde{f}_t(v^*)\,dv^*.$$

U_t^δ: score s_θ, measure f_t | $\bar{U}_t$: score s_θ, measure $\tilde{f}_t$ | $U[\tilde{f}_t]$: score $\tilde{s}_t$, measure $\tilde{f}_t$

Step 2: The Intermediate Drift Trick (change one ingredient at a time)

$$(\mathrm{I}) = \underbrace{\Big(U_t^\delta(\hat{v}_t) - \bar{U}_t(\hat{v}_t)\Big)}_{\text{Measure Mismatch } (f_t \to \tilde{f}_t)} + \underbrace{\Big(\bar{U}_t(\hat{v}_t) - U[\tilde{f}_t](\hat{v}_t)\Big)}_{\text{Score Error } (s_\theta \to \tilde{s}_t)}$$

- **Score Error:** Bounded directly by $\lambda_2\|s_\theta - \tilde{s}_t\|$ because the interaction kernel $A(z)$ is bounded ($\|A\| \le \lambda_2$).
- **Measure Mismatch (The clever part):** We rewrite this term as an integral with respect to $(f_t - \tilde{f}_t)$. By defining a test function $g(v^*) = A(\hat{v}_t - v^*)(s_\theta(\hat{v}_t) - s_\theta(v^*))$ and applying **Kantorovich-Rubinstein Duality**, we bound it exactly by the Wasserstein distance:

$$\|U_t^\delta(\hat{v}_t) - \bar{U}_t(\hat{v}_t)\| \le \mathrm{Lip}(g)\cdot W_1(f_t, \tilde{f}_t)$$

This seamlessly bridges the interacting particle dynamics to Optimal Transport, allowing us to close the Grönwall loop!

Master Certificate & Density Reconstruction

Master Certificate in W_1 (Theorem 4)

Assuming a finite-particle approximation bound $\mathcal{O}(N^{-1/2})$, we bridge the learned particles f_t to the exact PDE solution $\tilde{f}_t$:

$$\mathbb{E}W_1(f_t, \tilde{f}_t) \leq C(T)\left(\varepsilon_{\text{score}} + \varepsilon_{\text{phys}}\right)^{1/2} + C(T)N^{-1/2}$$

Density Reconstruction Bound (Theorem 5)

For KDE bandwidth $\varepsilon > 0$, the L_v^2 density error decomposes into bias, variance, and trajectory components:

$$\mathbb{E}\left[\|f_{t,N,\varepsilon}^{\text{approx}} - \tilde{f}_t\|_{L_v^2}^2\right] \leq C(T)\left(\underbrace{\varepsilon^4}_{\text{Bias}} + \underbrace{\frac{1}{N\varepsilon^d}}_{\text{Var}} + \underbrace{\varepsilon^{-(d+2)}\left(\varepsilon_{\text{score}} + \varepsilon_{\text{phys}} + N^{-1}\right)}_{\text{Trajectory-induced transport error}}\right)$$

Experimental Setup: Models and Architectures

Proposed Method

PINN-PM: Global-in-time trajectory network Φ_ξ and score network s_θ.

- **Optimizer:** Adam ($\mathrm{lr} = 10^{-4}$), **Activation:** SiLU.
- **Particle Count:** $N = 1,000$ for all PINN-PM experiments.

Baseline Methods (Explicit Time-Stepping with $\Delta t = 0.01$)

1. **SBP (Huang & Wang, 2025):** Score-Based Particle method using sequential neural score matching. Uses $N = 14,400 \sim 64,000$ particles.
2. **Blob (Deterministic Particle):** Classical kernel-based score estimation. Uses the same N as SBP.

Evaluation Metrics: Mathematical Definitions

To rigorously assess our framework, we track the following quantities evaluated over N_{eval} particles $(\hat{v}_t^{(i)})$:

1. Trajectory & Score Accuracy

- **Trajectory L^2 Error:** Deviation from the analytic characteristics $v_t^{(i)}$.

$$\text{Err}_{\text{traj}}(t) := \frac{\sum_{i=1}^{N_{\text{eval}}} \|\hat{v}_t^{(i)} - v_t^{(i)}\|^2}{\sum_{i=1}^{N_{\text{eval}}} \|v_t^{(i)}\|^2}$$

- **Relative Fisher Divergence (RFD):** Computes the L_v^2 score error.

$$\text{RFD}(t) := \frac{\sum_{i=1}^{N_{\text{eval}}} \|s_\theta(\hat{v}_t^{(i)}, t) - \tilde{s}_t(\hat{v}_t^{(i)})\|^2}{\sum_{i=1}^{N_{\text{eval}}} \|\tilde{s}_t(\hat{v}_t^{(i)})\|^2}$$

2. Physical Structure Diagnostics

- **Kinetic Energy:** Verifies the conservation law.

$$\mathcal{E}(t) := \frac{1}{2N_{\text{eval}}} \sum_{i=1}^{N_{\text{eval}}} \|\hat{v}_t^{(i)}\|^2$$

- **Entropy Dissipation Proxy:** Verifies the gradient-flow structure.

$$\mathcal{D}(t) := -\frac{1}{N_{\text{eval}}^2} \sum_{i,j} s_\theta(\hat{v}_t^{(i)}, t)^\top A(\hat{v}_t^{(i)} - \hat{v}_t^{(j)})\big(s_\theta(\hat{v}_t^{(i)}, t) - s_\theta(\hat{v}_t^{(j)}, t)\big)$$

Baseline Method: The Deterministic Blob Method

What is the Blob Method?

A classical deterministic particle method where the unknown density is reconstructed at every time step using **Kernel Density Estimation (KDE)** with a smooth "blob" kernel K_ε:

$$f_t^\varepsilon(v) = \frac{1}{N}\sum_{i=1}^{N} \varepsilon^{-d} K\left(\frac{v - v_t^{(i)}}{\varepsilon}\right)$$

The score is then explicitly approximated as $s_t(v) \approx \nabla_v \log f_t^\varepsilon(v)$.

Advantages

- **Structure-Preserving:** Conserves mass, momentum, and energy exactly (when using symmetric kernels).
- **Deterministic Evolution:** Free from the stochastic noise typical of Monte Carlo (DSMC) methods.
- **Theoretical Foundation:** Well-established convergence guarantees.

Limitations

- **Curse of Dimensionality:** KDE accuracy degrades exponentially as dimension d increases.
- **Computational Bottleneck:** Requires $\mathcal{O}(N^2)$ operations per time step for pairwise score evaluation.
- **Time-Stepping Error:** Still relies on sequential numerical integration, incurring $\mathcal{O}(\Delta t)$ error.

Benchmark: The BKW Exact Solution

Why BKW?

The **Bobylev–Krook–Wu (BKW)** solution is one of the very few *closed-form, time-dependent* solutions of a *nonlinear* kinetic collision equation (Maxwell molecules, $\gamma = 0$).

- **Shape:** a Maxwellian modulated by a polynomial correction that relaxes to the pure Maxwellian,

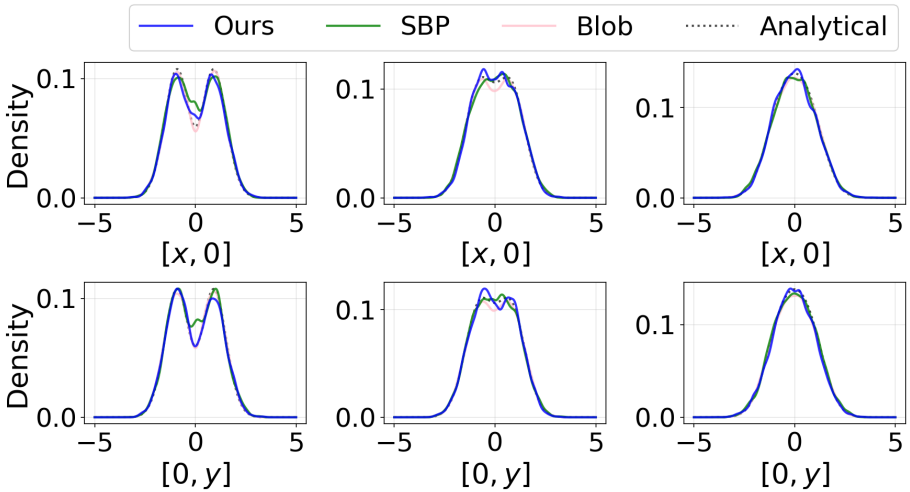
$$\tilde{f}_t(v) = M_{K(t)}(v)\, \underbrace{P\big(|v|^2;\, K(t)\big)}_{\to 1}, \qquad K(t) \to 1 \text{ as } t \to \infty,$$

  with the density $\tilde{f}_t$ *and* the score $\tilde{s}_t = \nabla_v \log \tilde{f}_t$ both known in closed form (see Huang & Wang, 2025).
- **Why it matters:** it provides an exact reference for the **density**, the **score**, and the **characteristic trajectories** — so we can measure *true* errors (density L^2, trajectory L^2, Fisher divergence), not just qualitative behavior.
- We evaluate both the **2D** and **3D** BKW solutions.

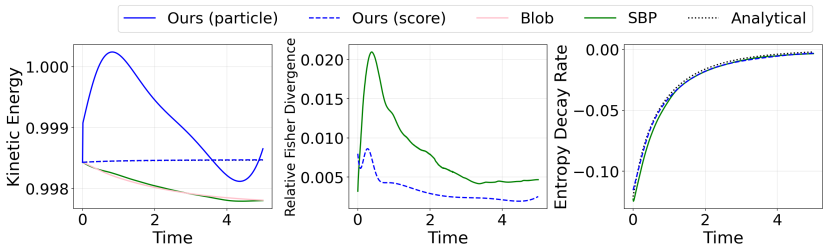
BKW 2D Benchmark (Maxwell Case, $\gamma = 0$)

Setup: Interaction kernel $A(z) = C_0(|z|^2 I - z \otimes z)$. Exact analytic density and score are known.

- **Efficiency:** PINN-PM uses $N = 1,000$ particles vs. SBP/Blob using $N = 22,500$.
- **Observation:** PINN-PM achieves competitive density matching without Δt error.

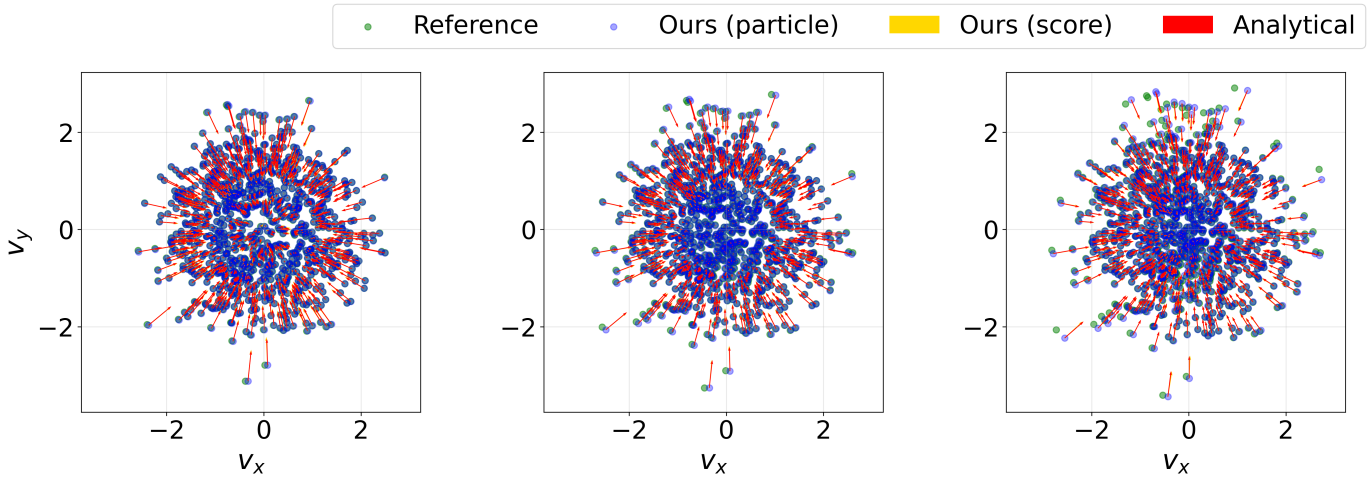


Density Slices vs. Analytic Solution



Conservation (Energy) & Score Accuracy (RFD)

BKW 2D: Simulation of the Learned Transport



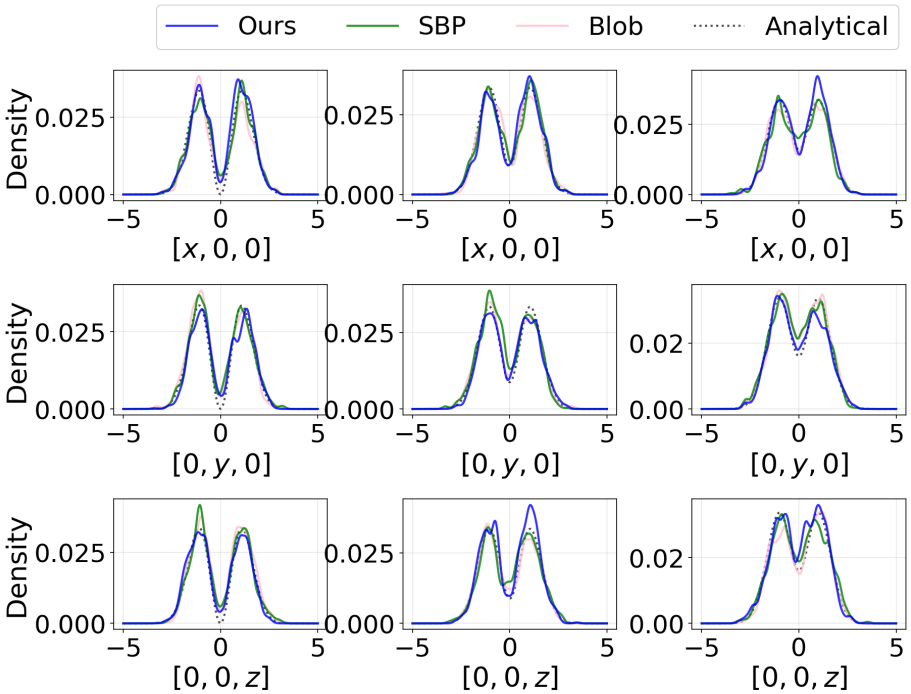
▶ **Click the figure to play** the BKW-2D evolution (opens in the system video player).

- The global flow map $\Phi_\xi(V_i, t)$ transports the $N = 1{,}000$ particles continuously in time — *no* sequential time-stepping.
- The learned density tracks the analytic BKW solution as it relaxes toward the Maxwellian.

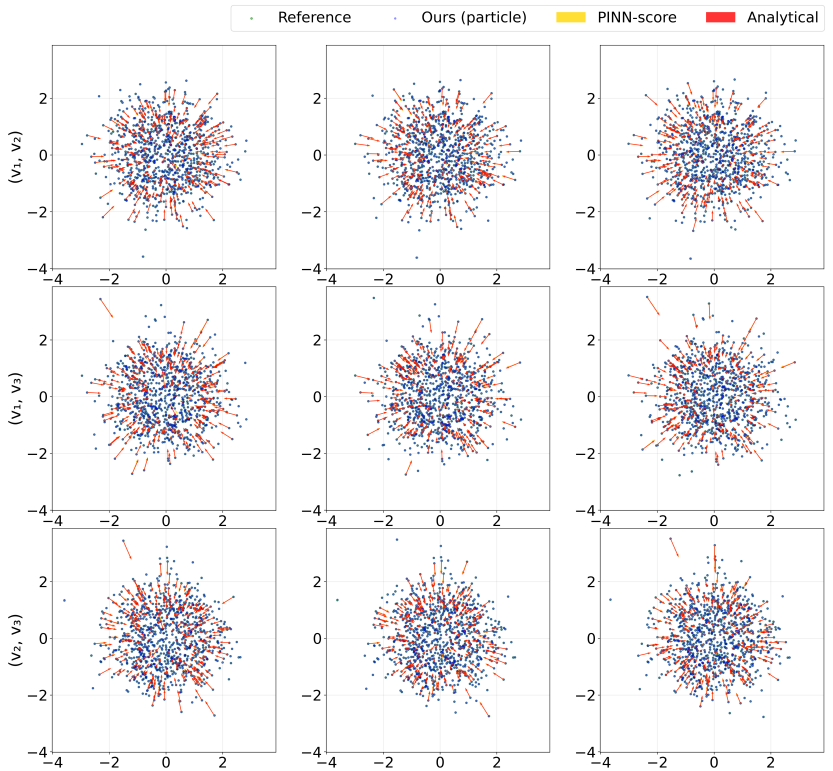
BKW 3D Benchmark (Maxwell Case, $\gamma = 0$)

Setup: Scaling to 3D. PINN-PM uses $N = 1,000$ vs. SBP/Blob using $N = 64,000$.

- **Observation:** Even with 64× fewer particles and no time-stepping, PINN-PM's global network stably captures the 3D characteristic flow.



1D Marginals along $(x, 0, 0)$, $(0, y, 0)$, $(0, 0, z)$



Particle Transport & Score Alignment

Quantitative Summary: Accuracy at a Fraction of the Particles

Method	**Particles** (2D / 3D)	**Time** stepping	**Density L^2 err** (2D / 3D)	**Traj. L^2 err** (2D / 3D)
PINN-PM (ours)	1,000	**none** (global)	0.031/0.063	**0.022**/**0.002**
SBP (Huang & Wang)	22,500/64,000	$\Delta t = 0.01$	0.036/0.075	0.050/0.008
Blob (deterministic)	22,500/64,000	$\Delta t = 0.01$	**0.025**/**0.033**	0.033/0.023

Relative L^2 errors at final time ($t = 5$ for 2D, $t = 6$ for 3D); Ours = particle variant. **Bold** = best in column. Values read from Figs. `bkw2d_l2`/`bkw3d_l2` — replace with exact numbers from your data.

- **Trajectory accuracy:** PINN-PM is *lowest* in both 2D and 3D, despite using 22–64× **fewer particles** and *no* time discretization.
- **Density accuracy:** competitive (Blob is slightly lower at final time, but with far more particles and sequential Δt-stepping).
- **Amortized inference:** any query time t is a single forward pass — cost is decoupled from the horizon length. Error trends track the W_1 and KDE bounds (Theorems 1–5).

Reference-Free Benchmarks: Purpose

Why Reference-Free?

In general cases (e.g., Coulomb potentials, arbitrary initial conditions), exact analytic solutions do not exist. We evaluate whether the models preserve the intrinsic **structural properties** of the Landau equation.

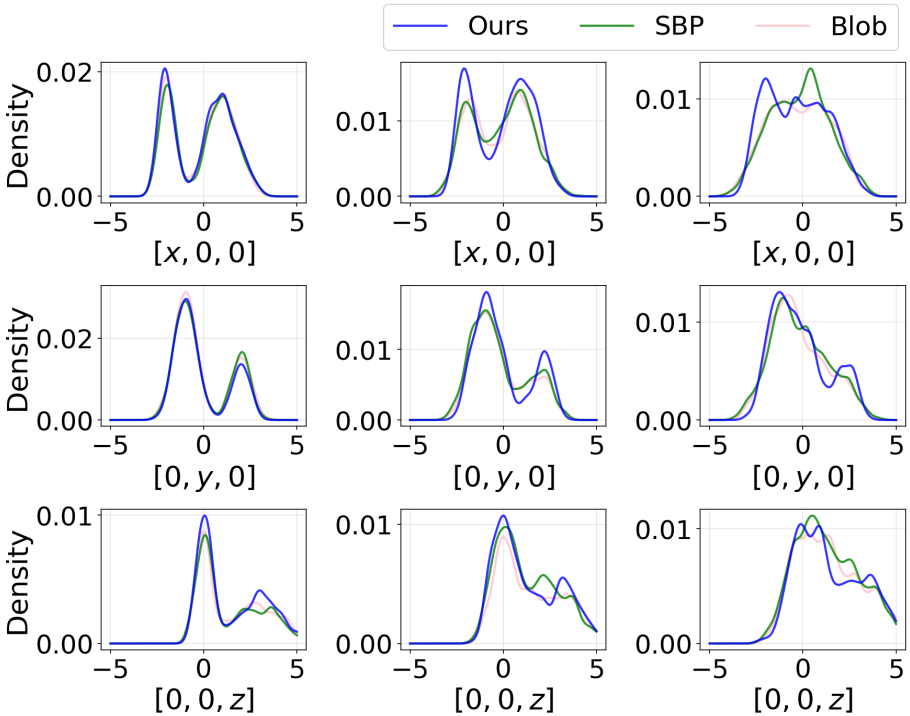
Key Structural Properties to Verify:

1. **Qualitative Density Evolution:** No artificial merging or spurious oscillations.
2. **Conservation Laws:** Strict preservation of mass and kinetic energy.
3. **H-Theorem:** Monotone relaxation (Entropy dissipation).

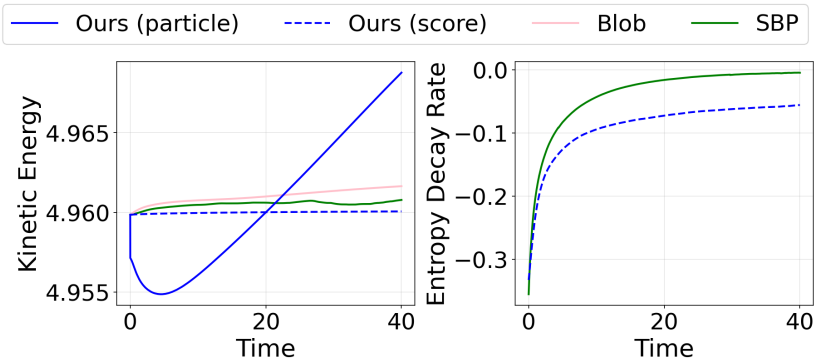
Gaussian Mixture (3D)

Setup: Initial density is a mixture of 8 Gaussians in 3D.

- **Goal:** Test multimodal transport and verify that nonlinear interactions do not cause artificial mode merging. ($N = 1,000$ vs. $64,000$)



Density Slices (Preservation of Multimodal Structure)

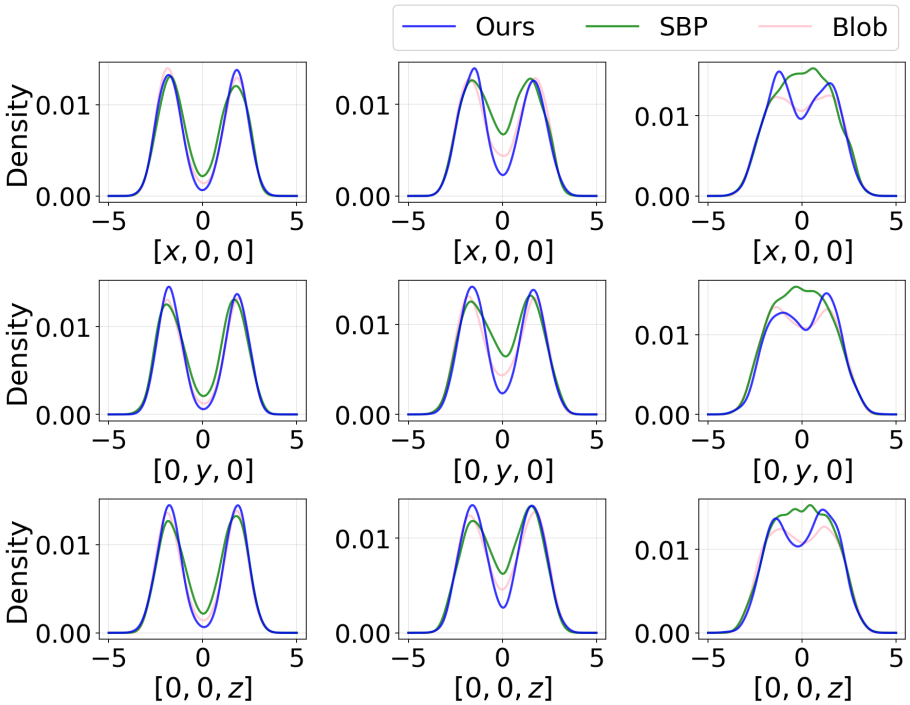


Stable Kinetic Energy & Entropy Decay

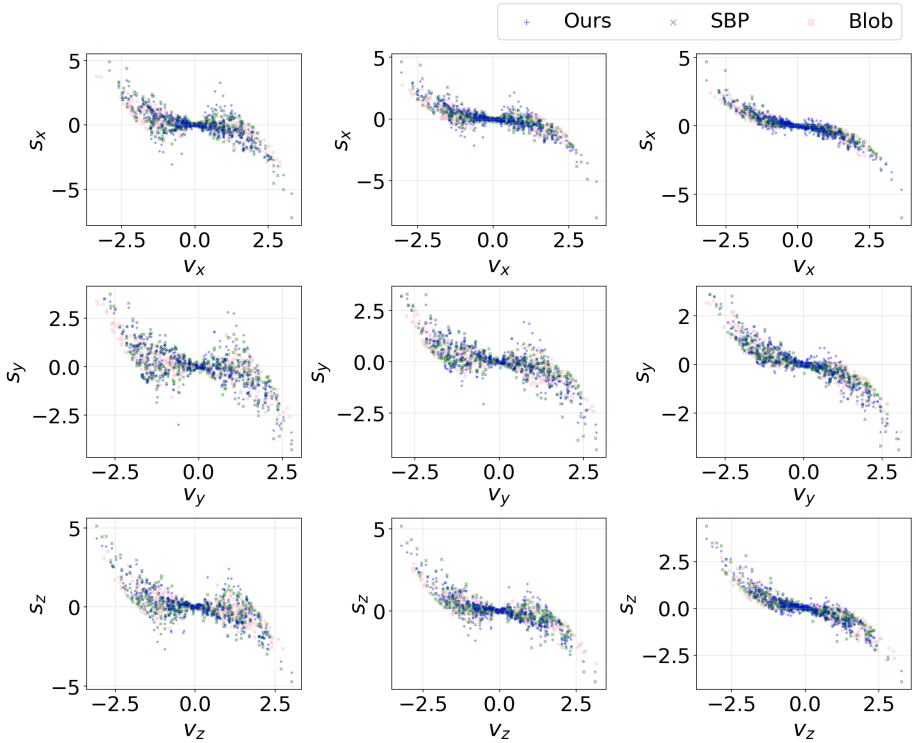
Rosenbluth Distribution (Coulomb Case, $\gamma = -3$)

Setup: Ring-shaped initial distribution with a Coulomb potential.

- **Goal:** The interaction kernel is near-singular. Tests the robustness of the learned score and trajectory residual control under extreme non-linearity.



Preservation of Singular Ring Geometry

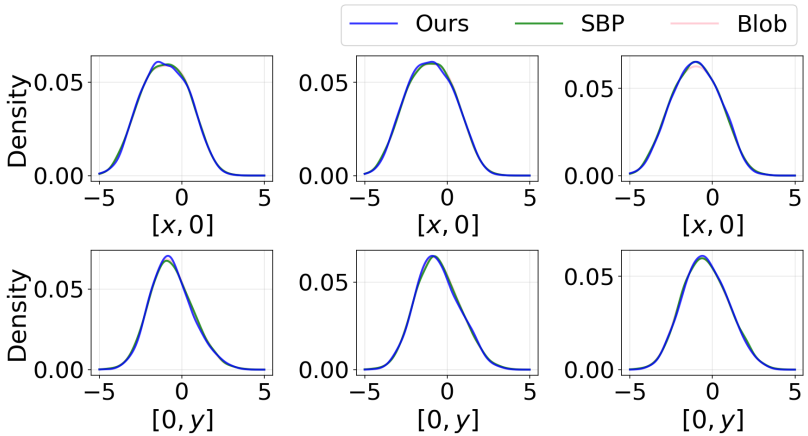


Score Regularity (No spurious oscillations)

Anisotropic & Truncated Distributions (2D)

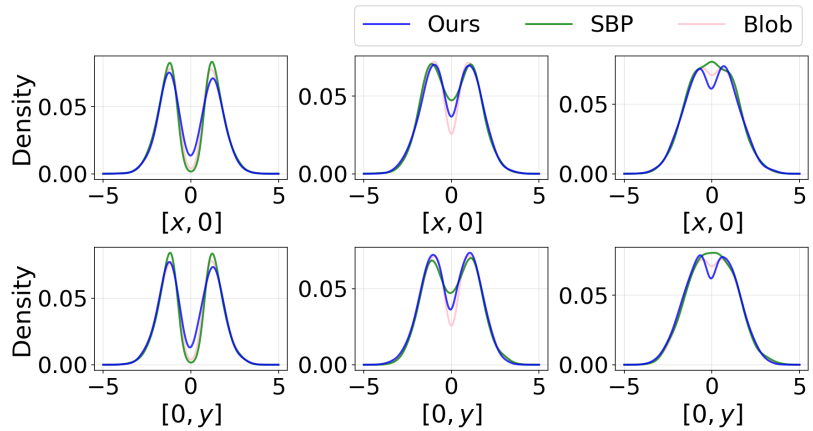
Anisotropic Initialization

Goal: Verify relaxation to isotropic equilibrium (Entropy-driven).



Truncated Initialization

Goal: Handle sharp boundary gradients without kernel smoothing artifacts.



- **Result:** PINN-PM naturally handles sharp gradients and accurately models isotropization, outperforming explicit kernel estimations.

Conclusion

Main Takeaways

- **Global-in-time Neural Simulator:** Replaces discrete, sequential time-stepping (SBP) with a continuous $\Phi_\xi(V_i, t)$ map.
- **Rigorous Error Propagation:** Established deterministic loss-to-dynamics certificates spanning trajectory error to W_1 and KDE density bounds.
- **High Efficiency:** Achieves highly competitive transport accuracy and physical conservation using significantly fewer particles than traditional score-based or Blob methods.

Thank you!